

THOMSON



COURSE TECHNOLOGY

# DISCRETE MATHEMATICAL STRUCTURES:

Theory and Applications

D.S. MALIK AND M.K. SEN

# List of Symbols

## Sets

$A = \{a, e, i, o, u\}$	set with elements $a, e, i, o, u$ ; Set roster method	3
$x \in X$	$x$ is an element of $X$	3
$x \notin X$	$x$ is not an element of $X$	3
$A = \{x \mid x \in S, P(x)\}$	set builder notation	3
$A = \{x \in S \mid P(x)\}$	set builder notation	3
$\mathbb{N}$	the set of all natural numbers	4
$\mathbb{Z}$	the set of all integers	4
$\mathbb{Z}^*$	the set of all nonzero integers	4
$\mathbb{E}$	the set of all even integers	4
$\mathbb{Q}$	the set of all rational numbers	4
$\mathbb{Q}^*$	the set of all non-zero rational numbers	5
$\mathbb{Q}^+$	the set of all positive rational numbers	5
$\mathbb{R}$	the set of all real numbers	5
$\mathbb{R}^*$	the set of all non-zero real numbers	5
$\mathbb{R}^+$	the set of all positive real numbers	5
$\mathbb{C}$	the set of all complex numbers	5
$\mathbb{C}^*$	the set of all non-zero complex numbers	5
$X \subseteq Y$	$X$ is a subset of $Y$	5
$X \not\subseteq Y$	$X$ is not a subset of $Y$	5
$Y \supseteq X$	$Y$ is a superset of $X$	5
$X \subset Y$	$X$ is a proper subset of $Y$	5
$X = Y$	sets $X$ and $Y$ are equal	6
$\emptyset$	empty set	6
$ S $	cardinality or the number of elements of $S$	6
$\mathcal{P}(X)$	power set of $X$	7
$X \cup Y$	$X$ union $Y$	8
$X \cap Y$	$X$ intersection $Y$	9
$\bigcup_{i=1}^n X_i$	union of $n$ sets	13
$X_1 \cup X_2 \cup \dots \cup X_n$	union of $n$ sets	13
$\bigcap_{i=1}^n X_i$	intersection of $n$ sets	13
$X_1 \cap X_2 \cap \dots \cap X_n$	intersection of $n$ sets	13
$\bigcup_{\alpha \in I} A_\alpha$	union of a family of sets	13
$\bigcap_{\alpha \in I} A_\alpha$	intersection of a family of sets	13
$X - Y$	difference of sets $X$ and $Y$	14
$X'$	complement of a set $X$	15
$X \Delta Y$	symmetric difference of sets $X$ and $Y$	16
$(x, y)$	ordered pair	17
$X \times Y$	Cartesian product of $X$ and $Y$	17
$\delta_X$	the diagonal of $X$	18
$(x_1, x_2, \dots, x_n)$	ordered $n$ -tuples	18
$\prod_{i=1}^n X_i$	set of ordered $n$ -tuples	18
$X^n$	$n$ -fold Cartesian product of $X$ with itself	18

## Logic

$\neg p$	negation of $p$	28
$p \wedge q$	$p$ and $q$	29
$p \vee q$	$p$ or $q$	30
$p \rightarrow q$	$p$ implies $q$	31
$p \Leftrightarrow q$	$p$ if and only if $q$	32
$\vdash A$	$A$ is a tautology	35
$A \rightarrow B$	$A$ logically implies $B$	35
$A \equiv B$	$A$ is logically equivalent to $B$	36

$A \Leftrightarrow B$	$A$ is logically equivalent to $B$	36
$P(x)$	predicate or propositional function	54
$P(x_1, x_2, \dots, x_n)$	$n$ -place predicate	55
$\forall x P(x)$	for all $x$ , $P(x)$	56
$\exists x P(x)$	there exists $x$ , $P(x)$	57

## Integers

---

$a > b$	$a$ is greater than $b$	94
$a \text{ div } b$	the quotient of $a$ and $b$ on dividing $a$ by $b$	98
$a \text{ mod } b$	the remainder of $a$ and $b$ on dividing $a$ by $b$	98
$a   b$	$a$ divides $b$	99
$a \nmid b$	$a$ does not divide $b$	99
$\gcd(a, b)$	greatest common divisor (gcd) of $a$ and $b$	101
$\text{lcm}[a, b]$	least common multiple (lcm) of $a$ and $b$	107
$(a_m a_{m-1} a_{m-2} \cdots a_1 a_0)_k$	base $k$ representation	114
$(a_m a_{m-1} a_{m-2} \cdots a_1 a_0)_2$	binary number	114

## Relations and Posets

---

$a R b$	$a$ is $R$ -related or related to $b$	175
$a \not R b$	$a$ is not $R$ -related to $b$	175
$\mathcal{D}(R)$	domain of $R$	178
$\text{Im}(R)$	range or image of $R$	178
$R^{-1}$	inverse of $R$	179
$S \circ R$	composition of $R$ and $S$	182
$[x]$	equivalence class containing $x$	188
$R^\infty$	transitive closure of $R$	195
$(A, \leq)$	partially ordered set	208
$(a, b) \preceq (c, d)$	$a < c$ or $a = c$ and $b \leq d$	209
$a \vee b$	lub of $\{a, b\}$	217
$a \wedge b$	glb of $\{a, b\}$	218

## Matrices

---

$A = [a_{ij}]_{m \times n}$	$A$ is an $m \times n$ matrix	237
$a_{ij}$	$(i, j)$ th element or entry of a matrix	237
$I_n$	identity matrix of size $n \times n$	239
$A + B$	sum of $A$ and $B$	240
$A - B$	difference of $A$ and $B$	241
$\mathbf{0}$	zero matrix	241
$AB$	multiplication of $A$ and $B$	242
$A^m$	multiplication of $A$ with itself $m$ times	244
$A^T$	transpose of $A$	245
$A \vee B$	Boolean join (or join) of $A$ and $B$	247
$A \wedge B$	Boolean meet (or meet) of $A$ and $B$	247
$(a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \cdots \vee (a_n \wedge b_n)$	join of meet expression	247
$A \odot B$	Boolean product (or product) of $A$ and $B$	249
$M_R = [m_{ij}]_{n \times p}$	matrix of the relation $R$	257

## Functions

---

$f : A \rightarrow B$	$f$ is a function from $A$ into $B$	281
$f(a) = b$	$a$ is mapped to $b$ under $f$	281
$f(A)$	range of the function $f$	282
$\text{Im}(f)$	image of $f$	282
$i_A$	identity function on the set $A$	285
$g \circ f$	composition of $f$ and $g$	289
$f^{-1}$	inverse of $f$	298
$f _{A'}$	restriction of $f$ to $A'$	302
$g _A = f$	$g$ is an extension of $f$	303
$f(P)$	image or direct image of $P$ under $f$	303
$f^{-1}(Q)$	inverse image of $Q$ under $f$	304
$\lfloor x \rfloor$	floor of $x$	304
$\lceil x \rceil$	ceiling of $x$	306

$f(x) = \lfloor x \rfloor$	floor function	306
$g(x) = \lceil x \rceil$	ceiling function	306
$ A  =  B $	$A$ and $B$ have the same cardinality	307
$\{a_n\}_{n=1}^{\infty}$	sequence	317
$\{a_n\}_{n=0}^{\infty}$	sequence	317
$\sum_{i=m}^n a_i$	sum of the terms $a_m, a_{m+1}, \dots, a_n$	320
$\prod_{i=m}^n a_i$	product of the terms $a_m, a_{m+1}, \dots, a_n$	324
$ s $	length of a string (word) $s$	325
$\lambda$	empty string or empty word	325
$s_1 s_2$	concatenation of $s_1$ and $s_2$	325
$(S, *)$	mathematical system	333

## Congruences

$a \equiv b \pmod{m}$	$a$ is congruent to $b$ modulo $m$	343
$a \not\equiv b \pmod{m}$	$a$ is not congruent to $b$ modulo $m$	343
$[a]$	congruence class modulo $m$ of the integer $a$	344
$\mathbb{Z}_m$	set of all congruence classes modulo $m$	345
$(a_1, a_2, \dots, a_k) \cdot (b_1, b_2, \dots, b_k)$	dot product	368
$a \leftrightarrow (a \pmod{m_1}, a \pmod{m_2}, \dots, a \pmod{m_k})$	modular representation	384
$\phi(n)$	Euler phi-function	403

## Counting

$P(n, r)$	$r$ -permutations of a set with $n$ distinct elements	439
$C(n, r)$	$r$ -combinations of a set with $n$ distinct elements	443
$\binom{n}{r}$	$r$ -combinations of a set with $n$ distinct elements	443
$P(A)$	probability of the occurrence of $A$	480
$P(A   B)$	conditional probability	484

## Algorithm Analysis

$f(x) = O(g(x))$	$f(x)$ is big- $O$ of $g(x)$	553
$f(n) = \Omega(g(n))$	$f(x)$ is omega of $g(x)$	556
$f(n) = \Theta(g(n))$	$f(x)$ is theta of $g(x)$	556

## Graph Theory

$G = (V, E, g)$	graph	604
$G = (V, E)$	graph	604
$\deg(v)$	degree of $v$	606
$K_n$	complete graph on $n$ vertices	611
$K_{m,n}$	complete bipartite graph on $m$ and $n$ vertices	611
$G - \{v\}$	subgraph obtained from $G$	612
$G - \{e\}$	by deleting the vertex $v$	
$G'$	subgraph obtained from $G$ by deleting the edge $e$	613
$(u = v_1, e_1, v_2, e_2, \dots, e_{n-1}, v_n, e_n, v_{n+1} = v)$	complement of the graph $G$	613
$P - Q$	walk from $u$ to $v$	619
$d(u, v)$	reduction of $P$ by $Q$	622
$N(A)$	distance between two vertices $u, v$	625
$G_1 \cong G_2$	the set of neighbors of $A$	630
$W[i, j]$	$G_1$ is isomorphic to $G_2$	663
$W$	weight of the edge $v_i - v_j$	670
$l(P)$	weight matrix	670
$L(v)$	length of the path $P$	670
	label of $v$	671

# IMPORTANT!

**Below is your registration code to access the student Web site for *Discrete Mathematical Structures: Theory and Applications* by D.S. Malik and M.K. Sen.**

## To Begin:

This unique key code will allow you to register and create an account that is accessible for 120 days.

Once you have registered, your key code will be invalidated and cannot be reused.\*

- 1 Visit [www.course.com/malikdiscrete](http://www.course.com/malikdiscrete)
- 2 Click the New User Registration link in the login box.
- 3 Choose your own User Name and Password and log in to the site.

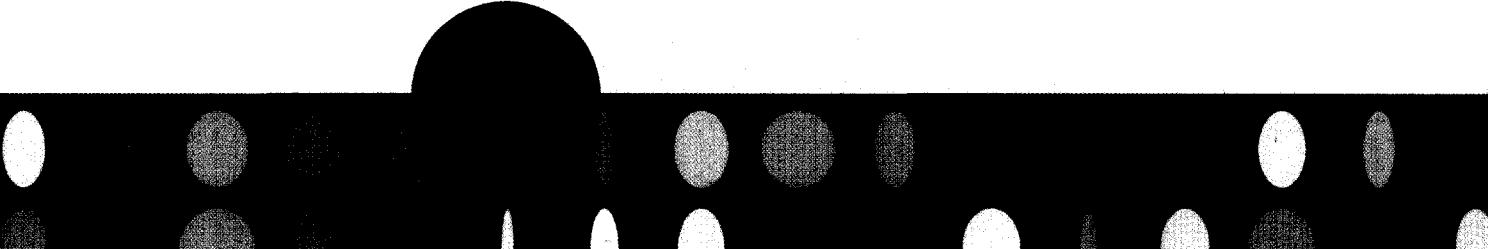
Scratch off to reveal your registration code.\*



## Robust Features of the Web Site Include:

- A Student Solutions Manual that supplies answers and detailed explanations to the odd-numbered exercises in the text.
- A Student Guide to Maple Software that contains chapter-by-chapter suggestions for using Maple to illustrate concepts in the text.
- Review Questions that allow the opportunity to think critically about the material in each chapter.
- Practice Tests for each chapter that provide extra practice in an interactive format.
- Useful Web links that offer additional information about the ideas discussed in each chapter.

**\*THIS BOOK CANNOT BE RETURNED OR RESOLD IF THE  
KEY CODE HAS BEEN ACCESSED.**



# **Discrete Mathematical Structures: Theory and Applications**

D.S. Malik  
M.K. Sen





## Discrete Mathematical Structures: Theory and Applications

by D.S. Malik and M.K. Sen

**Product Manager:**  
Alyssa Pratt

**Managing Editor:**  
Jennifer Muroff

**Senior Acquisitions Editor:**  
Amy Yarnevich

**Development Editor:**  
Laurie Brown

**Senior Production Editor:**  
Aimee Poirier

**Associate Product Manager:**  
Mirella Misiaszek

**Editorial Assistant:**  
Amanda Piantedosi

**Senior Manufacturing Coordinator:**  
Trevor Kallop

**Cover Designer:**  
Betsy Young

**Compositor:**  
Techsetters, Inc.

COPYRIGHT © 2004 Course Technology,  
a division of Thomson Learning, Inc.  
Thomson Learning™ is a trademark used  
herein under license.

Printed in the United States of America

1 2 3 4 5 6 7 8 9 QWT 08 07 06 05 04

For more information, contact Course  
Technology, 25 Thomson Place, Boston,  
Massachusetts, 02210.

Or find us on the World Wide Web at:  
[www.course.com](http://www.course.com)

ALL RIGHTS RESERVED. No part of this  
work covered by the copyright hereon may  
be reproduced or used in any form or by any  
means—graphic, electronic, or mechanical,  
including photocopying, recording, taping,  
Web distribution, or information storage  
and retrieval systems—with or without the written  
permission of the publisher.

For permission to use material from this text  
or product, submit a request online at  
[www.thomsonrights.com](http://www.thomsonrights.com).

Any additional questions about permissions  
can be submitted by e-mail to  
[thomsonrights@thomson.com](mailto:thomsonrights@thomson.com).

### Disclaimer

Course Technology reserves the right to  
revise this publication and make changes  
from time to time in its content without  
notice.

ISBN 0-619-21285-3 with software  
ISBN 0-619-21558-5 without software

*To*

*Sadhana Malik  
Monisha Sen*

# Contents

Preface	ix
<b>CHAPTER 1 Foundations: Sets, Logic, and Algorithms</b>	<b>1</b>
Sets	2
HISTORICAL NOTES: GEORG CANTOR	2
HISTORICAL NOTES: JOHN VENN	7
Mathematical Logic	26
Validity of Arguments	44
Quantifiers and First-Order Logic	53
Proof Techniques	63
Algorithms	73
HISTORICAL NOTES: MUHAMMEND IBN MÙSÀ AL-KHOWÀRIZMÌ	74
Programming Exercises	87
<b>CHAPTER 2 Integers and Mathematical Induction</b>	<b>89</b>
HISTORICAL NOTES: PYTHAGORAS	90
HISTORICAL NOTES: ANDREW WILES	91
Integers	92
Representation of Integers in Computer	111
Mathematical Induction	133
Prime Numbers	149
THE SEARCH FOR PRIME	150
HISTORICAL NOTES: EUCLID	152
HISTORICAL NOTES: PIERRE DE FERMAT	158
Linear Diophantine Equations	163
HISTORICAL NOTES: DIOPHANTUS	163
Programming Exercises	170
<b>CHAPTER 3 Relations and Posets</b>	<b>173</b>
Relations	174
Partially Ordered Sets	207
HISTORICAL NOTES: HELMUT HASSE	212
Application: Relational Database	227
HISTORICAL NOTES: EDGAR CODD	228
Programming Exercises	234
<b>CHAPTER 4 Matrices and Closures of Relations</b>	<b>235</b>
Matrices	236
HISTORICAL NOTES: ARTHUR CAYLEY	236

HISTORICAL NOTES: JAMES JOSEPH SYLVESTER	237
The Matrix of a Relation and Closures	257
HISTORICAL NOTES: STEPHEN MARSHALL	266
Programming Exercises	276
<b>CHAPTER 5 Functions</b>	<b>277</b>
Functions	278
HISTORICAL NOTES: JOHANN PETER GUSTAVE LEJEUNE DIRICHLET	278
HISTORICAL NOTES: GOTTFRIED WILHELM LEIBNITZ	279
Special Functions and Cardinality of a Set	298
Sequences and Strings	315
Binary Operations	331
Programming Exercises	340
<b>CHAPTER 6 Congruences</b>	<b>341</b>
Congruences	342
HISTORICAL NOTES: CARL FRIEDRICH GAUSS	342
Check Digits	358
HISTORICAL NOTES: GEORGE LAURER	367
Linear Congruences	378
Special Congruence Theorems	401
HISTORICAL NOTES: RSA KEY ENCRYPTION	407
Programming Exercises	413
<b>CHAPTER 7 Counting Principles</b>	<b>415</b>
Basic Counting Principles	416
Pigeonhole Principle	431
Permutations	438
Combinations	442
Generalized Permutations and Combinations	448
Binomial Coefficients	455
HISTORICAL NOTES: BLAISE PASCAL	460
Generating Permutations and Combinations	469
Discrete Probability	477
HISTORICAL NOTES: PIERRE SIMON DE LAPLACE	477
Programming Exercises	488
<b>CHAPTER 8 Recurrence Relations</b>	<b>489</b>
Sequences and Recurrence Relations	490
Linear Homogeneous Recurrence Relations	512
Linear Nonhomogeneous Recurrence Relations	527
Programming Exercises	545
<b>CHAPTER 9 Algorithms and Time Complexity</b>	<b>547</b>
Algorithm Analysis	548

Various Algorithms	564
Programming Exercises	600
<b>CHAPTER 10 Graph Theory</b>	<b>601</b>
HISTORICAL NOTES: LEONHARD EULER	602
Graph Definition and Notations	603
Walks, Paths, and Cycles	619
Matrix Representation of a Graph	636
Special Circuits	644
HISTORICAL NOTES: SIR WILLIAM ROWAN HAMILTON	653
Isomorphism	661
Graph Algorithms	669
HISTORICAL NOTES: EDGAR WYBE DIJKSTRA	671
Planar Graphs and Graph Coloring	684
HISTORICAL NOTES: KAZIMIERZ KURATOWSKI	691
Programming Exercises	702
<b>CHAPTER 11 Trees and Networks</b>	<b>703</b>
Trees	704
Rooted Tree	712
Spanning Trees	731
Networks	743
Programming Exercises	766
<b>CHAPTER 12 Boolean Algebra and Combinatorial Circuits</b>	<b>769</b>
Two-Element Boolean Algebra	770
HISTORICAL NOTES: GEORGE BOOLE	770
HISTORICAL NOTES: CLAUDE ELWOOD SHANNON	771
Boolean Algebra	785
Logical Gates and Combinatorial Circuits	794
HISTORICAL NOTES: MAURICE KARNAUGH	811
Programming Exercises	823
<b>CHAPTER 13 Finite Automata and Languages</b>	<b>825</b>
Finite Automata and Regular Languages	826
Finite State Machines with Input and Output	851
Grammars and Languages	860
Programming Exercises	874
<b>Appendix</b>	<b>875</b>
<b>Answers</b>	<b>879</b>
<b>References</b>	<b>893</b>
<b>Index</b>	<b>897</b>

*Discrete Mathematical Structures: Theory and Applications* is an innovative text that introduces a new way of teaching the Discrete Structures course. A course in discrete structures is an integral part of the computer science curriculum. The class may consist of both math and computer science majors and can be taught either by the mathematics department or by the computer science department. Therefore, it is important that a course in discrete structures present a balance of theoretical concepts as well as their relevant applications.

## Approach

The approach that we have taken in this book is a culmination of many years of experience. Our main objective is to make the learning of discrete mathematics easier and fun. Typically, in computer science, a course in discrete mathematics is taken just after programming courses. In many programs, this course becomes a prerequisite of other higher-level courses. In *Discrete Mathematical Structures: Theory and Applications*, we want to give students a solid foundation of theoretical concepts and their applications.

We have been teaching the discrete structures course for a number of years. The textbooks that we have come across tend to be either theory oriented or applications oriented. We do not believe in simply presenting the statement of a theorem and then showing its proof. Showing proof after proof is the surest way to discourage many students. On the other hand, showing application after application without the reinforcement of theoretical results is like following a cook book.

In *Discrete Mathematical Structures: Theory and Applications*, we show why theory is important and how theory connects with applications. Over the years, we have learned that giving an example before and after presenting a theoretical result makes learning easier and effective. Before writing a proof, we usually present examples to show the relevance of the concept. Moreover, we do not just show a proof, we show how the proof is constructed. The same methodology is followed when we present an algorithm. Before and/or after presenting an algorithm, we show how the algorithm works.

This book is written exclusively with students in mind. The language is user-friendly and conducive to learning. Very often we hear statements from students such as “How do I solve problems and write proofs?” To bridge this extremely important gap, we present a set of fully Worked-Out Exercises at the end of each section. These Worked-Out Exercises teach students how to solve problems as well as write proofs—they prepare students to do exercises on their own.

The book contains a rich collection of exercises. Furthermore, at the end of each chapter we include a set of Programming Exercises. Students are encouraged to solve these exercises in the programming language of their choice, such as Maple, C++, or Java.

Although this book is intended for a one-semester course, the book contains more material than could possibly be covered in this time frame. This gives the instructor flexibility in determining topic coverage. The book contains thirteen chapters, and they can be studied out of order depending on individual preference.

## Organization and Coverage

Chapter 1 covers the basics of set theory, logic, and algorithms. We present the basic terminology used in set theory and various results used throughout the book. In the logic section, after presenting the basic material, such as statements and rules of inference, we show various proof techniques. Finally, in the algorithm section, we set the syntax used to write algorithms throughout the book.

Chapter 2 is concerned with the properties of integers and principles of induction. Integers are by far the most important source of examples. We cover basic properties of integers and then show how integers are represented in computer memory. Next, we cover the principles of induction in detail, giving various examples and then discussing how induction is used to prove the correctness of programs, especially loops.

In Chapters 3 and 4 we cover relations, posets, and matrices in detail. We show how graphs and matrices are used to represent relations. Moreover, we use matrices to determine the transitive closure of relations on a finite set. Warshall's algorithm is covered in detail to find the transitive closure. We also show how relations are used in the design of relational databases.

Chapter 5 covers functions in detail. Other than covering various types of functions, we show the relationship between functions and strings.

Chapter 6 is concerned with congruences and their various applications. We focus on how congruences are used in the construction of ISBNs, UPCs, credit card numbers, the scheduling of round robin tournaments, hashing, and code words. This chapter can be studied after Chapter 3, and it is not a prerequisite for the remaining chapters in the book.

Chapter 7 focuses on counting techniques. More specifically, we discuss basic counting principles—the addition and the multiplication principle, pigeonhole principles, permutations, combinations, binomial coefficients, and discrete probability. We also give various algorithms to generate permutations, combinations, and binomial coefficients.

Chapter 8 is concerned with advanced counting techniques using recurrence relations. Following this, we focus on solving linear homogenous recurrence relations and certain linear nonhomogenous recurrence relations. We are especially interested in linear nonhomogenous recurrence relations as they frequently appear in the analysis of algorithms that use divide and conquer techniques. We present enough results so that we can analyze the various algorithms given in Chapter 9.

Chapter 9 focuses on the algorithms and their complexity. We start with showing why algorithm analysis is important and then develop theoretical concepts such as Big- $O$  and theta notations. In the second half of this chapter, we present and analyze various searching and sorting algorithms, as well as discuss algorithms to multiply matrices and an effective way to determine the order in which a sequence of matrices can be multiplied.

Chapter 10 covers graphs in detail. Starting with basic graph theory definitions and terminology, we discuss topics such as subgraphs, walks, paths, circuits, isomorphism of graphs, planer graphs, and graph coloring. Also covered is a way to represent graphs in computer memory as well as various graph algorithms.

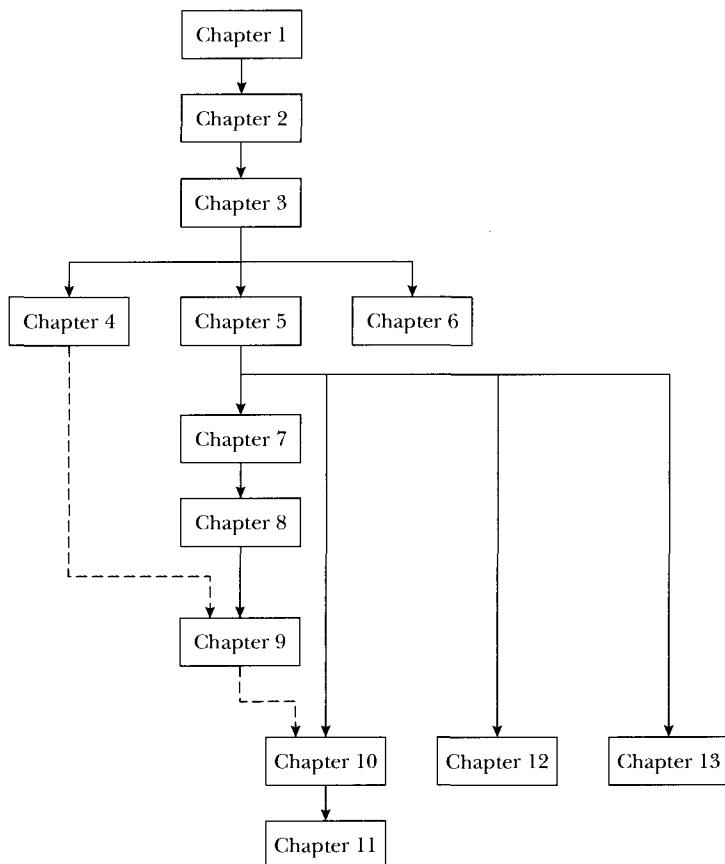
Chapter 11 focuses on trees, special types of trees, and determining spanning and minimal spanning trees. We close this chapter with a discussion of the transport network and present an algorithm to determine a maximal flow in a network.

Chapter 12 is concerned with Boolean algebra and its applications in the design of electric circuits.

Chapter 13 presents an introduction to automata theory and languages.

The Web site accompanying this book contains the following additional material: applications of Boolean algebra in the design of switching circuits, characterization of regular languages by right congruences, nondeterministic finite automata with lambda transitions, and generating functions.

The chapter dependency diagram in Figure 1 shows the dependency of chapters. A dotted line means that that the chapter is not necessarily a prerequisite for the subsequent chapter.



**FIGURE 1**

As shown in Figure 1, Chapters 1, 2, and 3 should be studied in sequence. After studying these chapters there are various choices. In Chapter 9, we describe certain algorithms related to matrices. The basic concept and basic operations of matrices, such as addition and multiplication, are needed to understand these algorithms. Therefore, only these parts from Chapter 4 are need for Chapter 9.

In Chapter 9, we present various notations used in algorithm analysis, such as Big-*O* and theta. In Chapter 10, other than discussing theoretical concepts related to graphs, we also discuss the matrix representation of graphs and applications of graphs in computer science such as shortest path algorithm and topological sort. Moreover, these algorithms are described in detail. Only the concept of theta notation from Chapter 9 is needed for the analysis of the shortest path algorithm. Other than this, Chapter 9 is not a prerequisite for Chapter 10. Similarly, only the basic properties of matrices are needed to study Chapter 10.

## Syllabus Planning

Some of the ways the chapters can be studied are (Chapter 6 can be studied any time after Chapter 3. Therefore, we do not list Chapter 6 in the following sequences.):

1. Study all the chapters in sequence.
2. Study the chapters in the sequence: 1, 2, 3, 4, 5, 10, 11, 7, 8, 9, 12, 13.
3. Study the chapters in the sequence: 1, 2, 3, 5, 4, 10, 11, 7, 8, 9, 12, 13.
4. Study the chapters in the sequence: 1, 2, 3, 5, 4, 10, 11, 12, 13, 6, 7, 8, 9.

## Features

Every chapter in this book includes the following features. These features are both conducive to learning and allow students to learn the material at their own pace.

- *Learning Objectives* offer an outline of the concepts discussed in detail in the chapter.
- *Remarks* highlight important facts about the concepts introduced in the chapter.
- More than 450 visual diagrams, both extensive and exhaustive, illustrate difficult concepts.
- *Numbered Examples* illustrate the key concepts.
- *Worked-Out Exercises* is a set of more than 325 *fully Worked-Out Exercises* at the end of each chapter. These exercises teach how to solve problems and write proofs. We strongly recommend that students study these Worked-Out Exercises very carefully in order to learn problem-solving techniques.
- *Section Review* offers a summary of the concepts covered in the chapter.
- *Exercises* further reinforce learning and ensure that students have, in fact, learned the material.
- *Programming Exercises* challenge students to write programs with a specified outcome.

## Student Resources

**Maple Software.** We are pleased to offer a 120-day trial version of Maple software with every saleable copy of this text. The CD in the back of the book provides access to a fully functional version of the latest release of Maple.

**Student Online Companion Web Site.** In the front of this text, you will find a scratch-off card with a key code that provides full access to a robust Web site, located at [www.course.com/malikdiscrete](http://www.course.com/malikdiscrete). This site includes the following features:

- **Student Solutions Manual** that supplies answers and detailed explanations to the odd-numbered problems in the text.
- **Student Guide to Maple Software** that contains chapter-by-chapter suggestions for using Maple to illustrate concepts in the text.
- **Review Questions** that allow the opportunity to think critically about the material in each chapter.

- **Practice Tests** for each chapter that provide extra practice in an interactive format.
- **Useful Web Links** that offer additional information about the ideas discussed in each chapter.

## Teaching Tools

*Discrete Mathematical Structures: Theory and Applications* includes teaching tools to support instructors in the classroom. The ancillaries that accompany the textbook include an Instructor's Manual, Solutions, Test Banks, and Test Engine, PowerPoint presentations, and Figure Files. All teaching tools available with this book are provided to the instructor on a single CD-ROM and are also available on the Web at [www.course.com](http://www.course.com).

**Electronic Instructor's Manual.** The Instructor's Manual that accompanies this textbook includes:

- Additional instructional material to assist in class preparation, including suggestions for lecture topics
- Solutions to all the exercises, including the Programming Exercises

**ExamView®** This objective-based test generator lets the instructor create paper, LAN, or Web-based tests from testbanks specifically designed for this Course Technology text. Instructors can use the QuickTest Wizard to create tests in fewer than five minutes by taking advantage of Course Technology's question banks—or create customized exams.

**Solutions.** The solution files for all programming exercises in C++ are available at [www.course.com](http://www.course.com), and are also available on the Teaching Tools CD-ROM.

**PowerPoint Presentations.** Microsoft PowerPoint slides are included for each chapter. Instructors might use the slides in a variety of ways, including as teaching aids during classroom presentations or as printed handouts for classroom distribution. Instructors can add their own slides for additional topics introduced to the class.

**Figure Files.** Figure files allow instructors to create their own presentations using figures taken directly from the text.

## Acknowledgements

There are many people that we must thank who, in one way or another, contributed to the success of this book. First, we would like to thank Dr. S.C. Cheng for his support and making suggestions to improve the text. We must also thank students who, during the preparation, were spontaneous in telling us if certain portions needed to be reworded for better understanding and clearer reading. We must thank Lee I. Fenicle, Director, Office of Technology Transfer, Creighton University, Dr. Randall L. Crist, and Dr. Ratish Basu Roy for their involvement, support, and for providing encouraging words when we needed them. We would like to acknowledge the feedback provided by Sunil Kumar Maity and Madhumita Mukherjee.

We owe a great deal to the following reviewers who patiently read each page of every chapter of the current version and made critical comments to improve on the book: Jim Ball, Indiana State University; Jose Cordova, University of Louisiana at Monroe; Joseph Klerlein, Western Carolina University; and Catherine Yan, Texas

A&M University. The reviewers will recognize that their criticisms have not been overlooked and, in fact, made this a better book. Thanks to Development Editor Laurie Brown for carefully editing and promptly returning each chapter. All this would not have been possible without the planning of Managing Editor Jennifer Muroff and Product Manager Alyssa Pratt. Our sincere thanks to Jennifer Muroff and Alyssa Pratt, as well as to Aimee Poirier, Senior Production Editor, and also to the QA department of Course Technology for carefully testing the code. We would especially like to thank Burt LaFountain, QA Tester, for carefully reading the manuscript, solutions to all the exercises, and programming exercises. We would also like to thank Kate Deibel, University of Washington, for providing answers to the programming exercises in Chapters 1–8 and 11, and Jim Bishop, Bryant College, for his help with the biographies.

We are thankful to our parents for their blessings.

Finally, we are thankful for the support of our wives Sadhana and Monisha and our children Shelly, Nilanjan, Debanjan, and Shubhashree. They cheered us whenever we were overwhelmed during the writing of this book. We welcome any comments concerning the text. In spite of our diligent efforts there may still be room for improvement. Comments may be forwarded to the following e-mail address: [malik@creightcr.edu](mailto:malik@creightcr.edu) or [senmk@cal3.vsnl.net.in](mailto:senmk@cal3.vsnl.net.in).

D.S. Malik

M.K. Sen

# Foundations: Sets, Logic, and Algorithms

**The objectives of this chapter are to:**

- Learn about sets
- Explore various operations on sets
- Become familiar with Venn diagrams
- Learn how to represent sets in computer memory
- Learn about statements (propositions)
- Learn how to use logical connectives to combine statements
- Explore how to draw conclusion using various argument forms
- Become familiar with quantifiers and predicates
- Learn various proof techniques
- Explore what an algorithm is

This chapter sets the stage for all that follows and also serves as an appropriate place for codifying certain technical terminologies used throughout the text. In the first section, we discuss sets and their basic properties. We then study mathematical logic in some detail. In this book, the focus is not just on theory but also on applications. When theoretical concepts are presented, we give various examples to clarify the concepts as well as to prove theoretical results, wherever appropriate. Therefore, after discussing sets, we study mathematical logic and describe various proof techniques.

Over the years a revolution in computer technology has changed the ways in which we live and communicate. Computer programs have made tedious computations easy to handle and have

enabled us to achieve results quickly and to a great degree of precision. Therefore, throughout the book we discuss various algorithms that can be implemented in a variety of programming languages, such as C++ and Java. In the last section of this chapter, we introduce algorithms and describe the syntax of the pseudocode used to describe algorithms in this book.

Natural numbers, integers, rational numbers, and real numbers are a great source of examples. We assume that the reader is familiar with these number systems.

## 1.1 SETS

The mathematical theory of sets grew out of the German mathematician Georg Cantor's study of trigonometric series and series of real numbers. The language of sets has since become an important tool for all branches of mathematics, serving as a basis for the precise description of higher concepts and for mathematical reasoning.

Let us begin with the question, what is a *set*? It is fascinating that the answer to this very basic and apparently simple question once jeopardized the very foundation of set theory. In this text, however, we adopt a naive and intuitive point of view and introduce the definition of a *set* according to his definition. According to his definition, *a set is a well-defined collection of distinct objects* of our perception or of our thoughts, to be conceived as a whole.



**Georg Cantor**  
(1845–1918)  
Although considered one of the great German mathematicians,

Cantor was born in St. Petersburg, Russia, in the winter of 1845 to a wealthy Danish merchant. At the age of 11, he moved with his family to Germany where he continued his education, earning a doctorate degree from the University of Berlin in 1867. In 1869 Cantor accepted a post at the University

### HISTORICAL NOTES

of Halle, an undistinguished school for women. His provocative ideas regarding concepts of infinity had put him in bad standing with his contemporaries, and many of them opposed his appointment to the prestigious University of Berlin. Cantor suffered fits of depression due in large part to stress related to his work. He spent the better part of his later years in and out of mental hospitals and ultimately died in a sanatorium.

Cantor is considered to be the founder of set theory, and he estab-

lished its relation to transfinite numbers. He explored paradoxes that had existed in mathematics for centuries and even stumbled upon one of his own, now known as Cantor's paradox. Although his theories were vehemently disputed by his peers, including Leopold Kronecker, his mentor at the University of Berlin, modern mathematicians completely accept Cantor's work.

To develop a perfectly balanced working idea of sets, it is sufficient for a beginner to concentrate on the first part (the italicized part) of this definition. Note that here *well-defined* is an adjective to the noun *collection* and not to the distinct objects that are to be collected to form a set. What this means is that there should be no ambiguity whatsoever regarding the membership of such a collection; *well-defined* means that we can tell for certain whether an object is a member of the collection or not. These objects are called *members* or *elements* of the set.

For example, we can talk about the set of all positive integers, even though no one really knows all of them. But a collection of *some* positive integers is *not* a set because it is not clear whether a particular positive integer, say 5, is a member of this collection or not. For another example, the collection of students taking the discrete mathematics course in your school is a set. On the other hand, the collection of best cars in a city cannot be a set because there is no well-defined notion of best.

We use italic uppercase letters,  $A, B, C, \dots, X, Y, Z$ , to denote sets. A set can be described in various ways, but the main point of any description is to specify the elements of the set in some unambiguous way. One common way, called the **roster method**, to describe a set is to list the elements of the set and enclose them within curly braces. For example, if  $A$  is a set of vowels, then we write

$$A = \{a, e, i, o, u\}.$$

For another example, we can describe the set  $B$  of all positive integers less than 11 as

$$B = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}.$$

Let  $X$  be a set. If  $x$  is an element of  $X$ , then we write  $x \in X$  and say that  $x$  belongs to  $X$ . The symbol  $\in$  stands for *belongs to*, which, like many other notations, was introduced in 1889 by the Italian mathematician Giuseppe Peano (1858–1932) and is believed to be a stylized form of the Greek epsilon. If  $x$  is not an element of  $X$ , then we write  $x \notin X$  and say that  $x$  is not an element of  $X$ . The symbol  $\notin$  stands for *does not belong to*.

### EXAMPLE 1.1.1

Let  $A$  be the set

$$A = \{1, 2, 3, 4, 5\}.$$

Then  $2 \in A$  and  $5 \in A$ . Also,  $6 \notin A$ .

### EXAMPLE 1.1.2

Let  $B$  be the set of first 10 positive odd integers. Then

$$B = \{1, 3, 5, 7, 9, 11, 13, 15, 17, 19\}.$$

It follows that  $9 \in B$  and  $2 \notin B$ .

We also describe sets in the following manner. Let  $S$  be a set. The notation

$$A = \{x \mid x \in S, P(x)\}$$

or

$$A = \{x \in S \mid P(x)\}$$

means that  $A$  is the set of all elements  $x$  of  $S$  such that  $x$  satisfies the property  $P$ . This way of describing a set is called the **set-builder method**.

For example, if  $\mathbb{Z}$  denotes the set of integers, then

$$\mathbb{N} = \{x \mid x \in \mathbb{Z}, x > 0\}$$

or

$$\mathbb{N} = \{x \in \mathbb{Z} \mid x > 0\}.$$

Here the property  $P(x)$  is

$$P(x) : x > 0.$$

### EXAMPLE 1.1.3

In set-builder notation, the set  $B$  of Example 1.1.2 can be described as

$$B = \{x \in \mathbb{Z} \mid x \text{ is odd and } 1 \leq x \leq 19\}.$$

### EXAMPLE 1.1.4

Let  $A = \{2, -2\}$ . Because 2 and  $-2$  are the only integers that satisfy the equation  $x^2 - 4 = 0$ , we can also write  $A$  as

$$A = \{x \mid x \in \mathbb{Z}, x^2 - 4 = 0\}$$

or

$$A = \{x \in \mathbb{Z} \mid x^2 - 4 = 0\}.$$

Here the property  $P(x)$  is

$$P(x) : x^2 - 4 = 0.$$

### EXAMPLE 1.1.5

Let  $A$  be the set described in set-builder form as:

$$A = \{x \mid x \text{ is a complex number and } x^4 = 1\}.$$

Now the equation  $x^4 = 1$ , i.e.,  $x^4 - 1 = 0$ , can be factored as

$$(x + 1)(x - 1)(x - i)(x + i) = 0,$$

where  $i^2 = -1$  or  $i = \sqrt{-1}$ . This implies that the solutions of the equation  $x^4 = 1$ , where  $x$  is a complex number, are  $x = 1, -1, i, -i$ . Therefore, using the roster form, the set  $A$  can be written as

$$A = \{1, -1, i, -i\}.$$

Throughout the book, we will use numbers to provide examples. Therefore, we would like to standardize the symbols to denote various sets of numbers as follows.

$\mathbb{N}$  : The set of all natural numbers (i.e., all positive integers)

$\mathbb{Z}$  : The set of all integers

$\mathbb{Z}^*$  : The set of all nonzero integers

$\mathbb{E}$  : The set of all even integers

$\mathbb{Q}$  : The set of all rational numbers

$\mathbb{Q}^*$  : The set of all nonzero rational numbers

$\mathbb{Q}^+$  : The set of all positive rational numbers

$\mathbb{R}$  : The set of all real numbers

$\mathbb{R}^*$  : The set of all nonzero real numbers

$\mathbb{R}^+$  : The set of all positive real numbers

$\mathbb{C}$  : The set of all complex numbers

$\mathbb{C}^*$  : The set of all nonzero complex numbers

We know that every integer is a real number; that is, every element of  $\mathbb{Z}$  is an element of  $\mathbb{R}$ . Similarly, every vowel is a letter in the set of English letters. In other words, if  $A = \{a, e, i, o, u\}$  and  $B$  is the set of all English letters, then every element of  $A$  is an element of  $B$ . When every element of a set, say  $A$ , is also an element of a set, say  $B$ , we say that  $A$  is a subset of  $B$ . More formally, we have the following definition.

---

**DEFINITION 1.1.6** ▶ Let  $X$  and  $Y$  be sets. Then  $X$  is said to be a **subset** of  $Y$ , written  $X \subseteq Y$ , if every element of  $X$  is an element of  $Y$ . If  $X$  is not a subset of  $Y$ , then we write  $X \not\subseteq Y$ .

#### EXAMPLE 1.1.7

- (i) Let  $X = \{0, 2, 4, 6, 8\}$ ,  $Y = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ , and  $Z = \{1, 2, 3, 4, 5\}$ . Then  $X \subseteq Y$  because every element of  $X$  is an element of  $Y$ . However, because  $0 \in X$  and  $0 \notin Z$ , we have  $X \not\subseteq Z$ .

Notice that we used the fact that  $0 \in X$  and  $0 \notin Z$  to conclude that  $X \not\subseteq Z$ . We could have also used the fact that  $6 \in X$  and  $6 \notin Z$  or  $8 \in X$  and  $8 \notin Z$  to conclude that  $X \not\subseteq Z$ . In other words, the elements 6 and 8 also prevent  $X$  from being a subset of  $Z$ .

- (ii) Let  $A = \{a, b, c\}$  and  $B = \{a, c, b\}$ . Now every element of  $A$  is also an element of  $B$  and so  $A \subseteq B$ . Also notice that  $B \subseteq A$ .  
 (iii) Let  $A = \{\text{Basic, Fortran, C++}\}$  and  $B = \{\text{Basic, Fortran, Pascal, C++, Java}\}$ . Then  $A \subseteq B$ .

**Note:** For every set  $X$ , we have  $X \subseteq X$ .

Let  $X$  and  $Y$  be sets. If  $X \subseteq Y$ , we also say that  $X$  is contained in  $Y$ , or  $Y$  contains  $X$ , or  $Y$  is a **superset** of  $X$  (written  $Y \supseteq X$ ).

Notice that in Example 1.1.7(i), every element of  $X$  is an element of  $Y$ . However, there are some elements in  $Y$  that are not in  $X$ . Such a set  $X$  is called a **proper subset** of  $Y$ .

---

**DEFINITION 1.1.8** ▶ Let  $X$  and  $Y$  be sets. Then  $X$  is a **proper subset** of  $Y$ , written  $X \subset Y$ , if  $X$  is a subset of  $Y$  and there exists at least one element in  $Y$  that is not in  $X$ .

#### EXAMPLE 1.1.9

Let  $A = \{a, b\}$  and  $B = \{a, b, c\}$ . Because every element of  $A$  is an element of  $B$ , we have  $A \subseteq B$ . Now  $c \in B$  and  $c \notin A$ . Therefore, there exists an element in  $B$  that is not in  $A$ . It now follows that  $A$  is a proper subset of  $B$ , i.e.,  $A \subset B$ .

#### EXAMPLE 1.1.10

The set of all even integers is a proper subset of the set of all integers. In set notation,  $\{2n \mid n \in \mathbb{Z}\} \subset \mathbb{Z}$ .

Note that  $\mathbb{N} \subset \mathbb{Z} \subset \mathbb{Q} \subset \mathbb{R} \subset \mathbb{C}$ .

**DEFINITION 1.1.11** ▶ Two sets  $X$  and  $Y$  are said to be **equal**, written  $X = Y$ , if every element of  $X$  is an element of  $Y$  and every element of  $Y$  is an element of  $X$ , i.e., if  $X \subseteq Y$  and  $Y \subseteq X$ .

**EXAMPLE 1.1.12**

- (i)  $\{a, b, c\} = \{a, c, b\}$ .
- (ii) Let  $A = \{1, 2, 3, 4\}$  and  $B = \{x \mid x \text{ is a positive integer and } x^2 < 18\}$ . Then  $A = B$ .
- (iii) The set  $A = \{x \mid x \text{ is an integer and } x^3 = 1\}$  and  $B = \{1\}$  are equal.

Now consider the set  $A := \{x \in \mathbb{Z} \mid x^2 - 2 = 0\}$ . Notice that  $x^2 - 2 = (x - \sqrt{2})(x + \sqrt{2})$ . Thus, the solutions of the equation  $x^2 - 2 = 0$  are  $\sqrt{2}$  and  $-\sqrt{2}$  and none of these is an integer. Therefore, it follows that  $A$  does not contain any elements. This is an empty collection of objects. We call it an empty set.

**DEFINITION 1.1.13** ▶ A set is said to be an **empty** (or **null**) **set** if it has no elements. We denote an empty set by the symbol  $\emptyset$ .

**Note:** We can consider the empty set a subset of every set. In fact, if  $\emptyset \not\subseteq A$  for some set  $A$ , then there exists an element  $x \in \emptyset$  such that  $x \notin A$ . However, there is no such element  $x$  because  $\emptyset$  is empty. Hence,  $\emptyset \subseteq A$  for every set  $A$ .

In Example 1.1.1, the set  $A$  has five elements.

**DEFINITION 1.1.14** ▶ Let  $X$  be a set.

- (i) If there exists a nonnegative integer  $n$  such that  $X$  has  $n$  elements, then  $X$  is called a **finite set** with  $n$  elements.
- (ii)  $X$  is called an **infinite set**, if  $X$  is not a finite set.

**EXAMPLE 1.1.15**

- (i) The set  $A = \{a, b, c\}$  has three elements, so it is a finite set.
- (ii) The set  $B$  of the first 10 positive odd integers is a finite set. Notice that

$$B := \{1, 3, 5, 7, 9, 11, 13, 15, 17, 19\}.$$

- (iii) The set of positive integers is an infinite set.

**REMARK 1.1.16** ▶ Note that an empty set is a finite set with 0 elements.

Let  $S$  be a finite set with  $n$  distinct elements, where  $n \geq 0$ . Then we write  $|S| = n$  and say that the *cardinality* (or the *number of elements*) of  $S$  is  $n$ .

If  $A = \{a, b, c, d, e\}$ , then  $A$  is a finite set with five elements and so  $|A| = 5$ .

Let

$$B = \{x \mid x \text{ is a positive even prime integer}\}.$$

The only positive even prime integer is 2. Therefore,  $B = \{2\}$  and so  $|B| = 1$ .

A set consisting of only one element is called a **singleton set** or, simply, a **singleton**. Thus,  $B$  is a singleton.

Sometimes an infinite set is described using the roster method. For example, if  $\mathbb{N}$  denotes the set of positive integers, then we can write  $\mathbb{N} = \{1, 2, 3, \dots\}$ .

Let  $X = \{1, 2\}$ . Then  $\emptyset \subseteq X$ ,  $\{1\} \subseteq X$ ,  $\{2\} \subseteq X$ , and  $X \subseteq X$ . Now each of the sets  $\emptyset$ ,  $\{1\}$ ,  $\{2\}$ ,  $X$  is well defined. We can therefore form the collection  $\{\emptyset, \{1\}, \{2\}, X\}$  of these sets, which would itself be a set.

**DEFINITION 1.1.17** ► For any set  $X$ , the **power set** of  $X$ , written  $\mathcal{P}(X)$ , is the set of all subsets of  $X$ . That is,

$$\mathcal{P}(X) = \{A \mid A \subseteq X\}.$$

For example, let  $X = \{a, b, c\}$ . Then

$$\mathcal{P}(X) = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, X\}.$$

Notice that  $|X| = 3$  and  $|\mathcal{P}(X)| = 8 = 2^3$ .

**REMARK 1.1.18** ► Let  $X$  be a finite set  $X$  such that  $|X| = k$ . In Chapter 2, we will show that  $|\mathcal{P}(X)| = 2^k$ .

**REMARK 1.1.19** ► Because  $\emptyset$  is a subset of every set, we have  $\emptyset \subseteq \emptyset$ . Therefore, we have  $\mathcal{P}(\emptyset) = \{\emptyset\}$ .

To avoid the logical difficulties that arise in the foundation of set theory, we further assume that each discussion involving a number of sets takes place with respect to an *arbitrarily chosen but fixed* set. This set is called a **universal set**<sup>1</sup> for that discussion and is generally denoted by  $U$ . All the sets under consideration in the problem must be subsets of  $U$ .

For example, in a discussion involving the sets  $X = \{1, 2, 3\}$ ,  $Y = \{2, 4, 6, 8\}$ ,  $Z = \{1, 3, 5, 7\}$ , one may choose  $U = \{x \in \mathbb{N} \mid 1 \leq x \leq 8\}$  as a universal set. Moreover, any superset of  $U$  can also be considered a universal set for these sets  $X$ ,  $Y$ , and  $Z$ .

## Venn Diagrams

Typically, it is not easy to visualize a set. In 1880, however, the English logician John Venn devised a pictorial representation for sets and their fundamental operations. Though admittedly loose and imprecise, and therefore somewhat contrary to the spirit of logical rigor at the heart of set theory, one may still find this diagrammatic approach very convenient in developing the so-called *abstract visualization*, which is essential to *seeing* the mental image of these abstract happenings. In these



**John Venn**  
(1834–1923)

He studied at Caius College at Cambridge University, and in 1857 earned a B.A. degree. He was also chosen as a Fellow to the college, where he later lectured in the moral sciences in a po-

### HISTORICAL NOTES

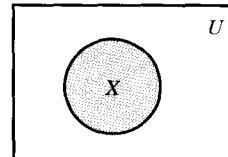
sition that he held for the remainder of his life. Also very important in Venn's life was religion; he was ordained first as deacon and then as priest, but given the climate of the times, he renounced his clerical orders in 1883.

Venn is best known for his diagrams that demonstrate sets and their unions, which he first introduced in a paper titled "On the Diagrammatic and

Mechanical Representation of Propositions and Reasoning." Chiefly interested in the workings of logic, Venn wrote three books on the subject, in which he explored the mathematical logic of Boole as well as symbolic logic. Venn also wrote extensively on the history of Cambridge University, a work that is still being continued today.

<sup>1</sup>We want to make it very clear that in spite of what the name may seem to suggest, we are by no means proposing a set that is *universal* for all the problems. Rather, it may vary from problem to problem and even more: For a problem involving certain sets, the choice of a universal set is not unique, but once chosen, subject to the conditions stated above, it must be kept fixed throughout the subsequent discussions of that problem.

representations, called **Venn diagrams**, the universal set  $U$  is shown as a rectangle and all of its subsets are shown by circles drawn within the rectangle. (See Figure 1.1.) The shaded portion in a Venn diagram represents the corresponding set.



**FIGURE 1.1** Set  $X$

## Operations on Sets

Given two sets  $A$  and  $B$ , there are various ways we can form new sets using the sets  $A$  and  $B$ . For example, we can form a set by taking all elements from  $A$  and all elements from  $B$  (in this case, an element will not be considered more than once). We can also form a set by taking elements that are common to both sets  $A$  and  $B$ . Similarly, we can form a set by taking all the elements of  $A$  that are not in  $B$ .

Next, we consider these and other ways to form new sets from existing sets by means of the so-called *algebraic operations on sets*.

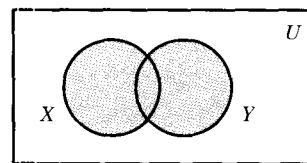
---

**DEFINITION 1.1.20** ► The **union** of two sets  $X$  and  $Y$ , denoted by  $X \cup Y$ , is defined to be the set

$$X \cup Y = \{x \mid x \in X \text{ or } x \in Y\}.$$

We would like to point out that in Definition 1.1.20,  $x \in X$  or  $x \in Y$  means that  $x$  is an element of  $X$ , or  $x$  is an element of  $Y$ , or  $x$  is an element of both  $X$  and  $Y$ . In other words,  $x \in X \cup Y$  if  $x$  is a member of at least one of the sets  $X$  and  $Y$ .

The Venn diagram of the union of sets is shown in Figure 1.2.



**FIGURE 1.2** Venn diagram of  $X \cup Y$

**EXAMPLE 1.1.21**

- (i) Let  $X = \{a, b, c, d, e\}$ ,  $Y = \{c, d, e, f, g, h\}$ , and  $Z = \{h, p, q, r\}$ . Then

$$X \cup Y = \{a, b, c, d, e, f, g, h\},$$

$$X \cup Z = \{a, b, c, d, e, h, p, q, r\},$$

and

$$Y \cup Z = \{c, d, e, f, g, h, p, q, r\}.$$

- (ii) Let  $A$  be the set of nonpositive integers; i.e.,  $A = \{x \in \mathbb{Z} \mid x \leq 0\}$ . Then

$$A \cup \mathbb{N} = \mathbb{Z}.$$

From Definition 1.1.20, it follows that every element of  $X$  must be an element of  $X \cup Y$ , so we have

$$X \subseteq X \cup Y.$$

Similarly, because every element of  $Y$  is an element of  $X \cup Y$ , we have  $Y \subseteq X \cup Y$ . We have thus proved the following theorem.

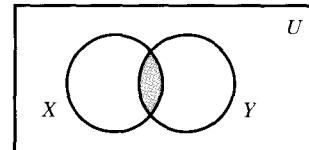
**Theorem 1.1.22:** Let  $X$  and  $Y$  be sets. Then  $X \subseteq X \cup Y$  and  $Y \subseteq X \cup Y$ .

**REMARK 1.1.23** ▶ Notice that the preceding statement is listed as a theorem. Later, in Section 1.2, Mathematical Logic, we will formally define the term theorem. Until then, by a theorem we mean a statement that can be proved to be true. In this chapter, we will use some commonly known techniques to prove theorems. In Section 1.5, Proof Techniques, we will present various proof techniques.

**DEFINITION 1.1.24** ▶ The **intersection** of two sets  $X$  and  $Y$ , denoted by  $X \cap Y$ , is defined to be the set

$$X \cap Y = \{x \mid x \in X \text{ and } x \in Y\}.$$

The Venn diagram of the intersection of sets is shown in Figure 1.3.



**FIGURE 1.3** Venn diagram of  $X \cap Y$

**EXAMPLE 1.1.25**

Let  $X = \{a, b, c, d, e\}$ ,  $Y = \{c, d, e, f, g, h\}$ , and  $Z = \{h, p, q, r\}$ . Then

$$\begin{aligned} X \cap Y &= \{c, d, e\}, \\ X \cap Z &= \emptyset, \end{aligned}$$

and

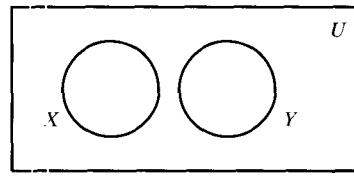
$$Y \cap Z = \{h\}.$$

From Definition 1.1.24, it follows that the set  $X \cap Y$  consists of the elements that are common to both the sets  $X$  and  $Y$ ; i.e., every element of  $X \cap Y$  must be an element of  $X$  as well as of  $Y$ . Because every element of  $X \cap Y$  is an element of  $X$ , we have  $X \cap Y \subseteq X$ . In a similar manner, we have  $X \cap Y \subseteq Y$ . We have thus proved the following theorem.

**Theorem 1.1.26:** Let  $X$  and  $Y$  be sets. Then  $X \cap Y \subseteq X$  and  $X \cap Y \subseteq Y$ .

**DEFINITION 1.1.27** ▶ Two sets  $X$  and  $Y$  are said to be **disjoint** if  $X \cap Y = \emptyset$ .

It is not possible to draw the Venn diagram of the null set by shading; however, disjoint sets are represented as in Figure 1.4.

FIGURE 1.4  $X \cap Y = \emptyset$ 

In Example 1.1.25, the sets  $X$  and  $Z$  are disjoint sets.

Theorem 1.1.29 enlists some fundamental properties of union and intersection of sets.

**REMARK 1.1.28 ▶**

- (i) We will prove some of the properties listed in Theorem 1.1.29. Moreover, we will use these properties to prove that two sets are equal. To prove two sets, say  $A$  and  $B$ , are equal, we typically prove that  $A \subseteq B$  and  $B \subseteq A$ . We then use the definition of the equality of sets. Now, to prove say  $A \subseteq B$ , we typically show that every element of  $A$  is also an element of  $B$ . For this, we choose an element, say  $x \in A$ , and show that  $x \in B$ .
- (ii) Sometimes we write statements such as  $P$  if and only if  $Q$ , where  $P$  and  $Q$  are statements. For example, let  $A$  and  $B$  be sets. Then  $A = B$  if and only if  $A \subseteq B$  and  $B \subseteq A$ . Here, we can think of  $P$  as  $A = B$  and  $Q$  as  $A \subseteq B$  and  $B \subseteq A$ . What we are saying is that if  $P$  is true, then  $Q$  is true and if  $Q$  is true, then  $P$  is true.

**Theorem 1.1.29:** Let  $X, Y, Z$  be subsets of a set  $U$ . Then the following assertions hold.

- (i) If  $X \subseteq Y$ , then

$$X \cup Y = Y, \quad X \cap Y = X.$$

- (ii) Laws of identity:

$$X \cup \emptyset = X, \quad X \cap \emptyset = \emptyset.$$

- (iii) Laws of idempotency:

$$X \cup X = X, \quad X \cap X = X.$$

- (iv) Laws of commutativity:

$$X \cup Y = Y \cup X, \quad X \cap Y = Y \cap X.$$

- (v) Laws of associativity:

$$(X \cup Y) \cup Z = X \cup (Y \cup Z),$$

$$(X \cap Y) \cap Z = X \cap (Y \cap Z).$$

- (vi) Laws of distributivity:

$$X \cup (Y \cap Z) = (X \cup Y) \cap (X \cup Z),$$

$$X \cap (Y \cup Z) = (X \cap Y) \cup (X \cap Z).$$

- (vii) Laws of absorptivity:

$$X \cap (X \cup Y) = X, \quad X \cup (X \cap Y) = X.$$

**Proof:** We prove the second equality of part (v), the second equality of part (vi), and the first equality of part (vii) and leave the rest as exercises (see Exercise 18, p. 25).

(v) We show that  $(X \cap Y) \cap Z = X \cap (Y \cap Z)$ .

From the definition of the equality of sets (Definition 1.1.11), we know that two sets  $A$  and  $B$  are equal if and only if every element of  $A$  is an element of  $B$  and every element of  $B$  is an element of  $A$ . That is,  $A = B$ , if and only if  $A \subseteq B$  and  $B \subseteq A$ . Therefore, to prove  $(X \cap Y) \cap Z = X \cap (Y \cap Z)$ , we prove that  $(X \cap Y) \cap Z \subseteq X \cap (Y \cap Z)$  and  $X \cap (Y \cap Z) \subseteq (X \cap Y) \cap Z$ .

Let us choose any element  $x \in (X \cap Y) \cap Z$ . Now,

$$\begin{aligned} x &\in (X \cap Y) \cap Z \\ \Rightarrow x &\in X \cap Y \quad \text{and} \quad x \in Z \\ \Rightarrow (x &\in X \text{ and } x \in Y) \quad \text{and} \quad x \in Z \\ \Rightarrow x &\in X \quad \text{and} \quad (x \in Y \text{ and } x \in Z) \\ \Rightarrow x &\in X \quad \text{and} \quad x \in Y \cap Z \\ \Rightarrow x &\in X \cap (Y \cap Z). \end{aligned}$$

(Here the symbol  $\Rightarrow$  means this implies that or this implies.) Hence,

$$(X \cap Y) \cap Z \subseteq X \cap (Y \cap Z) \tag{1.1}$$

Now let us choose any element  $x \in X \cap (Y \cap Z)$ . Then

$$\begin{aligned} x &\in X \cap (Y \cap Z) \\ \Rightarrow x &\in X \quad \text{and} \quad x \in Y \cap Z \\ \Rightarrow x &\in X \quad \text{and} \quad (x \in Y \text{ and } x \in Z) \\ \Rightarrow (x &\in X \text{ and } x \in Y) \quad \text{and} \quad x \in Z \\ \Rightarrow x &\in X \cap Y \quad \text{and} \quad x \in Z \\ \Rightarrow x &\in (X \cap Y) \cap Z. \end{aligned}$$

Hence,

$$X \cap (Y \cap Z) \subseteq (X \cap Y) \cap Z \tag{1.2}$$

Combining (1.1) and (1.2) we get,

$$X \cap (Y \cap Z) = (X \cap Y) \cap Z.$$

(vi) We show that  $X \cap (Y \cup Z) = (X \cap Y) \cup (X \cap Z)$ .

As in part (v), first we show that  $X \cap (Y \cup Z) \subseteq (X \cap Y) \cup (X \cap Z)$  and then that  $(X \cap Y) \cup (X \cap Z) \subseteq X \cap (Y \cup Z)$ . The result then follows by the definition of the equality of sets.

Let  $x$  be any element of  $X \cap (Y \cup Z)$ . We have

$$\begin{aligned} x &\in X \cap (Y \cup Z) \\ \Rightarrow x &\in X \quad \text{and} \quad x \in Y \cup Z \\ \Rightarrow x &\in X \quad \text{and} \quad (x \in Y \text{ or } x \in Z). \end{aligned}$$

If  $x \in X$  and  $x \in Y$ , then  $x \in X \cap Y$ . Similarly, if  $x \in X$  and  $x \in Z$ , then  $x \in X \cap Z$ . Therefore, it follows that

$$x \in X \cap Y \quad \text{or} \quad x \in X \cap Z.$$

Hence,  $x \in (X \cap Y) \cup (X \cap Z)$ . This shows that

$$X \cap (Y \cup Z) \subseteq (X \cap Y) \cup (X \cap Z).$$

Let us now show that  $(X \cap Y) \cup (X \cap Z) \subseteq X \cap (Y \cup Z)$ .

Suppose  $x$  is any element of  $(X \cap Y) \cup (X \cap Z)$ . Then

$$x \in X \cap Y \quad \text{or} \quad x \in X \cap Z.$$

Suppose  $x \in X \cap Y$ . Then  $x \in X$  and  $x \in Y$ . Since  $Y \subseteq Y \cup Z$  and  $x \in Y$ , we have  $x \in Y \cup Z$ . Thus,  $x \in X$  and  $x \in Y \cup Z$  and so  $x \in X \cap (Y \cup Z)$ .

Similarly, if  $x \in X$  and  $x \in Z$ , then  $x \in X \cap (Y \cup Z)$ . Hence,

$$(X \cap Y) \cup (X \cap Z) \subseteq X \cap (Y \cup Z).$$

Consequently,  $X \cap (Y \cup Z) = (X \cap Y) \cup (X \cap Z)$ .

(vii) We want to show that  $X \cap (X \cup Y) = X$ . As before, we show that  $X \cap (X \cup Y) \subseteq X$  and  $X \subseteq X \cap (X \cup Y)$ .

Let  $x \in X \cap (X \cup Y)$ . Then

$$x \in X \quad \text{and} \quad x \in X \cup Y.$$

This implies that  $x \in X$ . Hence,  $X \cap (X \cup Y) \subseteq X$ .

On the other hand, let  $x \in X$ . Because  $X \subseteq X \cup Y$ ,  $x \in X \cup Y$ . We therefore have  $x \in X$  and  $x \in X \cup Y$ , so  $x \in X \cap (X \cup Y)$ . Thus,  $X \subseteq X \cap (X \cup Y)$ . Consequently,

$$X \cap (X \cup Y) = X. \blacksquare$$

**REMARK 1.1.30** ▶ In the proof of Theorem 1.1.29(vi), we gave a direct proof to show that  $X \cap (X \cup Y) = X$ . However, this result can also be proved as follows: Let  $A = X \cup Y$ . By Theorem 1.1.22,  $X \subseteq X \cup Y := A$ , and so by Theorem 1.1.29(i), we have  $X \cap A = X$ , i.e.,  $X \cap (X \cup Y) = X$ .

Figure 1.5 shows the Venn diagrams of the union and intersection of three sets.

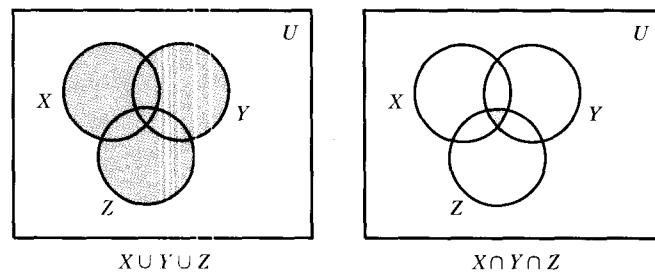


FIGURE 1.5 Various Venn diagrams involving three sets

Until now we have considered the union and intersection of only two sets. In fact, we can consider the union and intersection of three, four, or even infinitely many sets. That is, we can generalize the notions of union and intersection from two sets to an arbitrary collection of sets.

To begin with, let us consider a finite collection of  $n$  sets, say  $X_1, X_2, \dots, X_n$ ,  $n \geq 2$ . In this case, we write<sup>2</sup>

$$\bigcup_{i=1}^n X_i = X_1 \cup X_2 \cup \dots \cup X_n = \{x \mid x \in X_i \text{ for some } i, 1 \leq i \leq n\}$$

and

$$\bigcap_{i=1}^n X_i = X_1 \cap X_2 \cap \dots \cap X_n = \{x \mid x \in X_i \text{ for all } i, 1 \leq i \leq n\}.$$

To generalize it further to any arbitrary family of sets, finite or infinite, we introduce the notion of an index set.

---

**DEFINITION 1.1.31** ► A set  $I$  is said to be an **index set** for a family  $\mathcal{A}$  of sets, if for any  $\alpha \in I$ , there exists a set  $A_\alpha \in \mathcal{A}$  and  $\mathcal{A} = \{A_\alpha \mid \alpha \in I\}$ .

Note that in Definition 1.1.31,  $I$  can be any nonempty set, finite or infinite. We now define the union or intersection of the sets  $A_\alpha, \alpha \in I$ , as follows:

$$\bigcup_{\alpha \in I} A_\alpha = \{x \mid x \in A_\alpha \text{ for at least one } \alpha \in I\},$$

$$\bigcap_{\alpha \in I} A_\alpha = \{x \mid x \in A_\alpha \text{ for all } \alpha \in I\}.^3$$

Notice that  $\bigcup_{\alpha \in I} A_\alpha$  contains all those elements  $x$  such that  $x$  is an element of at least one  $A_\alpha$ . Similarly,  $\bigcap_{\alpha \in I} A_\alpha$  contains all those elements  $x$  such that  $x$  is an element of every  $A_\alpha$ .

Let  $\mathcal{A} = \{A_\alpha \mid \alpha \in I\}$  be a family of sets. The sets  $A_\alpha$  are said to be **mutually disjoint**, or **pairwise disjoint**, if for  $\alpha, \beta \in I, \alpha \neq \beta$  implies  $A_\alpha \cap A_\beta = \emptyset$ .

### EXAMPLE 1.1.32

Let  $A_1 = \{a, b, c\}$ ,  $A_2 = \{d, e\}$ , and  $A_3 = \{u, v, w\}$ . Then  $A_1 \cap A_2 = \emptyset$ ,  $A_2 \cap A_3 = \emptyset$ , and  $A_1 \cap A_3 = \emptyset$ . This implies that the sets  $A_1$ ,  $A_2$ , and  $A_3$  are pairwise disjoint.

### EXAMPLE 1.1.33

Let  $n \in \mathbb{N}$ . Consider the set

$$I_n = \left\{ x \in \mathbb{R} \mid -\frac{1}{n} < x < \frac{1}{n} \right\}.$$

For example,

$$\begin{aligned} I_1 &= \left\{ x \in \mathbb{R} \mid -\frac{1}{1} < x < \frac{1}{1} \right\} \\ &= \{x \in \mathbb{R} \mid -1 < x < 1\}, \end{aligned}$$

<sup>2</sup>It is important to understand that Theorem 1.1.29(v) plays a very important role in enabling us to extend the scope of the definition of union and intersection beyond three sets. Indeed, in both of these definitions two sets were combined to yield a third set, and it is the associativity of both of these operations that allows us to dispense with the parentheses, as by virtue of associativity, we come to the conclusion that the order in which the operations are made is of no significance.

<sup>3</sup>While the index set  $I$  needs to be nonempty, such a restriction is not required for the family  $\mathcal{A}$ . In fact if  $\mathcal{A}$  is an empty family (i.e., there is no member of this family), then the above definitions of union and intersection give (keeping in mind that all the sets under discussion are subsets of a universal set  $U$ )  $\bigcup_{\alpha \in I} A_\alpha = \emptyset$  and  $\bigcap_{\alpha \in I} A_\alpha = U$ . While the first equality is quite obvious, the second one may be puzzling. Observe that, to be in  $\bigcap_{\alpha \in I} A_\alpha$ , an element is required to belong to each member  $A_\alpha$  of the family  $\mathcal{A}$ , and if there does not exist any such  $A_\alpha$  in  $\mathcal{A}$  (as  $\mathcal{A}$  is empty), then every element in  $U$  satisfies this requirement *vacuously*.

$$I_2 = \left\{ x \in \mathbb{R} \mid -\frac{1}{2} < x < \frac{1}{2} \right\},$$

$$I_3 = \left\{ x \in \mathbb{R} \mid -\frac{1}{3} < x < \frac{1}{3} \right\},$$

and so on. Let us consider the family  $\mathcal{F} = \{I_n \mid n \in \mathbb{N}\}$ . Notice that here  $\mathbb{N}$  is the index set. Moreover, observe that

$$I_1 \supset I_2 \supset I_3 \supset \dots$$

From this it follows that

$$\bigcup_{n \in \mathbb{N}} I_n = \{x \in \mathbb{R} \mid -1 < x < 1\}.$$

Now what about  $\cap_{n \in \mathbb{N}} I_n$ ? We claim that  $\cap_{n \in \mathbb{N}} I_n = \{0\}$ . First note that for each  $n$ ,  $-\frac{1}{n} < 0 < \frac{1}{n}$ , so  $0 \in I_n$  for all  $n \in \mathbb{N}$ . This implies that  $0 \in \cap_{n \in \mathbb{N}} I_n$ . Next, let  $x \in \cap_{n \in \mathbb{N}} I_n$  and suppose that  $x \neq 0$ . Now  $x$  is such that  $x \in I_n$  for all  $n \in \mathbb{N}$ ,  $-1 < x < 1$  and  $x \neq 0$ . Let us assume that  $0 < x < 1$ . Then  $x = \frac{a}{b}$  for some positive integers  $a$  and  $b$  such that  $a < b$ . It follows that  $x = \frac{a}{b} \geq \frac{1}{b}$ . This implies that  $x \not< \frac{1}{b}$  so  $x \notin I_b$ , which is a contradiction to the fact that  $x \in I_n$  for all  $n \in \mathbb{N}$ . In a similar manner, if  $-1 < x < 0$ , we can get a contradiction. Thus, our assumption that  $x \neq 0$  is incorrect, so we must have  $x = 0$ . It now follows that

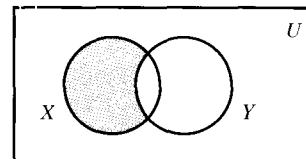
$$\bigcap_{n \in \mathbb{N}} I_n = \{0\}.$$

---

**DEFINITION 1.1.34** ▶ Let  $X$  and  $Y$  be sets. The **difference** of  $X$  and  $Y$  (or the **relative complement** of  $Y$  in  $X$ ), written  $X - Y$ , is the set

$$X - Y = \{x \mid x \in X \text{ but } x \notin Y\}.$$

The Venn diagram of the difference of sets is shown in Figure 1.6.



**FIGURE 1.6** Venn diagram of  $X - Y$

**EXAMPLE 1.1.35**

Let  $X = \{1, 2, 3, 4\}$  and  $Y = \{3, 4, 5, 6\}$ . Then

$$X - Y = \{1, 2\},$$

and

$$Y - X = \{5, 6\}.$$

Notice that  $X - Y \neq Y - X$ ; i.e., the difference of sets is noncommutative.

**Note:** For any sets  $X$  and  $Y$ , it can be shown that  $X - X = \emptyset$ ,  $X - \emptyset = X$ ,  $\emptyset - X = \emptyset$ , and

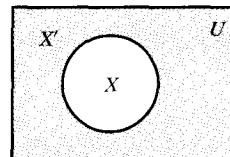
$$X \cup Y = (X - Y) \cup (Y - X) \cup (X \cap Y).$$

**DEFINITION 1.1.36** ► The **complement** of a set  $X$  with respect to a universal set  $U$ , denoted by  $X'$  is defined to be

$$X' = \{x \in U \mid x \notin X\}.$$

From Definition 1.1.36, it follows that  $X' = U - X$ .

The Venn diagram of the complement of a set is shown in Figure 1.7.

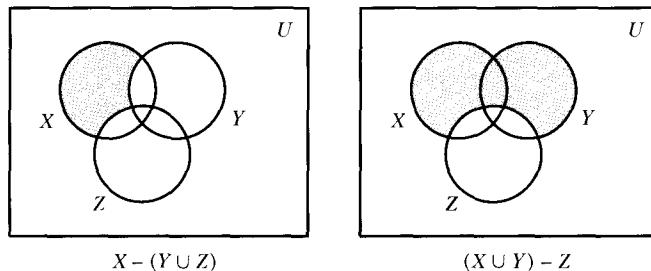


**FIGURE 1.7** Venn diagram of  $X'$

**EXAMPLE 1.1.37**

- (i) Let  $U = \{a, b, c, d, e, f, g, h\}$  and  $A = \{a, c, g, h\}$ . Then  $A' = \{b, d, e, f\}$ .
- (ii)  $\mathbb{E}' = \mathbb{Z} - \mathbb{E} = \{x \in \mathbb{Z} \mid x \text{ is an odd integer}\}$ .

Figure 1.8 shows the Venn diagrams of the sets  $X - (Y \cup Z)$  and  $(X \cup Y) - Z$ .



**FIGURE 1.8** Venn diagrams of the sets  $X - (Y \cup Z)$  and  $(X \cup Y) - Z$

In the next theorem, we enlist some important properties of the complement of a set.

**Theorem 1.1.38:** Let  $X$  and  $Y$  be sets and let  $U$  be a universal set under consideration. Then

- (i)  $X \cup X' = U$  and  $X \cap X' = \emptyset$ .
- (ii)  $(X')' = X$ .
- (iii)  $X - Y = X \cap Y'$ .
- (iv) (DeMorgan's laws)  $(X \cup Y)' = X' \cap Y'$  and  $(X \cap Y)' = X' \cup Y'$ .

**Proof:** We prove only the second equality of part (iv) and leave the others as exercises (see Exercise 32, p. 26).

To prove  $(X \cap Y)' = X' \cup Y'$ , we show that  $(X \cap Y)' \subseteq X' \cup Y'$  and  $X' \cup Y' \subseteq (X \cap Y)'$ . The result then will follow from the equality of sets.

Let  $x$  be any element of  $(X \cap Y)'$ . Now

$$\begin{aligned} x &\in (X \cap Y)' \\ \Rightarrow x &\in U \text{ and } x \notin X \cap Y \\ \Rightarrow x &\in U \text{ and } (x \notin X \text{ or } x \notin Y) \\ \Rightarrow (x &\in U \text{ and } x \notin X) \text{ or } (x \in U \text{ and } x \notin Y) \\ \Rightarrow x &\in X' \text{ or } x \in Y' \\ \Rightarrow x &\in X' \cup Y'. \end{aligned}$$

We therefore have

$$(X \cap Y)' \subseteq X' \cup Y'. \quad (1.3)$$

In a similar manner, we can show that

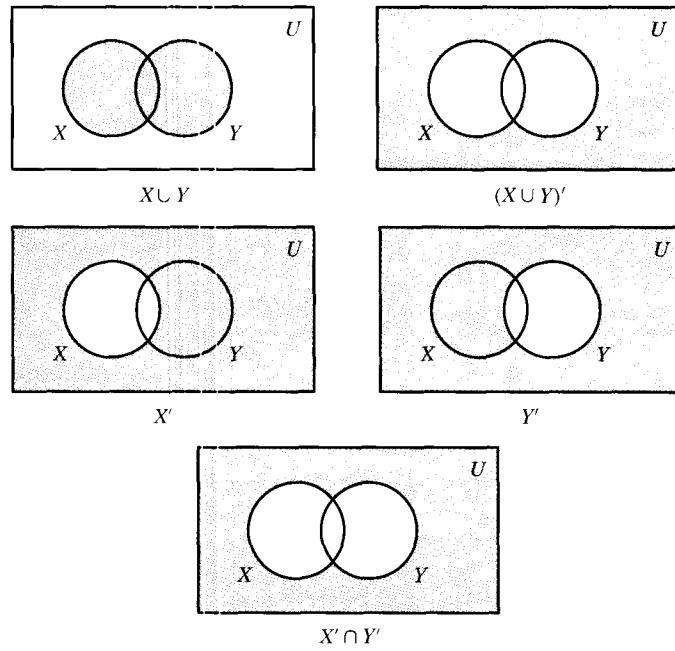
$$X' \cup Y' \subseteq (X \cap Y)'. \quad (1.4)$$

By the definition of the equality of sets, we can conclude that

$$(X \cap Y)' = X' \cup Y'. \blacksquare$$

### EXAMPLE 1.1.39

Using Venn diagrams (see Figure 1.9), let us verify the DeMorgan's law  $(X \cup Y)' = X' \cap Y'$ , listed in Theorem 1.1.38(iv).



**FIGURE 1.9** DeMorgan's law  $(X \cup Y)' = X' \cap Y'$

Notice that the last diagram in the first row and the last diagram in the third row are the same. Thus,  $(X \cup Y)' = X' \cap Y'$ .

**DEFINITION 1.1.40** ► The **symmetric difference** of two sets  $X$  and  $Y$ , denoted by  $X \Delta Y$ , is defined to be

$$X \Delta Y = (X - Y) \cup (Y - X).$$

Hence,  $X \Delta Y$  is the set of those elements that are either only in  $X$  or only in  $Y$ , but not in both.

**Note:** Let  $X$  and  $Y$  be sets. From the definition of the symmetric difference, it follows that the symmetric difference of two sets is commutative, i.e.,

$$X \Delta Y = Y \Delta X.$$

The symmetric difference is also associative; i.e., for sets  $X$ ,  $Y$ , and  $Z$ ,

$$X \Delta (Y \Delta Z) = (X \Delta Y) \Delta Z.$$

## Ordered Pair and Cartesian Cross Product

Let us now define a concept that is of fundamental importance in the development of our subject.

Let  $X$  and  $Y$  be sets. Intuitively, an **ordered pair** of the elements  $x \in X$  and  $y \in Y$ , written  $(x, y)$ , is a listing of the elements  $x$  and  $y$  in a specific order. The ordered pair  $(x, y)$  specifies that  $x$  is the first element and  $y$  is the second element. Moreover, we use the convention that  $(x, y) = (z, w)$  if and only if  $x = z$  and  $y = w$ , for all  $x, z \in X$  and  $y, w \in Y$ .

---

**REMARK 1.1.41** ▶ Let  $X$  and  $Y$  be two nonempty sets and  $x \in X, y \in Y$ . We would like to point out that the set-theoretic definition of the ordered pair  $(x, y)$  is the set  $\{\{x\}, \{x, y\}\}$ . That is,  $(x, y) = \{\{x\}, \{x, y\}\}$ . Using this definition we can, in fact, prove that  $(x, y) = (z, w)$  if and only if  $x = z$  and  $y = w$ , for all  $x, z \in X$  and  $y, w \in Y$ .

---

**DEFINITION 1.1.42** ▶ The **Cartesian product** of two sets  $X$  and  $Y$ , written  $X \times Y$ , is the set

$$X \times Y = \{(x, y) \mid x \in X, y \in Y\}.$$

For any set  $X$ ,  $X \times \emptyset = \emptyset = \emptyset \times X$ .

### EXAMPLE 1.1.43

Let  $X = \{1, 2\}$ ,  $Y = \{3, 4\}$ . Then

$$X \times Y = \{(1, 3), (1, 4), (2, 3), (2, 4)\}$$

and

$$Y \times X = \{(3, 1), (3, 2), (4, 1), (4, 2)\}.$$

### EXAMPLE 1.1.44

For the set  $\mathbb{R}$  of real numbers, the Cartesian product  $\mathbb{R} \times \mathbb{R}$  is the Euclidean plane, where the coordinate of each point is an ordered pair of real numbers.

Note that if  $X \neq Y$ , then  $X \times Y \neq Y \times X$ . For example, in Example 1.1.43, we have  $X \neq Y$  and  $X \times Y \neq Y \times X$ .

Suppose that  $X = \{a, b, c\}$  and  $Y = \{1, 2\}$ . Then

$$X \times Y = \{(a, 1), (a, 2), (b, 1), (b, 2), (c, 1), (c, 2)\}.$$

Thus, we have  $|X| = 3$ ,  $|Y| = 2$ , and  $|X \times Y| = 3 \cdot 2 = 6$ .

In fact, it is generally true that if  $|X| = m$  and  $|Y| = n$ , then  $|X \times Y| = mn$ . Let us prove this fact.

Let  $X = \{x_1, x_2, \dots, x_m\}$  and  $Y = \{y_1, y_2, \dots, y_n\}$ . Then

$$X \times Y = \{(x_i, y_j) \mid i = 1, 2, \dots, m, j = 1, 2, \dots, n\}.$$

Next, we show that no two elements of  $X \times Y$  are the same. Suppose that  $(x_i, y_j) = (x_k, y_l)$ . Then by the definition of the ordered pair, we must have  $x_i = x_k$  and  $y_j = y_l$ . Because the  $x_i$ 's are distinct and the  $y_j$ 's are distinct, we must have,  $i = k$  and  $j = l$ . It now follows that no two elements of  $X \times Y$  are the same. Consequently,  $|X \times Y| = mn$ .

**DEFINITION 1.1.45** ► For a set  $X$ , the set  $\delta_X$  defined by

$$\delta_X = \{(x, x) \mid x \in X\}$$

is called the **diagonal** of  $X$ .

For example, if  $X = \{1, 2, 3\}$ , then

$$\delta_X = \{(1, 1), (2, 2), (3, 3)\}.$$

The concept of the Cartesian product of two sets may be extended to any finite number of sets. For example, for any sets  $X_1, X_2, \dots, X_n$  the Cartesian product, denoted by  $X_1 \times X_2 \times \dots \times X_n$  or, equivalently, by  $\prod_{i=1}^n X_i$ , is the set of all **ordered  $n$ -tuples**  $(x_1, x_2, \dots, x_n)$ , where  $x_i \in X_i$  for all  $i = 1, 2, \dots, n$ . Here we use the convention that if  $(x_1, x_2, \dots, x_n), (y_1, y_2, \dots, y_n) \in X_1 \times X_2 \times \dots \times X_n$ . Then

$$(x_1, x_2, \dots, x_n) = (y_1, y_2, \dots, y_n)$$

if and only if  $x_1 = y_1, x_2 = y_2, \dots$ , and  $x_n = y_n$ ; i.e.,  $x_i = y_i$  for all  $i = 1, 2, \dots, n$ .

For convenience we shall write  $X^n$  to represent  $X \times X \times \dots \times X$ , which is the  **$n$ -fold Cartesian product** of  $X$  with itself. It follows that  $\mathbb{R}^3 = \mathbb{R} \times \mathbb{R} \times \mathbb{R}$  denotes the usual three-dimensional space.

## Computer Representation of Sets

In the preceding sections, we discussed sets and described their basic operations, including determining if a set is a subset of another set, union and intersection of sets, and the complement of a set. In this section, we describe an effective way to represent sets in a computer's memory so that we can write computer programs to perform basic operations on sets.

Recall that when we study sets we usually study them in the context of a universal set. For example, if  $A$  is a set of first 10 positive integers and  $B$  is a set of first 15 positive integers, we can take the universal set to be a set of first 20 positive integers. When we represent sets in computer memory, we represent them in the context of a universal set.

A simple way to represent a set in computer memory is to store the elements of the set in an array as an unordered list. The problem with doing this is that the operations of determining the union, intersection, and so on, becomes time-consuming. We describe sets as a sequence of bit strings.

A **bit string** is a sequence of 0's and 1's. The sequences 10010001010 and 10111000110 are examples of bit strings. The **length** of a bit string is the number of occurrences of 0's and 1's in it. For example, the length of the bit string 10010001010 is 11 and the length of the bit string 10111000110110 is 14.

To increase the readability of bit strings, we break them into blocks of 4 bits in which all except (possibly) the last block have 4 bits. For example, 10111000110110 is written as 1011 1000 1101 10.

Consider the set

$$A = \{a, b, c, d\}.$$

Some other ways  $A$  can be written are as follows:

$$A = \{b, c, a, d\},$$

and

$$A = \{c, d, b, a\},$$

and

$$A = \{d, a, b, c\}.$$

As remarked above, if we arbitrarily represent  $A$  into computer memory, then the operations of determining the union, intersection, and so on, would be time-consuming. Therefore, before we represent sets in computer memory, we fix the order in which the elements of the set would appear.

Let  $U$  be the following set:

$$U = \{a, b, c, d, e, f, g, h\}.$$

We assume that the elements of  $U$  and any of its subsets would appear in the same order. For example, we assume that we will always write  $a$  first,  $b$  second,  $c$  third, and so on. If  $A$  is a subset of  $U$  consisting of the elements  $b, a, g, d$ , and  $f$ , then we would write  $A$  as:

$$A = \{a, b, d, f, g\}.$$

To further clarify, let us write  $a_1 = a, a_2 = b, a_3 = c, a_4 = d, a_5 = e, a_6 = f, a_7 = g, a_8 = h$ . Then we assume that the elements of the set  $U$  and any of its subsets would appear in the order  $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8$ , and so on.

Now the set  $U$  has eight elements. Therefore, we will use a bit string of 8 bits to represent a subset of  $U$ . The  $i$ th bit of the bit string is 1 if the  $i$ th element of  $U$  is in the subset; otherwise the  $i$ th bit is zero.

For example, because  $a_1, a_2, a_4, a_6$ , and  $a_7$  are in  $A$ , the bit string representing  $A$  is 1101 0110.

Thus, in general, in computer memory a set can be represented as a bit string.

Suppose  $U$  is the universal set with  $n$  elements. We use bit strings of length  $n$  to represent the set  $U$  and its subsets.

---

**DEFINITION 1.1.46** ▶ Let  $U = \{a_1, a_2, \dots, a_n\}$  be the universal set with  $n$  elements, and let  $A$  be a subset of  $U$ . Let  $s_A$  denote the  $n$  bit string of  $A$  and  $s_{Ai}$  denote the  $i$ th bit of  $s_A$  for all  $i = 1, 2, \dots, n$ . Then the  $i$ th bit of  $s_A$  of  $A$  is 1 if the  $i$ th element,  $a_i$ , of  $U$  is in  $A$ ; otherwise the  $i$ th bit of  $s_A$  is 0, i.e.,

$$s_{Ai} = \begin{cases} 1 & \text{if } a_i \in A, \\ 0 & \text{if } a_i \notin A. \end{cases}$$

### EXAMPLE 1.1.47

Let the universal set  $U = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14\}$ . Suppose  $a_i = i$ ,  $i = 1, 2, \dots, 14$ . Because the set  $U$  has 14 elements, we use a bit string of length 14 to represent a subset of  $U$ .

Let  $A = \{1, 3, 5, 7, 9, 11, 13\}$ . Then the bit string representing  $A$  is 1010 1010 1010 10. Similarly, if  $B = \{2, 5, 6, 8, 10, 14\}$ , then the bit string representing  $B$  is 0100 1101 0100 01.

**EXAMPLE 1.1.48**

Let the universal set  $U = \{a, b, c, d, e, f, g, h, i, j\}$ . We assume that the elements of any subset of  $U$  appear in the order  $a, b, c, d, e, f, g, h, i$ , and  $j$ . Let  $A$  be the subset of  $U$  such that the bit string of  $A$  is  $s_A = 1101\ 0010\ 10$ . Because the first bit  $s_{A1} = 1$ , it follows that  $a \in A$ . Similarly, because  $s_{A7} = 1$ , it follows that  $g \in A$ . Now  $s_{A3} = 0$  and so  $c \notin A$ .

Next, we describe how to determine the union, intersection, and complement of sets using bit strings.

Let the universal set  $U$  have  $n$  elements. Then the bit string needed to represent a subset of  $U$  is of length  $n$ . Let  $A$  and  $B$  be subsets of  $U$ . Then  $s_A$  denotes the bit string of  $A$ ,  $s_B$  denotes the bit string of  $B$ , and  $s_{A \cup B}$  denotes the bit string of  $A \cup B$ . Moreover,  $s_{Ai}$  denotes the  $i$ th bit of  $s_A$  and  $s_{Bi}$  denotes the  $i$ th bit of  $s_B$ .

Let us write  $U = \{u_1, u_2, \dots, u_n\}$ . Now if  $u_i \in A$ , then  $s_{Ai} = 1$ ; otherwise  $s_{Ai} = 0$ , i.e.,

$$s_{Ai} = \begin{cases} 1 & \text{if } u_i \in A, \\ 0 & \text{if } u_i \notin A. \end{cases}$$

Similarly,

$$s_{Bi} = \begin{cases} 1 & \text{if } u_i \in B, \\ 0 & \text{if } u_i \notin B. \end{cases}$$

Now consider  $s_{(A \cup B)i}$ , the  $i$ th bit of  $s_{A \cup B}$ . If  $s_{Ai} = 1$ , then  $u_i \in A \subseteq A \cup B$  and so  $s_{(A \cup B)i} = 1$ . Similarly, if  $s_{Bi} = 1$ , then  $u_i \in B \subseteq A \cup B$  and so  $s_{(A \cup B)i} = 1$ . It now follows that

$$s_{(A \cup B)i} = \begin{cases} 1 & \text{if either } s_{Ai} = 1 \text{ or } s_{Bi} = 1, \\ 0 & \text{otherwise.} \end{cases}$$

In a similar manner, we can show that if  $s_{A \cap B}$  denotes the bit string of  $A \cap B$ , then

$$s_{(A \cap B)i} = \begin{cases} 1 & \text{if } s_{Ai} = 1 \text{ and } s_{Bi} = 1, \\ 0 & \text{otherwise.} \end{cases}$$

Moreover, if  $s_{A'}$  denotes the bit string of  $A'$ , the complement of  $A$ , then

$$s_{A'i} = \begin{cases} 1 & \text{if } s_{Ai} = 0, \\ 0 & \text{if } s_{Ai} = 1. \end{cases}$$

Notice that to find the bit string of the complement, we change 0 to 1 and 1 to 0.

**EXAMPLE 1.1.49**

Let  $U = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$  be the universal set. Let

$$A = \{2, 4, 6, 8, 11, 12\}$$

and

$$B = \{1, 2, 5, 7, 8, 10\}.$$

Then

$$s_A = 0101\ 0101\ 0011$$

and

$$s_B = 1100\ 1011\ 0100.$$

To determine  $s_{A \cup B}$ , let us write the bit strings as follows:

$$\begin{array}{ccccccccccccc} s_A & = & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ & & \downarrow & \text{Compare the corresponding bits.} \\ s_B & = & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ s_{A \cup B} & = & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \end{array}$$

We compare the corresponding bits and if either of the two bits is 1, we make the corresponding bit of  $s_{A \cup B}$  1.

To determine  $s_{A \cap B}$ , we again compare the corresponding bits of  $s_A$  and  $s_B$ . If both of the corresponding bits are 1, we make the corresponding bit of  $s_{A \cap B}$  1. Thus,

$$\begin{array}{ccccccccccccc} s_A & = & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ & & \downarrow & \text{Compare the corresponding bits.} \\ s_B & = & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ s_{A \cap B} & = & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{array}$$

Similarly,

$$\begin{aligned} s_A &= 0101\ 0101\ 0011 \\ s_{A'} &= 1010\ 1010\ 1100. \end{aligned}$$

## WORKED-OUT EXERCISES

**Exercise 1:** In each case describe the set  $A$  in the notation  $A = \{x \mid P(x)\}$ .

- (a)  $A$  is the set of all positive integers that are multiples of 7.
- (b)  $A$  is the set of all real numbers that are solutions of the equation  $x^4 = 1$ .

**Solution:**

- (a)  $A = \{7n \mid n \in \mathbb{N}\}$
- (b)  $A = \{x \mid x \in \mathbb{R} \text{ and } x^4 = 1\}$

**Exercise 2:** List the elements of the following sets.

- (a)  $A = \{x \mid x \in \mathbb{R} \text{ and } x^4 = 1\}$
- (b)  $A = \{n \mid n \in \mathbb{N} \text{ and } n < 7\}$

**Solution:**

- (a) Notice that  $x^4 = 1$  is equivalent to  $x^4 - 1 = 0$ . Now  $x^4 - 1 = (x - 1)(x + 1)(x - i)(x + i)$ . This implies that the solutions of the equation  $x^4 - 1 = 0$  are  $1, -1, i, -i$ . However,  $1, -1 \in \mathbb{R}$  and  $i, -i \notin \mathbb{R}$ . Thus, the elements of the set  $A$  are  $1, -1$ . (Note that here  $i = \sqrt{-1}$ .)
- (b)  $1, 2, 3, 4, 5, 6$

**Exercise 3:** Let  $A$  and  $B$  be sets. Prove that  $A \subseteq B$  if and only if  $A \cup B = B$ .

**Solution:** In this exercise, we are saying that if  $A \subseteq B$ , then  $A \cup B = B$  and if  $A \cup B = B$ , then  $A \subseteq B$ . Therefore, we have to prove two things. Assume that  $A \subseteq B$  and show that  $A \cup B = B$ . Then assume that  $A \cup B = B$  and show that  $A \subseteq B$ .

First suppose  $A \subseteq B$ . We now show that  $A \cup B = B$ . To do so, we show that  $A \cup B \subseteq B$  and  $B \subseteq A \cup B$ . The result will then follow from the equality of sets.

Let  $x$  be any element of  $A \cup B$ . We have

$$\begin{aligned} x &\in A \cup B \\ \Rightarrow x &\in A \text{ or } x \in B \\ \Rightarrow x &\in B. \text{ because } A \subseteq B \text{ and if } x \in A, \text{ then } x \in B \end{aligned}$$

Thus, we find that every element of  $A \cup B$  is an element of  $B$ , so  $A \cup B \subseteq B$ . Also,  $B \subseteq A \cup B$  by Theorem 1.1.22. Hence,  $A \cup B = B$ .

Conversely, suppose  $A \cup B = B$ . Now by Theorem 1.1.22,  $A \subseteq A \cup B$ . We therefore have  $A \subseteq A \cup B = B$ , so  $A \subseteq B$ .

**Exercise 4:** Let  $A$ ,  $B$ , and  $C$  denote the subsets of a set  $S$  and let  $C'$  denote the complement of  $C$  in  $S$ . If  $A \cap C = B \cap C$  and  $A \cap C' = B \cap C'$ , then prove that  $A = B$ .

**Solution:** By Theorem 1.1.29(i),  $A = A \cap S$  and by Theorem 1.1.38(i),  $S = C \cup C'$ . Therefore,

$$\begin{aligned} A &= A \cap S \\ &= A \cap (C \cup C') \\ &= (A \cap C) \cup (A \cap C') \quad \text{by distributivity} \\ &= (B \cap C) \cup (B \cap C') \quad \text{by the given conditions} \\ &= B \cap (C \cap C') \quad \text{by distributivity} \\ &= B \cap S \\ &= B. \end{aligned}$$

**Exercise 5:** Let  $A$ ,  $B$ , and  $C$  be sets such that  $A \cap B = A \cap C$  and  $A \cup B = A \cup C$ . Prove that  $B = C$ .

**Solution:** We have

$$\begin{aligned} B &= B \cup (A \cap B) && \text{by absorptivity} \\ &= B \cup (A \cap C) && \text{by the given condition} \\ &= (B \cup A) \cap (B \cup C) && \text{by distributivity} \\ &= (A \cup B) \cap (B \cup C) && \text{by commutativity} \\ &= (A \cup C) \cap (B \cup C) && \text{by the given condition} \\ &= (A \cup B) \cup C && \text{by distributivity} \\ &= (A \cap C) \cup C && \text{by the given condition} \\ &= C. && \text{by absorptivity} \end{aligned}$$

**Exercise 6:** Let  $A$ ,  $B$ , and  $C$  be sets such that  $(A \cap C) \cup (B \cap C') = \emptyset$ . Prove that  $A \cap B = \emptyset$ , where  $C'$  is the complement of  $C$ .

**Solution:** Because  $(A \cap C) \cup (B \cap C') = \emptyset$ , we must have  $A \cap C = \emptyset$  and  $B \cap C' = \emptyset$ . By Theorem 1.1.38(iii), we have  $B - C = B \cap C'$  and so  $B - C = \emptyset$ . Now  $B - C = \emptyset$  implies that  $B = C$  or  $B \subset C$  or  $B = \emptyset$ .

If  $B = C$ , then  $A \cap C = \emptyset$  implies that  $A \cap B = A \cap C = \emptyset$ .

If  $B \subset C$ , then  $A \cap B \subseteq A \cap C = \emptyset$  gives  $A \cap B = \emptyset$ .

If  $B = \emptyset$ , then, of course,  $A \cap B = \emptyset$ .

Hence, in either case we have  $A \cap B = \emptyset$ .

**Exercise 7:** Prove that an equivalent definition of the symmetric difference of two sets  $A$  and  $B$  is

$$A \Delta B = (A \cup B) - (A \cap B).$$

**Solution:** We know by definition that  $A \Delta B = (A - B) \cup (B - A)$ . Hence, we are to show that  $(A - B) \cup (B - A) = (A \cup B) - (A \cap B)$ . We show this by using Theorem 1.1.38. Indeed,

$$\begin{aligned} &(A \cup B) - (A \cap B) \\ &= (A \cup B) \cap (A \cap B)' && \text{as } X - Y = X \cap Y' \\ &= (A \cup B) \cap (A' \cup B') && \text{by DeMorgan's laws} \\ &= ((A \cup B) \cap A') \cup ((A \cup B) \cap B') && \text{by distributivity} \\ &= (A \cap A') \cup (B \cap A') \cup \\ &\quad (A \cap B') \cup (B \cap B') && \text{by distributivity} \\ &= \emptyset \cup (B - A) \cup (A - B) && \text{for any set } X, X \cap X' = \emptyset \\ &= (B - A) \cup (A - B) && \text{for any set } X, X \cup \emptyset = X \\ &= (A - B) \cup (B - A). && \text{by commutativity} \end{aligned}$$

**Exercise 8:** Prove that for any nonempty sets  $A$ ,  $B$ , and  $C$ ,

$$A \times (B - C) = (A \times B) - (A \times C).$$

**Solution:** Let  $(x, y) \in A \times (B - C)$ . We have

$$\begin{aligned} &(x, y) \in A \times (B - C) \\ &\Rightarrow x \in A; y \in B - C \\ &\Rightarrow x \in A; y \in B \text{ and } y \notin C \end{aligned}$$

$$\begin{aligned} &\Rightarrow (x, y) \in A \times B \text{ and } (x, y) \notin A \times C \text{ as } y \notin C \\ &\Rightarrow (x, y) \in (A \times B) - (A \times C). \end{aligned}$$

$$\text{So, } A \times (B - C) \subseteq (A \times B) - (A \times C).$$

On the other hand, let  $(x, y) \in (A \times B) - (A \times C)$ . This gives

$$\begin{aligned} &(x, y) \in A \times B \text{ and } (x, y) \notin A \times C \\ &\Rightarrow x \in A; y \in B \text{ and } (x, y) \notin A \times C \\ &\Rightarrow x \in A; y \in B \text{ and } y \notin C \text{ (as } x \in A) \\ &\Rightarrow x \in A; y \in B - C \\ &\Rightarrow (x, y) \in A \times (B - C). \end{aligned}$$

Hence,  $(A \times B) - (A \times C) \subseteq A \times (B - C)$ . Consequently, we have

$$A \times (B - C) = (A \times B) - (A \times C).$$

**Exercise 9:** Justify the following statements or else give an example to disprove the result. Let  $A$ ,  $B$ , and  $C$  be subsets of a set  $U$ .

- (a)  $A \Delta C = B \Delta C \Rightarrow A = B$
- (b)  $(A - C) - (B - C) = (A - B) - C$
- (c)  $(A - B)' = (B - A)'$

**Solution:**

- (a) True.  
Let  $x \in A$ . We shall consider two cases separately.

**Case 1:**  $x \notin C$ .

Then

$$x \in A - C \subseteq (A - C) \cup (C - A) = A \Delta C = B \Delta C \text{ given}$$

So  $x \in (B - C) \cup (C - B)$ . But because  $x \notin C$ , we have  $x \notin C - B$ , from which it follows that  $x \in B - C \subseteq B$ . Consequently,  $A \subseteq B$ .

**Case 2:**  $x \in C$ .

Here  $x \in A \cap C$ , so  $x \notin (A \cup C) - (A \cap C) = A \Delta C$  (by Worked-Out Exercise 7 (of this section)). Now, if possible, let  $x \notin B$ . Then,  $x \in C - B \subseteq (C - B) \cup (B - C) = B \Delta C = A \Delta C$  (given)—a contradiction. Hence,  $x \in B$  and, consequently, we have  $A \subseteq B$ .

In an essentially similar manner, it can be proved that  $B \subseteq A$ . Consequently,  $A = B$ .

- (b) True.  
Indeed,
- $$\begin{aligned} &(A - C) - (B - C) \\ &= (A \cap C') - (B \cap C') && \text{because } X - Y = X \cap Y' \\ &= (A \cap C') \cap (B \cap C')' \\ &= (A \cap C') \cap (B' \cup (C')') && \text{by DeMorgan's laws} \\ &= (A \cap C') \cap (B' \cup C) && \text{because } (X')' = X \\ &= ((A \cap C') \cap B') \cup ((A \cap C') \cap C) && \text{by distributivity} \\ &= (A \cap (C' \cap B')) \cup (A \cap (C' \cap C)) && \text{by associativity} \\ &= (A \cap (B' \cap C')) \cup (A \cap \emptyset) && \text{by commutativity} \\ &= ((A \cap B') \cap C') \cup \emptyset && \text{by associativity} \\ &= (A - B) \cap C' && \text{because } X \cup \emptyset = X \\ &= (A - B) - C. \end{aligned}$$

(c) False.

Let  $U = \{a, b, c, d, e, f, g\}$ ,  $A = \{a, b, e, f\}$ , and  $B = \{b, f, g\}$ . Then,

$$A - B = \{a, e\} \quad \text{and} \quad B - A = \{g\}.$$

Now,

$$(A - B)' = U - (A - B) = \{b, c, d, f, g\}$$

and

$$(B - A)' = U - (B - A) = \{a, b, c, d, e, f\}.$$

It follows that

$$(A - B)' \neq (B - A)'.$$

## SECTION REVIEW

### Key Terms

set	power set	complement of a set
roster method	universal set	symmetric difference
set-builder method	Venn diagrams	ordered pair
subset	union of sets	Cartesian product
superset	intersection of sets	diagonal of a set
proper subset	disjoint sets	ordered $n$ -tuples
equal sets	index set	$n$ -fold Cartesian product
empty (null) set	set difference	bit string
finite set	mutually disjoint	length
infinite set	pairwise disjoint	
singleton set	relative complement	

### Some Key Definitions

1. A set is a well-defined collection of objects.
2. Let  $X$  and  $Y$  be sets. Then  $X$  is said to be a subset of  $Y$ , written  $X \subseteq Y$ , if every element of  $X$  is an element of  $Y$ .
3. Two sets  $X$  and  $Y$  are said to be equal, written  $X = Y$ , if every element of  $X$  is an element of  $Y$  and every element of  $Y$  is an element of  $X$ ; i.e., if  $X \subseteq Y$  and  $Y \subseteq X$ .
4. If there exists a nonnegative integer  $n$  such that  $X$  has  $n$  elements, then  $X$  is called a finite set with  $n$  elements; otherwise  $X$  is called an infinite set.
5. For any set  $X$ , the power set of  $X$ , written  $\mathcal{P}(X)$ , is the set of all subsets of  $X$ .
6. The union of two sets  $X$  and  $Y$ , denoted by  $X \cup Y$ , is defined to be the set  $X \cup Y = \{x \mid x \in X \text{ or } x \in Y\}$ .
7. The intersection of two sets  $X$  and  $Y$ , denoted by  $X \cap Y$ , is defined to be the set  $X \cap Y = \{x \mid x \in X \text{ and } x \in Y\}$ .
8. Two sets  $X$  and  $Y$  are said to be disjoint if  $X \cap Y = \emptyset$ .
9. Let  $X$  and  $Y$  be sets. The difference of  $X$  and  $Y$  (or the relative complement of  $Y$  in  $X$ ), written  $X - Y$ , is the set  $X - Y = \{x \mid x \in X \text{ but } x \notin Y\}$ .
10. The Cartesian product of two nonempty sets  $X$  and  $Y$ , denoted by  $X \times Y$ , is the set  $X \times Y = \{(x, y) \mid x \in X, y \in Y\}$ .

## Some Key Results

1. Let  $X$  and  $Y$  be sets. Then  $X \subseteq X \cup Y$  and  $Y \subseteq X \cup Y$ .
2. Let  $X$  and  $Y$  be sets. Then  $X \cap Y \subseteq X$  and  $X \cap Y \subseteq Y$ .
3. Let  $X, Y, Z$  be subsets of a set  $U$ . Then
  - (i) If  $X \subseteq Y$ , then  $X \cup Y = Y$  and  $X \cap Y = X$ .
  - (ii)  $X \cup \emptyset = X$  and  $X \cap \emptyset = \emptyset$ .
  - (iii)  $X \cup X = X$  and  $X \cap X = X$ .
  - (iv)  $X \cup Y = Y \cup X$  and  $X \cap Y = Y \cap X$ .
  - (v)  $(X \cup Y) \cup Z = X \cup (Y \cup Z)$  and  $(X \cap Y) \cap Z = X \cap (Y \cap Z)$ .
  - (vi)  $X \cup (Y \cap Z) = (X \cup Y) \cap (X \cup Z)$  and  $X \cap (Y \cup Z) = (X \cap Y) \cup (X \cap Z)$ .
  - (vii)  $X \cap (X \cup Y) = X$  and  $X \cup (X \cap Y) = X$ .
4. Let  $X$  and  $Y$  be sets and  $U$  be a universal set under consideration. Then
  - (i)  $X \cup X' = U$  and  $X \cap X' = \emptyset$ .
  - (ii)  $(X')' = X$ .
  - (iii)  $X - Y = X \cap Y'$ .
  - (iv) (DeMorgan's laws)  $(X \cup Y)' = X' \cap Y'$  and  $(X \cap Y)' = X' \cup Y'$ .

## EXERCISES

---

1. Let  $A = \{1, 2, 3, 4, 5, 6\}$ ,  $B = \{x \in \mathbb{Z} \mid x \text{ is divisible by } 6\}$ , and  $C = \{x \in \mathbb{R} \mid x^2 = 2 \text{ or } x^3 = 1\}$ . Mark the following true or false.
  - a.  $3 \in A$
  - b.  $6 \in A$
  - c.  $2 \notin A$
  - d.  $2 \in B$
  - e.  $6 \in B$
  - f.  $24 \in B$
  - g.  $28 \notin B$
  - h.  $2 \in C$
  - i.  $1 \in C$
  - j.  $-\sqrt{2} \in C$
  - k.  $5 \in A \cup B$
  - l.  $6 \in A \cap B$
  - m.  $1 \in A \cap C$
  - n.  $\sqrt{2} \in B \cup C$
2. Mark the following true or false.
  - a.  $28 \in \mathbb{Z}$
  - b.  $-5 \in \mathbb{N}$
  - c.  $\sqrt{2} \notin \mathbb{Q} \cap \mathbb{R}$
  - d.  $\mathbb{Z} \cup \mathbb{Q} = \mathbb{R}$
  - e.  $\mathbb{R} \cap \mathbb{C} = \mathbb{R}$
3. Let  $U = \{a, b, c, d, e, f, g\}$ ,  $A = \{a, d, e, f\}$ , and  $B = \{b, e, g\}$  be sets, where  $U$  acts as the universal set. Determine the following.
  - a.  $(A \cup B)'$
  - b.  $A \cap B$
  - c.  $A - B$
  - d.  $B - A$
4. Let  $U$  be the set of all students in a college. Let  $A$  be the set of students taking the discrete mathematics course and  $B$  be the set of students taking the calculus course. Describe the following.
  - a.  $A \cup B$
  - b.  $A \cap B$
  - c.  $A - B$
  - d.  $B - A$
5. Let  $P = \{x \in \mathbb{N} \mid 2 < x \leq 8\}$ ,  $Q = \{x \in \mathbb{N} \mid 0 \leq x < 5\}$ ,  $R = \{x \in \mathbb{N} \mid 1 \leq x \leq 10\}$ . Let  $U = \{x \in \mathbb{Z} \mid -2 \leq x < 12\}$  be the universal set. Determine the following.
  - a.  $P \cup R$
  - b.  $Q \cap R$
  - c.  $P \Delta R$
  - d.  $Q'$
6. Let  $P, Q, R$ , and  $U$  be the same as in Exercise 5. Verify the following.
  - a.  $(P \cup Q)' = P' \cap Q'$
  - b.  $P \cap (P \cup R) = P$
  - c.  $P \cup (Q \cap R) = (P \cup Q) \cap (P \cup R)$
7. Let  $A = \{x \in \mathbb{R} \mid 1 < x \leq 5\}$  and  $B = \{x \in \mathbb{R} \mid 3 \leq x \leq 8\}$ . Find  $A \cup B$ ,  $A \cap B$ ,  $A - B$ ,  $B - A$ .
8. Determine whether the following pairs of sets are equal. Justify your answer.
 
$$A = \left\{ n \in \mathbb{Z} \mid n = \frac{1}{n} \right\} \quad \text{and} \quad B = \{x \in \mathbb{R} \mid x^2 = 1\}.$$
9. Does every set has a subset? Give an example of a set that has only one proper subset.
10. Let  $X$  be a set with 4 elements. Find  $|\mathcal{P}(X)|$ .
11. Find  $\mathcal{P}(\mathcal{P}(\mathcal{P}(\emptyset)))$ .
12. Let  $I_n = \{1, 2, \dots, n\}$ , the set of first  $n$  natural numbers.
  - a. Describe the set  $I_{10} - I_5$ .
  - b. Describe the set  $I_n - I_m$  if
    - (i)  $n > m$
    - (ii)  $n = m$
    - (iii)  $n < m$
13. Let  $A$  and  $B$  be subsets of the set  $U$ . Draw the Venn diagram of the following sets.
  - a.  $(A \cup B)'$
  - b.  $(A \cap B)'$
  - c.  $A \Delta B$
  - d.  $(A \cup B) - (A \cap B)$
14. Let  $A, B$ , and  $C$  be subsets of the set  $U$ . Draw the Venn diagram of the following sets.
  - a.  $(A \cup B) \cap C$
  - b.  $(A \cap B) \cup C$
  - c.  $(A \cap B) - C$
  - d.  $(A - B) - C$
  - e.  $(A - (B \cup C)) \cup (B - (A \cup C))$

15. Let  $A, B, C$ , and  $D$  be subsets of the set  $U$ . Draw the Venn diagram of the following sets.
- $A \cap B \cap C \cap D$
  - $(A \cup B \cup C) \cap D$
  - $(A \cup B) \cap (C \cap D)$
16. What sets do each of the Venn diagrams in Figure 1.10 represent?

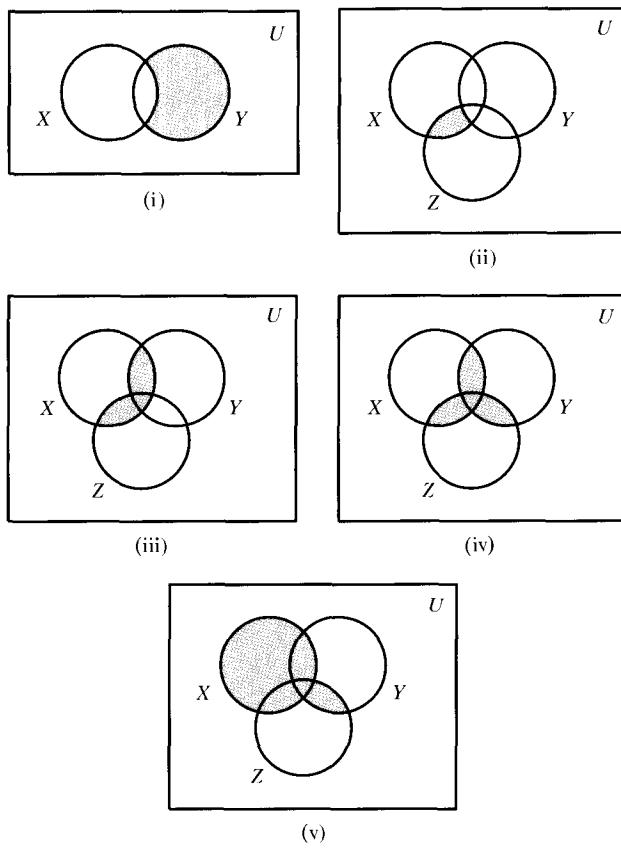


FIGURE 1.10

17. Let  $A$  and  $B$  be sets. Prove that  $A \subseteq B$  if and only if  $A \cap B = A$ .
18. Prove those parts of Theorem 1.1.29 that are not proved in this section.
19. Suppose  $P$  and  $Q$  are two sets. Let  $R$  be a set that contains elements belonging to  $P$  or  $Q$  but not both. Let  $T$  be a set that contains elements belonging to  $Q$  or the complement of  $P$  but not both. Show that  $R$  is the complement of  $T$ .
20. Let  $A$  and  $B$  be sets. Prove that  $A - (A - B) = A \cap B$ .
21. Let  $A, B$ , and  $C$  be subsets of a set  $U$ . Prove the following.
- $((A - B) \cup (A \cap B)) \cap ((B - A) \cup (A \cup B)') = \emptyset$
  - $A - (B \cap C) = (A - B) \cup (A - C)$
  - $A - (B \cup C) = (A - B) \cap (A - C)$
  - $A \cap (B - C) = (A \cap B) - (A \cap C)$
22. Let  $U = \{a, b, c, d, e, f, g\}$ ,  $A = \{a, e, f\}$ ,  $B = \{b, g\}$ , and  $C = \{d, e, g\}$  be sets, where  $U$  acts as the universal set. Determine the following sets.

- $A \times B$
  - $B \times C$
  - $A \times C$
  - $A \times B \times C$
23. Let  $U = \{a, b, c, d, e, f, g\}$ ,  $A = \{a, d, e, f\}$ ,  $B = \{b, e, g\}$ , and  $C = \{a, c, e, g\}$  be sets, where  $U$  acts as the universal set. Verify that

$$A \times (B - C) = (A \times B) - (A \times C).$$

24. Let  $A, B, C$  be sets. Prove the following.
- $A \times (B \cap C) = (A \times B) \cap (A \times C)$
  - $A \times (B \cup C) = (A \times B) \cup (A \times C)$
  - $A \times B = \bigcup_{b \in B} A \times \{b\}$
25. Let  $A$  and  $B$  be sets as in Exercise 23. Verify that  $(A - B) \cup (B - A) = (A \cup B) - (A \cap B)$ .
26. If  $A, B, C$  are subsets of a set  $U$ , then prove that  $A - C = B - C$  if and only if  $A \cup C = B \cup C$ .
27. Let  $A, B, C$  be sets. Prove that  $A \cap (B \Delta C) = (A \cap B) \Delta (A \cap C)$ .
28. Let  $A$  and  $B$  be finite subsets of a set  $U$ . Show that
- $|A - B| = |A| - |A \cap B|$ .
  - $|A \cup B| \leq |A| + |B|$ . Moreover, show that  $|A \cup B| = |A| + |B|$  if and only if  $A \cap B = \emptyset$ .
  - $|A \cup B| = |A| + |B| - |A \cap B|$ .
29. In Figure 1.11,  $A$  is the set of people who go to a resort area for vacation,  $B$  is the set of people who take a cruise for vacation, and  $C$  is the set of people who go to a national park for vacation. The numbers in the figure represent the number of people in that region. Suppose that  $|A| = 150$ ,  $|B| = 100$ , and  $|C| = 300$ .

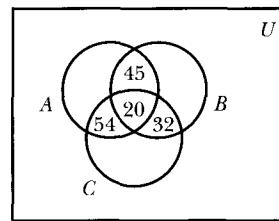


FIGURE 1.11

- How many people only go to a resort area for vacation?
  - How many people only take a cruise for vacation?
  - How many people only go to a national park for vacation?
  - How many people either go to a resort area for vacation or take a cruise for vacation?
  - How many people use one of the three methods to take a vacation?
30. In Figure 1.12,  $A$  is the set of students taking algebra,  $B$  is the set of students who play basketball,  $C$  is the set of students taking the computer programming course, and the numbers in each region represent the number of students in that region.

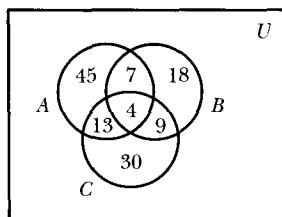


FIGURE 1.12

- a. What is the number of students taking both the algebra course and the computer programming course and playing basketball?
  - b. What is the number students taking the algebra course and playing basketball?
  - c. What is the number of students taking the algebra course and the computer programming course?
  - d. What is the number of students taking the computer programming course and playing basketball?
  - e. What is the number of students taking the algebra course?
  - f. What is the number of students taking the computer programming course?
  - g. What is the number of students playing basketball?
  - h. What is the number of students taking the algebra course or the computer programming course or playing basketball?
31. In an examination, 70% of the candidates passed in mathematics, 73% passed in physics, and 64% passed in both subjects. If 63 candidates failed in both subjects, use a Venn diagram to find the total number of candidates who appeared at the examination.

32. Prove those parts of Theorem 1.1.38 that are not proved in this section.
33. Let  $U$  be a universal set of 20 elements.
  - a. What is the bit string of the empty set?
  - b. What is the bit string of  $U$ ?
34. Let  $U = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13\}$ . Let  $A = \{2, 5, 8, 9, 10, 13\}$  and  $B = \{1, 3, 7, 9, 12, 13\}$ . Determine the following.
  - a.  $s_A$
  - b.  $s_{A'}$
  - c.  $s_B$
  - d.  $s_{A \cup B}$
  - e.  $s_{A \cap B}$
35. Let  $U = \{x \in \mathbb{Z} \mid 1 \leq x \leq 30\}$ . Let  $A = \{x \in U \mid 2 \text{ divides } x\}$  and  $B = \{x \in U \mid 3 \text{ divides } x\}$ . Determine the following.
  - a.  $s_A$
  - b.  $s_{A'}$
  - c.  $s_B$
  - d.  $s_{A \cup B}$
  - e.  $s_{A \cap B}$
36. Let  $U, P, Q$ , and  $R$  be as given in Exercise 5. Determine the following.
  - a.  $s_{P \cup Q}$
  - b.  $s_{P \cup R}$
  - c.  $s_{P \cup Q \cup R}$
  - d.  $s_{Q \cap R}$
  - e.  $s_{P \cap Q \cap R}$
  - f.  $s_{P \cap (Q \cup R)}$
37. Prove the following set-theoretic statements if you find them correct or else give an example to disprove the result. The sets  $A, B$  and  $C$  are subsets of a set  $U$ .
  - a.  $A \cup (B - C) = (A \cup B) - (A \cup C)$
  - b.  $(A - B) - C = A - (B \cup C)$
  - c.  $(A \cup B) - A = A - B$
  - d.  $A - C = B - C$  if and only if  $A \cup C = B \cup C$

## 1.2 MATHEMATICAL LOGIC

Intuitively, logic is the discipline that considers the methods of reasoning. It provides the rules and techniques for determining whether an argument is valid or not. In everyday life, we use reasoning to prove different points. For example, to prove to our parents that we passed an exam, we might show the test and the score. Or to prove to the utility company that our bill has been paid, we might show the cancelled check. Similarly, in mathematics and computer science, mathematical logic or logic is used to prove results. To be specific, in mathematics we use logic or logical reasoning to prove theorems, and in computer science we use logic or logical reasoning to prove the correctness of programs and also to prove theorems. (Mathematical logic is a discipline in its own right, and it is worthy of full treatment. In this section, however, we present and discuss only those aspects of mathematical logic that are necessary to the study of mathematics and computer science.)

Throughout the book not only are results given, but whenever a computer algorithm is available to solve the problem, an algorithm is presented. Therefore, our main objective is to use logic to prove theorems and the correctness of the programs. We therefore begin by defining the word theorem.

A *theorem* is a statement that can be shown to be true (under certain conditions). For example, in mathematics the following statement is a theorem,

If  $x$  is an even integer, then  $x + 1$  is an odd integer.

It can be proved that the above statement under the condition that  $x$  is an integer is true. A proof of a theorem is an argument consisting of a sequence of statements aimed at demonstrating the truth of the assertion. It is not easy to give a complete and rigorous definition of a statement, but the following definition will serve our purpose.

**DEFINITION 1.2.1** ► A **statement**, or a **proposition**, is a declarative sentence that is either true or false, but not both.

### EXAMPLE 1.2.2

In this example, we consider the following sentences.

- (i) 4 is an integer.
- (ii)  $\sqrt{5}$  is an integer.
- (iii) Washington, DC, is the capital of the United States.

Each of these sentences is a declarative sentence. Sentence (i) is true, sentence (ii) is not true, and sentence (iii) is true. Hence, these are examples of statements.

We typically use lowercase letters, with or without subscripts, such as  $p$ ,  $q$ , and  $r$  to denote statements. For example, we might write

$p$  : 4 is an integer.

$q$  :  $\sqrt{5}$  is an integer.

$r$  : Washington, DC, is the capital of the United States.

### EXAMPLE 1.2.3

Consider the following sentences.

$p$  : 5 is less than 3.

$q$  : 7 is an even integer.

$r$  : Every even integer greater than 4 is a sum of two odd primes.

Now, 5 is less than 3 is false, so  $p$  is a statement. Similarly,  $q$  is a statement.

Consider sentence  $r$ . So far, no one has been able to prove that this is true. At the same time, no one has proved that it is false. Nevertheless,  $r$  is a statement because it is either true or false, but not both. This is known as the *Goldbach's conjecture*.

### EXAMPLE 1.2.4

Consider the following sentences.

$p$  : Will you go?

$q$  : Enjoy the lovely weather!

Here the sentences  $p$  and  $q$  are not declarative sentences, so these are not statements.

### REMARK 1.2.5

► In this chapter, we assume that the reader is familiar with the basic terminology and properties of integers. For example, even and odd integers, prime integers, and statements such as 2 divides 4 and 2 does not divide 5. (We use the convention that an integer  $x$  divides another integer  $y$ , if  $x \neq 0$  and  $y = xz$  for some integer  $z$ .) In Chapter 2, we will formally discuss various properties of integers.

By definition, a statement is a declarative sentence that can be classified as true or false, but not both. Thus, one of the values “*truth*” or “*falsity*” that is assigned to

a statement is called its **truth value**. We abbreviate “truth” to  $T$  or 1 and “falsity” to  $F$  or 0. If a statement  $p$  is true, we say that the (*logical*) *truth value* of  $p$  is true and write  $p$  is  $T$  (or  $p$  is 1); otherwise, we say the (*logical*) *truth value* of  $p$  is false and write  $p$  is  $F$  (or  $p$  is 0).

In the above discussion, we assumed intuitively the idea of the words “sentence,” “true,” and “false,” and we defined a “statement” with the help of these words.

New statements can be constructed from existing statements. Next, we describe the different ways of doing so. In the process, we also define various logical operations.

## Negation

Consider the following statements:

$p$  : 2 is positive.

$q$  : It is not the case that 2 is positive.

We see that statement  $p$  is true and statement  $q$  is false. Statement  $q$  is obtained by negating statement  $p$ , and the truth values of  $p$  and  $q$  are opposite. Statement  $q$  is the negation of statement  $p$ . More formally, we have the following definition.

---

**DEFINITION 1.2.6** ▶ Let  $p$  be a statement. The **negation** of  $p$ , written  $\sim p$ , is the statement obtained by negating statement  $p$ .

It follows that the truth values of  $p$  and  $\sim p$  are opposite. The symbol  $\sim$  is called “*not*.” We read  $\sim p$  as “not  $p$ .”

Typically, if  $p$  is a statement, then its negation is formed by writing “it is not the case that  $p$ .” For example, if

$p$  : 2 is positive,

then

$\sim p$  : It is not the case that 2 is positive.

Sometimes,  $\sim p$  is also written as follows:

$\sim p$  : 2 is not positive.

By the definition of the negation of a statement  $p$ , the truth value of  $\sim p$  is opposite to the truth value of  $p$ ; i.e., if  $p$  is  $T$ , then  $\sim p$  is  $F$  and if  $p$  is  $F$ , then  $\sim p$  is  $T$ . We record this in a table, called a *truth table*, as follows:

$p$	$\sim p$
$T$	$F$
$F$	$T$

## Conjunction

Consider the following statements:

$p$  : 2 is an even integer.

$q$  : 7 divides 14.

Now consider the sentence

$r$  : 2 is an even integer and 7 divides 14.

Because  $r$  is true,  $r$  is a statement. Notice that  $r$  is, in fact, a combination of statements  $p$  and  $q$ . In other words,  $r$  can be expressed by joining  $p$  and  $q$  using the word “*and*.” A statement such as  $r$  is called the conjunction of  $p$  and  $q$ . More formally, we have the following definition.

---

**DEFINITION 1.2.7** ▶ Let  $p$  and  $q$  be statements. The **conjunction** of  $p$  and  $q$ , written  $p \wedge q$ , is the statement formed by joining statements  $p$  and  $q$  using the word “*and*.” The statement  $p \wedge q$  is true if both  $p$  and  $q$  are true; otherwise  $p \wedge q$  is false.

The symbol  $\wedge$  is called “*and*.” Let  $p$  and  $q$  be statements. The truth table of  $p \wedge q$  is given by:

$p$	$q$	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

### EXAMPLE 1.2.8

- (i) Let  $p$  : Washington, DC, is the capital of the United States and  $q$  : The United States is in North America. Then  $p \wedge q$  is the statement:

$p \wedge q$  : Washington, DC, is the capital of the United States and the United States is in North America.

Notice that  $p \wedge q$  is  $T$ .

- (ii) Let  $p$  : 2 divides 4 and  $q$  : 3 is greater than 5. Then  $p \wedge q$  is the statement:

$p \wedge q$  : 2 divides 4 and 3 is greater than 5.

Because  $p$  is  $T$  and  $q$  is  $F$ , it follows that  $p \wedge q$  is  $F$ .

- (iii) Let  $r$  : 2 divides 4 and  $s$  : 2 divides 6. Then  $r \wedge s$  is the statement:

$r \wedge s$  : 2 divides 4 and 2 divides 6.

Because  $r$  is  $T$  and  $s$  is  $T$ , it follows that  $r \wedge s$  is  $T$ .

- (iv) Let  $p$  : 5 is an integer and  $q$  : 5 is not an odd integer. Then  $p \wedge q$  is the statement:

$p \wedge q$  : 5 is an integer and 5 is not an odd integer.

Because  $p$  is  $T$  and  $q$  is  $F$ , it follows that  $p \wedge q$  is  $F$ .

Let us consider the statements in Example 1.2.8(iii). The statement  $r \wedge s$  is:

$r \wedge s$  : 2 divides 4 and 2 divides 6.

Sometimes, the statement  $r \wedge s$  is written

$r \wedge s$  : 2 divides both 4 and 6.

Similarly, the statement  $p \wedge q$  of Example 1.2.8(iv),

$p \wedge q$  : 5 is an integer and 5 is not an odd integer,

is also written as

$p \wedge q$  : 5 is an integer but 5 is not an odd integer.

## Disjunction

Given two statements  $p$  and  $q$ , we can form the statement “ $p$  or  $q$ ” by putting the word “or” between the statements such that the statement  $p$  or  $q$  is true if at least one of the statements  $p$  or  $q$  is true. For example, suppose we have the statements:

$$p : 2 \text{ is an integer.}$$

$$q : 3 \text{ is greater than } 5.$$

Then we can form the statement:

$$r : 2 \text{ is an integer or } 3 \text{ is greater than } 5.$$

Because  $p$  is  $T$ , it follows that  $r$  is true. Statement  $r$  is called the disjunction of  $p$  and  $r$ .

**DEFINITION 1.2.9** ▶ Let  $p$  and  $q$  be statements. The **disjunction** of  $p$  and  $q$ , written  $p \vee q$ , is the statement formed by putting statements  $p$  and  $q$  together using the word “or.” The truth value of the statement  $p \vee q$  is  $T$  if at least one of statements  $p$  and  $q$  is true.

The symbol  $\vee$  is called “or.” For statements  $p$  and  $q$ , the truth table of  $p \vee q$  is given by:

$p$	$q$	$p \vee q$
$T$	$T$	$T$
$T$	$F$	$T$
$F$	$T$	$T$
$F$	$F$	$F$

### EXAMPLE 1.2.10

Let

$$p : 2^2 + 3^3 \text{ is an even integer,}$$

$$q : 2^2 + 3^3 \text{ is an odd integer.}$$

Then

$$p \vee q : 2^2 + 3^3 \text{ is an even integer or } 2^2 + 3^3 \text{ is an odd integer.}$$

Sometimes, for better readability, we write  $p \vee q$  as:

$$p \vee q : \text{Either } 2^2 + 3^3 \text{ is an even integer or } 2^2 + 3^3 \text{ is an odd integer}$$

or

$$p \vee q : 2^2 + 3^3 \text{ is an even integer or an odd integer.}$$

**REMARK 1.2.11** ▶ We now make a comment about the use of the word “or.” In ordinary language “or” is sometimes used in an exclusive sense (i.e., “ $p$  or  $q$ ” means either  $p$  or  $q$  but not both) and sometimes in an inclusive sense (i.e., “ $p$  or  $q$ ” means either  $p$  or  $q$  or both  $p$  and  $q$ ). Consider the statement “2 is an even integer or 2 is an odd integer.” In this statement “or” is used in exclusive sense. However, in the statement “Laurie is a good student or Alyssa plays well,” “or” is used in the inclusive sense because either Laurie is a good student or Alyssa plays well or Laurie is a good student and Alyssa plays well. In mathematics and computer science, we agree to use the word “or” in the inclusive sense.

## Implication

In everyday life, we encounter statements such as “If it is cold, then I will wear a jacket,” and “If I get bonus, then I will buy a car.” Similarly, in mathematics and computer science, we frequently encounter statements such as:

1. If  $ABC$  is a triangle, then  $\angle A + \angle B + \angle C = 180^\circ$ .
2. If 49,301 is divisible by 6, then 49,301 is divisible by 3.
3. If 89,302 is divisible by 3, then it is divisible by 6.
4. If my program has no syntax errors, it will compile.

In each of these statements, two statements are connected by “if . . . then” to form a new statement.

---

**DEFINITION 1.2.12** ▶ Let  $p$  and  $q$  be two statements. Then “if  $p$ , then  $q$ ” is a statement called an **implication**, or a **condition**, written  $p \rightarrow q$ .

The statement  $p \rightarrow q$  is also to be read as

$p$  implies  $q$ ,

or

$p$  is sufficient for  $q$ ,

or

$q$  if  $p$ ,

or

$q$  whenever  $p$ .

The implication  $p \rightarrow q$  is considered false when  $p$  is true and  $q$  is false; otherwise, it is considered true.

In the implication  $p \rightarrow q$ ,  $p$  is called the *hypothesis* and  $q$  is called the *conclusion*. The truth table of the implication  $p \rightarrow q$  is given by:

$p$	$q$	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

A word of warning about the use of “if . . . then” in computer science: Many programming languages contain statements such as if  $p$ , then  $Q$ , where  $p$  is a statement and  $Q$  is a program segment consisting of one or more statements. During the execution of this kind of program, when a computer encounters such a statement,  $Q$  is executed if  $p$  is true, but nothing will be done if  $p$  is false.

---

**DEFINITION 1.2.13** ▶ Let  $p$  and  $q$  be statements.

- (i) The statement  $q \rightarrow p$  is called the **converse** of the implication  $p \rightarrow q$ .
- (ii) The statement  $\sim p \rightarrow \sim q$  is called the **inverse** of the implication  $p \rightarrow q$ .
- (iii) The statement  $\sim q \rightarrow \sim p$  is called the **contrapositive** of the implication  $p \rightarrow q$ .

**EXAMPLE 1.2.14**

Consider the statement “If today is Sunday, then I will go for a walk.” Let  $p$  and  $q$  be the following statements:

$p$  : Today is Sunday.

$q$  : I will go for a walk.

Then the given statement can be written as  $p \rightarrow q$ . The converse of this implication is  $q \rightarrow p$ , which is

$q \rightarrow p$  : If I will go for a walk, then today is Sunday.

The inverse of the above implication is  $\sim p \rightarrow \sim q$ , which is

$\sim p \rightarrow \sim q$  : If today is not Sunday, then I will not go for a walk.

The contrapositive of the above implication is  $\sim q \rightarrow \sim p$ , which is

$\sim q \rightarrow \sim p$  : If I will not go for a walk, then today is not Sunday.

## Biimplication

In the preceding section, we discussed implications. Let us now consider the following sentence: “You can get this shirt if and only if you pay for it.” Here we are saying both “If you get this shirt, then you pay for it” and “If you pay, then you can get this shirt.” That is, our sentence is a combination of two implications.

Similarly, consider the following statements.

1.  $ABCD$  is a cyclic quadrilateral if and only if  $\angle A + \angle C = 180^\circ$ .
2. 19,302 is divisible by 6 if and only if 19,302 is divisible by 2 and 3.
3. An integer  $n$  is divisible by 3 if and only if  $n$  is divisible by 9.
4. My program will compile if and only if it has no syntax errors.

We see that from the given two statements we form a new statement by putting the phrase “if and only if” between the two statements; the new statement is called the biimplication or biconditional of the given statements.

---

**DEFINITION 1.2.15** ▶ Let  $p$  and  $q$  be two statements, then “ $p$  if and only if  $q$ ,” written  $p \leftrightarrow q$ , is called the **biimplication**, or **biconditional**, of statements  $p$  and  $q$ .

The statement  $p \leftrightarrow q$  may also be read as “ $p$  is necessary and sufficient for  $q$ ,” or “ $q$  is necessary and sufficient for  $p$ ,” or “ $q$  if and only if  $p$ ,” or “ $q$  when and only when  $p$ .” We define that the biimplication  $p \leftrightarrow q$  is considered to be true when both  $p$  and  $q$  have the same truth values and false otherwise. The following table shows the truth values of the biimplication  $p \leftrightarrow q$ .

$p$	$q$	$p \leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

## Statement Formulas (Formulas)

In our discussions, we have used letters such as  $p$  and  $q$  to denote statements. From statements  $p$  and  $q$  we constructed the statements: negation  $\sim p$ , conjunction  $p \wedge q$ , disjunction  $p \vee q$ , implication  $p \rightarrow q$ , and biimplication  $p \leftrightarrow q$ .

The symbols  $\sim$ ,  $\wedge$ ,  $\vee$ ,  $\rightarrow$ , and  $\leftrightarrow$  are called **logical connectives**. Henceforth, we use the symbols  $p$ ,  $q$ ,  $r$ ,  $\dots$ , called *statement variables*, with or without subscripts to denote statements.

---

**DEFINITION 1.2.16** ► A *statement formula*, or *formula*, is defined as follows.

- (i) A statement variable is a statement formula.
- (ii) If  $A$  and  $B$  are statement formulas, then the expressions  $(\sim A)$ ,  $(A \wedge B)$ ,  $(A \vee B)$ ,  $(A \rightarrow B)$ , and  $(A \leftrightarrow B)$  are statement formulas.
- (iii) Those expressions are statement formulas that are constructed only by using (i) and (ii).

---

**REMARK 1.2.17** ► In mathematical logic, a statement formula is also called a **well-formed formula** (wff).

---

**REMARK 1.2.18** ► The logical connectives  $\vee$ ,  $\wedge$ ,  $\rightarrow$ , and  $\leftrightarrow$  can be considered as binary operators; that is, they have two operands. In an expression such as  $p \vee q$ , you can think of  $p$  and  $q$  as the operands of  $\vee$ . On the other hand  $\sim$  is a unary operator; that is, it has only one operand.

### EXAMPLE 1.2.19

- (i) Consider the expression  $((\sim(p \vee q)) \rightarrow (p \wedge r))$ .

By Definition 1.2.16(i),  $p$ ,  $q$ , are  $r$  are formulas. Next, by Definition 1.2.16(ii),  $(p \vee q)$  and  $(p \wedge r)$  are formulas. Because  $(p \vee q)$  is a formula,  $(\sim(p \vee q))$  is a formula, by Definition 1.2.16(ii). Now  $(\sim(p \vee q))$  and  $(p \wedge r)$  are formulas. So by Definition 1.2.16(ii),  $((\sim(p \vee q)) \rightarrow (p \wedge r))$  is a formula.

- (ii) As in part (i), we can show that the expressions  $((\sim(p \wedge q)) \leftrightarrow (p \wedge q))$ ,  $((\sim(p \wedge q)) \wedge r) \rightarrow (p \wedge q)$  are all statement formulas.  
 (iii) The expression  $(p \vee)$  is not a formula, because it cannot be constructed using Definition 1.2.16. (Recall that the logical connective  $\vee$  requires two operands, while in the expression  $(p \vee)$ ,  $\vee$  has only one operand.)

Similarly, the expressions  $(\sim(p \vee q)) \rightarrow (p \wedge)$  and  $(\sim(\vee)) \rightarrow (p \wedge r)$  are not statement formulas.

To avoid the use of so many parentheses in a statement formula when no confusion should arise, we adopt the following conventions:

1. We omit the outer pair of parentheses in a statement formula. For example, we write  $\sim p$  for  $(\sim p)$ .
2. If there is any statement formula of the form  $(\sim p) \rightarrow (\sim q)$ , we write it as  $\sim p \rightarrow \sim q$ .

Similarly, we write  $(\sim p) \vee (\sim q)$  as  $\sim p \vee \sim q$ ,  $(\sim p) \wedge (\sim q)$  as  $\sim p \wedge \sim q$ ,  $(\sim p) \leftrightarrow (\sim q)$  as  $\sim p \leftrightarrow \sim q$ ,  $\sim(\sim p)$  as  $\sim \sim p$ , and so on.

In a statement formula without parentheses that contains logical connectives, the logical connectives are evaluated in the following order; i.e., the precedence of logical connectives is:

- $\sim$  highest,
- $\wedge$  second highest,
- $\vee$  third highest,
- $\rightarrow$  fourth highest,
- $\Leftrightarrow$  fifth highest.

Consider a statement formula  $A$ . For every assignment of truth values  $T$  or  $F$  to the statement variables occurring in  $A$ , there corresponds by virtue of the truth tables for the logical connectives a truth value for  $A$ . Thus, we can construct a truth table for  $A$  by considering different combinations of truth values for the statement variables occurring in  $A$ . To compute the truth value of  $A$  for a given assignment of truth values to the statement variables occurring in  $A$ , we follow these rules: First compute the truth value of the statement formula within innermost parentheses, then determine the truth value of the statement formula within the next innermost set of parentheses, and continue the process until we determine the truth value for  $A$ . We show this computation in the following examples.

### EXAMPLE 1.2.20

Let  $A$  be the statement formula

$$(\sim(p \vee q)) \rightarrow (q \wedge p).$$

To construct the truth table for  $A$ , we first set up columns labeled  $p, q, (p \vee q), \sim(p \vee q), (q \wedge p)$ , and  $A$ . Write in the  $p, q$  columns all possible combinations of the truth values  $T$  and  $F$ . Then from the truth table of  $\vee$  and  $\wedge$ , we write the truth values in the columns of  $(p \vee q)$  and  $(q \wedge p)$ . Next, with the help of the truth table for  $\sim$ , we fill in the column of  $\sim(p \vee q)$ . Finally, using the truth table of  $\rightarrow$ , we fill in the column of  $A$ . Thus, we obtain the truth table for  $A$ . The truth table for  $A$  is:

$p$	$q$	$(p \vee q)$	$(\sim(p \vee q))$	$(q \wedge p)$	$A$
$T$	$T$	$T$	$F$	$T$	$T$
$T$	$F$	$T$	$F$	$F$	$T$
$F$	$T$	$T$	$F$	$F$	$T$
$F$	$F$	$F$	$T$	$F$	$F$

Consider the formula

$$A : (\sim p \wedge q) \rightarrow r.$$

The variables in this formula are  $p, q$ , and  $r$ . Now each  $p, q$ , and  $r$  can be assigned the value  $T$  or  $F$ . Suppose that the truth values of  $p, q$ , and  $r$  are  $T, F$ , and  $T$ , respectively. Then the truth value of  $\sim p$  is  $F$ , and by the truth table of  $\wedge$ , the truth value of  $\sim p \wedge q$  is  $F$ . Thus, from the truth table of  $\rightarrow$  the truth value of  $(\sim p \wedge q) \rightarrow r$  for this assignment is  $T$ . Notice that because each  $p, q$ , and  $r$  can be assigned the value  $T$  or  $F$ , there are  $2^3 = 8$  assignments of formula  $A$ . This means that to construct the truth table for  $(\sim p \wedge q) \rightarrow r$ , we have to consider all eight assignments and hence there will be eight rows for this table.

The following is the truth table for  $(\sim p \wedge q) \rightarrow r$ .

$p$	$q$	$r$	$\sim p$	$\sim p \wedge q$	$(\sim p \wedge q) \rightarrow r$
T	T	T	F	F	T
T	T	F	F	F	T
T	F	T	F	F	T
T	F	F	F	F	T
F	T	T	T	T	T
F	T	F	T	T	F
F	F	T	T	F	T
F	F	F	T	F	T

- DEFINITION 1.2.21** ► (i) A statement formula  $A$  is said to be a **tautology** if the truth value of  $A$  is  $T$  for any assignment of the truth values  $T$  and  $F$  to the statement variables occurring in  $A$ .  
(ii) A statement formula  $A$  is said to be a **contradiction** if the truth value of  $A$  is  $F$  for any assignment of the truth values  $T$  and  $F$  to the statement variables occurring in  $A$ .

**Notation 1.2.22:** For a statement formula  $A$ , we use the notation  $\models A$  to indicate that  $A$  is a tautology.

### EXAMPLE 1.2.23

Let  $A$  be the statement formula  $(\sim p \wedge q) \rightarrow (\sim(q \rightarrow p))$ . We construct the truth table for  $A$ . This statement formula contains two statement variables, so to construct the truth table of  $A$  we have to consider four different assignments of truth values. The following is the truth table of  $A$ .

$p$	$\sim p$	$q$	$(\sim p \wedge q)$	$q \rightarrow p$	$\sim(q \rightarrow p)$	$A$
T	F	T	F	T	F	T
T	F	F	F	T	F	T
F	T	F	F	T	F	T
F	T	T	T	F	T	T

From the truth table it follows that the truth value of  $A$  is  $T$  for any assignments of truth values  $T$  and  $F$  to  $p$  and  $q$ . Hence,  $A$  is a tautology.

### EXAMPLE 1.2.24

Let  $B$  be the statement formula  $\sim p \wedge p$ . We construct the truth table for  $B$ .

$p$	$\sim p$	$\sim p \wedge p$
T	F	F
F	T	F

From the table, it follows that  $B$  is a contradiction.

### DEFINITION 1.2.25

- (i) A statement formula  $A$  is said to *logically imply* a statement formula  $B$  if the statement formula  $A \rightarrow B$  is a tautology. If  $A$  logically implies  $B$ , then symbolically we write  $A \rightarrow B$ .

- (ii) A statement formula  $A$  is said to be *logically equivalent* to a statement formula  $B$  if the statement formula  $A \leftrightarrow B$  is a tautology. If  $A$  is logically equivalent to  $B$ , then symbolically we write  $A \equiv B$  (or  $A \Leftrightarrow B$ ).

**EXAMPLE 1.2.26**

Let  $A$  denote the statement formula  $p \wedge (p \rightarrow q)$  and  $B$  be  $q$ . We show that  $A$  logically implies  $B$ . For this we construct the truth table of  $A \rightarrow B$ .

$p$	$q$	$p \rightarrow q$	$p \wedge (p \rightarrow q)$	$A \rightarrow B$
T	T	T	T	T
T	F	F	F	T
F	T	T	F	T
F	F	T	F	T

From the truth table it follows that  $A \rightarrow B$  is a tautology and hence  $A$  logically implies  $B$ .

**EXAMPLE 1.2.27**

In this example, we show that the implication  $p \rightarrow q$  is equivalent to  $\sim p \vee q$ . For this we construct the truth table of  $(p \rightarrow q) \leftrightarrow (\sim p \vee q)$ .

$p$	$q$	$\sim p$	$(p \rightarrow q)$	$\sim p \vee q$	$(p \rightarrow q) \leftrightarrow (\sim p \vee q)$
T	T	F	T	T	T
T	F	F	F	F	T
F	T	T	T	T	T
F	F	T	T	T	T

From this table, it follows that  $(p \rightarrow q) \leftrightarrow (\sim p \vee q)$  is a tautology and hence  $p \rightarrow q$  is equivalent to  $\sim p \vee q$ ; that is,

$$p \rightarrow q \equiv \sim p \vee q.$$

**EXAMPLE 1.2.28**

In this example, we show that the statement formula

$$A = \sim(p \wedge q)$$

is logically equivalent to the statement formula

$$B = (\sim p) \vee (\sim q).$$

We construct the truth table for  $\sim(p \wedge q) \leftrightarrow (\sim p) \vee (\sim q)$ .

$p$	$q$	$(p \wedge q)$	$\sim(p \wedge q)$	$(\sim p)$	$(\sim q)$	$(\sim p) \vee (\sim q)$	$A \leftrightarrow B$
T	T	T	F	F	F	F	T
T	F	F	T	F	T	T	T
F	T	F	T	T	F	T	T
F	F	F	T	T	T	T	T

From the truth table, it follows that  $A \leftrightarrow B$  is a tautology and hence  $A$  is logically equivalent to  $B$ .

We now list some basic logical equivalences regarding statement formulas.

**Theorem 1.2.29:** Let  $p$ ,  $q$ , and  $r$  be statements. Then the following logical equivalences hold.

- (i) **Commutative laws:**  $p \wedge q \equiv q \wedge p$  and  $p \vee q \equiv q \vee p$
- (ii) **Associative laws:**  $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$  and  $(p \vee q) \vee r \equiv p \vee (q \vee r)$
- (iii) **Distributive laws:**  $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$  and  $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
- (iv) **Absorption laws:**  $p \wedge (p \vee q) \equiv p$  and  $p \vee (p \wedge q) \equiv p$
- (v) **Idempotent laws:**  $p \wedge p \equiv p$  and  $p \vee p \equiv p$
- (vi) **Double negation law:**  $\sim\sim p \equiv p$
- (vii) **DeMorgan's laws:**  $\sim(p \wedge q) \equiv (\sim p) \vee (\sim q)$  and  $\sim(p \vee q) \equiv (\sim p) \wedge (\sim q)$

**Proof:** We prove the first part of (iii) and the second part of (vii):

$$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$$

and

$$\sim(p \vee q) \equiv (\sim p) \wedge (\sim q).$$

We first construct the truth table of the statement formula

$$p \vee (q \wedge r) \leftrightarrow (p \vee q) \wedge (p \vee r).$$

Let us write  $A = p \vee (q \wedge r) \leftrightarrow (p \vee q) \wedge (p \vee r)$ . We have

$p$	$q$	$r$	$q \wedge r$	$p \vee (q \wedge r)$	$p \vee q$	$p \vee r$	$(p \vee q) \wedge (p \vee r)$	$A$
T	T	T	T	T	T	T	T	T
T	T	F	F	T	T	T	T	T
T	F	T	F	T	T	T	T	T
T	F	F	F	T	T	T	T	T
F	T	T	T	T	T	T	T	T
F	T	F	F	F	T	F	F	T
F	F	T	F	F	F	T	F	T
F	F	F	F	F	F	F	F	T

From the table, it follows that the statement formula

$$p \vee (q \wedge r) \leftrightarrow (p \vee q) \wedge (p \vee r)$$

is a tautology. Hence,

$$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r).$$

Next we construct the truth value of the statement formula

$$\sim(p \vee q) \leftrightarrow (\sim p) \wedge (\sim q).$$

Now,

$p$	$q$	$p \vee q$	$\sim(p \vee q)$	$\sim p$	$\sim q$	$(\sim p) \wedge (\sim q)$	$\sim(p \vee q) \leftrightarrow (\sim p) \wedge (\sim q)$
T	T	T	F	F	F	F	T
T	F	T	F	F	T	F	T
F	T	T	F	T	F	F	T
F	F	F	T	T	T	T	T

From the table, it follows that

$$\sim(p \vee q) \leftrightarrow (\sim p) \wedge (\sim q)$$

is a tautology. Hence,

$$\sim(p \vee q) \equiv (\sim p) \wedge (\sim q). \blacksquare$$

**Theorem 1.2.30:** Let  $A$  and  $B$  be two statement formulas. If  $\models (A \rightarrow B)$  and  $\models A$ , then  $\models B$ . That is if  $A \rightarrow B$  and  $A$  are tautologies, then  $B$  is a tautology.

**Proof:** Let  $\models (A \rightarrow B)$  and  $\models A$ . We show that the truth value of  $B$  is  $T$  for any assignment of the truth values to statement variables occurring in  $B$ . To establish this, we use a proof technique known as proof by contradiction. For this, we assume that the conclusion is false and then arrive at a contradiction.

Suppose for some assignment of the truth values to the statement variables of  $B$  the truth value of  $B$  is  $F$ . Now the truth value of  $A$  is always  $T$ . Hence, for any assignment of truth values to the statement variables of  $A \rightarrow B$  that contain the above assignment of truth values to  $B$ , the truth value of  $(A \rightarrow B)$  is  $F$ . This contradicts the assumption that  $\models (A \rightarrow B)$ . ■

We leave the proof of the following theorem as an exercise.

**Theorem 1.2.31:** Let  $A$  and  $B$  be two statement formulas. If  $\models (A \leftrightarrow B)$  and  $\models (B \leftrightarrow C)$ , then  $\models (A \leftrightarrow C)$ .

**Theorem 1.2.32: Principle of Substitution.** If a statement formula  $A$  is a tautology containing statement letters  $p_1, p_2, \dots, p_n$  and statement  $B$  is obtained from statement  $A$  by substituting statement formulas  $A_1, A_2, \dots, A_n$  for  $p_1, p_2, \dots, p_n$ , respectively, then statement  $B$  is also a tautology.

**Proof:** Let  $\models A$ . To show that  $B$  is a tautology, we consider an arbitrary assignment of truth values to the statement letters of  $B$ . For this assignment, let the truth values of  $A_1, A_2, \dots, A_n$  be  $x_1, x_2, \dots, x_n$ , respectively, where each  $x_1, x_2, \dots, x_n$  is either  $T$  or  $F$ . Now if we assign the truth values  $x_1, x_2, \dots, x_n$  to  $p_1, p_2, \dots, p_n$ , respectively, in  $A$ , then the resulting truth value of  $A$  is the same as the truth value of  $B$ . Now the truth value of  $A$  is  $T$  for the above assignment since  $A$  is a tautology. Thus, the truth value of  $B$  is  $T$ . Hence, we find that the truth value of  $B$  is  $T$  for any

assignment of the truth values to the statement letters occurring in  $B$ . Therefore,  $B$  is a tautology. ■

Let  $A$  and  $B$  be statement formulas such that either both are tautologies or both are contradictions. Then  $A \equiv B$ .

Suppose  $A$  is a tautology and  $B$  is an arbitrary statement formula. Then it can be proved that

$$\begin{aligned} A \vee B &\equiv A, \\ A \wedge B &\equiv B, \end{aligned}$$

which we also write as

$$\begin{aligned} T \vee B &\equiv T, \\ T \wedge B &\equiv B. \end{aligned}$$

Now suppose  $A$  is a contradiction and  $B$  is an arbitrary statement formula. Then it can be proved that

$$\begin{aligned} A \vee B &\equiv B, \\ A \wedge B &\equiv A, \end{aligned}$$

which we also write as

$$\begin{aligned} F \vee B &\equiv B, \\ F \wedge B &\equiv F. \end{aligned}$$

**REMARK 1.2.33** ▶ To verify that a formula is a tautology, we have constructed its truth table. However, there are situations where the formula can be simplified using the results of Theorem 1.2.29 and other proven results and it can be shown that it is a tautology or contradiction without constructing the truth table. This is especially useful if we have a large number of variables in the formula.

For example, consider the formula  $(\sim p \wedge q) \rightarrow (\sim(q \rightarrow p))$  of Example 1.2.23. Let us simplify using the results of Theorem 1.2.29 and other proven results. We have

$$\begin{aligned} &(\sim p \wedge q) \rightarrow (\sim(q \rightarrow p)) \\ &\equiv \sim(\sim p \wedge q) \vee (\sim(q \rightarrow p)) && \text{by Example 1.2.27} \\ &\equiv (\sim\sim p \vee \sim q) \vee (\sim(q \rightarrow p)) && \text{by DeMorgan's law} \\ &\equiv (p \vee \sim q) \vee (\sim(q \rightarrow p)) && \text{by the double negation law} \\ &\equiv (p \vee \sim q) \vee (\sim(\sim q \vee p)) && \text{by Example 1.2.27} \\ &\equiv (p \vee \sim q) \vee (\sim\sim q \wedge \sim p) && \text{by DeMorgan's law} \\ &\equiv (p \vee \sim q) \vee (q \wedge \sim p) && \text{by the double negation law} \\ &\equiv p \vee (\sim q \vee (q \wedge \sim p)) && \text{by associativity} \\ &\equiv p \vee ((\sim q \vee q) \wedge (\sim q \vee \sim p)) && \text{by distributivity} \\ &\equiv p \vee (T \wedge (\sim q \vee \sim p)) && \text{because } \sim q \vee q \equiv T \\ &\equiv p \vee (\sim q \vee \sim p) && \text{because } T \wedge (\sim q \vee \sim p) \equiv (\sim q \vee \sim p) \\ &\equiv (p \vee \sim q) \vee \sim p && \text{by associativity} \\ &\equiv (\sim q \vee p) \vee \sim p && \text{by commutativity} \\ &\equiv \sim q \vee (p \vee \sim p) && \text{by associativity} \\ &\equiv \sim q \vee T && \text{because } p \vee \sim p \equiv T \\ &\equiv T && \text{because } \sim q \vee T \equiv T \end{aligned}$$

This shows that the formula  $(\sim p \wedge q) \rightarrow (\sim(q \rightarrow p))$  is always true and hence a tautology.

## WORKED-OUT EXERCISES

**Exercise 1:** Which of the following are statements?

- (a) 2 is an even integer.
- (b) On January 6, 1986, the temperature of Omaha dropped below freezing.
- (c) Why should we study mathematics?
- (d) There is an integer  $x$  such that  $x^2 = 3$ .
- (e)  $7 + 3 = 11$ .
- (f) New Delhi is the capital of India.
- (g) Please be quiet.
- (h) Dogs can fly.
- (i) There will be snow in December.

**Solution:**

- (a) This is a declarative sentence and it is also true. Hence, it is a statement.
- (b) This is a declarative sentence and it is either true or false but not both. Hence, it is a statement.
- (c) This is not a declarative sentence. Hence, it is not a statement.
- (d) This is a declarative sentence and because there is no integer  $x$  such that  $x^2 = 3$ , we find that the given sentence is false. Hence, the given sentence is a statement.
- (e)  $7 + 3 = 11$  is a declarative sentence, which is also false. Hence, it is a statement.
- (f) This is a declarative sentence, which is also true. Hence, it is a statement.
- (g) This is not a declarative sentence. Hence, it is not a statement.
- (h) This is a declarative sentence, which is also false. Hence, it is a statement.
- (i) This is a declarative sentence and it is either true or false but not both. Hence, it is a statement.

**Exercise 2:** Write the negation of each of the following statements.

- (a) 7 is an even integer.
- (b)  $5 + 8 > 3$ .
- (c) It is hot.

**Solution:**

- (a) 7 is not an even integer.
- (b)  $5 + 8 \not> 3$ . This can also be written as  $5 + 8 \leq 3$ .
- (c) It is not hot.

**Exercise 3:** Determine the truth value of each of the following statements.

- (a) It is not the case that 5 is an even integer.

- (b)  $5 < 7$  and  $\sqrt{7}$  is a real number.
- (c)  $9 \geq 11$  and 45 is a multiple of 5.
- (d)  $7 < 5$  or 3 is a prime integer.
- (e) 5 times 7 is 15 or  $9 - 7 = 5$ .
- (f) If 2 is an odd integer, then Andy is in New York. (Moreover, write the converse and inverse of this statement.)
- (g) If Lisa is in New York, then 2 is an even integer. (Moreover, write the converse and inverse of this statement.)

**Solution:**

- (a) Let  $p$  denote the statement: 5 is an even integer. Then the given statement is  $\sim p$ . Now the truth value of  $p$  is  $F$ . Hence, the truth value of the given statement is  $T$ .
- (b) Let  $p$  denote the statement:  $5 < 7$  and  $q$  denote the statement:  $\sqrt{7}$  is a real number. Hence, the given statement is  $p \wedge q$ . Now both  $p$  and  $q$  are true. Hence, the truth value of the given statement is  $T$ .
- (c) Let  $p : 9 \geq 11$  and  $q : 45$  is a multiple of 5. Then the given statement is  $p \wedge q$ . Here  $p$  is false and  $q$  is true and  $F \wedge T = F$ . Hence, the truth value of the given statement is  $F$ .
- (d) Let  $p : 7 < 5$  and  $q : 3$  is a prime integer. Then the given statement is  $p \vee q$ . Here  $p$  is false and  $q$  is true and  $F \vee T = T$ . Hence, the truth value of the given statement is  $T$ .
- (e) Let  $p$  be the statement: 5 times 7 is 15, and  $q$  denote the statement:  $9 - 7 = 5$ . Thus, the given statement is  $p \vee q$ . Now both  $p$  and  $q$  are false and  $F \vee F = F$ . Hence, the truth value of the statement  $p \vee q = F \vee F$  is  $F$ .
- (f) Let  $p : 2$  is an odd integer, and  $q : \text{Andy is in New York}$ . Hence, the given statement is  $p \rightarrow q$ . Here  $p$  is false. Hence, the truth value of the given statement is  $T$  because  $F \rightarrow T$  is  $T$  and  $F \rightarrow F$  is  $T$ .

The converse of this implication is " $q \rightarrow p$ ," which is "If Andy is in New York, then 2 is an odd integer." The inverse of the implication is " $\sim p \rightarrow \sim q$ ," which is "If 2 is not an even integer, then Andy is not in New York."

- (g) Let  $p$  denote the statement: Lisa is in New York, and  $q$  denote the statement: 2 is an even integer. Hence, the given statement is  $p \rightarrow q$ . Here the  $q$  statement is true. Hence, the truth value of the given statement is  $T$ .

The converse of this implication is " $q \rightarrow p$ ," which is "If 2 is an even integer, then Lisa is in New York." The inverse of the implication is " $\sim p \rightarrow \sim q$ ," which is "If Lisa is not in New York, then 2 is not an even integer."

**Exercise 4:** In the following exercise let

- $p : 2$  is an even integer,
- $q : -3$  is a negative integer.

Write each of the following sentences in terms of  $p$ ,  $q$ , and logical connectives, and find the truth values of the given statements.

- 2 is an even integer and  $-3$  is a negative integer.
- 2 is not an even integer and  $-3$  is a negative integer.
- 2 is not an even integer and  $-3$  is not a negative integer.
- If 2 is not an even integer, then  $-3$  is not a negative integer.
- If 2 is an even integer, then  $-3$  is not a negative integer.
- 2 is an even integer if and only if  $-3$  is a negative integer.
- 2 is an not an even integer if and only if  $-3$  is a negative integer.

### Solution:

- $p \wedge q$ . Here the truth value of  $p$  is  $T$  and the truth value of  $q$  is  $T$ . Therefore, the truth value of  $p \wedge q$  is  $T$ .
- $\sim p \wedge q$ . Here the truth value of  $p$  is  $T$  and so the truth value of  $\sim p$  is  $F$ . Also the truth value of  $q$  is  $T$ . Therefore, the truth value of  $\sim p \wedge q$  is  $F$ .
- $\sim p \wedge (\sim q)$ . Here the truth value of  $p$  is  $T$  and so the truth value of  $\sim p$  is  $F$ . Also the truth value of  $q$  is  $T$  and so the truth value of  $\sim q$  is  $F$ . Hence, the truth value of  $\sim p \wedge (\sim q)$  is  $F$ .
- $\sim p \rightarrow (\sim q)$ . Here the truth value of  $p$  is  $T$  and so the truth value of  $\sim p$  is  $F$ . Also the truth value of  $q$  is  $T$  and so the truth value of  $\sim q$  is  $F$ . Hence, the truth value of  $\sim p \rightarrow (\sim q)$  is  $T$ .
- $p \rightarrow (\sim q)$ . Here the truth value of  $p$  is  $T$ . The truth value of  $q$  is  $T$  and so the truth value of  $\sim q$  is  $F$ . Hence, the truth value of  $p \rightarrow (\sim q)$  is  $F$ .
- $p \leftrightarrow q$ . Here the truth value of  $p$  is  $T$ , and the truth value of  $q$  is  $T$ . Therefore, the truth value of  $p \leftrightarrow q$  is  $T$ .
- $\sim p \leftrightarrow q$ . Here the truth value of  $p$  is  $T$  and so the truth value of  $\sim p$  is  $F$ . Also the truth value of  $q$  is  $T$ . Hence, the truth value of  $\sim p \leftrightarrow q$  is  $F$ .

**Exercise 5:** Construct the truth tables of the following formulas.

- $\sim p \wedge q$
- $\sim(p \vee q) \rightarrow q$
- $\sim(\sim p \wedge q) \vee q$
- $(p \rightarrow q) \rightarrow (\sim q \rightarrow \sim p)$

### Solution:

- The following is the truth table for  $(\sim p) \wedge q$ .

$p$	$q$	$\sim p$	$\sim p \wedge q$
$T$	$T$	$F$	$F$
$T$	$F$	$F$	$F$
$F$	$T$	$T$	$T$
$F$	$F$	$T$	$F$

- The truth table of  $\sim(p \vee q) \rightarrow q$  is the following.

$p$	$q$	$p \vee q$	$\sim(p \vee q)$	$\sim(p \vee q) \rightarrow q$
$T$	$T$	$T$	$F$	$T$
$T$	$F$	$T$	$F$	$T$
$F$	$T$	$T$	$F$	$T$
$F$	$F$	$F$	$T$	$F$

- The truth table of  $\sim(\sim p \wedge q) \vee q$  is the following.

$p$	$q$	$\sim p$	$\sim p \wedge q$	$\sim(\sim p \wedge q)$	$\sim(\sim p \wedge q) \vee q$
$T$	$T$	$F$	$F$	$T$	$T$
$T$	$F$	$F$	$F$	$T$	$T$
$F$	$T$	$T$	$F$	$F$	$T$
$F$	$F$	$T$	$F$	$T$	$T$

- The truth table of  $(p \rightarrow q) \rightarrow (\sim q \rightarrow \sim p)$  is the following.

$p$	$q$	$\sim p$	$\sim q$	$p \rightarrow q$	$\sim q \rightarrow \sim p$	$(p \rightarrow q) \rightarrow (\sim q \rightarrow \sim p)$
$T$	$T$	$F$	$F$	$T$	$T$	$T$
$T$	$F$	$F$	$T$	$F$	$F$	$T$
$F$	$T$	$T$	$F$	$T$	$T$	$T$
$F$	$F$	$T$	$T$	$T$	$T$	$T$

**Exercise 6:** From the truth table in Worked-Out Exercise 5(d), we find the statement formula  $(p \rightarrow q) \rightarrow (\sim q \rightarrow \sim p)$  is a tautology. Prove this without constructing the truth table.

**Solution:** We have

$$\begin{aligned}
 & (p \rightarrow q) \rightarrow (\sim q \rightarrow \sim p) \\
 & \equiv (\sim p \vee q) \rightarrow (\sim \sim q \vee \sim p) \quad \text{by Example 1.2.27, } p \rightarrow q \equiv \sim p \vee q, \text{ and, similarly,} \\
 & \quad \sim q \rightarrow \sim p \equiv \sim \sim q \vee \sim p \\
 & \equiv (\sim p \vee q) \rightarrow (q \vee \sim p) \quad \text{because } \sim \sim q \equiv q \\
 & \equiv \sim(\sim p \vee q) \vee (q \vee \sim p) \quad \text{by applying the result of} \\
 & \quad \text{Example 1.2.27} \\
 & \equiv (\sim \sim p \wedge \sim q) \vee (q \vee \sim p) \quad \text{because } \sim(\sim p \vee q) = \sim \sim p \wedge \sim q \\
 & \quad \text{(DeMorgan's law)} \\
 & \equiv (p \wedge \sim q) \vee (q \vee \sim p) \quad \text{because } \sim \sim p \equiv p \\
 & \equiv ((p \wedge \sim q) \vee q) \vee \sim p \quad \text{by the associativity of } \vee \\
 & \equiv ((p \vee q) \wedge (\sim q \vee q)) \vee \sim p \quad \text{by distributivity} \\
 & \equiv ((p \vee q) \wedge T) \vee \sim p \quad \text{because } \sim q \vee q = T \\
 & \equiv (p \vee q) \vee \sim p \quad \text{because } (p \vee q) \wedge T \equiv p \vee q \\
 & \equiv \sim p \vee (p \vee q) \quad \text{by the commutativity of } \vee \\
 & \equiv (\sim p \vee p) \vee q \quad \text{by the associativity of } \vee \\
 & \equiv T \vee q \quad \text{because } \sim p \vee p = T \\
 & \equiv T \quad \text{because } T \vee q = T
 \end{aligned}$$

## SECTION REVIEW

---

### Key Terms

statement	condition	converse
proposition	biimplication	inverse
truth value	biconditional	contrapositive
negation	logical connectives	statement formula
conjunction	well-formed formulas	formula
disjunction	tautology	
implication	contradiction	

### Some Key Definitions

1. A statement or a proposition is a declarative sentence that is either true or false, but not both.
2. A statement formula or formula is defined as follows:
  - (i) Any statement variable is a statement formula.
  - (ii) If  $A$  and  $B$  are statement formulas, then the expressions  $(\sim A)$ ,  $(A \wedge B)$ ,  $(A \vee B)$ ,  $(A \rightarrow B)$ , and  $(A \leftrightarrow B)$  are statement formulas.
  - (iii) Only those expressions are statement formulas which are constructed only by using (i) and (ii).
3. A statement formula  $A$  is said to be a tautology if the truth value of  $A$  is  $T$  for any assignment of the truth values  $T$  and  $F$  to the statement variables occurring in  $A$ .
4. A statement formula  $A$  is said to be a contradiction if the truth value of  $A$  is  $F$  for any assignment of the truth values  $T$  and  $F$  to the statement variables occurring in  $A$ .
5. A statement formula  $A$  is said to logically imply a statement formula  $B$  if the statement formula  $A \rightarrow B$  is a tautology. If  $A$  logically implies  $B$ , then symbolically we write  $A \rightarrow B$ .
6. A statement formula  $A$  is said to be logically equivalent to a statement formula  $B$  if the statement formula  $A \leftrightarrow B$  is a tautology. If  $A$  is logically equivalent to  $B$ , then symbolically we write  $A \equiv B$  (or  $A \leftrightarrow B$ ).

### Some Key Results

1. Let  $p$ ,  $q$ , and  $r$  be statements. Then the following logical equivalences hold.
  - (i)  $p \wedge q \equiv q \wedge p$  and  $p \vee q \equiv q \vee p$
  - (ii)  $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$  and  $(p \vee q) \vee r \equiv p \vee (q \vee r)$
  - (iii)  $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$  and  $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
  - (iv)  $p \wedge (p \vee q) \equiv p$  and  $p \vee (p \wedge q) \equiv p$
  - (v)  $p \wedge p \equiv p$  and  $p \vee p \equiv p$
  - (vi)  $\sim \sim p \equiv p$

(vii) (DeMorgan's laws)  $\sim(p \wedge q) \equiv (\sim p) \vee (\sim q)$  and  
 $\sim(p \vee q) \equiv (\sim p) \wedge (\sim q)$ .

2. Let  $A$  and  $B$  be two statement formulas. If  $\models (A \rightarrow B)$  and  $\models A$ , then  $\models B$ . That is, if  $A \rightarrow B$  and  $A$  are tautologies, then  $B$  is a tautology.
3. Let  $A$  and  $B$  be two statement formulas. If  $\models A \leftrightarrow B$  and  $\models B \leftrightarrow C$ , then  $\models A \leftrightarrow C$ .

## EXERCISES

---

1. Which of the following are statements?
  - a. 5 is an odd integer.
  - b. There is an integer  $x$  such that  $x^2 = 2$ .
  - c.  $5 + 6 = 11$ .
  - d.  $\emptyset \subseteq \{1\}$ .
  - e. Every integer is a real number.
2. Which of the following are statements?
  - a. Every set can be represented in computer memory.
  - b. Today is Sunday.
  - c. Why should we study computer science?
  - d. Boston is the capital of the United States.
  - e. Please do this work.
  - f. There will be a rainy day in September.
3. Write the negation of each of the following statements.
  - a. 13 is an even integer.
  - b.  $5 + 8 < 18$ .
  - c. This flower is beautiful.
4. Suppose  $x$  is a particular real number. Let  $p, q, r$  denote the statements  $2 < x$ ,  $x = 5$ , and  $x < 5$ , respectively. Write the following inequalities by  $p, q, r$  and logical connectives.
  - a.  $x \leq 5$
  - b.  $2 < x < 5$
  - c.  $2 < x \leq 5$
5. Use DeMorgan's laws to write negation of the statements of Exercise 4.
6. Determine the truth value of each of the following statements.
  - a. It is not the case that 15 is a multiple of 7.
  - b. Either today is Monday or  $\sqrt{7}$  is a real number.
  - c.  $72 > 15$  and 33 is a prime integer.
  - d.  $19 - 4 = 15$  or today's temperature is below freezing.
  - e. If Mickey is in Florida, then 17 is an odd integer.
7. Construct the truth table for each of the following statement formulas.
  - a.  $(\sim p \vee q) \wedge p$
  - b.  $(\sim p \wedge q) \rightarrow p$
8. Construct the truth table for each of the following statement formulas.
  - a.  $(p \rightarrow q \wedge r) \vee (\sim p)$
  - b.  $(p \vee q) \leftrightarrow (q \rightarrow r)$
  - c.  $(p \rightarrow r) \leftrightarrow (q \rightarrow r)$
9. Show by truth table that the following statement formulas are tautologies.
  - a.  $\sim p \rightarrow (p \rightarrow q)$
  - b.  $(p \wedge q) \rightarrow (p \rightarrow q)$
  - c.  $(p \wedge (p \rightarrow q)) \rightarrow q$
  - d.  $(p \wedge \sim p) \rightarrow q$
10. Show without constructing the truth table that the formulas in Exercise 9 are tautologies.
11. Prove without constructing the truth table that  $p \wedge (\sim q \rightarrow \sim p) \rightarrow q$  is a tautology.
12. In Worked-Out Exercise 5(c) of this section we used a truth table to show that  $\sim(\sim p \wedge q) \vee q$  is a tautology. Prove this without constructing the truth table.
13. Prove that  $((p \rightarrow q) \wedge \sim q) \rightarrow \sim p$  is a tautology.
14. Prove or disprove that  $(p \wedge q) \rightarrow (p \vee q)$  is a tautology.
15. Prove or disprove that  $(p \vee q) \rightarrow (p \wedge q)$  is a tautology.
16. Show by truth table that the following statement formula is a tautology:  

$$(p \rightarrow q) \leftrightarrow (\sim p \vee q).$$
17. Show by truth table that the following statement formula is a tautology:  

$$((p \rightarrow q) \wedge (q \rightarrow r)) \rightarrow (p \rightarrow r).$$
18. Show that the formula  $(p \rightarrow q) \rightarrow (\sim q \rightarrow \sim p)$  is a tautology.
19. Find a formula  $A$  that uses the variables  $p$  and  $q$  such that  $A$  is a contradiction.
20. Find a formula  $A$  that uses the variables  $p, q$ , and  $r$  such that  $A$  is a contradiction.
21. Find a formula  $A$  that uses the variables  $p, q$ , and  $r$  such that  $A$  is a tautology.
22. Find a formula  $A$  that uses the variables  $p$  and  $q$  such that  $A$  is true only when exactly one of  $p$  and  $q$  is true.
23. In the following exercises, show that statement formula  $A$  logically implies statement formula  $B$ .
  - a.  $A : \sim(p \rightarrow q)$ ,  $B : p \wedge (\sim q)$
  - b.  $A : \sim p$ ,  $B : (p \rightarrow q)$
  - c.  $A : \sim q \wedge (p \wedge q)$ ,  $B : \sim p$
  - d.  $A : q$ ,  $B : p \vee q$
  - e.  $A : p \rightarrow (q \rightarrow p)$ ,  $B : \sim q$
24. Prove the following.
  - a.  $p \rightarrow q$  and  $\sim q \rightarrow \sim p$  are logically equivalent.
  - b.  $p \leftrightarrow q$  and  $\sim q \leftrightarrow \sim p$  are logically equivalent.

25. In the following exercise, show that statement formula  $A$  logically implies statement formula  $B$ .

$$A : p \rightarrow (q \rightarrow r), B : (p \wedge q) \rightarrow r$$

26. Show that statement formula  $A$  is logically equivalent to statement formula  $B$ , where  $A : (p \leftrightarrow q)$ ,  $B : (p \rightarrow q) \wedge (q \rightarrow p)$ .  
 27. Show that formula  $A : (p \rightarrow q) \wedge (p \rightarrow r)$  is logically equivalent to formula  $B : p \rightarrow (q \wedge r)$ .

28. Show that formula  $A : (p \rightarrow q) \vee (p \rightarrow r)$  is logically equivalent to formula  $B : p \rightarrow (q \vee r)$ .  
 29. Show that formulas  $A : (p \rightarrow q) \rightarrow r$  and  $B : p \rightarrow (q \rightarrow r)$  are not logically equivalent.  
 30. Prove the remaining parts of Theorem 1.2.29.  
 31. Prove that  $\sim(p \vee q \vee r) \equiv \sim p \wedge \sim q \wedge \sim r$ .  
 32. Prove that  $\sim(p \wedge q \wedge r) \equiv \sim p \vee \sim q \vee \sim r$ .  
 33. Prove or disprove that  $(p \rightarrow q) \vee (\sim p \rightarrow q)$  and  $q$  are logically equivalent.

## 1.3 VALIDITY OF ARGUMENTS

Consider the following argument and conclusion: If Sheila solved seven problems correctly, then Sheila obtained the grade A. Sheila solved seven problems correctly. Therefore, Sheila obtained the grade A.

For a more complex example, consider the following argument: If my checkbook is on my office table, then I paid my phone bill. I was looking at the phone bill for payment at breakfast or I was looking at the phone bill for payment in my office. If I was looking at the phone bill at breakfast, then the checkbook is on the breakfast table. I did not pay my phone bill. If I was looking at the phone bill in my office, then the checkbook is on my office table. Where was my checkbook?

Similarly, in mathematics, we encounter arguments such as: If  $x$  is a positive even integer, it is divisible by 2.  $x$  is an even integer and  $x$  is not prime. Therefore,  $x$  is divisible by 4. Such a set of statements is called a **proof**.

In mathematics, an argument or a proof of a theorem consists of a finite sequence of statements ending in a conclusion. In this section, we define what is meant by a logically valid argument and discuss how to determine the validity of arguments.

---

**DEFINITION 1.3.1** ▶ A finite sequence  $A_1, A_2, A_3, \dots, A_{n-1}, A_n$  of statements is called an **argument**. The final statement  $A_n$  is the **conclusion**, and the statements  $A_1, A_2, A_3, \dots, A_{n-1}$  are called the **premises** of the argument.

An argument  $A_1, A_2, A_3, \dots, A_{n-1}, A_n$  is called *logically valid* if the statement formula

$$(A_1 \wedge A_2 \wedge A_3 \wedge \cdots \wedge A_{n-1}) \rightarrow A_n$$

is a tautology.

Sometimes we write an argument in the following form

$$\begin{array}{c} A_1 \\ A_2 \\ A_3 \\ \vdots \\ A_{n-1} \\ \therefore A_n. \end{array}$$

To test the logical validity of an argument written in a natural language, we first write each of the premises and the conclusion with the help of statement

letters and logical connectives. Then we check whether the conjunction

$$A_1 \wedge A_2 \wedge A_3 \wedge \cdots \wedge A_{n-1}$$

logically implies  $A_n$ . If it does, then the argument is logically valid, otherwise not. We consider the following example to explain this.

**EXAMPLE 1.3.2**

Consider the following argument. If Sheila solved seven problems correctly, then Sheila obtained the grade A. Sheila solved seven problems correctly. Therefore, Sheila obtained the grade A.

We first write the arguments by statement letters and logical connectives. Let

$p$  : Sheila solved seven problems correctly.

$q$  : Sheila obtained the grade A.

So the argument takes the form

$$\begin{aligned} p &\rightarrow q \\ p \\ \therefore q. \end{aligned}$$

Now consider the truth table for the statement formula  $((p \rightarrow q) \wedge p) \rightarrow q$ .

$p$	$q$	$p \rightarrow q$	$(p \rightarrow q) \wedge p$	$((p \rightarrow q) \wedge p) \rightarrow q$
T	T	T	T	T
T	F	F	F	T
F	T	T	F	T
F	F	T	F	T

Because  $((p \rightarrow q) \wedge p) \rightarrow q$  is a tautology, it follows that the given argument is valid.

We now consider another example.

**EXAMPLE 1.3.3**

If Peter solved seven problems correctly, then Peter obtained the grade A. Peter obtained the grade A. Therefore, Peter solved seven problems correctly. To check the validity of this argument, we first write the arguments by statement letters and logical connectives. Let

$p$  : Peter solved seven problems correctly.

$q$  : Peter obtained the grade A.

So the argument takes the form

$$\begin{aligned} p &\rightarrow q \\ q \\ \therefore p. \end{aligned}$$

Now consider the truth table for the statement formula  $((p \rightarrow q) \wedge q) \rightarrow p$ .

$p$	$q$	$p \rightarrow q$	$(p \rightarrow q) \wedge q$	$((p \rightarrow q) \wedge q) \rightarrow p$
T	T	T	T	T
T	F	F	F	T
F	T	T	T	F
F	F	T	F	T

Because  $((p \rightarrow q) \wedge q) \rightarrow p$  is not a tautology, it follows that the given argument is not valid.

In testing the validity of an argument  $A_1, A_2, A_3, \dots, A_{n-1}, A_n$ , we do not show that the truth value of  $A_n$  is true, we show only that

$$(A_1 \wedge A_2 \wedge A_3 \wedge \dots \wedge A_{n-1}) \rightarrow A_n$$

is a tautology; i.e., we verify that the truth value of the statement formula

$$(A_1 \wedge A_2 \wedge A_3 \wedge \dots \wedge A_{n-1}) \rightarrow A_n$$

is  $T$  for all possible assignments of truth values to the statement letters occurring in the above statement formula. Therefore, if the above statement formula contains  $n$  statement letters, we have to construct a truth table of  $2^n$  rows (and also the table may consist of  $m$  columns, where  $m$  may be a large number). So it may take time to check the validity by the truth table. Let us discuss another method.

One thing that is clear is that the statement formula

$$(A_1 \wedge A_2 \wedge A_3 \wedge \dots \wedge A_{n-1}) \rightarrow A_n$$

is not a tautology if there is an assignment of truth values to the statement letters such that the truth value of

$$A_1 \wedge A_2 \wedge A_3 \wedge \dots \wedge A_{n-1}$$

is  $T$  and the truth value of  $A_n$  is  $F$ . If we can show that there is no such assignment, then we can conclude that the argument is valid. Let us explain this with the help of the following example.

#### EXAMPLE 1.3.4

Consider the formula  $((p \rightarrow q) \wedge p) \rightarrow q$ . Suppose there is an assignment of truth values to  $p$  and  $q$  such that  $q$  is  $F$  and  $(p \rightarrow q) \wedge p$  is  $T$ . Now from the truth table of the connective  $\wedge$  it follows that the truth value of both  $(p \rightarrow q)$  and  $p$  is  $T$ . Now  $p$  is  $T$  and  $p \rightarrow q$  is  $T$ . Thus, from the truth table of  $p \rightarrow q$  and the fact that  $p$  is  $T$ , it follows that  $q$  is  $T$ . So for this assignment, the truth value of  $q$  is  $T$  as well as  $F$ . This is impossible because  $q$  is a statement, and so for any assignment of the truth values  $q$  is either  $T$  or  $F$ , but not both. Hence, there is no such assignment, so  $((p \rightarrow q) \wedge p) \rightarrow q$  is a tautology.

## Some Valid Argument Forms

Let us now list some useful, valid argument forms.

1. Consider the following argument form.

$$\begin{array}{c} p \rightarrow q \\ p \\ \therefore q \end{array}$$

We proved in Example 1.3.2 that  $((p \rightarrow q) \wedge p) \rightarrow q$  is a tautology, so this is a valid argument form. This argument form is called **modus ponens**. The Latin meaning of modus ponens is *method of affirming*.

2. We now consider the following argument form.

$$\begin{array}{c} p \rightarrow q \\ \sim q \\ \therefore \sim p \end{array}$$

We can show that the statement formula  $((p \rightarrow q) \wedge (\sim q)) \rightarrow (\sim p)$  is a tautology (see Exercise 13). Hence, the above argument form is valid. This argument form is called **modus tollens**. The Latin meaning of modus tollens is *method of denying*.

3. Next we consider the following argument forms.

$$\begin{array}{ll} \text{a. } & p \vee q \\ & \sim p \\ & \therefore q \\ \text{b. } & p \vee q \\ & \sim q \\ & \therefore \sim p \end{array}$$

These two argument forms are also valid forms. They are called **disjunctive syllogisms**.

4. The following argument form is also a valid argument form.

$$\begin{array}{c} p \rightarrow q \\ q \rightarrow r \\ \therefore p \rightarrow r \end{array}$$

This is called **hypothetical syllogism**.

5. Consider the following argument form.

$$\begin{array}{c} p \vee q \\ p \rightarrow r \\ q \rightarrow r \\ \therefore r \end{array}$$

We can verify that this is also a valid argument form. It is called **dilemma**.

6. Consider the following two argument forms.

$$\begin{array}{ll} \text{a. } & p \wedge q \\ & \therefore p \\ \text{b. } & p \wedge q \\ & \therefore q \end{array}$$

These two forms are valid argument forms. They are called the **conjunctive simplifications**.

7. Consider the following two argument forms.

$$\begin{array}{ll} \text{a. } & p \\ & \therefore p \vee q \\ \text{b. } & q \\ & \therefore p \vee q \end{array}$$

These two forms are valid argument forms. They are called the **disjunctive additions**.

8. The following argument form is also a valid argument form.

$$\begin{array}{c} p \\ q \\ \therefore p \wedge q \end{array}$$

This is called the **conjunctive addition**.

**DEFINITION 1.3.5** ► A statement formula  $A$  is said to *follow logically from the statement formulas  $A_1, A_2, A_3, \dots, A_{n-1}, A_n$* , written as

$$A_1, A_2, A_3, \dots, A_{n-1}, A_n \models A$$

if there exists an argument  $B_1, B_2, B_3, \dots, B_{m-1}, B_m$  satisfying the following conditions:

1.  $B_m$  is  $A$ .
2. For  $1 \leq i \leq m$ , either
  - (i)  $B_i$  is one of  $A_1, A_2, A_3, \dots, A_{n-1}, A_n$  (we say  $B_i$  is a hypothesis), or
  - (ii)  $B_i$  is a tautology, or
  - (iii) for  $i \geq 2$ , there exist  $B_{i_1}, B_{i_2}, B_{i_3}, \dots, B_{i_t}$ , where  $\{i_1, i_2, i_3, \dots, i_t\} \subseteq \{1, 2, 3, \dots, i-1, i\}$  such that  $B_{i_1}, B_{i_2}, B_{i_3}, \dots, B_{i_t}$  is a logically valid argument form and  $B_{i_t}$  is  $B_i$ ; i.e.,

$$B_{i_1} \wedge B_{i_2} \wedge B_{i_3} \wedge \cdots \wedge B_{i_{t-1}} \rightarrow B_{i_t}$$

is a tautology.

### EXAMPLE 1.3.6

In this example, we show that  $p, q, p \rightarrow r, q \rightarrow s \models r \wedge s$ . For this we write the following argument.

$B_1 : p \rightarrow r$	hypothesis
$B_2 : p$	hypothesis
$B_3 : r$	$B_1, B_2, B_3$ is a logically valid argument, by modus ponens; sometimes we write $B_3$ follows from $B_2$ and $B_1$ , by modus ponens.
$B_4 : q \rightarrow s$	hypothesis
$B_5 : q$	hypothesis
$B_6 : s$	$B_4, B_5, B_6$ is a logically valid argument, by modus ponens.
$B_7 : r \wedge s$	$B_3, B_6, B_7$ is a logically valid argument, by conjunctive addition.

Thus, we find that there exists an argument  $B_1, B_2, B_3, B_4, B_5, B_6, B_7$  satisfying the conditions of the Definition 1.3.5. Hence,  $p, q, p \rightarrow r, q \rightarrow s \models r \wedge s$ .

### EXAMPLE 1.3.7

Consider the following statements.

- (i) If my checkbook is on my office table, then I paid my phone bill.
- (ii) I was looking at the phone bill for payment at breakfast or I was looking at the phone bill for payment in my office.
- (iii) If I was looking at the phone bill at breakfast, then the checkbook is on the breakfast table.

- (iv) I did not pay my phone bill.  
 (v) If I was looking at the phone bill in my office, then the checkbook is on my office table.

Where was my checkbook?

Let

$p$  : My checkbook is on my office table.

$q$  : I paid my phone bill.

$r$  : I was looking at the phone bill for payment at breakfast.

$s$  : I was looking at the phone bill for payment in my office.

$t$  : The checkbook is on the breakfast table.

$s$  : I was looking at the phone bill in my office.

Hence, in symbolic notation, the given argument takes the form

$$p \rightarrow q$$

$$r \vee s$$

$$r \rightarrow t$$

$$\sim q$$

$$s \rightarrow p.$$

We now consider the following argument.

$$B_1 : s \rightarrow p \quad \text{hypothesis}$$

$$B_2 : p \rightarrow q \quad \text{hypothesis}$$

$$B_3 : s \rightarrow q \quad \text{from } B_1, B_2, \text{ and by hypothetical syllogism}$$

$$B_4 : \sim q \quad \text{hypothesis}$$

$$B_5 : \sim s \quad \text{from } B_3, B_4, \text{ and by modus tollens}$$

$$B_6 : r \vee s \quad \text{hypothesis}$$

$$B_7 : r \quad \text{from } B_5, B_6 \text{ and by disjunctive syllogism}$$

$$B_8 : r \rightarrow t \quad \text{hypothesis}$$

$$B_9 : t \quad \text{from } B_7, B_8, \text{ and by modus ponens}$$

Conclusion: The checkbook was on the breakfast table.



## WORKED-OUT EXERCISES

**Exercise 1:** Test whether the following argument is valid:  
 If 10,836 is divisible by 12, then 10,836 is divisible by 3. If 10,836 is divisible by 3, then the sum of the digits of 10,836 is divisible by 3. Therefore, if 10,836 is divisible by 12, then the sum of the digits of 10,836 is divisible by 3.

**Solution:** To check the validity of the argument we symbolize it using statement letters. Let

$p$  : 10,836 is divisible by 12.

$q$  : 10,836 is divisible by 3.

$r$  : The sum of the digits 10,836 is divisible by 3.

Then the whole argument may be symbolized as

$$p \rightarrow q$$

$$q \rightarrow r$$

$$\therefore p \rightarrow r.$$

The result here follows from hypothetical syllogism. However, let us construct the truth table of the statement formula  $(p \rightarrow q) \wedge (q \rightarrow r) \rightarrow (p \rightarrow r)$ . (Let us write  $A = (p \rightarrow q) \wedge (q \rightarrow r) \rightarrow (p \rightarrow r)$ .)

$p$	$q$	$r$	$p \rightarrow q$	$q \rightarrow r$	$p \rightarrow r$	$(p \rightarrow q) \wedge (q \rightarrow r)$	$A$
$T$	$T$	$T$	$T$	$T$	$T$	$T$	$T$
$T$	$T$	$F$	$T$	$F$	$F$	$F$	$T$
$T$	$F$	$T$	$F$	$T$	$T$	$F$	$T$
$T$	$F$	$F$	$F$	$T$	$F$	$F$	$T$
$F$	$T$	$T$	$T$	$T$	$T$	$T$	$T$
$F$	$T$	$F$	$T$	$F$	$T$	$F$	$T$
$F$	$F$	$T$	$T$	$T$	$T$	$T$	$T$
$F$	$F$	$F$	$T$	$T$	$T$	$T$	$T$

statements as follows:

$p$  : I (will) go to my office tomorrow.

$q$  : I must get up before 7 A.M..

$r$  : I (will) attend the dinner party at the club.

$s$  : I return home late.

$t$  : I (will) sleep well.

The whole argument may be symbolized as

$$(p \rightarrow q) \wedge (r \rightarrow s)$$

$$(s \wedge q) \rightarrow (\sim t)$$

$$t$$

$$\therefore (\sim p) \vee (\sim r).$$

Hence, the statement formula

$$(p \rightarrow q) \wedge (q \rightarrow r) \rightarrow (p \rightarrow r)$$

is a tautology. So it follows that the given argument is valid.

**Exercise 2:** Use a truth table to determine whether the following argument form is valid.

$$\begin{aligned} & p \vee q \\ & p \rightarrow r \\ & q \rightarrow r \\ \therefore & r \end{aligned}$$

**Solution:** We construct the truth table for the statement formula

$$A = (p \vee q) \wedge (p \rightarrow r) \wedge (q \rightarrow r) \rightarrow r.$$

$p$	$q$	$r$	$p \vee q$	$p \rightarrow r$	$(p \vee q) \wedge (p \rightarrow r)$	$q \rightarrow r$	$(p \vee q) \wedge (p \rightarrow r) \wedge (q \rightarrow r)$	$A$
$T$	$T$	$T$	$T$	$T$	$T$	$T$	$T$	$T$
$T$	$T$	$F$	$T$	$F$	$F$	$F$	$F$	$T$
$T$	$F$	$T$	$T$	$T$	$T$	$T$	$T$	$T$
$T$	$F$	$F$	$F$	$F$	$F$	$T$	$F$	$T$
$F$	$T$	$T$	$T$	$T$	$T$	$T$	$T$	$T$
$F$	$T$	$F$	$T$	$T$	$T$	$F$	$F$	$T$
$F$	$F$	$T$	$T$	$F$	$F$	$T$	$F$	$T$
$F$	$F$	$F$	$T$	$F$	$F$	$T$	$F$	$T$

Because statement formula  $A$  is a tautology, the given argument form is valid.

**Exercise 3:** Test whether the following argument is valid:

If I go to my office tomorrow, then I must get up before 7 A.M., and if I attend the dinner party at the club, I will return home late. If I return home late and get up before 7 A.M., I will not sleep well. I want to sleep well. Therefore, either I will not go to the office or I will not attend the dinner party.

**Solution:** To check the validity of the argument we symbolize it using statement letters. Let  $p$ ,  $q$ ,  $r$ ,  $s$ , and  $t$  be the

Now assume that there is an assignment of truth values to  $p$ ,  $q$ ,  $r$ ,  $s$ , and  $t$  such that the truth value of each of the statement formulas  $(p \rightarrow q) \wedge (r \rightarrow s)$ ,  $(s \wedge q) \rightarrow (\sim t)$ , and  $t$  is  $T$ , but the truth value of  $(\sim p) \vee (\sim r)$  is  $F$ . Then for this assignment each of  $\sim p$  and  $\sim r$  has the truth value  $F$ , and the truth value of  $t$  is  $T$ . This implies that for this assignment each of  $p$ ,  $r$ , and  $t$  has the truth value  $T$ . Because the truth value of  $(p \rightarrow q) \wedge (r \rightarrow s)$  is  $T$ , it follows that each of  $q$  and  $s$  has the truth value  $T$ . Then the truth value of  $(s \wedge q) \rightarrow (\sim t)$  is  $F$ .

So there is no assignment of truth values to  $p$ ,  $q$ ,  $r$ ,  $s$ , and  $t$  such that the truth value of each of the statement formulas  $(p \rightarrow q) \wedge (r \rightarrow s)$ ,  $(s \wedge q) \rightarrow (\sim t)$ , and  $t$  is  $T$ , but the truth value of  $(\sim p) \vee (\sim r)$  is  $F$ . Therefore, the given argument is valid.

**Exercise 4: Requires Calculus.** Test whether the following argument is valid for the three real-valued functions  $f$ ,  $g$ ,  $h$ .

If  $f$  is integrable, then  $g$  or  $h$  is differentiable. If  $g$  is not differentiable, then  $f$  is not integrable, but it is bounded. If  $f$  is bounded, then either  $g$  or  $h$  is differentiable. Hence,  $g$  is differentiable.

**Solution:** To check the validity of the argument we symbolize it using statement letters.

- $p : f$  is integrable.  
 $q : g$  is differentiable.  
 $r : h$  is differentiable.  
 $s : f$  is bounded.

The whole argument may be symbolized as

$$\begin{aligned} p &\rightarrow (q \vee r) \\ \sim q &\rightarrow (\sim p \wedge s) \\ s &\rightarrow (q \vee r) \\ \therefore q. \end{aligned}$$

Now assume that there is an assignment of truth values to  $p$ ,  $q$ ,  $r$ , and  $s$  such that the truth value of each of the statement forms  $p \rightarrow (q \vee r)$ ,  $\sim q \rightarrow (\sim p \wedge s)$ , and  $s \rightarrow (q \vee r)$  is  $T$ , but the truth value of  $q$  is  $F$ .

Because the truth value of each of  $\sim q$  and  $\sim q \rightarrow (\sim p \wedge s)$  is  $T$ , it follows that for this assignment  $(\sim p \wedge s)$  has the truth value  $T$ , so the truth value of  $\sim p$  and  $s$  is  $T$ . This implies that the truth value of  $p$  is  $F$  and the truth value of  $s$  is  $T$ . Then, from  $s \rightarrow (q \vee r)$ , we find that the truth value  $(q \vee r)$  is  $T$ . But the truth value of  $q$  is  $F$ . Thus, the truth value of  $r$  is  $T$ .

Hence, we find that there is an assignment  $F, F, T, T$  for  $p, q, r, s$ , respectively, such that the truth value of each

of the statement forms  $p \rightarrow (q \vee r)$ ,  $\sim q \rightarrow (\sim p \wedge s)$ , and  $s \rightarrow (q \vee r)$  is  $T$ , but the truth value of  $q$  is  $F$ . Therefore, the given argument is not valid.

**Exercise 5:** Show that  $\sim p, (\sim q \vee p), \sim r \vee q \models \sim r$ .

**Solution:** We write the following argument:

$B_1 : \sim p$	hypothesis
$B_2 : (\sim q \vee p)$	hypothesis
$B_3 : (\sim q \vee p) \rightarrow (q \rightarrow p)$	tautology
$B_4 : q \rightarrow p$	$B_1, B_3, B_4$ is a logically valid argument, by modus ponens.
$B_5 : \sim q$	$B_1$ and $B_4$ and modus tollens
$B_6 : \sim r \vee q$	hypothesis
$B_7 : \sim r \vee q \rightarrow (r \rightarrow q)$	tautology
$B_8 : r \rightarrow q$	$B_7, B_6, B_8$ is a logically valid argument, by modus ponens.
$B_9 : \sim r$	$B_8, B_5, B_9$ is a logically valid argument, by modus tollens.

Hence, we find that there exists an argument  $B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8, B_9$  satisfying the conditions of the Definition 1.3.5. Hence,  $\sim p, (\sim q \wedge p), \sim r \vee q \models \sim r$ .

## SECTION REVIEW

### Key Terms

proof	modus tollens	disjunctive additions
argument	disjunctive syllogisms	conjunctive addition
conclusion	hypothetical syllogism	logically valid
premise	dilemma	
modus ponens	conjunctive simplifications	

### Some Key Definitions

1. A finite sequence  $A_1, A_2, A_3, \dots, A_{n-1}, A_n$  of statements is called an argument. The final statement,  $A_n$ , is the conclusion, and the statements  $A_1, A_2, A_3, \dots, A_{n-1}$  are called the premises of the argument.
2. An argument  $A_1, A_2, A_3, \dots, A_{n-1}, A_n$  is called logically valid if the statement formula

$$(A_1 \wedge A_2 \wedge A_3 \wedge \dots \wedge A_{n-1}) \rightarrow A_n$$

is a tautology.

## EXERCISES

---

1. Use a truth table to determine whether the following argument form is valid.

$$\begin{aligned} p &\rightarrow q \\ p &\rightarrow r \\ \therefore p &\rightarrow q \vee r \end{aligned}$$

2. Use a truth table to determine whether the following argument form is valid.

$$\begin{aligned} p &\rightarrow q \\ \sim(p \vee r) &\\ \therefore \sim p & \end{aligned}$$

3. Use a truth table to determine whether the following argument form is valid.

$$\begin{aligned} \sim p \vee q & \\ r \rightarrow (\sim q) & \\ \therefore p \rightarrow (\sim r) & \end{aligned}$$

4. Use a truth table to determine whether the following argument form is valid.

$$\begin{aligned} p \vee q & \\ p \rightarrow (\sim q) & \\ p \rightarrow r & \\ \therefore r & \end{aligned}$$

5. Determine whether the following argument form is valid.

$$\begin{aligned} p &\rightarrow q \\ \sim p & \\ \therefore \sim q & \end{aligned}$$

6. Prove that the following argument form is invalid.

$$\begin{aligned} p &\rightarrow q \\ q & \\ \therefore p & \end{aligned}$$

7. Test the validity of the following argument: For a particular real number  $x$ :  $x$  is positive or  $x$  is negative. If  $x$  is positive, then  $x^2 > 0$ . If  $x$  is negative, then  $x^2 > 0$ . Therefore,  $x^2 > 0$ .

*In Exercises 8–21, test whether the given arguments are logically valid.*

8. If the budget is not cut, then prices remain stable if and only if taxes will be raised. If the budget is not cut, then taxes will be raised. If prices remain stable, then

taxes will not be raised. Therefore, taxes will not be raised.

9. If Rita works hard and has talent, then she will get a good job. If she gets a good job, then she will be happy. Hence, if Rita is not happy, then she did not work hard or she does not have talent.
10. If it snows, then the streets become slippery. If the streets become slippery, then accidents happen. Accidents do not happen. Therefore, it does not snow.
11. If it rains, the prices of vegetables go up. The prices of vegetables go up. So it rains.
12. If capital investment remains unchanged, then government spending will increase or unemployment will result. If government spending does not increase, taxes can be reduced. If taxes can be reduced and capital investment remains unchanged, then unemployment will not result. Hence, government spending will increase.
13. If Chris studies, then he will pass the class test. If Chris does not play cards, then he will study. Chris did not pass in the class test. Therefore, Chris played cards.
14. If Lisa's job performance for the year is good, she will get a bonus. If she gets a bonus, she will take a vacation. If she takes a vacation, she will take a cruise. Lisa did not take a cruise. Therefore, Lisa did not get a bonus.
15. If I do all the exercises in this chapter, I will understand the material. If I understand the material, I will do well on the exam. If I do well on the exam, I will pass. I passed the exam. Therefore, I did all the exercises in the chapter.
16. During the summer Laurie will go to New York or Paris. If she goes to New York, she will not visit the Eiffel Tower. If she does not visit the Eiffel Tower, she will visit the Statue of Liberty. She did not go to Paris. Therefore, she visited the Statue of Liberty.
17. If I save money, I will buy a house. I did not buy a house. Therefore, I did not save money.
18. If interest rates go up, then the prices of houses go down. The prices of houses did not go down. Therefore, interest rates went up.
19. Shelly is a computer science major or a chemistry major. If Shelly is a chemistry major, then she must take the organic chemistry course. Therefore, Shelly is a computer science major or she must take organic chemistry.
20. Anne plays golf or Anne plays basketball. Therefore, Anne plays golf.
21. I met Brandon at our university library or I met him at the football field. If I met Brandon at the football field, then I talked about our football team. If I met Brandon at our university library, then I talked about the discrete structure course. I did not talk about the discrete structure course. Prove that I talked about our football team.

## 1.4 QUANTIFIERS AND FIRST-ORDER LOGIC

In the preceding sections, we defined and discussed basic properties of statements (also called propositions). There, we were interested only in the truth or falsity of the statement. The structure of the statement was not taken into account. The logic that we discussed in the preceding section is categorized as the **statement logic**, or **propositional logic**. More formally, in statement logic, we look at the truth and falsity of a statement. The statement is considered a single unit; its structure and composition are suppressed. There are many justified arguments whose validity cannot be tested within the framework of propositional logic. For example, consider the following argument.

Every integer is a rational number.

3 is an integer.

Therefore, 3 is a rational number.

In mathematics, this is a justified argument. In statement logic, to check the validity of this argument we symbolize it using statement letters. Let  $p$  denote “Every integer is a rational number,”  $q$  denote “3 is an integer,” and  $r$  denote “3 is a rational number.” Hence, in symbolic notation, the above argument takes the form

$$\begin{array}{c} p \\ q \\ \therefore r. \end{array}$$

Now this argument is valid if the statement formula  $(p \wedge q) \rightarrow r$  is a tautology. But according to the statement logic, this statement formula is not a tautology. For if  $(p \wedge q) \rightarrow r$  is a tautology, then for any assignment of truth values to  $p$ ,  $q$ , and  $r$ , the truth value of  $(p \wedge q) \rightarrow r$  must be  $T$ . However, if we assign  $T$  to  $p$ ,  $T$  to  $q$ , and  $F$  to  $r$ , then the truth value of  $(p \wedge q) \rightarrow r$  is  $F$ . Hence, according to the propositional logic, the argument is not valid. Once again, we point out that the validity of an argument form depends only on the structure of the sentence in terms of component sentences; it does not depend on the analysis of sentence structure along the subject predicate lines.

For example, if we analyze the sentence

“Every integer is a rational number,”

then we find that it is equivalent to the following sentence:

“For all  $x$ , if  $x$  is an integer, then  $x$  is a rational number.”

The sentences “ $x$  is an integer” and “ $x$  is a rational number” are both declarative sentences.

Next, consider the sentence

“ $x$  is an integer.”

This is a declarative sentence, but its truth or falsity depends on a particular value of  $x$ . For example, if  $x = 5$ , then the sentence is true, and if  $x = 2.5$ , then the sentence is false. It follows that the sentence “ $x$  is an integer” is not a statement in propositional logic.

We next show that if we introduce logical notions called *predicates* and *quantifiers*, then most of the everyday arguments—and most of the arguments in mathematics and computer science—can be symbolized in such a way that we can verify the validity of the arguments.

Again consider the sentence

“ $x$  is an integer.”

Let us denote this sentence by  $P(x)$ ; i.e.,

$$P(x) : x \text{ is an integer.} \quad (1.5)$$

Then  $P(5)$ ; i.e., 5 is an integer, is true and  $P(2.5)$ ; i.e., 2.5 is an integer, is false. To study the properties of such sentences, we need to extend the framework of propositional logic. The discussion of logic that follows is categorized as **first-order logic**.

Once again, consider the sentence in (1.5). There are two parts in this sentence— $x$ , the variable, and “is an integer,” the relation. We call the relation “is an integer”—the predicate, which, here, we denote by  $P$ . Moreover,  $P(x)$  is called a predicate or propositional function. Notice that there is a set of values, (in this case, say real numbers), associated with  $P(x)$ , called the domain.

---

**DEFINITION 1.4.1** ▶ Let  $x$  be a variable and  $D$  be a set;  $P(x)$  is a sentence. Then  $P(x)$  is called a **predicate** or **propositional function** with respect to the set  $D$  if for each value of  $x$  in  $D$ ,  $P(x)$  is a statement; i.e.,  $P(x)$  is true or false. Moreover,  $D$  is called the **domain** of the discourse and  $x$  is called the **free variable**.

---

**REMARK 1.4.2** ▶ In Definition 1.4.1, the predicate  $P(x)$  is also called a propositional function because for each value  $x$  in the domain  $D$ ,  $P(x)$  is either true or false. So  $P(x)$  acts as a rule that assigns to each value in the domain either the value  $T$  or the value  $F$ . Although some prefer to use the term propositional function, we prefer the term predicate as it is a commonly used terminology in mathematical logic.

### EXAMPLE 1.4.3

Consider the sentence  $P(x)$ ,

$$P(x) : x \text{ is an even integer,}$$

where the domain of the discourse is the set of integers. Then

$$P(4); \text{ i.e., } 4 \text{ is an even integer, is } T,$$

and

$$P(3); \text{ i.e., } 3 \text{ is an even integer, is } F,$$

The predicates that we considered until now involved only one variable. We can also have predicates involving two or more variables. For example, consider the following:

$$P(x, y) : x \text{ equals } y + 1.$$

Here the predicate  $P(x, y)$  involves two variables and represents the relation “equal.” Let the domain be the set of integers. Consider  $P(2, 1)$ . Here  $x = 2$  and  $y = 1$ . Because

$$x = 2 = 1 + 1 = y + 1,$$

it follows that  $P(2, 1)$  is  $T$ . Similarly, consider  $P(5, 4)$ . Here  $x = 5$  and  $y = 4$ . Because

$$x = 5 = 4 + 1 = y + 1,$$

it follows that  $P(5, 4)$  is  $T$ .

Now consider  $P(6, 4)$ . Here  $x = 6$  and  $y = 4$ . Because

$$x = 6 \neq 4 + 1 = y + 1,$$

it follows that  $P(6, 4)$  is  $F$ .

#### EXAMPLE 1.4.4

Let  $Q(x, y)$  denote the sentence

$$Q(x, y) : x^2 \text{ is greater than or equal to } y.$$

Let the domain be the set of integers. Here the predicate  $Q(x, y)$  involves two variables. Consider  $Q(2, 3)$ . Here  $x = 2$  and  $y = 3$ . Because  $x^2 = 4 > 3$ , it follows that  $Q(2, 3)$  is true. However,  $Q(2, 5)$  is false because  $2^2 = 4$  is neither greater than 5 nor equal to 5.

---

**DEFINITION 1.4.5** ▶ Let  $x_1, x_2, \dots, x_n$  be  $n$  variables. An  **$n$ -place predicate** is a sentence  $P(x_1, x_2, \dots, x_n)$  containing  $x_1, x_2, \dots, x_n$  such that on assignment of values to the variables  $x_1, x_2, \dots, x_n$  from appropriate domains, a statement results.

For example,  $Q(x, y)$  in Example 1.4.4 is a two-place predicate.

In addition to predicates, in first-order calculus, we deal with another term—*quantifiers*. There are two types of quantifiers, *universal* and *existential*. We describe them next.

Let  $P(x)$  be a predicate. Then for each value  $x$  in the domain  $P(x)$  is a statement. Certain predicates are true for each value of the domain, while others are not. Even though a  $P(x)$  is not a statement, it can be turned into a statement through a process called *quantification*.

Suppose that  $P(x)$  is a predicate with domain  $D$ . Consider the sentence

$$P_1(x) : \text{for all } x, P(x).$$

If  $P(x)$  is true at each value  $x$  of the domain  $D$ , then  $P_1(x)$  is true. However, if  $P(x)$  is false for at least one value  $x$  of the domain  $D$ , then  $P_1(x)$  is false. We therefore see that  $P_1(x)$  is a statement because it is either true or false.  $P_1(x)$  is called the universal quantification of  $P(x)$ . Notice that  $P_1(x)$  is also written as

$$P_1(x) : \text{for every } x, P(x).$$

---

**DEFINITION 1.4.6** ▶ Let  $P(x)$  be a predicate and let  $D$  be the domain of the discourse. The universal quantification of  $P(x)$  is the statement

$$\text{for all } x, P(x)$$

or

$$\text{for every } x, P(x).$$

The symbol used to denote the adjectives for all (for every) is  $\forall$ , and it is called the **universal quantifier**. Thus, in notation, the universal quantification of the

predicate  $P(x)$  is

$$\forall x P(x).$$

In fact, the symbol  $\forall$  is read as “for all or for every.”

---

**REMARK 1.4.7** ▶ Remember that for the predicate  $P(x)$  the universal quantification  $\forall x P(x)$  is a statement, so it is either true or false. That is, the value of the statement  $\forall x P(x)$  is either  $T$  or  $F$ .

**EXAMPLE 1.4.8**

Let  $P(x)$  be the predicate given by

$$P(x) : x^2 \geq x$$

and let the domain be the set of all integers. Consider the universal quantification of  $P(x)$ :

$$\forall x P(x).$$

Because for all integers  $x$ ,  $x^2 \geq x$  is true, it follows that  $P(x)$  is true for all integers  $x$ . We can now conclude that the value of the universal quantification  $\forall x P(x)$  is  $T$ .

**EXAMPLE 1.4.9**

Let  $P(x)$  be the predicate given by

$$P(x) : x \geq 3$$

and let the domain of discourse be the set of real numbers. Consider the universal quantification of  $P(x)$ :

$$\forall x P(x).$$

If we take  $x = 2$ , then the statement  $P(2)$ , i.e.,  $2 \geq 3$ , is false. Because the predicate  $P(x)$  is false when  $x$  is replaced with 2, it follows that the value of the universal quantification  $\forall x P(x)$  is  $F$ .

---

**REMARK 1.4.10** ▶ Suppose that we have a two-place predicate  $P(x, y)$ . Then the universal quantification of  $P(x, y)$  is the sentence

$$\forall x \forall y P(x, y).$$

In this case, the universal quantification  $\forall x \forall y P(x, y)$  is true if  $P(x, y)$  is true for all  $x$  and for all  $y$ ; otherwise it is false.

**EXAMPLE 1.4.11**

Let  $P(x, y)$  be the sentence  $xy > 0$  and let the domain of discourse be the set of all nonnegative real numbers. We find that  $P(x, y)$  is not true for  $x = 0$  or  $y = 0$ , but it is true when both  $x$  and  $y$  are positive. Hence  $\forall x \forall y P(x, y)$  is false.

Next we discuss the second way of quantifying the predicate  $P(x)$ .

Suppose that  $P(x)$  is a predicate with domain  $D$ . Consider the sentence

$$P_2(x) : \text{there exists } x, P(x).$$

If  $P(x)$  is true for at least one value  $x$  in the domain  $D$ , then  $P_2(x)$  is true. However, if  $P(x)$  is false for all values  $x$  in the domain  $D$ , then  $P_2(x)$  is false. We therefore

see that  $P_2(x)$  is a statement because it is either true or false.  $P_2(x)$  is called the existential quantification of  $P(x)$ .

**DEFINITION 1.4.12** ▶ Let  $P(x)$  be a predicate and let  $D$  be the domain of the discourse. The existential quantification of  $P(x)$  is the statement

$$\text{there exists } x, P(x).$$

The symbol used to denote “there exists” is  $\exists$ , and it is called the **existential quantifier**. Thus, in notation, the existential quantification of the predicate  $P(x)$  is

$$\exists x P(x).$$

In fact, the symbol  $\exists$  is read as “there exists.”

**REMARK 1.4.13** ▶ Remember that for the predicate  $P(x)$  the existential quantification  $\exists x P(x)$  is a statement, so it is either true or false. That is, the value of the statement  $\exists x P(x)$  is either  $T$  or  $F$ .

### EXAMPLE 1.4.14

Let  $P(x)$  be the predicate given by

$$P(x) : x^2 > x$$

and let the domain of discourse be the set of all real numbers. Consider the existential quantification of  $P(x)$ :

$$\exists x P(x).$$

Let  $x = 2$ . Now  $2^2 = 4 > 2$  is true, so  $P(2)$  is true. Because we have found a value in the domain at which the predicate is true, we can conclude that the value of  $\exists x P(x)$  is  $T$ . We would like to remark that  $P(1)$  is false, so the universal quantification  $\forall x P(x)$  is  $F$ .

### EXAMPLE 1.4.15

Let  $P(x)$  be the predicate given by

$$P(x) : x^2 < x$$

and let the domain of discourse be the set of all integers. Consider the existential quantification of  $P(x)$ :

$$\exists x P(x).$$

Because for all integers  $x$ ,  $x^2 \geq x$  is true, it follows that for all integers  $x$ ,  $x^2 < x$  is false. Therefore, there is no integer for which the predicate  $P(x)$  is true. Hence, it follows that the value of  $\exists x P(x)$  is  $F$ .

The variable  $x$  appearing in

$$\forall x P(x)$$

or

$$\exists x P(x)$$

is called a **bound variable**. (It is considered bounded by the quantifiers  $\forall$  and  $\exists$ .)

If  $P(x, y)$  is a sentence, then in the sentence  $\forall x P(x, y)$  only the variable  $x$  is bounded.

## Negation of Predicates

Let  $P(x)$  be the following predicate:

$$P(x) : x \text{ has taken the programming course}, \quad (1.6)$$

where the domain of discourse is the set of all students in the discrete structures course. The universal quantification of  $P(x)$  is  $\forall x P(x)$ ; i.e.,

$$\forall x P(x) : \text{Every student in discrete structures has taken the programming course.} \quad (1.7)$$

Let us consider the negation of statement (1.7), which is:

$$\sim \forall x P(x) : \text{It is not the case that every student in discrete structures has taken the programming course.}$$

This means that there exists at least one student in discrete structures who has not taken the programming course. In symbols, we write this as  $\exists x \sim P(x)$ . Thus,

$$\sim \forall x P(x) \equiv \exists x \sim P(x). \quad (1.8)$$

Next let us consider the existential quantification of  $P(x)$ , which is the statement

$$\exists x P(x) : \text{There exists a student in discrete structures who has taken the programming course.} \quad (1.9)$$

The negation of statement (1.9) is the statement

$$\sim \exists x P(x) : \text{No student in discrete structures has taken the programming course.}$$

Here we are saying that for all students  $x$  in discrete structures,  $x$  has not taken the programming course, i.e.,  $\forall x \sim P(x)$ . It now follows that

$$\sim \exists x P(x) \equiv \forall x \sim P(x). \quad (1.10)$$

The results obtained in statements (1.8) and (1.10) are true in general and are known as the generalized DeMorgan's laws. We record them in the following theorem.

**Theorem 1.4.16: Generalized DeMorgan's Laws.** Let  $P(x)$  be a predicate with domain of discourse  $D$ . Then

- (i)  $\sim \forall x P(x) \equiv \exists x \sim P(x).$
- (ii)  $\sim \exists x P(x) \equiv \forall x \sim P(x).$

**Proof:** We only prove part (i) and leave part (ii) as an exercise.

- (i) Suppose that the statement  $\sim \forall x P(x)$  is true. Then the statement  $\forall x P(x)$  is false. This means there exists some value of  $x$ , say  $a$ , in domain  $D$  such that  $P(a)$  is false. Then for this  $a$  the statement  $\sim P(a)$  is true. This implies that  $\sim P(x)$  is true for some  $x$  in domain  $D$ . Therefore, the statement  $\exists x \sim P(x)$  is true in  $D$ .

Next assume that  $\exists x \sim P(x)$  is true in  $D$ . This implies that  $\sim P(x)$  is true for some value, say  $a$ , in  $D$ . This means that  $\sim P(a)$  is true and so  $P(a)$  is false. Therefore, the statement  $\forall x P(x)$  is false. This implies that  $\sim \forall x P(x)$  is true.

It now follows that the statements  $\sim \forall x P(x)$  and  $\exists x \sim P(x)$  are logically equivalent. ■

As in the case of statement logic, next we define formulas in the predicate logic in the following way:

1. All statement formulas are considered formulas.
2. Each  $n$ ,  $n = 1, 2, \dots, n$ -place predicate  $P(x_1, x_2, \dots, x_n)$  containing the variables  $x_1, x_2, \dots, x_n$  is a formula.
3. If  $A$  and  $B$  are formulas, then the expressions  $\sim A$ ,  $A \wedge B$ ,  $A \vee B$ ,  $A \rightarrow B$ , and  $A \leftrightarrow B$  are statement formulas, where  $\sim$ ,  $\wedge$ ,  $\vee$ ,  $\rightarrow$  and  $\leftrightarrow$  are logical connectives.
4. If  $A$  is a formula and  $x$  is a variable, then  $\forall x A$  and  $\exists x A$  are formulas.
5. All formulas that are constructed using only (1), (2), (3), and (4) are considered formulas in predicate logic.

## Additional Rules of Inference

To verify the validity of a logical consequence in predicate logic, we introduce four more additional rules of inference to the rules of inference already presented in our discussion of propositional logic.

1. If the statement  $\forall x P(x)$  is assumed to be true, then  $P(a)$  is also true, where  $a$  is an arbitrary member of the domain of the discourse. This rule is called the *universal specification* (*US*).
2. If  $P(a)$  is true, where  $a$  is an arbitrary member of the domain of the discourse, then  $\forall x P(x)$  is true. This rule is called the *universal generalization* (*UG*).
3. If the statement  $\exists x P(x)$  is true, then  $P(a)$  is true, for some member of the domain of the discourse. This rule is called the *existential specification* (*ES*).
4. If  $P(a)$  is true for some member  $a$  of the domain of the discourse, then  $\exists x P(x)$  is also true. This rule is called the *existential generalization* (*EG*).

We now verify the following argument, which cannot be verified in propositional logic.

Every integer is a rational number.

3 is an integer.

Therefore, 3 is a rational number.

We translate the above argument in the following form.

For all  $x$ , if  $x$  is an integer, then  $x$  is a rational number.

3 is an integer.

Therefore, 3 is a rational number.

We now symbolize the above argument: Let

$P(x) : x$  is an integer.

$Q(x) : x$  is a rational number.

So we can write the above argument in the following form:

$$\begin{aligned} & \forall x (P(x) \rightarrow Q(x)) \\ & P(3) \\ & \text{Therefore, } Q(3). \end{aligned}$$

The domain of discourse is the set of all real numbers.

To verify the validity, we now consider the following sequence of formulas.

$$\begin{array}{ll} B_1 : \forall x (P(x) \rightarrow Q(x)) & \text{hypothesis} \\ B_2 : P(3) \rightarrow Q(3) & \text{by the rule of inference US} \\ B_3 : P(3) & \text{hypothesis} \\ B_4 : Q(3) & B_2, B_3, B_4 \text{ is a logically valid argument, by modus ponens.} \end{array}$$

Hence, the given argument is a valid argument.

Consider now a statement of the form  $\forall x (P(x) \rightarrow Q(x))$ , where the domain of the discourse is  $D$ . To show that this implication is not true in the domain  $D$ , we have to show that there exists some  $x$  in  $D$  such that  $P(x) \rightarrow Q(x)$  is not true. This means that there exists some  $x$  in  $D$  such that  $P(x)$  is true but  $Q(x)$  is not true. Such an  $x$  is called a **counterexample** of the above implication. To show that  $\forall x (P(x) \rightarrow Q(x))$  is false by finding an  $x$  in  $D$  such that  $P(x) \rightarrow Q(x)$  is false is called the **disproof** of the given statement by counterexample.

### EXAMPLE 1.4.17

In the set of all real numbers, for all  $x$ , if  $x^2 < 9$ , then  $0 < x < 3$ .

Let  $P(x)$  denote  $x^2 < 9$  and  $Q(x)$  denote  $0 < x < 3$ . Hence, the given statement can be written as

$$\forall x (P(x) \rightarrow Q(x))$$

in the domain of all real numbers. Now for  $x = -2$ ,  $(-2)^2 < 9$ , but the inequality  $0 < -2 < 3$  is not true. Therefore, there exists  $x$ , which is  $-2$ , such that  $P(-2)$  is true, but  $Q(-2)$  is false. Hence,  $\forall x (P(x) \rightarrow Q(x))$  is not true in the given domain. We say that  $x = -2$  is a counterexample of the given implication.

## WORKED-OUT EXERCISES

**Exercise 1:** Symbolize the following by using quantifiers, predicates, and logical connectives.

- (a) The square of any real number is greater than or equal to zero.
- (b) Some integers are multiples of 7.
- (c) There is an integer  $x$  such that  $x^2 = 16$ .
- (d) All birds can fly.
- (e) There exists an integer such that it is even and prime.
- (f) For any integer, there exists an integer such that their sum is 0.

### Solution:

- (a) Let

$$P(x) : x \text{ is a real number.}$$

$$Q(x) : x^2 \text{ is greater than or equal to zero.}$$

Then in symbols, the given sentence takes the form  $\forall x (P(x) \rightarrow Q(x))$ .

- (b) Let

$$P(x) : x \text{ is an integer.}$$

$$Q(x) : x \text{ is multiple of 7.}$$

Then in symbols, the given sentence takes the form  $\exists x (P(x) \rightarrow Q(x))$ .

- (c) Let

$$Q(x) : x^2 = 16.$$

Then in symbols, the given sentence takes the form  $\exists x Q(x)$ . The domain of discourse is the set of integers.

- (d) Let

$$P(x) : x \text{ is a bird.}$$

$$Q(x) : x \text{ can fly.}$$

Then in symbols, the given sentence takes the form  
 $\forall x(P(x) \rightarrow Q(x))$ .

(e) Let

$P(x)$  :  $x$  is even.

$Q(x)$  :  $x$  is prime.

Then in symbols, the given sentence takes the form  
 $\exists x(P(x) \wedge Q(x))$ . The domain of discourse is the set of integers.

(f) Let  $P(x, y)$  denote the sentence:  $x + y = 0$ . Hence, in the domain of integers the given sentence may be symbolized as  $\forall x \exists y P(x, y)$ .

**Exercise 2:** In the following, use  $P(x)$  :  $x$  is an odd integer;  $Q(x)$  :  $x$  is a prime integer; and  $R(x)$  :  $x^2$  is an odd integer. Write a statement in English corresponding to each symbolic statement.

- (a)  $\forall x(P(x) \rightarrow R(x))$       b.  $\forall x(P(x) \wedge Q(x))$   
 (c)  $\exists x(P(x) \wedge Q(x))$

### Solution:

- (a) The squares of all odd integers are odd.  
 (b) All integers are odd and prime.  
 (c) Some odd integers are prime.

**Exercise 3:** Let  $P(x, y)$  denote the sentence:  $xy = x$ . What is the truth value of  $\forall x \exists y P(x, y)$ , where the domain of  $x, y$  is the set of all integers?

**Solution:** Since  $P(x, 1) : x1 = x$  and it is true for all integers  $x$ , we find that the truth value of  $\forall x \exists y P(x, y)$  is  $T$ .

**Exercise 4:** Let  $P(x, y)$  denote the sentence:  $xy = 1$ . What is the truth value of  $\forall x \exists y P(x, y)$ , where the domain of  $x, y$  is the set of all integers?

**Solution:** For  $x = 2$ ,  $P(2, y) : 2y = 1$ . There are no integers  $y$  such that  $2y = 1$ . Hence, the statement  $\forall x \exists y P(x, y)$  is false.

**Exercise 5:** Let  $P(x, y)$  denote the sentence:  $x + y = 1$ . What are the truth values of  $\forall x \exists y P(x, y)$ ,  $\forall x \forall y P(x, y)$ , and  $\exists x \exists y P(x, y)$ , where the domain of  $x$  and  $y$  is the set of all integers?

**Solution:** Let  $x$  be an integer. Then  $y = 1 - x$  is an integer such that  $x + (1 - x) = 1$ . Thus, for any integer  $x$  there exists an integer  $y = 1 - x$  such that  $x + y = 1$ . Therefore, we find that the truth value of  $\forall x \exists y P(x, y)$  is  $T$ .

To find the truth value of  $\forall x \forall y P(x, y)$ , we consider the integers 2 and 3. Because  $2 + 3 \neq 1$ , the truth value of  $\forall x \forall y P(x, y)$  is  $F$ .

We now consider the statement  $\exists x \exists y P(x, y)$ . Because  $0 + 1 = 1$ , we find that there are integers  $x$  and  $y$  such that  $x + y = 1$ . Hence, the truth value of the statement  $\exists x \exists y P(x, y)$  is  $T$ .

**Exercise 6:** Test the validity of the following argument: Some rational numbers are powers of 5. All integers are rational numbers. Therefore, some integers are powers of 5.

**Solution:** We first translate the given argument in the following form.

There exists  $x$ , if  $x$  is a rational number, then  $x$  is a power of 5.

For all  $x$ , if  $x$  is an integer, then  $x$  is a rational number. 5 is an integer.

Therefore, there exists  $x$  such that if  $x$  is a integer, then  $x$  is a power of 5.

We now symbolize the above arguments: Let

$P(x)$  :  $x$  is an integer.

$Q(x)$  :  $x$  is a rational number.

$R(x)$  :  $x$  is a power of 5.

We can write the above argument in the following form:

$\exists x(Q(x) \rightarrow R(x))$

$\forall x(P(x) \rightarrow Q(x))$

$P(5)$

Therefore,  $\exists x(P(x) \rightarrow R(x))$ .

To verify the validity we now consider the following sequence of formulas.

$B_1 : \exists x(Q(x) \rightarrow R(x))$  hypothesis

$B_2 : Q(a) \rightarrow R(a)$  for some member of the domain  
the set of rational numbers, by  
the rule of inference ES

$B_3 : \forall x(P(x) \rightarrow Q(x))$  hypothesis

$B_4 : P(a) \rightarrow Q(a)$  by the rule of inference US

$B_5 : P(a) \rightarrow R(a)$   $B_4, B_2, B_5$  is a logically valid argument,  
by hypothetical syllogism.

Therefore,

$B_6 : \exists x(P(x) \rightarrow R(x))$  by the rule of inference EG

Hence, the consequence is valid.

## SECTION REVIEW

### Key Terms

statement logic

predicate

domain

propositional logic

propositional function

free variable

$n$ -place predicate	bound variable	first-order logic
universal quantifier	counterexample	
existential quantifier	disproof	

## Some Key Definitions

- Let  $x$  be a variable and let  $D$  be a set.  $P(x)$  is a sentence. Then  $P(x)$  is called a predicate or propositional function with respect to the set  $D$  if each value of  $x$  in  $D$ ,  $P(x)$  is a statement; i.e.,  $P(x)$  is true or false. Moreover,  $D$  is called the domain of the discourse and  $x$  is called the free variable.
- Let  $P(x)$  be a predicate and let  $D$  be the domain of the discourse. The universal quantification of  $P(x)$  is the statement for all  $x$ ,  $P(x)$  or for every  $x$ ,  $P(x)$ . In symbols, the universal quantification of the predicate  $P(x)$  is written as  $\forall x P(x)$ .
- Let  $P(x)$  be a predicate and let  $D$  be the domain of the discourse. The existential quantification of  $P(x)$  is the statement there exists  $x P(x)$ . In symbols, the existential quantification of the predicate  $P(x)$  is written as  $\exists x P(x)$ .

## Some Key Results

- Let  $P(x)$  be a predicate with domain of discourse  $D$ . Then
  - $\sim \forall x P(x) \equiv \exists x \sim P(x)$ .
  - $\sim \exists x P(x) \equiv \forall x \sim P(x)$ .

## EXERCISES

---

- Symbolize the following by using quantifiers, predicates, and logical connectives.
  - All integers are rational numbers.
  - Some rational numbers are integers.
  - All positive integers are multiples of 5.
  - Some rectangles are square.
  - For all integers  $n$ ,  $2n + 1$  is an odd integer.
  - Every integer is either odd or even.
  - Every integer is a multiple of 6 if and only if it is a multiple of both 3 and 2.
  - There is no integer  $n$  such that  $n^2$  is 5.
- In parts (a)–(d), use  $P(x)$ :  $x$  is an integer,  $Q(x)$ :  $x$  is a rational number, and  $R(x)$ :  $x$  is a prime integer. Write a statement in English corresponding to the following symbolic statements.
 

a. $P(5)$	b. $\forall x \sim P(x)$
c. $\exists x R(x)$	d. $\exists x \sim Q(x)$
- What is the universal quantification of the sentence:  $x^2 + x$  is an even integer, where  $x$  is an odd integer? Is the universal quantification a true statement?
- What is the existential quantification of the sentence:  $x$  is a prime integer, where  $x$  is an odd integer? Is the existential quantification a true statement?
- What is the existential quantification of the sentence:  $x < 0$ , where  $x$  is an integer? Is the existential quantification a true statement?
- What is the truth value of the quantification  $\forall x P(x)$ ? The domain of the discourse is the set of all positive integers.
  - $P(x)$ :  $(x + 1)(x + 2)$  is an even integer.
  - $P(x)$ :  $x + 1 > x$
  - $P(x)$ :  $x + 2 > 5$
  - $P(x)$ :  $x^2 + 2 = 3$
- What is the truth value of the quantification  $\exists x P(x)$ ? The domain of the discourse is the set of all real numbers.
  - $P(x)$ :  $x + 1 = 1$
  - $P(x)$ :  $x^3 + 1 < x$
  - $P(x)$ :  $x \cdot \frac{1}{2} = 1$
  - $P(x)$ :  $x^2 + 2x + 1 < 0$

9. Let  $P(x, y)$  denote the sentence:  $x + y = 7$ . What are the truth values of  $\forall x \exists y P(x, y)$ ,  $\forall x \forall y P(x, y)$ , and  $\exists x \exists y P(x, y)$ , where the domain of  $x, y$  is the set of all integers?
10. Let  $P(x, y)$  denote the sentence:  $2x + y = 1$ . What are the truth values of  $\forall x \exists y P(x, y)$ ,  $\forall x \forall y P(x, y)$ , and  $\exists x \exists y P(x, y)$ , where the domain of  $x, y$  is the set of all integers?
11. Let  $P(x, y)$  denote the sentence:  $x$  divides  $y$ . What are the truth values of  $\forall x \exists y P(x, y)$ ,  $\forall x \forall y P(x, y)$ , and  $\exists x \exists y P(x, y)$ , where the domain of  $x, y$  is the set  $\{1, 2, 4, 6, 12\}$ ?
12. Symbolize the following sentences by using predicates, quantifiers, and logical connectives.
- Any finite set with  $n$  elements has  $2^n$  subsets.
  - Not all real numbers are rational numbers.
13. Symbolize the following sentences by using predicates, quantifiers, and logical connectives.
- Every computer science major takes a programming course.
  - If you buy a car, then you must pay a sales tax.
  - Some people are vegetarians.
14. Express the following using predicates, quantifiers, and logical connectives. Also verify the validity of the consequence.
- Everyone who graduates gets a job.  
Jennifer graduated.  
Therefore, Jennifer got a job.
- In Exercises 15–23, test the validity of the logical consequences.*
15. All men are mortal.  
Randy is a man.  
Therefore, Randy is mortal.
16. All birds can fly.  
A crow is a bird.  
Therefore, a crow can fly.
17. All polynomials with real coefficients are differentiable functions.  
All differentiable functions are continuous.  
Therefore, all polynomials with real coefficients are continuous.
18. Everyone who studies logic is good in reasoning.  
Lance is good in reasoning.  
Therefore, Lance studies logic.
19. All employers pay their employees.  
Juan is an employer.  
Therefore, Juan pays his employees.
20. All athletes exercise.  
Emily is an athlete.  
Therefore, Emily exercises.
21. All drivers take a driving test.  
Tom did not take the driving test.  
Therefore, Tom is not a driver.
22. All athletes are healthy.  
All healthy people take vitamins.  
Grant is an athlete.  
Therefore, Grant takes vitamins.
23. All dogs fetch.  
Kitty does not fetch.  
Therefore, Kitty is not a dog.
24. Show that  $\forall x (P(x) \wedge Q(x))$  and  $\forall x P(x) \wedge \forall x Q(x)$  are equivalent.
25. Show that  $\exists x (P(x) \vee Q(x))$  and  $\exists x P(x) \vee \exists x Q(x)$  are equivalent.
26. Show that  $\forall x (P(x) \rightarrow Q(x))$  is not equivalent to  $\forall x P(x) \rightarrow \forall x Q(x)$ .
27. Find a counterexample to show that the following propositions are false.
- $\forall x \in \mathbb{R}, x < x^2$
  - $\forall m, n \in \mathbb{Z}, m \cdot n > m + n$
28. Find a counterexample to show that the following propositions are false.
- $\forall a, b \in \mathbb{R}, \sqrt{ab} = \sqrt{a}\sqrt{b}$
  - $\forall a, b \in \mathbb{R}, \sqrt{a+b} = \sqrt{a} + \sqrt{b}$
29. Find a counterexample to show that the proposition  $\forall a, b, c \in \mathbb{R}, c \neq 0, \frac{ac+bc}{c} = a + b$  is false.

## 1.5 PROOF TECHNIQUES

In the preceding sections, we presented various ways of using logical arguments and deriving conclusions. As stated earlier, in mathematics and computer science, mathematical logic is used to prove theorems and the correctness of programs. In this section, after formally defining the term *theorem*, we describe some general

techniques that are used in proving theorems. (We already used some of these techniques in earlier sections when we proved some of the theorems.) In the next section, we discuss algorithms and programs, and in later chapters we show how to prove the correctness of an algorithm (program).

Recall that a **theorem** is a statement that can be shown to be true (under certain conditions). For example, consider the following statement:

If  $x$  is an integer and  $x$  is odd, then  $x^2$  is odd,

or, equivalently,

For all integers  $x$ , if  $x$  is odd, then  $x^2$  is odd.

This statement can be shown to be true. We will prove below that it is a true statement.

Theorems are typically stated as follows:

1. As facts. For example, 6 is an even integer. As another example, the equation  $x^2 + 1 = 0$  has no solutions in real numbers.
2. As implications. For example, for all integers  $x$ , if  $x$  is even, then  $x + 1$  is odd.
3. As biimplications. For example, for all integers  $x$ ,  $x$  is even if and only if  $x$  is divisible by 2.

A proof may consist of previously known facts, proved results, or previous statements of the proof. There are several known techniques for constructing a proof. In the remainder of this section, we illustrate some of these techniques.

## Direct Proofs

We first discuss the proof of those theorems that can be expressed in the form

$\forall x (P(x) \rightarrow Q(x))$ ,  $D$  is the domain of discourse.

For example, for all integers  $x$ , if  $x$  is even, then  $x + 1$  is odd.

To construct a proof of the theorem

$\forall x (P(x) \rightarrow Q(x))$ ,  $D$  is the domain of discourse,

we start by selecting a particular but arbitrarily chosen member  $a$  of the domain  $D$ . Then we show that the statement  $P(a) \rightarrow Q(a)$  is true. For this we assume that  $P(a)$  is true. We now show that  $Q(a)$  is true. If we do this, then by the rule of universal generalization (UG), it follows that

$\forall x (P(x) \rightarrow Q(x))$ ,  $D$  is the domain of discourse

is true. This procedure is called the *proof by direct method*, or **direct proof**.

### EXAMPLE 1.5.1

In this example, we use direct proof to prove the following theorem.

For all integers  $x$ , if  $x$  is odd, then  $x^2$  is odd.

Before writing the proof, let us verify the theorem for certain values of  $x$ . If  $x = 3$ , then  $x^2 = 9$  so  $x^2$  is odd. If  $x = 511$ , then  $x^2 = 26,1121$  so  $x^2$  is odd. Similarly, if  $x = -25$ , then  $x^2 = 625$  so  $x^2$  is odd. This gives us some indication that the given theorem is true. We want to stress here that verifying a given theorem

for a particular value is not a proof. We must still prove that the theorem is true for an arbitrary value of the domain of discourse.

Let  $P(x)$  denote “ $x$  is an odd integer” and  $Q(x)$  denote “ $x^2$  is an odd integer.” Then in symbolic notation we have following statement of the theorem.

$$\forall x (P(x) \rightarrow Q(x)), \quad \text{the domain of discourse is the set } \mathbb{Z} \text{ of all integers.}$$

We will start the proof by assuming that  $a$  is a particular but arbitrarily chosen element of  $\mathbb{Z}$ . For this  $a$ , we assume that  $P(a)$  is true and then we show that  $Q(a)$  is true.

**Proof:** Let  $a$  be an integer such that  $a$  is odd. Then we can write  $a = 2n + 1$  for some integer  $n$ . This implies that  $a^2 = (2n + 1)^2 = 4n^2 + 4n + 1 = 2(2n^2 + 2n) + 1$ . Let  $m = 2n^2 + 2n$ . Because  $n$  is an integer,  $m = 2n^2 + 2n$  is also an integer. We can therefore write  $a^2 = 2m + 1$  for some integer  $m$ . This implies that  $a^2$  is odd. This completes the proof.

Sometimes such a proof can also be written as: Let  $a$  be an odd integer. Then

$$\begin{aligned} & a \text{ is an odd integer} \\ \Rightarrow & a = 2n + 1 \quad \text{for some integer } n \\ \Rightarrow & a^2 = (2n + 1)^2 \\ \Rightarrow & a^2 = 4n^2 + 4n + 1 \\ \Rightarrow & a^2 = 2(2n^2 + 2n) + 1 \\ \Rightarrow & a^2 = 2m + 1, \quad \text{where } m = 2n^2 + 2n \text{ is an integer} \\ \Rightarrow & a^2 \text{ is an odd integer.} \end{aligned}$$

Therefore, for all integers  $x$ , if  $x$  is odd, then  $x^2$  is odd. ■

**REMARK 1.5.2** ► In Example 1.5.1, we showed two ways to write a proof. For readability, wherever possible, we will use the second form to write a proof or a portion of a proof. (The symbol  $\Rightarrow$  stands for this implies or this implies that.)

**REMARK 1.5.3** ► Notice that in Example 1.5.1, we first verified that the theorem is true for a certain value of the variable and then supplied a proof using an arbitrary value. We also stressed that verifying a statement for particular values is not a proof; it only indicates that the theorem might be true. If we are not careful, then, verifying a statement for a particular value of the variable can give a wrong indication. For example, consider the following sentence:

$$\text{for all real numbers } x, x^2 \geq x.$$

If we choose  $x$  to be a real number greater than or equal to 1 or less than or equal to  $-1$  or  $x$  to be 0, then we will find that the sentence is true. However, if we choose  $x$  to be, say  $\frac{1}{2}$ , then we find that  $x^2 = \frac{1}{4} \not\geq \frac{1}{2} = x$ . Therefore, we must be very careful when writing proofs.

#### EXAMPLE 1.5.4

In this example, we use direct proof to show that product of two odd integers is an odd integer.

The equivalent statement of this theorem is

For all integers  $x$  and  $y$ , if  $x$  and  $y$  are odd, then the product  $xy$  is odd.

To change it into symbolic notation, let  $P(x)$  denote “ $x$  is an odd integer,”  $Q(y)$  denote “ $y$  is an odd integer,” and  $R(x, y)$  denote “the product  $xy$  is an odd integer.” Then in symbolic notation we have following statement of the theorem.

$$\forall x \forall y ((P(x) \wedge Q(y)) \rightarrow R(x, y)), \quad \text{the domain of the discourse is the set } \mathbb{Z} \text{ of all integers.}$$

**Proof:** Suppose  $a$  and  $b$  are odd integers. Then  $a = 2m + 1$  and  $b = 2n + 1$  for some integers  $m$  and  $n$ . Let us evaluate  $ab$ . We have

$$\begin{aligned} ab &= (2m + 1)(2n + 1) \\ &= 4mn + 2m + 2n + 1 \\ &= 2(2mn + m + n) + 1. \end{aligned}$$

Let us write  $t = 2mn + m + n$ . Then  $t$  is an integer. Hence,  $ab = 2t + 1$  for some integer  $t$ , so  $ab$  is odd. Therefore, for all integers  $x$  and  $y$ , if  $x$  and  $y$  are odd, then the product  $xy$  is odd. ■

## Indirect Proof

Consider the implication  $p \rightarrow q$ . This implication is equivalent to the implication  $\sim q \rightarrow \sim p$ . This means that in order to show that  $p \rightarrow q$  is true, we can also show that the implication  $\sim q \rightarrow \sim p$  is true. Now to show that  $\sim q \rightarrow \sim p$  is true, we assume that the negation of  $q$  is true and prove that the negation of  $p$  is true. This type of proof is called **indirect proof**.

### EXAMPLE 1.5.5

In this example, we use indirect proof to prove the following: Let  $n$  be an integer. If  $n^2 + 3$  is odd, then  $n$  is even.

Let us express the statement using symbolic notation. For this purpose, let  $P(n)$  denote “ $n^2 + 3$  is an odd integer” and  $Q(n)$  denote “ $n$  is an even integer.” Then in symbolic notation we have following statement of the theorem.

$$\forall n (P(n) \rightarrow Q(n)), \quad \text{the domain of discourse is the set } \mathbb{Z} \text{ of all integers.}$$

**Proof:** We will start the proof by assuming that  $n$  is a particular but arbitrarily chosen element of  $\mathbb{Z}$ . For this  $n$ , we show that  $P(n) \rightarrow Q(n)$  is true. Now  $P(n) \rightarrow Q(n)$  is logically equivalent to  $\sim Q(n) \rightarrow \sim P(n)$ . Therefore, we show that  $\sim Q(n) \rightarrow \sim P(n)$  is true. Suppose  $\sim Q(n)$  is true. We show that  $\sim P(n)$  is true.

Because  $\sim Q(n)$  is true,  $n$  is not even. Then  $n$  is odd, so  $n = 2k + 1$  for some integer  $k$ . Thus,

$$\begin{aligned} n^2 + 3 &= (2k + 1)^2 + 3, && \text{substitute } n = 2k + 1 \\ &= 4k^2 + 4k + 1 + 3 \\ &= 4k^2 + 4k + 4 \\ &= 2(2k^2 + 2k + 2). \end{aligned}$$

Let us write  $t = 2k^2 + 2k + 2$ . Then  $t$  is an integer, so  $n^2 + 3 = 2t$  for some integer  $t$ . This implies that  $n^2 + 3$  is an even integer; i.e.,  $n^2 + 3$  is not an odd integer. Thus,  $\sim P(n)$  is true. We have thus shown that the implication  $\sim Q(n) \rightarrow \sim P(n)$  is true. Hence,  $P(n) \rightarrow Q(n)$ . Therefore,  $\forall n (P(n) \rightarrow Q(n))$ , in the domain  $\mathbb{Z}$  of all integers (by the rule universal generalization (UG); i.e., if  $n^2 + 3$  is odd, then  $n$  is even).

Notice that the argument that  $n^2 + 3$  is an even integer can also be written as:

$$\begin{aligned} n &= 2k + 1 \\ \Rightarrow n^2 &= (2k + 1)^2 \\ \Rightarrow n^2 &= 4k^2 + 4k + 1 \\ \Rightarrow n^2 + 3 &= 4k^2 + 4k + 1 + 3 \\ \Rightarrow n^2 + 3 &= 4k^2 + 4k + 4 \\ \Rightarrow n^2 + 3 &= 2(2k^2 + 2k + 2). \end{aligned}$$

Because  $k$  is an integer,  $t = 2k^2 + 2k + 2$  is an integer, so  $n^2 + 3 = 2t$ , which is a multiple of 2. Thus,  $n^2 + 3$  is an even integer. ■

## Proof by Contradiction

In a **proof by contradiction**, we assume that the conclusion is not true and then arrive at a contradiction.

### EXAMPLE 1.5.6

In this example, we use proof by contradiction to prove the following:

Let  $A$  and  $B$  be sets such that  $A \subseteq B$ . Then  $A \cap B = A$ .

The conclusion is that  $A \cap B = A$ . Let us deny the conclusion and assume that  $A \cap B \neq A$ . We know, by Theorem 1.1.26, that  $A \cap B \subseteq A$ . From this it follows that if  $A \cap B \neq A$ , then  $A \not\subseteq A \cap B$ . This implies that there exists  $x \in A$  such that  $x \notin A \cap B$ . This in turn implies that either  $x \notin A$  or  $x \notin B$ . However,  $x \in A$  and so  $x \notin A \cap B$  implies that  $x \notin B$ . We have therefore found an element  $x$  such that  $x \in A$  but  $x \notin B$ . This implies that  $A \not\subseteq B$ . This is a contradiction to our hypothesis. Hence, we can now conclude that if  $A \subseteq B$ , then  $A \cap B = A$ .

### EXAMPLE 1.5.7

In this example, we use proof by contradiction to show that  $\sqrt{2}$  is an irrational number.

**Proof:** Let us assume that  $\sqrt{2}$  is not an irrational number. Then  $\sqrt{2}$  is a rational number, so we can write

$$\sqrt{2} = \frac{a}{b},$$

where  $a$  and  $b$  are integers and  $b \neq 0$ . Moreover, we may assume that the fraction  $\frac{a}{b}$  is in the lowest term; that is,  $a$  and  $b$  have no common factors other than 1. (If  $a$  and  $b$  have common factors other than 1, then we can cancel those common factors and get the fraction in which the numerator and denominator have no common factors other than 1.)

Now

$$\begin{aligned} \sqrt{2} &= \frac{a}{b} \\ \Rightarrow (\sqrt{2})^2 &= \left(\frac{a}{b}\right)^2 \\ \Rightarrow 2 &= \frac{a^2}{b^2} \\ \Rightarrow a^2 &= 2b^2 \\ \Rightarrow a^2 &\text{ is an even integer.} \end{aligned}$$

Because  $a^2$  is an even integer, we must have  $a$  an even integer (because, as proven in Example 1.5.1, if  $a$  is odd, then  $a^2$  is also odd). Therefore, we can write  $a = 2n$  for some integer  $n$ . This implies that  $a^2 = 4n^2$ . We now substitute this value of  $a^2$  into  $a^2 = 2b^2$  to obtain

$$2b^2 = a^2 = 4n^2.$$

This implies that

$$b^2 = 2n^2,$$

so  $b^2$  is even. We can now conclude that  $b$  is even. Thus, we have proved that both  $a$  and  $b$  are even and therefore have 2 as a common factor. This contradicts our assumption that  $a$  and  $b$  have no common factor other than 1. We have now arrived at a contradiction. Consequently, we can conclude that  $\sqrt{2}$  is an irrational number. ■

## Proving Biimplications

There are theorems that can be expressed by using universal quantifiers and logical connective biimplication.

Consider the following theorem:

An integer  $x$  is even if and only if  $x + 1$  is odd.

If we let  $P(x) : x$  is even and  $Q(x) : x + 1$  is odd, then we are saying that, for all integers  $x$ ,  $P(x)$  if and only if  $Q(x)$  or, in symbolic notation  $\forall x (P(x) \leftrightarrow Q(x))$ , in the domain  $\mathbb{Z}$ . To prove a theorem of the form  $\forall x (P(x) \leftrightarrow Q(x))$ , where  $D$  is the domain of the discourse, we consider an arbitrary but fixed element  $a$  from  $D$ . For this  $a$ , we will prove that the biimplication  $P(a) \leftrightarrow Q(a)$  is true. So how do we prove this biimplication?

Recall that the biimplication  $p \leftrightarrow q$  is equivalent to  $(p \rightarrow q) \wedge (q \rightarrow p)$ . From this it follows to prove that  $p \leftrightarrow q$  we only need to prove that the implications  $p \rightarrow q$  and  $q \rightarrow p$  are true. Therefore, first we assume that  $p$  is true and show that  $q$  is true. Then we assume that  $q$  is true and show that  $p$  is true.

### EXAMPLE 1.5.8

In this example, we prove the theorem: An integer  $x$  is even if and only if  $x + 1$  is odd.

To do so we do the following: Assume that  $x$  is a particular but arbitrary integer such that  $x$  is even and show that  $x + 1$  is odd. Then assume that  $x + 1$  is odd and prove that  $x$  is even.

Let us first suppose that  $x$  is even. Then  $x = 2n$  for some integer  $n$ . This implies that  $x + 1 = 2n + 1$ . From this it follows that  $x + 1$  is an odd integer.

Let us now suppose that  $x + 1$  is odd. Then  $x + 1 = 2m + 1$  for some integer  $m$ . This implies that  $x = 2m$ . From this it follows that  $x$  is an even integer.

We can now conclude that an integer  $x$  is even if and only if  $x + 1$  is odd.

### EXAMPLE 1.5.9

In this example, we prove the following theorem: Let  $A$  and  $B$  be nonempty subsets of a set  $U$ . Then  $A = B$  if and only if  $A \times B = B \times A$ .

**Proof:** To prove that  $A = B$  if and only if  $A \times B = B \times A$ , we first show that if  $A = B$ , then  $A \times B = B \times A$ . Next we show that if  $A \times B = B \times A$ , then  $A = B$ .

Suppose that  $A = B$ . Then  $A \times B = A \times A$  and  $B \times A = A \times A$  (on the right side substitute  $A$  for  $B$ ). It now follows that  $A \times B = B \times A$ .

Now suppose that  $A \times B = B \times A$ . To show that  $A = B$ , we show that every element of  $A$  is an element of  $B$  and every element of  $B$  is an element of  $A$ ; i.e.,  $A \subseteq B$  and  $B \subseteq A$ .

Let  $a \in A$ . Because  $B \neq \emptyset$ , there exists  $b \in B$ . Then  $(a, b) \in A \times B$ . Because  $A \times B = B \times A$ , we have  $(a, b) \in B \times A$ . This implies that  $(a, b) = (u, v)$  for some  $u \in B$  and  $v \in A$ . Because  $(a, b) = (u, v)$ , by the equality of ordered pairs,  $a = u$  and  $b = v$ . Thus,  $a = u \in B$ . Because  $a$  is an arbitrary element of  $A$ , it follows that  $A \subseteq B$ .

We now show that  $B \subseteq A$ . Let  $y \in B$ . Because  $A \neq \emptyset$ , there exists  $x \in A$ . Then  $(y, x) \in B \times A$ . Because  $A \times B = B \times A$ , we have  $(y, x) \in A \times B$ . This implies that  $(y, x) = (s, t)$  for some  $s \in A$  and  $t \in B$ . Because  $(y, x) = (s, t)$ , by the equality of ordered pairs,  $y = s$  and  $x = t$ . Thus,  $y = s \in A$ . Because  $y$  is an arbitrary element of  $B$ , it follows that  $B \subseteq A$ .

Consequently,  $A = B$ . ■

## Proving Equivalent Statements

Consider the following statements: Let  $x$  be an integer.

$p$ :  $x$  is divisible by 6.

$q$ :  $x$  is divisible by 2 and 3.

$r$ :  $x$  is an even number and  $x$  is divisible by 3.

Here we can prove that  $p$  if and only if  $q$ ;  $p$  if and only if  $r$ , and  $q$  if and only if  $r$ . In other words, the statements  $p$ ,  $q$ , and  $r$  are equivalent statements.

Sometimes theorems are stated in the form of equivalent statements. So how do we prove a theorem that is statement in the form of equivalent statements? For example, consider the theorem that says that statements  $p$ ,  $q$ , and  $r$  are equivalent. For this, we normally show that  $p \rightarrow q$ ,  $q \rightarrow r$ , and  $r \rightarrow p$ . That is, we have a cycle  $p \rightarrow q \rightarrow r \rightarrow p$ . In other words, we assume  $p$  and prove  $q$ . Then we assume  $q$  and prove  $r$ . Finally, we assume  $r$  and prove  $p$ .

Is this the only way to prove equivalent statements? Certainly not! We can also prove that  $p$  if and only if  $q$ , and then  $q$  if and only if  $r$ . There are also other ways that can be used to prove a theorem that consists of equivalent statements. For example, we can prove that  $p$  if and only if  $r$ , and  $q$  if and only if  $r$ ; or  $p$  if and only if  $q$ , and  $p$  if and only if  $r$ .

### EXAMPLE 1.5.10

In this example, we prove the above equivalent statements: that is, let  $x$  be an integer. Then the following statements are equivalent.

- (i)  $p$ :  $x$  is divisible by 6.
- (ii)  $q$ :  $x$  is divisible by 2 and 3.
- (iii)  $r$ :  $x$  is an even number and  $x$  is divisible by 3.

**Proof:** To prove that these statements are equivalent, we show that (i) implies (ii), (ii) implies (iii), and (iii) implies (i). In symbols, (i)  $\Rightarrow$  (ii), (ii)  $\Rightarrow$  (iii), and (iii)  $\Rightarrow$  (i), which sometimes we write as (i)  $\Rightarrow$  (ii)  $\Rightarrow$  (iii)  $\Rightarrow$  (i).

(i)  $\Rightarrow$  (ii): Suppose that  $x$  is divisible by 6. Then  $x = 6n$  for some integer  $n$ .

Then  $x = 6n = 2 \cdot 3n = 3 \cdot 2n$ . From this it follows that  $x$  is divisible by 2 and 3.

(ii)  $\Rightarrow$  (iii): Suppose that  $x$  is divisible by 2 and 3. Because  $x$  is divisible by 2,  $x$  is an even integer. Hence,  $x$  is an even integer and  $x$  is divisible by 3.

(iii)  $\Rightarrow$  (i): Suppose that  $x$  is an even number and  $x$  is divisible by 3. Because  $x$  is an even integer, we have  $x = 2n$  for some integer  $n$ . This implies that 3 divides  $2n$  and thus  $2n = 3t$  for some integer  $t$ . Now

$$n = 3n - 2n.$$

This implies that

$$n = 3n - 3t,$$

that is,

$$n = 3(n - t).$$

Let us write  $n - t = s$ . Then  $s$  is an integer and we can write  $n = 3s$  for some integer  $s$ . It now follows that

$$x = 2n = 2 \cdot 3s = 6s \quad \text{for some integer } s.$$

This implies that  $x$  is divisible by 6. ■

### EXAMPLE 1.5.11

In this example, we prove that the following statements are equivalent. Let  $A$  and  $B$  be subsets of a set  $U$ . Then the following statements are equivalent.

- (i)  $A \subseteq B$
- (ii)  $A - B = \emptyset$
- (iii)  $A \cup B = B$

**Proof:** We show the following implications: (i)  $\Rightarrow$  (ii)  $\Rightarrow$  (iii)  $\Rightarrow$  (i).

(i)  $\Rightarrow$  (ii): Suppose that  $A \subseteq B$ . We prove  $A - B = \emptyset$  by contradiction. So suppose that  $A - B \neq \emptyset$ . Then there exists an element  $x$  such that  $x \in A - B$ . This implies that  $x \in A$  and  $x \notin B$ . Thus, there exists an element in  $A$  that is not in  $B$ . From this it follows that  $A \not\subseteq B$ , which is a contradiction to our hypothesis. Consequently,  $A - B = \emptyset$ .

(ii)  $\Rightarrow$  (iii): Suppose that  $A - B = \emptyset$ . We show that  $A \cup B = B$ . By Theorem 1.1.22, we have  $B \subseteq A \cup B$ . Next, we show that  $A \cup B \subseteq B$ . Let  $x \in A \cup B$ . Then  $x \in A$  or  $x \in B$ . Suppose  $x \notin B$ . Then we must have  $x \in A$ . We therefore have  $x \in A$  and  $x \notin B$ . This implies that  $x \in A - B$ , so  $A - B \neq \emptyset$ , which is a contradiction. Hence,  $x \in B$ . Because  $x$  is an arbitrary element of  $A \cup B$ , it follows that  $A \cup B \subseteq B$ . Consequently,  $A \cup B = B$ .

(iii)  $\Rightarrow$  (i): Suppose that  $A \cup B = B$ . By Theorem 1.1.22,  $A \subseteq A \cup B$ . Hence,  $A \subseteq A \cup B = B$ . ■

### Fallacies (Errors) in the Proofs

As remarked above, a proof may consist of previously known facts, proved results, or previous statements of the proof. However, if we are not careful, errors can occur in the proofs. For example, consider the following proof:

$$\begin{aligned} 1 &= \sqrt{1} \\ &= \sqrt{(-1) \cdot (-1)} \end{aligned}$$

$$\begin{aligned}
 &= \sqrt{(-1)} \cdot \sqrt{(-1)} \\
 &= (\sqrt{(-1)})^2 \\
 &= -1.
 \end{aligned}$$

We have just shown that  $1 = -1$ , which of course is not true. So where is the error? In this proof, the error is in the equality  $\sqrt{(-1)} \cdot (-1) = \sqrt{(-1)} \cdot \sqrt{(-1)}$ . We were trying to use the fact that  $\sqrt{ab} = \sqrt{a}\sqrt{b}$  for all real numbers  $a$  and  $b$ . However, the expression  $\sqrt{ab} = \sqrt{a}\sqrt{b}$  is true when  $a$  and  $b$  are nonnegative real numbers.

We have demonstrated once more that we must be careful when writing proofs.



## WORKED-OUT EXERCISES

**Exercise 1:** Let  $a$  and  $b$  be real numbers such that  $a < b$ . Then there exists a real number  $c$  such that  $a < c < b$ .

**Solution:** Let us imagine the numbers  $a$  and  $b$  on a real-number line. Because  $a < b$ , we can visualize infinitely many numbers between  $a$  and  $b$ . However, we must find at least one specific real number  $c$  such that  $a < c < b$ . If we think about the midpoint theorem from a geometry course, then we know that one such number is  $\frac{a+b}{2}$ . Let us try to establish this fact.

We prove

$$\begin{aligned}
 &a < b \\
 \Rightarrow &a + a < a + b \quad \text{Add } a \text{ to both sides.} \\
 \Rightarrow &2a < a + b \quad \text{Simplify.} \\
 \Rightarrow &a < \frac{a+b}{2} \quad \text{Divide both sides by 2.}
 \end{aligned}$$

Also:

$$\begin{aligned}
 &a < b \\
 \Rightarrow &a + b < b + b \quad \text{Add } b \text{ to both sides.} \\
 \Rightarrow &a + b < 2b \quad \text{Simplify.} \\
 \Rightarrow &\frac{a+b}{2} < b \quad \text{Divide both sides by 2.}
 \end{aligned}$$

We have thus proved that  $a < \frac{a+b}{2} < b$ . Let  $c = \frac{a+b}{2}$ . Because  $a$  and  $b$  are real numbers,  $a + b$  is a real number, so  $\frac{a+b}{2}$  is a real number; i.e.,  $c$  is a real number. Consequently, there exists a real number  $c$  such that  $a < c < b$ .

**Exercise 2:** Prove that  $\forall x \in \mathbb{Z}$ ,  $x^2 - x$  is an even integer.

**Solution:** Let  $x$  be an arbitrary but fixed element of  $\mathbb{Z}$ . We want to prove that  $x^2 - x$  is an even integer. Now  $x$  is an integer. This is a situation in which we do the proof using cases. For our problem, there are two cases: one when  $x$  is an even integer and the other is when  $x$  is an odd integer.

**Case 1:** Suppose  $x$  is even. Then  $x = 2n$  for some integer  $n$ . Thus,

$$x^2 - x = (2n)^2 - 2n = 4n^2 - 2n = 2(2n^2 - n).$$

Let us write  $m = 2n^2 - n$ . Because  $n$  is an integer,  $n^2$  is an integer and therefore  $2n^2 - n$  is an integer. That is,  $m$  is an integer. Hence,  $x^2 - x = 2m$  for some integer  $m$ , so  $x^2 - x$  is an even integer.

**Case 2:** Suppose  $x$  is odd. Then  $x = 2n + 1$  for some integer  $n$ . Thus,

$$\begin{aligned}
 x^2 - x &= (2n + 1)^2 - (2n + 1) \\
 &= (4n^2 + 4n + 1) - (2n + 1) \\
 &= 4n^2 + 4n + 1 - 2n - 1 \\
 &= 4n^2 + 2n \\
 &= 2(2n^2 + n).
 \end{aligned}$$

Let us write  $t = 2n^2 + n$ . Because  $n$  is an integer,  $n^2$  is an integer and therefore  $2n^2 + n$  is an integer. That is,  $t$  is an integer. Hence,  $x^2 - x = 2t$  for some integer  $t$ , so  $x^2 - x$  is an even integer.

**Exercise 3:** Find errors in the following proof: Let  $a$ ,  $b$ , and  $c$  be real numbers such that  $a = b + c$ . Then

$$\begin{aligned}
 &a = b + c \\
 \Rightarrow &a(a - b) &= (b + c)(a - b) &\text{Multiply both sides by } (a - b). \\
 \Rightarrow &a^2 - ab &= ab - b^2 + ac - bc &\text{Simplify.} \\
 \Rightarrow &a^2 - ab - ac &= ab - b^2 - bc &\text{Subtract } ac \text{ from both sides.} \\
 \Rightarrow &a(a - b - c) &= b(a - b - c) &\text{Take common factors out.} \\
 \Rightarrow &a = b. &&\text{Divide both sides by } (a - b - c).
 \end{aligned}$$

This shows that any two real numbers are the same. For example, suppose  $a = 7$  and  $b = 4$ . We can take  $c = 3$  and repeat this process to conclude that  $7 = 4$ .

**Solution:** The error in the proof is in the step where we divided both sides by  $a - b - c$ . Notice that because  $a = b + c$ , we have  $a - b - c = 0$ . Therefore, we are in fact dividing by 0, which of course is not valid.

## SECTION REVIEW

---

### Key Terms

theorem	indirect proof	proof by direct method
direct proof	proof by contradiction	

### Some Key Definitions

1. A theorem is a statement that can be shown to be true under certain conditions.
2. In a direct proof of the implication  $\forall x(P(x) \rightarrow Q(x))$ ,  $D$  is the domain of discourse; we start by selecting a particular but arbitrarily chosen member  $a$  of the domain  $D$ . Then we show that the statement  $P(a) \rightarrow Q(a)$  is true. For this we assume that  $P(a)$  is true. Now show that  $Q(a)$  is true.
3. In an indirect proof of the implication  $p \rightarrow q$ , we assume that the negation of  $q$  is true and prove that the negation of  $p$  is true.
4. In a proof by contradiction, we assume that the conclusion is not true and then arrive at a contradiction.

## EXERCISES

---

1. Express the statements of the following theorems by universal quantifier and logical connectives, then prove each of the theorems by direct method.
  - a. If  $n$  is an even integer, then  $n^2$  is an even integer.
  - b. If the sum of two integers is even, then their difference is even.
  - c. The sum of two odd integers is even.
2. Express the statements of the following theorems by universal quantifier and logical connectives, then prove each of the theorems by the indirect method.
  - a. If  $n^2$  is an even integer, then  $n$  is an even integer.
  - b. For all integers  $n$ , if  $5n + 2$  is odd, then  $n$  is odd.
3. Give a direct proof to show that if  $x$  and  $y$  are even integers, then  $x + y$  is an even integer.
4. Give a direct proof to show that if  $x$  is an even integer, then  $x^2 + x$  is an even integer.
5. Give a direct proof to show that the product of two consecutive integers is an even integer.
6. Prove that the sum of two consecutive integers is an odd integer.
7. Prove that the product of even integers is an even integer.
8. Let  $A$ ,  $B$ , and  $C$  be subsets of a set  $U$ . Give a direct proof to show that if  $A \subseteq B$  and  $B \subseteq C$ , then  $B - A \subseteq C - A$ .
9. Prove by contradiction that if  $n^2$  is an odd integer, then  $n$  is odd.
10. Give an indirect proof to show that if  $n$  is an even integer, then  $n + 1$  is odd.
11. Prove that  $n$  is an even integer if and only if  $5n^2 + 12$  is even.
12. Prove that  $m^2 = n^2$  if and only if  $m = n$  or  $m = -n$  for all real numbers  $m$  and  $n$ .
13. Prove that for all real numbers  $x$ , if  $x > 5$ , then  $x^2 > 25$ .
14. Prove the following theorem: For all integers  $n$ ,  $m$ , if  $n + m > 20$ , then either  $n > 10$  or  $m > 10$ .
15. Let  $x$  and  $y$  be nonzero rational numbers. Prove that  $\frac{5x+2y}{3y}$  is a rational number.
16. Prove or disprove: If  $x$  is an irrational number, then  $x^2$  is irrational.
17. Prove or disprove: The product of two irrational numbers is an irrational number.
18. Prove or disprove that the difference of two odd integers is an even integer.
19. Prove that the following statements are equivalent.
  - a.  $n$  is an odd integer.
  - b.  $5n + 4$  is an odd integer.
  - c.  $n^2$  is an odd integer.
20. Prove that the following statements are equivalent.
  - a.  $x$  is a rational number.
  - b.  $x/3$  is a rational number.
  - c.  $2x + 5$  is a rational number.
21. Prove that the following statements are equivalent. Let  $a$  and  $b$  be real numbers.
  - a.  $a < b$
  - b.  $a < \frac{a+b}{2}$
  - c.  $\frac{a+b}{2} < b$

22. Let  $A$  and  $B$  be subsets of a set  $U$ . Prove that the following statements are equivalent.
- $A \subseteq B$
  - $A \cap B = A$
  - $A \cup B = B$
23. Let  $A$  and  $B$  be subsets of a set  $U$ . Prove that the following statements are equivalent.
- $A \subseteq B$
  - $A - B = \emptyset$
  - $A \cap B = A$
24. Let  $A$ ,  $B$ , and  $C$  be subsets of a set  $U$ . Prove that the following statements are equivalent.
- $A \subseteq B$  and  $B \subseteq C$ .
  - $A \cap B \cap C = A$  and  $B \cap C = B$ .
  - $A \cup B \cup C = C$  and  $A \cup B = B$ .
25. Let  $A$  and  $B$  be subsets of a set  $U$ . Prove that the following statements are equivalent.
- $A \subseteq B$
  - $A \Delta B = B - A$
  - $A \cap B' = \emptyset$
26. Let  $x$  and  $y$  be real numbers. Prove that  $\min(x, y) + \max(x, y) = x + y$ .
27. Let  $x$  and  $y$  be real numbers. Prove that  $\min(x, y) \leq \frac{x+y}{2} \leq \max(x, y)$ .
28. Let  $x$ ,  $y$ , and  $z$  be real numbers. Prove that  $\min(x, \min(y, z)) = \min(\min(x, y), z)$ .
29. Let  $x$ ,  $y$ , and  $z$  be real numbers. Prove that  $\max(x, \max(y, z)) = \max(\max(x, y), z)$ .
30. Let  $x$  and  $y$  be real numbers. Prove that  $|xy| = |x||y|$ .
31. Let  $x$  and  $y$  be real numbers. Prove that  $|x+y| \leq |x| + |y|$ .
32. Let  $x$  and  $y$  be nonzero real numbers. Prove that  $|x+y| = |x| + |y|$  if and only if either both  $x$  and  $y$  are positive or both are negative.
33. Let  $x$  be an integer. Prove that there exists a unique integer  $y$  such that  $x + y = 0$ .
34. Let  $x$  be a nonzero real number. Prove that there exists a unique real number  $y$  such that  $xy = 1$ .
35. Find errors in the following proof that proves that if  $A$  and  $B$  are subsets of a set  $U$ , then  $A \cup B \subseteq A$ .
- Let  $A$  and  $B$  be subsets of a set  $U$ . Let  $x \in A \cup B$ . Then  $x \in A$  or  $x \in B$ . Thus,  $x \in A$ . Because  $x$  is an arbitrary element of  $A \cup B$ , it follows that  $A \cup B \subseteq A$ .
36. Find errors in the following proof that proves that if  $x$  is an even integer and  $y$  is an odd integer, then  $\frac{x+1}{y} = 1$ .
- Suppose  $x$  an even integer and  $y$  is an odd integer. Then  $x = 2n$  and  $y = 2n + 1$  for some integer  $n$ . Thus,  $\frac{x+1}{y} = \frac{2n+1}{2n+1} = 1$ .
37. Find errors in the following proof that proves that if  $p$  and  $q$  are nonzero rational numbers, then  $\frac{p}{q} = 1$ .
- Let  $p$  and  $q$  be nonzero rational numbers. Then  $p = \frac{a}{b}$  and  $q = \frac{a}{b}$  for some integers  $a \neq 0, b \neq 0$ . Thus,
- $$\frac{p}{q} = \frac{\frac{a}{b}}{\frac{a}{b}} = \frac{a}{b} \cdot \frac{b}{a} = 1.$$
38. Find errors in the following proof that proves that if  $A$ ,  $B$ , and  $C$  are subsets of a set  $U$ , then  $A \subseteq B$  and  $A \subseteq C$  implies that  $B \subseteq C$ .
- Let  $x \in A$ . Then  $x \in B$  because  $A \subseteq B$ , and  $x \in C$  because  $A \subseteq C$ . Thus,  $x \in B$  and  $x \in C$ . Hence,  $B \subseteq C$ .

## 1.6 ALGORITHMS

In a college algebra course, one learns how to solve systems of linear equations. However, in that course, most of the problems are limited to two equations and two variables or three equations and three variables. Because the number of equations and variables is small, it is possible to do all the calculations by hand using paper and pencil. However, as the number of variables and equations increases, solving the equations by hand becomes very time-consuming. Let us consider the following problem: We are asked whether the number  $1 + 2^{2^9}$  can be written as a product,  $ab$ , of integers  $a$  and  $b$  such that  $a > 1$  and  $b > 1$ . It would be extremely tedious, if not impossible, to do all the computations necessary to solve this problem by hand. Or, let us consider the problem of determining the smallest number in a list of numbers. If there are only a few numbers, we could look over the entire list and choose the smallest element. But if the list contains lots of numbers, say over a million, then checking the list one element at a time and keeping track of the smallest element would be an overwhelming task.

Fortunately, there are computer programs that can effectively solve a variety of such problems. Several professionally written software programs are available in the market, and it is our objective in this book to present algorithms to solve various problems.

First, let us define the term algorithm: An **algorithm** is a step-by-step problem-solving process in which a solution is arrived at in a finite amount of time. According to this definition, a cooking recipe is an algorithm. Or, to take another example, giving a friend directions to your house is an algorithm. The word algorithm is derived from the name of the ninth-century Persian mathematician al-Khowârizmî.

Typically, all algorithms have the following properties.

- **Input:** There is an input to the algorithm (for example, a set of numbers to find the sum of the numbers).
- **Output:** There is an output of the algorithm (for example, the sum of the numbers).
- **Precision:** Each step of the algorithm is precisely defined.
- **Uniqueness:** The results of each step of the algorithm are unique and depend on the input and the results of the previous step.
- **Finiteness:** The algorithm must terminate after executing a finite number of steps.
- **Generality:** The algorithm is general in the sense that it applies to a set of inputs.

Once an algorithm is written, the next step is to verify that it works properly. For certain algorithms, correctness can be verified by using some mathematical techniques; that is, we can prove the correctness of the algorithm. In the next chapter, we will introduce the proof technique *mathematical induction*, which is very useful in determining the correctness of algorithms. After introducing mathematical induction for certain algorithms, we will establish its correctness. For the remainder of this section, we discuss some well-known algorithms and introduce the syntax used to describe algorithms.

Let us consider the problem of determining the smallest number from a list of three numbers, say  $a, b, c$ . We can accomplish this by using the following algorithm.

1. Let  $x := a$ .
2. If  $x > b$ , then  $x := b$ .
3. If  $x > c$ , then  $x := c$ .

In this algorithm, at step 1 we assume that  $a$  is the smallest number and copy its value into  $x$ . In step 2, we compare  $x$  with  $b$ , and if  $x > b$ , we copy the value of  $b$  into  $x$ . Next, in step 3, the new value of  $x$  is compared with  $c$ , and if  $c$  is less than the current value of  $x$ , we copy the value of  $c$  into  $x$ . After step 3,  $x$  contains the smallest number.



**Muhammed ibn Mûsâ al-Khowârizmî**  
(c. 800–850 A.D.)

Little is known of the life of al-Khowârizmî, though his contributions to the study of mathematics are beyond measure. Al-Khowârizmî was

### HISTORICAL NOTES

born in a part of Central Asia currently known as Uzbekistan. He moved to Baghdad to work as a mathematics and astronomy scholar at the House of Wisdom in the court of al-Mamun.

There he wrote about the use of base-10 numerals. Owing to a Latin translation of his work, credit for this

invention was mistakenly attributed to Arabic rather than Indian sources. Al-Khowârizmî's writings also included a book called *Al-jabr w'al muqabala*, which gave algebra its name and established the basic tenets of algebra still practiced to this day.

The algorithm for determining the smallest of three numbers satisfies the properties of an algorithm. The input to the program is three numbers. The output is the smallest of three numbers. There are three steps in this algorithm, as described in the preceding paragraph. Each step is either an assignment or a comparison and assignment, so each step is defined precisely. Step 1 is defined using the input (the value of  $a$ ), step 2 depends on step 1 and the input (the value of  $b$ ), step 3 depends on step 2 and the input (the value of  $c$ ). It follows that each step is defined uniquely. Because each step is either an assignment or a comparison and (possibly) an assignment, it follows that the algorithm terminates after executing a finite number of steps. Next, because  $a$ ,  $b$ , and  $c$  can be any numbers, it follows that the algorithm is general.

## Pseudocode Conventions

To describe algorithms, we use pseudocodes, the syntax of which is described in this section.

The symbol  $:=$  is called the **assignment operator**. The statement  $x := a$  copies the value of  $a$  into  $x$ ; in fact, this statement replaces the old value of  $x$  with the value of  $a$ . The statement  $x := a$  is read as “ $x$  gets the value of  $a$ ,” or “assign the value of  $a$  to  $x$ ,” or “copy the value of  $a$  into  $x$ .” This statement is called the **assignment statement**.

The general form of the assignment statement is

```
var := expression;
```

The expression is evaluated and its value is assigned to the var.

## Control Structures

We use the following syntax to describe selection and repetition control structures.

**One way-selection** takes the form:

```
if booleanExpression then
    statement
```

If booleanExpression evaluates to true, statement is evaluated.

**Two-way selection** takes the following form:

```
if booleanExpression then
    statement1
else
    statement2
```

If booleanExpression evaluates to true, statement1 executes, otherwise statement2 executes.

The while loop takes the form:

```
while booleanExpression do
    loopBody
```

The booleanExpression is evaluated. If it evaluates to true, loopBody executes. There after loopBody continues to execute as long as booleanExpression is true.

The for loop takes the form:

```
for var := start to limit do
    loopBody
```

where `var` is an integer variable. The variable `var` is set to the value specified by `start`. If `var ≤ limit`, `loopBody` executes. After executing the `loopBody`, `var` is incremented by 1. The statement continues to execute until `var > limit`.

The `do/while` loop takes the form:

```
do
    loopBody
while booleanExpression;
```

The `loopBody` is executed first and then the `booleanExpression` is evaluated. The `loopBody` continues to execute as long as the `booleanExpression` is true.

### Block of Statement

To consider a set of statements a single statement, we write the statements between the words `begin` and `end`.

```
begin
    statement1
    statement2
    :
    statementn;
end;
```

### Return Statement

The return statement is used to return the value computed by the algorithm and it takes the following form:

```
return expression;
```

The value specified by `expression` is returned. Notice that in an algorithm, the execution of a `return` statement also terminates the algorithm.

### Arrays (List)

A list is a set of elements of the same type. The length of the list is the number of elements in the list. A convenient way to store and manipulate a list is by using an array. We will use the following notation to denote arrays:

`L[1...n]`—`L` is an array of `n` components, indexed 1 to `n`. `L[i]` denotes the `i`th element of `L`.

If we are dealing with data given in a tabular form, we use a two-dimensional array to store such data. We use the following notation to denote a **two-dimensional array**:

`M[1...m, 1...n]`—`M` is a two-dimensional array of `m` rows and `n` columns. The rows are indexed 1 to `m` and the columns are indexed 1 to `n`. `M[i, j]` denotes the  $(i, j)$ th element of `M`, that is, the element at the `i`th row and `j`th column position.

### Read and Print Statements

The statement

```
read x;
```

means read the next value and store it in the variable `x`.

The statement

```
print x;
```

means output the value of `x`.

### Subprograms (Procedures)

In a programming language, an algorithm is implemented in the form of a **subprogram**. Other terms for subprograms are subroutines, and modules. Typically, there are two types of subprograms. One does some computations and calculates a unique value, which becomes the value of the subroutine, and this unique value is returned via the **return** statement. Such subroutines in the programming language C++ are called value-returning functions, in Java they are called value-returning methods, and in Pascal they are called functions. The other type of subroutine only performs tasks, such as ordering the elements in a list. Such subroutines in the programming language C++ are called void functions, in Java they are called void methods, and in Pascal they are called procedures.

In this book, we use the term **function** to designate a subprogram that returns a unique value and the term **procedure** for other types of subroutines. We will enclose the body of the function or procedure between the words **begin** and **end**.

Because algorithms are typically implemented with the help of functions or procedures, the execution of a **return** statement in a **function** would terminate the **function**.

### Comments

In describing the steps of an algorithm, we include comments wherever necessary to clarify the steps. There are two types of comments—single-line and multi-line. Single-line comments start anywhere in the line with the pair of symbols `//`. In other words, anything in a line after the pair of symbols `//` is a comment. Multi-line comments are enclosed between the pair of symbols `/*` and `*/`.

Whenever we write an algorithm using this syntax, we specify what the algorithm does, as well as the input and output. For example, using these conventions, we can write the algorithm for determining the smallest of three numbers as follows:

#### ALGORITHM 1.1: Determine the smallest of three numbers.

*Input:* Three numbers  $a$ ,  $b$ , and  $c$

*Output:* Output the smallest of  $a$ ,  $b$ , and  $c$

```
1. function minimum( $a, b, c$ )
2. begin
3.    $min := a;$ 
4.   if  $min > b$  then
5.      $min := b;$ 
6.   if  $min > c$  then
7.      $min := c;$ 
8.   return  $min;$ 
9. end
```

Let us consider the problem of determining the smallest element in a list. Let  $L$  be a list of  $n$  elements.

**ALGORITHM 1.2:** Determine the smallest element in a list.

*Input:*  $L$ —list of  $n$  elements  
 $n$ —the size of  $L$

*Output:* Smallest element of  $L$

```

1. function smallest( $L$ ,  $n$ )
2. begin
3.    $min := L[1];$ 
4.   for  $i := 2$  to  $n$  do
5.     if  $min > L[i]$  then
6.        $min := L[i];$ 
7.   return  $min;$ 
8. end

```

**EXAMPLE 1.6.1**

**Sequential Search.** In this example, we describe the sequential search algorithm to search a list. Let  $L$  be a list of  $n$  elements, indexed 1 to  $n$ . Let  $x$  be an element. We want to determine whether  $x$  is in  $L$ ; i.e., if there exists  $i$ ,  $1 \leq i \leq n$ , such that  $x = L[i]$ . The sequential search algorithm (also called linear search), starts the search by comparing  $x$  with the first element,  $L[1]$ , of  $L$ . If  $x = L[1]$ , then the search stops, otherwise, we compare  $x$  with the next element of  $L$ . The search continues until either we have found an element in  $L$  that is the same as  $x$  or we have searched the entire list. The following algorithm implements the sequential search algorithm.

**ALGORITHM 1.3: Sequential Search.** Determine whether an item is in a list.

*Input:*  $L[1 \dots n]$ —list of  $n$  elements  
 $n$ —the size of  $L$   
 $x$ —the search item

*Output:* The index of the first element of  $L$  that is the same as  $x$ , otherwise  $-1$  (indicating an unsuccessful search).

```

1. function sequentialSearch( $L$ ,  $n$ ,  $x$ )
2. begin
3.   for  $i := 1$  to  $n$  do
4.     if  $x = L[i]$  then
5.       return  $i;$ 
6.   return  $-1;$ 
7. end

```

The **for** loop, at Line 3, executes at most  $n$ , the number of elements in  $L$ , times. Each time through the loop, the statement in Line 4 compares  $x$  with  $L[i]$ , the  $i$ th element of  $L$ . If  $x = L[i]$ , then the **return** statement, in Line 5, returns  $i$ , the index of the (first) element in  $L$  that is the same as  $x$ . If  $x$  is not in  $L$ , then the

**for** loop executes  $n$  times and the statement in Line 6 returns  $-1$ , indicating an unsuccessful search.

**EXAMPLE 1.6.2**

**Bubble Sort.** In this example, we describe a simple sorting algorithm, called *bubble sort*, to sort a list. Let  $L[1 \dots n]$  be a list of  $n$  elements, indexed 1 to  $n$ . We want to rearrange, that is, sort, the elements of  $L$  in increasing order. The bubble sort algorithm works as follows: In a series of  $n - 1$  iterations, successive elements  $L[j]$  and  $L[j + 1]$  of  $L$  are compared. If  $L[j] > L[j + 1]$ , then the elements  $L[j]$  and  $L[j + 1]$  are swapped, that is, interchanged. It follows that the smaller elements move toward the top and the larger elements move toward the bottom. In the first iteration, we consider the list  $L[1 \dots n]$ ; in the second iteration, we consider the list  $L[1 \dots n - 1]$ ; in the third iteration, we consider the list  $L[1 \dots n - 2]$ , and so on. For example, consider the following list  $L[1 \dots 5]$  of five elements.

$L$
$L[1]$
10
$L[2]$
7
$L[3]$
19
$L[4]$
5
$L[5]$
16

**Iteration 1:** Sort  $L[1 \dots 5]$ . Figure 1.13 shows how the elements of  $L$  get rearranged in the first iteration.

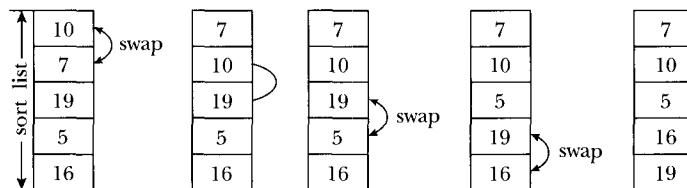


FIGURE 1.13 List during first iteration

Notice that in the first diagram of Figure 1.13,  $L[1] > L[2]$ . Therefore,  $L[1]$  and  $L[2]$  are swapped. In the second diagram,  $L[2] < L[3]$ , they do not get swapped. The third diagram of Figure 1.13 compares  $L[3]$  with  $L[4]$ ; because  $L[3] > L[4]$ ,  $L[3]$  is swapped with  $L[4]$ . Then, in the fourth diagram, we compare  $L[4]$  with  $L[5]$ . Because  $L[4] > L[5]$ ,  $L[4]$  and  $L[5]$  are swapped.

After the first iteration, the largest element is in the last position. Therefore, in the next iteration, we consider the list  $L[1 \dots 4]$ .

**Iteration 2:** Sort  $L[1 \dots 4]$ . Figure 1.14 shows how the elements of  $L$  get rearranged in the second iteration.

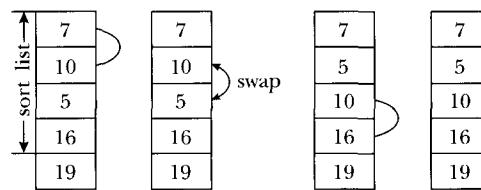


FIGURE 1.14 List during second iteration

After the second iteration, the last two elements are in the right place. Therefore, in the next iteration, we consider the list  $L[1 \dots 3]$ .

**Iteration 3:** Sort  $L[1 \dots 3]$ . Figure 1.15 shows how the elements of  $L$  get rearranged in the third iteration.

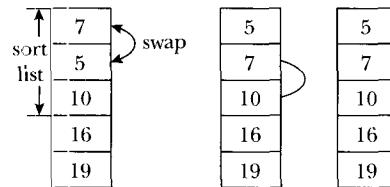


FIGURE 1.15 List during third iteration

After the third iteration, the last three elements are in the right place. Therefore, in the next iteration, we consider the list  $L[1 \dots 2]$ .

**Iteration 4:** Sort  $L[1 \dots 2]$ . Figure 1.16 shows how the elements of  $L$  get rearranged in the fourth iteration.

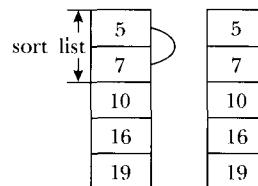


FIGURE 1.16 List during fourth iteration

After the fourth iteration,  $L$  is sorted. The following algorithm implements the bubble sort.

#### ALGORITHM 1.4: Bubble Sort.

*Input:*  $L$ —list of  $n$  elements  
 $n$ —the number of elements in  $L$

*Output:*  $L$  with elements arranged in increasing order

1. **procedure** bubbleSort( $L$ ,  $n$ )
2. **begin**
3.     **for**  $i := 1$  **to**  $n - 1$  **do**
4.         **for**  $j := 1$  **to**  $n - i$  **do**
5.             **if**  $L[j] > L[j + 1]$  **then**
6.                 swap( $L[j]$ ,  $L[j + 1]$ );
7.     **end**

## Polynomial Operations

We learned in college algebra or calculus that a polynomial  $p(x)$  in one variable,  $x$ , is an expression of the form:

$$p(x) = a_0 + a_1x + \cdots + a_{n-1}x^{n-1} + a_nx^n,$$

where  $a_i$  are real (or complex) numbers and  $n$  is a nonnegative integer. If  $p(x) = a_0$ , then  $p(x)$  is called a *constant polynomial*. If  $p(x)$  is a nonzero constant polynomial, then the degree of  $p(x)$  is defined to be 0. Even though in mathematics the degree of the zero polynomial is undefined, for the purpose of this example, we will consider the degree of such polynomials to be zero. If  $p(x)$  is not constant and  $a_n \neq 0$ , then  $n$  is called the *degree* of  $p(x)$ ; that is, the degree of a nonconstant polynomial is defined to be the exponent of the highest power of  $x$ .

The basic operations performed on polynomials are to add, subtract, multiply, divide polynomials, and evaluate a polynomial at a given point. For example, suppose that

$$p(x) = 1 + 2x + 3x^2$$

and

$$q(x) = 4 + x.$$

The degree of  $p(x)$  is 2 and the degree of  $q(x)$  is 1. Moreover,

$$p(2) = 1 + 2 \cdot 2 + 3 \cdot 2^2 = 17,$$

$$p(x) + q(x) = 5 + 3x + 3x^2,$$

$$p(x) - q(x) = -3 + x + 3x^2,$$

and

$$p(x) \cdot q(x) = 4 + 9x + 14x^2 + 3x^3.$$

In the following pages, we discuss algorithms that can be used to perform various polynomial operations. Specifically, we will implement the following operations on polynomials:

1. Evaluate a polynomial at a given value
2. Add polynomials
3. Subtract polynomials
4. Multiply polynomials

We assume that the coefficients of polynomials are real numbers.

A convenient way to store a polynomial in computer memory is to use an array as follows: Suppose  $p(x)$  is a polynomial of degree  $n \geq 0$ . Let  $L$  be an array of size  $n + 1$ . The coefficient  $a_i$  of  $x^i$  is stored in  $L[i]$ . See Figure 1.17.

$[0]$	$[1]$	$[i]$	$[n - 1]$	$[n]$
$p(x)$	$a_0$	$a_1$	$\dots$	$a_i$

**FIGURE 1.17** The polynomial  $p(x)$  and its coefficients

From this figure it is clear that if  $p(x)$  is a polynomial of degree  $n$ , then we need an array of size  $n + 1$  to store the coefficients of  $p(x)$ . Suppose that

$$p(x) = 1 + 8x - 3x^2 + 5x^4 + 7x^8.$$

Then the array storing the coefficient of  $p(x)$  is given in Figure 1.18.

	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]
$p(x)$	1	8	-3	0	5	0	0	0	7

FIGURE 1.18 The polynomial  $p(x)$  and its coefficients

Similarly, if  $q(x) = -5x^2 + 16x^5$ , then the array storing the coefficient of  $q(x)$  is given in Figure 1.19.

	[0]	[1]	[2]	[3]	[4]	[5]
$q(x)$	0	0	-5	0	0	16

FIGURE 1.19 The polynomial  $q(x)$  and its coefficients

Next, we define the operations  $+$ ,  $-$ , and  $\cdot$  (multiplication). Suppose that

$$p(x) = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1} + a_n x^n$$

and

$$q(x) = b_0 + b_1 x + \cdots + b_{m-1} x^{m-1} + b_m x^m.$$

Let  $t = \max(n, m)$ . Then

$$p(x) + q(x) = c_0 + c_1 x + \cdots + c_{t-1} x^{t-1} + c_t x^t,$$

where for  $i = 0, 1, 2, \dots, t$

$$c_i = \begin{cases} a_i + b_i & \text{if } i \leq \min(n, m), \\ a_i & \text{if } i > m, \\ b_i & \text{if } i > n. \end{cases}$$

The difference,  $p(x) - q(x)$ , of  $p(x)$  and  $q(x)$  can be defined similarly. It follows that the degree of the polynomials  $p(x) + q(x)$  and  $p(x) - q(x)$  is  $\leq \max(n, m)$ .

The product,  $p(x) \cdot q(x)$ , of  $p(x)$  and  $q(x)$  is defined as follows:

$$p(x) \cdot q(x) = d_0 + d_1 x + \cdots + d_{n+m} x^{n+m}.$$

The coefficient  $d_k$ , for  $k = 0, 1, 2, \dots, n + m$ , is given by the formula

$$d_k = a_0 b_k + a_1 b_{k-1} + \cdots + a_k b_0,$$

where if either  $a_i$  or  $b_i$  does not exist, it is assumed to be zero. For example,

$$\begin{aligned} d_0 &= a_0 b_0 \\ d_1 &= a_0 b_1 + a_1 b_0 \\ &\vdots \\ d_{n+m} &= a_n b_m. \end{aligned}$$

### ALGORITHM 1.5: Evaluate a polynomial.

*Input:*  $p$ —polynomial  
 $n$ —degree of  $p$   
 $a$ —the value at which to evaluate  $p(x)$

*Output:*  $p(a)$

```

1. function evaluate(p, n, a)
2. begin
3.   value = 0.0;
4.   for i := 0 to n do
5.     if p[i] ≠ 0.0 then
6.       value = value + p[i] * ai;
7.   return value;
8. end

```

Suppose  $p(x)$  is a polynomial of degree  $n$  and  $q(x)$  is a polynomial of degree  $m$ . If  $n = m$ , then to calculate  $p(x) + q(x)$ , we add the corresponding coefficients of  $p(x)$  and  $q(x)$ . If  $n > m$ , then the first  $m$  coefficients of  $p(x)$  are added with the corresponding coefficients of  $q(x)$ . The remaining coefficients of  $p(x)$  are copied into the polynomial containing the sum of  $p(x)$  and  $q(x)$ . Similarly, if  $n < m$ , the first  $n$  coefficients of  $q(x)$  are added with the corresponding coefficients of  $p(x)$ . The remaining coefficients of  $q(x)$  are copied into the polynomial containing the sum. Similarly, we can determine  $p(x) - q(x)$ .

#### **ALGORITHM 1.6:** Add polynomials.

*Input:*  $p, q$ —polynomials  
 $n$ —degree of  $p$   
 $m$ —degree of  $q$

*Output:*  $p(x) + q(x)$

```

1. function add(p,q,n,m)
2. begin
3.   size = max(n,m);
4.   for i := 0 to min(n,m) do
5.     t[i] = p[i] + q[i];
6.   if size = n then
7.     for i := min(n,m) to n do
8.       t[i] = p[i];
9.   else
10.    for i := min(n,m) to m do
11.      t[i] = q[i];
12.   return t;
13. end

```

#### **ALGORITHM 1.7:** Subtract polynomials.

*Input:*  $p, q$ —polynomials  
 $n$ —degree of  $p$   
 $m$ —degree of  $q$

*Output:*  $p(x) - q(x)$

```

1. function subtract(p,q,n,m)
2. begin
3.   size = max(n,m);
4.   for i := 0 to min(n,m) do
5.     t[i] = p[i] - q[i];
6.   if size = n then
7.     for i := min(n,m) to n do
8.       t[i] = p[i];
9.   else
10.    for i := min(n,m) to m do
11.      t[i] = -q[i];
12.   return t;
13. end

```

The following algorithm multiplies two polynomials.

**ALGORITHM 1.8:** Multiply polynomials.

*Input:*  $p, q$ —polynomials  
 $n$ —degree of  $p$   
 $m$ —degree of  $q$

*Output:*  $p(x) \cdot q(x)$

```

1. function multiply(p,q,n,m)
2. begin
3.   for i := 0 to n + m do
4.     t[i] := 0;
5.   for i := 0 to n do
6.     for k := 0 to m do
7.       t[i + k] = t[i + k] + p[i] * q[k];
8.   return t;
9. end;

```

An alternative way of writing the multiplication algorithm is:

*Input:*  $p, q$ —polynomials

$n$ —degree of  $p$

$m$ —degree of  $q$

*Output:*  $p(x) \cdot q(x)$

```

1. function multiply(p,q,n,m)
2. begin
3.   for k := 0 to n + m do
4.     begin
5.       t[k] := 0;
6.       for i := 0 to min(k,n) do
7.         if (i ≤ n and k - i ≤ m) then
8.           t[k] = t[k] + p[i] * q[k - i];
9.     end
10.    return t;
11. end

```

## WORKED-OUT EXERCISES

**Exercise 1:** Let  $L$  be a list of  $n$  elements. In this section, we designed an algorithm to determine the smallest element in  $L$ . Sometimes, rather than determining the smallest element, we are interested in knowing the index, that is, the position of the smallest element in the list. Of course, if we know the position of the smallest element, then we can access the smallest element. Write an algorithm that returns the position of the smallest element in the list.

**Solution:**

**ALGORITHM 1.9:** Index of the smallest element in a list.

*Input:*  $L$ —list of  $n$  elements  
 $n$ —the size of  $L$

*Output:* Smallest element of  $L$

```

1. function indexSmallest(L, n)
2. begin
3.   indexMin := 1;
4.   for i := 2 to n do
5.     if L[indexMin] > L[i] then
6.       indexMin := i;
7.   return indexMin;
8. end

```

**Exercise 2:** What does the following algorithm do?

```

1. procedure mystery()
2. begin
3.   secret := 0;
4.   for i := 1 to 20 do
5.     begin
6.       read x;
7.       secret := secret + x;
8.     end
9.   print secret;
10.  end

```

**Solution:** The statement in Line 3 sets the variable *secret* to 0. The **for** loop in Line 4 executes 20 times. The body of the **for** loop is between Lines 5 and 8. The statement in Line 6 reads a number and stores it in *x*. The statement in Line 7 adds the value of *x* into *secret* and the value is assigned to *secret*. Notice that after the first iteration of the **for** loop the value of *secret* is *x*, which is the first number read. After the second iteration of the **for** loop, the value of *secret* is the sum of the first two elements. It follows that this algorithm reads and adds 20 numbers. Finally, the statement in Line 9 outputs the value of *secret*.

## SECTION REVIEW

---

### Key Terms

algorithm	two-way selection	list
input	if	two-dimensional array
output	then	read
precision	else	print
uniqueness	while	subprograms
finiteness	do	procedure
generality	for	function
assignment operator	begin	constant polynomial
assignment statement	end	degree
control structures	return	
one-way selection	arrays	

### Some Key Definitions

1. An algorithm is a step-by-step problem-solving process in which a solution is arrived at in a finite amount of time.

## EXERCISES

---

1. What does the following algorithm do?

```
function secret(x:integer)
begin
    prod = 1;
    for i := 1 to 3 do
        prod = prod * x;
    return prod;
end
```

2. What does the following algorithm do?

```
function mystery(k:integer)
begin
    y = k;
    for x := 1 to k - 1 do
        y = y * (k - x);
    return y;
end
```

3. Let  $m$  and  $n$  be positive integers. Write an algorithm that uses repeated addition to calculate  $mn$ .

4. Let  $n$  be a positive integer. Write an algorithm that computes  $n!$ , the factorial of  $n$ .

5. Let  $x$  be a nonzero real number and let  $n$  be an integer. Write an algorithm that computes  $x^n$ .

6. Consider the quadratic equation  $ax^2 + bx + c = 0$ , where  $a$ ,  $b$ , and  $c$  are real numbers and  $a \neq 0$ . Write an algorithm that calculates the roots of the equation that are real numbers. If the equation has no real roots,

then it outputs a message indicating that the equation has no real roots.

7. Let  $L$  be a list of  $n$  elements. The list  $L$  may contain duplicates and its elements are in no particular order. Write an algorithm that returns the position of the last occurrence of the smallest element.
8. Let  $L$  be a list of  $n$  elements,  $n \geq 2$ . Write an algorithm that returns the second smallest element in  $L$ .
9. Let  $L$  be a list of  $n$  elements. Write an algorithm that returns the largest element in  $L$ .
10. Let  $L$  be a list of  $n$  elements. Write an algorithm that returns the position of the largest element in  $L$ .
11. Let  $L$  be a list of  $n$  elements. Assume that the elements of  $L$  are in ascending or descending order. Write an algorithm that returns true if  $L$  does not contain any duplicates; otherwise it returns false.
12. Let  $L$  be a list of  $n$  numbers. Assume that the elements of  $L$  are in ascending or descending order. Write an algorithm that returns the smallest difference between successive elements of  $L$ .
13. Let  $L$  be a list of  $n$  numbers. Assume that the elements of  $L$  are in ascending or descending order. Write an algorithm that returns the largest difference between successive elements of  $L$ .
14. Let  $L$  be a list of  $n$  numbers. Write an algorithm that finds the sum of the elements of  $L$ .

15. Let  $L$  be a list of  $n$  numbers. Write an algorithm that finds the sum of the squares of the elements of  $L$ .
16. In the bubble sort algorithm, if successive elements  $L[j]$  and  $L[j + 1]$  are such that  $L[j] > L[j + 1]$ , then they are interchanged, that is, swapped. Therefore, the bubble sort algorithm may require elements to be swapped. Let  $x$  and  $y$  be elements. Write an algorithm that swaps the values of  $x$  and  $y$ .
17. Show how bubble sort sorts the elements 5 4 3 2 1 in increasing order. Draw figures as in Example 1.6.2.
18. Show how bubble sort sorts the elements 7 5 6 3 1 4 2 in increasing order. Draw figures as in Example 1.6.2.
19. Many programming languages provide a function to determine the square root of a nonnegative real number. Using Newton's method, we can also write an algorithm to find the square root of a nonnegative real number within a given tolerance as follows: Suppose  $x$  is a nonnegative real number,  $a$  is the approximate square root of  $x$ , and  $\varepsilon$  is the tolerance. Start with  $a := x$ ;
  - a. If  $|a^2 - x| \leq \varepsilon$ , then  $a$  is the square root of  $x$  within the tolerance; otherwise
  - b. Replace  $a$  with  $(a^2 + x)/(2a)$  and repeat step (a).
20. Let  $p(x) = a_0 + a_1x + \cdots + a_{n-1}x^{n-1} + a_nx^n$  be a polynomial of degree  $n$ , where  $a_i$  are real numbers and  $n$  is a nonnegative integer. The derivative of  $p(x)$ , written  $p'(x)$ , is defined to be  $p'(x) = a_1 + 2a_2x + \cdots + na_nx^{n-1}$ . If  $p(x)$  is constant, then  $p'(x) = 0$ . Write an algorithm that computes the derivative of a polynomial.

## PROGRAMMING EXERCISES

---

Use any programming language to write the following programs.

1. Write a program to perform various operations on finite sets.
2. Write a program that generates a truth table of statements involving up to three variables and two logical connectives.
3. Write a program to implement the bubble sort algorithm.

4. **Polynomial Calculator.** Write a program to perform various operations of polynomials in one variable. Some of the operations that the program must perform are: addition, subtraction, multiplication, division (output the quotient and remainder), evaluation a polynomial at a given value, and finding the derivative of a polynomial.

## Integers and Mathematical Induction

The objectives of this chapter are to:

- Learn about the basic properties of integers
- Become aware how integers are represented in computer memory
- Explore how addition and subtraction operations are performed on binary numbers
- Learn how the principle of mathematical induction is used to solve problems
- Learn about loop invariants and how they are used to prove the correctness of loops
- Explore various properties of prime numbers
- Learn about linear Diophantine equations and how to solve them

Since the days of Pythagoras, it was known to the Greeks—and everyone since then—that if the lengths of the three sides of a triangle are 3, 4, and 5 units, then it must be a right-angled triangle. The search to find all the right-angled triangles with integral side-lengths then amounts to the search to find all positive integer solutions  $(x, y, z)$ , commonly called the **Pythagorean triple**, to the Pythagorean equation,

$$x^2 + y^2 = z^2.$$

Given a Pythagorean triple, one can, with a little bit of calculation, derive infinitely many other Pythagorean triples. This naturally encourages an inquisitive mind to think about the possibility of finding nonzero integers  $x, y, z$  such that  $x^n + y^n = z^n$ , where  $n$  is an integer greater than 2.

Tremendous efforts were put forth by many renowned mathematicians, including Leonhard Euler, Adrien-Marie Legendre, Niels Henrik Abel, Carl Friedrich Gauss, Peter Gustav Lejeune Dirichlet, Augustin-Louis Cauchy, Ernst Kummer, Leopold Kronecker, and David Hilbert, to prove that the equation  $x^n + y^n = z^n$  has no solutions in integers except for the trivial ones (in which one of  $x, y, z$  is zero) with positive integral exponents higher than 2.

In 1637, French scholar Pierre de Fermat, as he was going through a copy of Diophantus' *Arithmetic*, wrote in the margin of book:

On the other hand it is impossible to separate a cube into two cubes or a biquadratic (fourth power) into two biquadratics, or generally any power except a square into two powers with the same exponent. I have discovered a truly marvelous proof of this, however, the margin is not large enough to contain.

What Fermat is stating in symbols is that the equation  $x^n + y^n = z^n$ , where  $x, y, z, n$  are positive integers and  $n > 2$ , has no nontrivial solutions in integers. Because Fermat gave no proof, many mathematicians labored to find a proof of his statement, popularly known as *Fermat's Last Theorem*, which says that *the equation*

$$x^n + y^n = z^n$$

*has no nontrivial solution in positive integers, where  $n$  is an integer such that  $n > 2$ .*



**Pythagoras**  
(ca. 500 B.C.)

Pythagoras was born on the Greek isle of Samos, where he was well-educated in music, poetry, and philosophy. It is believed that he traveled widely with his father, who was a merchant. His quest for knowledge continued to lead him abroad, but he eventually came back to Samos and es-

#### Historical Notes

tablished a school called the Semicircle, which was a forum for political, ethical, and philosophical ideas. However, politics began to overtake all other areas of his life, so in the tradition of philosophers before him, Pythagoras left Samos for southern Italy. There he created another school built on the beliefs that reality was mathematically based and that philosophy held the key to spiritual ascension.

Even though no specific writings by Pythagoras exist today, he is credited with proving that in right-angled triangles, the square of the length,  $c$ , of the hypotenuse is the sum of the squares of the other two sides, i.e.,  $a^2 + b^2 = c^2$ . He is also credited with establishing the abstract concept of numbers as well as defining concepts in music theory.

Fermat gave an explicit proof of the case  $n = 4$ , but the general proof was never found in his works. It is debatable that if he could have dealt with the general case successfully, he probably would not have bothered with the particular case  $n = 4$  later. The case  $n = 3$  was proved initially in the eighteenth century by Euler, and later Gauss provided a corrected proof. Mathematicians soon realized that to prove Fermat's theorem, it would be sufficient to prove it for odd prime exponents only. The case  $n = 5$  was initiated by Dirichlet and settled by Legendre around 1825, and the French engineer cum mathematician Gabriel Lamé proved it for  $n = 7$ , in 1839. Sophie Germain proved that if  $n$  is an odd prime  $< 100$ , the equation  $x^n + y^n = z^n$  is not solvable in integers not divisible by  $n$ . In the middle of the nineteenth century, Kummer proved the theorem for all primes less than 37.

With the introduction of computers, mathematicians gained access to a tremendous improvement in calculating power, and in 1993, Joe Buhler and Richard Crandall proved that Fermat's Last Theorem is true for all prime exponents below 4,000,000. But the mathematical community worldwide was still awaiting a general proof. Then, on June 23, 1993, a truly historic day in mathematics, nearly 355 years after the inception of this theorem, Andrew Wiles of Princeton University came forward with a proof. Unfortunately, a fatal logical gap in the argument was soon discovered, but in October 1994, Richard Taylor and Wiles released a corrected version of the proof, almost 130 pages long, which was published in May 1995 in the *Annals of Mathematics*.

**Andrew Wiles**

(b. 1953)

Even as a child, Andrew Wiles was interested in mathematics.

As a boy growing up in England, he stumbled across a book describing Fermat's Last Theorem in a local library. The challenge to prove the theorem would become his life's work.

### HISTORICAL NOTES

Wiles graduated from Oxford in 1974 with a B.A. He was awarded his doctorate in 1980 from Cambridge. Throughout the 1980s and 1990s, Wiles had the opportunity and distinction to teach in some of the most prestigious universities in the United States, Great Britain, France, and Germany, all the while continuing his work to solve Fermat's Last Theorem. Focused and

driven, Wiles maintained that he could not concentrate on anything besides his work and his family. Finally, in 1994, Wiles found the key that unlocked Fermat's Last Theorem, a discovery that until now had evaded mathematicians for over three centuries.

The proof is based on different areas of mathematics and is highly technical. Wiles's proof is generally regarded as the high point of twentieth-century number theory.

As we can see, numbers, especially integers, have challenged mathematicians for centuries. In the literature, one can find many interesting problems relating to integers. In fact, number theory is an active area of research in its own right.

In Chapter 1, we discussed sets, logic, and algorithms. In this chapter, we discuss some basic properties of integers.

## 2.1 INTEGERS

---

Since we have been old enough to count, we have been familiar with natural numbers. The mathematician Kronecker once remarked, "God created the integers all else is the work of man." Integers are by and large the building blocks of mathematics, particularly of abstract algebra. In fact, many algebraic abstractions come from the set of integers. Throughout this book, the set of all integers will be used as a major source of examples of different algebraic systems. In this section, we do not intend to give an axiomatic development of the integers. We assume that the reader is familiar with the set  $\mathbb{N} = \{1, 2, \dots\}$  of all positive integers (i.e., natural numbers), the set  $\mathbb{Z}^* = \{0, 1, 2, \dots\}$  of all nonnegative integers, and the set  $\mathbb{Z} = \{0, \pm 1, \pm 2, \dots\}$  of all integers, together with the usual properties of *addition* and *multiplication* along with the usual *order* prevailing among these numbers. In particular, for all  $a, b, c \in \mathbb{Z}$  the following properties apply.

- Closure:  $a + b \in \mathbb{Z}$   
 $a \cdot b \in \mathbb{Z}$
- Commutative laws:  $a + b = b + a,$   
 $a \cdot b = b \cdot a$
- Associative laws:  $a + (b + c) = (a + b) + c,$   
 $a \cdot (b \cdot c) = (a \cdot b) \cdot c$
- Identity elements:  $a + 0 = a = 0 + a,$   
 $a \cdot 1 = a = 1 \cdot a$
- The element 0 is called the **additive identity** and 1 is called the **multiplicative identity**.
- Additive inverse: For each integer  $a$ , there exists an integer  $b$  such that

$$a + b = 0 = b + a.$$

This integer  $b$  is called the additive inverse of  $a$  and is denoted by  $-a$ . Moreover, we write  $a - b$  for  $a + (-b)$ .

- Distributive laws:  $a \cdot (b + c) = a \cdot b + a \cdot c,$   
 $(b + c) \cdot a = b \cdot a + c \cdot a$

- Cancellation laws: Suppose that  $a \neq 0$ . Then  $a \cdot b = a \cdot c$  implies that  $b = c$ ; and  $b \cdot a = c \cdot a$  implies that  $b = c$ .

Using these basic properties of integers, we can prove other properties of integers, some of which are listed in the following theorem.

### Theorem 2.1.1:

- (i) Let  $a$ ,  $b$  and  $c$  be integers. Then

$$\begin{aligned} \text{a. } & b + a = c + a \Rightarrow b = c. \\ \text{b. } & a + b = a + c \Rightarrow b = c. \end{aligned}$$

- (ii) For any integer  $a$ ,  $a \cdot 0 = 0 = 0 \cdot a$ .  
 (iii) If  $a$  and  $b$  are two integers such that  $a \cdot b = 0$ , then either  $a = 0$  or  $b = 0$ .  
 (iv) For any integer  $a$ ,  $-(-a) = a$ .

### Proof:

- (i) a. Suppose  $a$ ,  $b$ , and  $c$  are integers such that  $b + a = c + a$ . Then

$$\begin{aligned} b + a &= c + a \\ \Rightarrow (b + a) + (-a) &= (c + a) + (-a) \quad \text{add } -a \text{ to both sides} \\ \Rightarrow b + (a + (-a)) &= c + (a + (-a)) \quad \text{by associativity laws} \\ \Rightarrow b + 0 &= c + 0 \quad \text{by additive inverse} \\ \Rightarrow b &= c \quad \text{because 0 is an additive identity.} \end{aligned}$$

Hence,  $b + a = c + a$  implies that  $b = c$ .

- b. Suppose  $a + b = a + c$ . By the commutative law, we have  $a + b = b + a$  and  $a + c = c + a$ . Therefore,  $a + b = a + c$  implies  $b + a = c + a$ . Hence, using part i(a), we can conclude that  $b = c$ .

- (ii) In this proof, we use the fact that for all integers  $x$ ,  $x + 0 = x$ .

Let  $a$  be an integer. Then

$$\begin{aligned} a \cdot 0 + 0 &= a \cdot 0 \quad \text{because 0 is an additive identity} \\ \Rightarrow a \cdot 0 + 0 &= a \cdot (0 + 0) \quad \text{because } 0 = 0 + 0 \\ \Rightarrow a \cdot 0 + 0 &= a \cdot 0 + a \cdot 0 \quad \text{by distributivity.} \end{aligned}$$

We now apply part (i) to conclude that  $a \cdot 0 = 0$ . Moreover, by the commutative law,  $0 \cdot a = a \cdot 0$  and so  $0 \cdot a = 0$ .

- (iii) Suppose  $a$  and  $b$  are integers such that  $a \neq 0$  and  $a \cdot b = 0$ . Then

$$\begin{aligned} a \cdot b &= 0 \\ \Rightarrow a \cdot b &= a \cdot 0 \quad \text{by part (ii), } a \cdot 0 = 0 \\ \Rightarrow b &= 0 \quad \text{by the cancellation law.} \end{aligned}$$

- (iv) Let  $a \in \mathbb{Z}$ . Then by the inverse property, we have  $(-a) + a = 0$ . Now  $-a \in \mathbb{Z}$ . We apply the inverse property on  $-a$ . Therefore,  $(-a) + (-(-a)) = 0$ . Hence, we have

$$(-a) + (-(-a)) = 0 = (-a) + a,$$

i.e.,

$$(-a) + (-(-a)) = (-a) + a.$$

We can now apply part (i) to conclude that  $-(-a) = a$ . ■

**DEFINITION 2.1.2** ▶ Let  $a$  and  $b$  be integers. Then  $a$  is said to be **greater than**  $b$ , written  $a > b$ , if  $a - b$  is a positive integer. If  $a > b$ , then sometimes we also write  $b < a$ .

We leave the proof of the following theorem as an exercise. (See Exercise 11.)

**Theorem 2.1.3:** Let  $a$  and  $b$  be integers such that  $a > b$ . Then

- (i)  $a + c > b + c$  for any integer  $c$ , and
- (ii)  $ad > bd$  for any positive integer  $d$ .
- (iii)  $ad < bd$  for any negative integer  $d$ .

The proofs of many results of algebra depend on the following basic principle of the integers. Before stating the principle, let us consider the following. Suppose  $S$  is a finite set of positive integers. Because there are only a finite number of elements, comparing one integer with another integer we can find in a finite number of steps a smallest element in this set. Suppose  $S$  contains an infinite number of elements. Intuitively,  $S$  has a smallest element. However, it is a difficult task to consider the proof of the statement. To overcome the problem, we consider the following property of positive integers as an axiom and prove some other properties of integers by using this axiom.

**Well-Ordering Principle:** Any nonempty subset of nonnegative integers has a least (i.e., smallest) element. That is, if  $S$  is a nonempty subset of the set of nonnegative integers, then there exists  $n \in S$  such that  $n \leq m$  for all  $m \in S$ .

**REMARK 2.1.4** ▶ To apply the well-ordering principle to a problem, one of the first things we do is construct a set  $S$  of some nonnegative integers and then show that  $S$  is nonempty.

Consider the integers 25 and 7. Now  $25 \not< 1 \cdot 7$ ,  $25 \not< 2 \cdot 7$ ,  $25 \not< 3 \cdot 7$ , but  $25 < 4 \cdot 7$ . Therefore, for the integers 25 and 7, we have the integer  $n = 4$  such that  $25 < 7n$ . So the obvious question is the following: Is this result true for any positive integers; i.e., given two positive integers  $a$  and  $b$ , can we find a positive integer  $n$  such that  $b < na$ ? In the next theorem, this property of integers is proved using the well-ordering principle.

**Theorem 2.1.5:** Let  $a$  and  $b$  be two positive integers. Then there exists a positive integer  $n$  such that  $b < na$ .

**Proof:** We prove this theorem by the method of contradiction.

(In this proof, we use the well-ordering principle. To do so we need to construct a set  $S$  of some nonnegative integers. So what should this  $S$  be? Our objective

is to show the existence of a positive integer  $n$  such that  $b < na$ . The proof that we are constructing is by contradiction. Therefore, we deny the conclusion and assume that  $b \geq na$  for all positive integers. Now  $b \geq na$  implies that  $b - na \geq 0$ , i.e., the  $b - na$  is a nonnegative integer. This suggests that to apply the well-ordering principle, we can take  $S$  to be the set of all such nonnegative integers.)

Suppose  $b \geq na$  for all positive integers  $n$ . Consider the following set  $S$ .

$$S = \{b - na \mid n \text{ is a positive integer}\}.$$

Now  $b - a = b - 1a$  and so  $b - a \in S$ . Thus,  $S$  is nonempty.

For any integer  $n$ ,  $b - na$  is an integer and so  $S$  is a subset of  $\mathbb{Z}$ . By our assumption, for any integer  $n$ ,  $b \geq na$  and so  $b - na \geq 0$ . It now follows that  $S$  is a nonempty subset of the set of nonnegative integers. Therefore, by the well-ordering principle,  $S$  has a least element, say  $b - ta$ , where  $t > 0$ .

Because  $t > 0$ , we have  $t + 1 > 0$  and so  $b - (t + 1)a \in S$ . Now  $b - ta$  is the smallest element of  $S$  and so we have  $b - ta \leq b - (t + 1)a$ . This implies that

$$b - (t + 1)a - (b - ta) \geq 0.$$

However,  $b - (t + 1)a - (b - ta) = -a$  and so  $-a \geq 0$  or  $a \leq 0$ . This contradicts the fact that  $a$  is a positive integer. Therefore, our assumption that  $b \geq na$  for all positive integers is incorrect. Hence, there exists a positive integer  $n$  such that  $b < na$ . ■

In the remainder of this section, we present some basic concepts about integers that we have been using since our first algebra course, such as division of integers, greatest common divisors, and least common multiple.

## The Division Algorithm

Consider the integers 38 and 5. We can write

$$38 = 7 \cdot 5 + 3.$$

Such a result is true for any integers  $a$  and  $b$  such that  $b \geq 1$ . By using the well-ordering principle in the next theorem, we prove this result.

**Theorem 2.1.6: The Division Algorithm.** Let  $a$  and  $b$  ( $\geq 1$ ) be integers.

Then there exist unique integers  $q$  and  $r$  such that

$$a = bq + r,$$

where  $0 \leq r < b$ .

**Proof:** (The proof of this theorem uses the well-ordering principle. Therefore, to apply the well-ordering principle, we need to construct a set of some nonnegative integers. Now  $a$  and  $b$  are given, and our first objective is find  $q$  and  $r$  so that  $a = bq + r$ , i.e.,  $a - bq = r$ . We also need to show that  $0 \leq r < b$ , i.e.,  $0 \leq a - bq < b$ . This suggests that we can start with the set of all elements of the form  $a - tb$ ,  $t$  is an integer, such that  $a - tb \geq 0$ . Then show that this set is nonempty and then apply the well-ordering principle.)

Consider the set

$$S = \{a - tb \mid t \in \mathbb{Z}, a - tb \geq 0\}.$$

Note that if  $a$ ,  $b$ , and  $t$  are integers, then  $a - tb$  is also an integer. It follows that

$S$  is a subset of the set of nonnegative integers. Next we show that  $S$  is nonempty. For this we consider two cases:  $a \geq 0$  and  $a < 0$ .

First suppose  $a \geq 0$ . Then

$$a - 0b = a \geq 0.$$

This implies that  $a = a - 0b \in S$ .

Now suppose  $a < 0$ . Then

$$a - ab = a(1 - b) \geq 0,$$

and so  $a - ab \in S$ .

Thus,  $S$  is a nonempty subset of the set of nonnegative integers. Hence, by the well-ordering principle,  $S$  has a least element, say  $r$ .

Now  $r \in S$  and so  $r = a - bq$  for some integer  $q$ . This implies that

$$a = qb + r \quad (2.1)$$

such that  $r \geq 0$ .

Next we show that  $r < b$ . We prove this by the method of contradiction.

Suppose that  $r \geq b$ . Then

$$a - (q + 1)b = a - qb - b = r - b \geq 0.$$

Therefore, by the definition of  $S$ ,  $a - (q + 1)b \in S$  or  $r - b \in S$ .

Now,  $r$  is the least element of  $S$  and  $r - b \in S$ . Therefore,  $r < r - b$ , which in turn implies that  $0 < -b$  or  $b < 0$ . This contradicts the fact that  $b \geq 1$ . Thus, our assumption that  $r \geq b$  is incorrect. Hence,  $r < b$ .

We have thus shown the existence of  $q$  and  $r$ . Next, we show that for the elements  $a$  and  $b$ , these elements  $q$  and  $r$  are unique.

To show the uniqueness of  $q$  and  $r$ , suppose that there exist integers  $q_1$  and  $r_1$  such that

$$a = q_1b + r_1 \quad (2.2)$$

and  $0 \leq r_1 < b$ . Thus, from (2.1) and (2.2), we have

$$qb + r = q_1b + r_1.$$

This implies

$$(q - q_1)b = r_1 - r. \quad (2.3)$$

Now,  $0 \leq r < b$  implies that

$$-b < -r \leq 0. \quad (2.4)$$

Also,

$$0 \leq r_1 < b. \quad (2.5)$$

Thus, by adding the corresponding sides of the inequalities (2.4) and (2.5), we have

$$-b < r_1 - r < b.$$

This implies, using (2.3), that

$$-b < (q - q_1)b < b.$$

Because  $b \geq 1$ , canceling  $b$ , we get

$$-1 < q - q_1 < 1.$$

Now  $q - q_1$  is an integer and  $-1 < q - q_1 < 1$ . The only integer between  $-1$  and  $1$  is  $0$ , so we must have  $q - q_1 = 0$ , i.e.,  $q = q_1$ . This in turn implies that

$$r_1 - r = (q - q_1)b = 0$$

and so  $r = r_1$ . We have thus proved that  $q$  and  $r$  are unique. ■

In Theorem 2.1.6, the integer  $q$  is called the **quotient** of  $a$  and  $b$  on dividing  $a$  by  $b$ , and the integer  $r$  is called the **remainder** of  $a$  and  $b$  on dividing  $a$  by  $b$ .

In Theorem 2.1.6, it is assumed that  $b$  is a positive integer. However, such a result is also true if  $b$  is negative, as shown in the following result.

**Corollary 2.1.7:** If  $a$  and  $b$  are two integers with  $b \neq 0$ , then there exist unique integers  $q$  and  $r$  such that  $a = qb + r$ , where  $0 \leq r < |b|$ .

### Proof:

**Case I:** Suppose  $b > 0$ . In this case, the corollary follows from the Theorem 2.1.6.

**Case II:** Suppose  $b < 0$ . Then  $|b| = -b > 0$ . We can apply Theorem 2.1.6 to the integers  $a$  and  $-b$ . Thus, it follows that there exist unique integers  $q_1$  and  $r$  such that

$$a = q_1(-b) + r$$

and  $0 \leq r < |b|$ . Let  $-q_1 = q$ . Thus,

$$a = qb + r,$$

where  $0 \leq r < |b|$ . Because  $q_1$  is unique,  $q$  is unique. Hence, there exist unique integers  $q$  and  $r$  such that  $a = qb + r$ , where  $0 \leq r < b$ . ■

Typically, we are accustomed to applying the division algorithm when  $a$  and  $b$  are positive integers. The next example illustrates the division algorithm when  $a$  and  $b$  may be negative.

### EXAMPLE 2.1.8

- (i) Consider the integers  $a = 95$  and  $b = -30$ . Now

$$95 = (-3)(-30) + 5, \quad 0 < 5 < |-30|.$$

Here,  $q = -3$  and  $r = 5$ .

- (ii) Consider the integers  $a = -95$  and  $b = -30$ . Now

$$-95 = 4(-30) + 25, \quad 0 < 25 < |-30|.$$

Here,  $q = 4$ ,  $r = 25$ .

Throughout we will see many applications of the division algorithm. The following two examples show its two simple applications.

### EXAMPLE 2.1.9

We show that any integer is one of the forms  $3k$ ,  $3k + 1$ , or  $3k + 2$ , and the square of any integer is one of the forms  $3k$  or  $3k + 1$ , where  $k$  is an integer.

Let  $n$  be any integer. Then by the division algorithm, there exist integers  $k$  and  $r$  such that

$$n = 3k + r, \tag{2.6}$$

where  $0 \leq r < 3$ . Because  $0 \leq r < 3$  and  $r$  is an integer, it follows that  $r = 0, 1$ , or  $2$ . We have from (2.6),

$$\begin{aligned} \text{if } r = 0, \text{ then } n &= 3k + r = 3k + 0 = 3k; \\ \text{if } r = 1, \text{ then } n &= 3k + 1 = 3k + 1; \\ \text{if } r = 2, \text{ then } n &= 3k + r = 3k + 2. \end{aligned}$$

Hence,  $n$  is one of the forms  $3k, 3k + 1$ , or  $3k + 2$ .

If  $n = 3k$ , then

$$n^2 = 9k^2 = 3(3k^2) = 3r, \text{ where } r = 3k^2.$$

If  $n = 3k + 1$ , then

$$n^2 = (3k + 1)^2 = 9k^2 + 6k + 1 = 3(3k^2 + 2k) + 1 = 3t + 1,$$

where  $t = 3k^2 + 2k$ .

If  $n = 3k + 2$ , then

$$\begin{aligned} n^2 &= (3k + 2)^2 = 9k^2 + 12k + 4 = 3(3k^2 + 4k) + 3 + 1 \\ &= 3(3k^2 + 4k + 1) + 1 = 3s + 1, \end{aligned}$$

where  $s = 3k^2 + 4k + 1$ .

### EXAMPLE 2.1.10

In this example, we show another interesting application of the division algorithm.

Suppose today is Thursday. We want to know the day just after 90 days from today. By the division algorithm,

$$90 = 12 \cdot 7 + 6.$$

Because today is Thursday, it follows that the day just after 7 days from today is again Thursday. Hence, just after  $12 \cdot 7$  days it is again Thursday, and so just after 90 days from today it will be Wednesday.

### The **div** and **mod** Operators

Programming languages such as C++ and Java provide operators to calculate the quotient and remainder when an integer  $a$  is divided by a nonzero integer  $b$ . Similarly, software such as *Mathematica* and *Maple* also provide such operators. We briefly describe how these operators work the notation we use to specify them.

We use the word **div** to obtain the quotient and the word **mod** to obtain the remainder. The syntax to use these operators is: (Let  $a$  and  $b$  be integers such that  $b \neq 0$ .)

$a \text{ div } b$  = the quotient of  $a$  and  $b$  on dividing  $a$  by  $b$ .

$a \text{ mod } b$  = the remainder of  $a$  and  $b$  on dividing  $a$  by  $b$ .

For example,

$$30 \text{ div } 4 = 7,$$

$$308 \text{ div } 5 = 61,$$

$$30 \text{ mod } 4 = 2,$$

$$308 \text{ mod } 5 = 3.$$

**REMARK 2.1.11** ▶ The programming languages C++ and Java use the symbol `/` for `div` and the symbol `%` for `mod`. Mathematica uses the word `Quotient`, in the form `Quotient[a, b]`, and the word `Mod`, in the form `Mod[a, b]`, to determine the quotient and remainder, respectively.

Let us now write an algorithm to compute the remainder and quotient.

### ALGORITHM 2.1: Division Algorithm.

*Input:* Integers  $m, n$  such that  $n \geq 0$  and  $m > 0$   
*Output:* Integers  $q$  and  $r$  such that  $n = mq + r$   
 and  $0 \leq r < m$

1. **procedure** **divisionAlgorithm**( $m, n, q, r$ )
2. **begin**
3.    $r := n;$
4.    $q := 0;$
5.   **while**  $r \geq m$  **do**
6.     **begin**
7.        $r := r - m;$
8.        $q := q + 1;$
9.     **end**
10. **end**

Let us apply this algorithm to the integers  $m = 15$  and  $n = 57$ . Before the loop executes,  $r = 57$  and  $q = 0$ .

	$r$	$q$
Iteration 1:	$57 - 15 = 42$	1
Iteration 2:	$42 - 15 = 27$	2
Iteration 3:	$27 - 15 = 12$	3

After the third iteration,  $r = 12 < 15$  and so the loop terminates. Hence, the remainder is 12 and the quotient is 3. Therefore,

$$57 = 3 \cdot 15 + 12.$$

### Divisibility

Consider the integers 28 and 7. Using the division algorithm, we can write  $28 = 4 \cdot 7 + 0$ . Here  $q = 4$  and  $r = 0$ ; that is, the remainder is 0. When an integer  $x$  is divided by a nonzero integer  $y$  and the remainder is 0, we say that  $y$  divides  $x$ . More formally, we have the following definition.

**DEFINITION 2.1.12** ▶ Let  $a$  and  $b$  be two integers such that  $a \neq 0$ . If there exists an integer  $c$  such that  $b = ac$ , then  $a$  is said to **divide**  $b$  or  $a$  is said to be a **divisor** of  $b$  and we write  $a | b$ .

Whenever we write  $a | b$  we mean  $a \neq 0$  and  $a$  divides  $b$ . The notation  $a \nmid b$  means that  $a$  does not divide  $b$ .

- REMARK 2.1.13** ▶ (i) For any integer  $a$ , we have  $a = 1a$  and so  $1 \mid a$ .  
(ii) Let  $b \neq 0$  be an integer. Now  $0 = 0b$  and  $b = 1b$ . It follows that  $b \mid 0$  and  $b \mid b$ .

In the following theorem, we list some basic properties of divisibility.

**Theorem 2.1.14:** Let  $a$ ,  $b$ , and  $c$  be integers.

- (i) Suppose  $a \neq 0$  and  $c \neq 0$ . If  $a \mid b$ , then  $ca \mid cb$  and  $ac \mid bc$ .
- (ii) Suppose  $a \neq 0$  and  $b \neq 0$ . If  $a \mid b$  and  $b \mid c$ , then  $a \mid c$ .
- (iii) Suppose  $a \neq 0$ . If  $a \mid b$  and  $a \mid c$ , then  $a \mid (bx + cy)$  for any integers  $x$  and  $y$ .
- (iv) Suppose  $a \neq 0$ . If  $a \mid b$  and  $a \mid (b + c)$ , then  $a \mid c$ .
- (v) Suppose  $a \neq 0$  and  $b \neq 0$ . If  $a \mid b$  and  $b \mid a$ , then  $a = \pm b$ .

### Proof:

- (i) Suppose  $a \mid b$ . Then there exists an integer  $m$  such that  $b = am$ . This implies that  $cb = cam$ . Because  $a \neq 0$  and  $c \neq 0$ , we have  $ca \neq 0$ . Therefore,  $cb = cam$  implies that  $ca \mid cb$ .  
By commutativity,  $ac = ca$  and  $bc = cb$ . So we also have  $ac \mid bc$ .
- (ii) Suppose  $a \mid b$  and  $b \mid c$ . Then there exist integers  $m$  and  $n$  such that  $b = na$  and  $c = mb$ . Now

$$c = mb = m(na) = (mn)a, \text{ and } mn \text{ is an integer.}$$

This implies that  $a \mid c$ .

- (iii) Suppose  $a \mid b$  and  $a \mid c$ . Then there exist integers  $n$  and  $m$  such that  $b = na$  and  $c = ma$ . Let  $x, y \in \mathbb{Z}$ . Thus,

$$bx + cy = anx + amy = a(nx + my) = at,$$

where  $t = nx + my \in \mathbb{Z}$ . It follows that  $a$  divides  $bx + cy$ .

- (iv) Suppose  $a \mid b$  and  $a \mid (b + c)$ . Then there exist integers  $m$  and  $n$  such that  $b = na$  and  $b + c = ma$ . Now

$$c = ma - b = ma - na = (m - n)a, \text{ and } m - n \text{ is an integer.}$$

This implies that  $a \mid c$ .

- (v) Suppose  $a \mid b$  and  $b \mid a$ . Then there exist integers  $m$  and  $n$  such that  $b = ma$  and  $a = nb$ . Thus,

$$a = nb = nma.$$

Because  $a \neq 0$  and  $a = nma$ , by the cancellation law,  $nm = 1$ . As  $m$  and  $n$  are integers and  $nm = 1$ , we conclude that either  $n = m = 1$  or  $n = m = -1$ . If  $n = 1$ , then  $a = b$  and if  $n = -1$ , then  $a = -b$ . Consequently,  $a = \pm b$ . ■

- REMARK 2.1.15** ▶ The properties of integers listed in Theorem 2.1.14 are very useful. Quite often, we have applied these results to prove other properties as well as solve problems. The next example illustrates an instance where we apply these results.

**EXAMPLE 2.1.16**

Consider the integers 2, 8, and  $r$ . Suppose that  $2 \mid (8 + r)$ . We can apply Theorem 2.1.14(iii) to conclude that  $2 \mid r$  as follows. Let  $a = 2$ ,  $b = 8 + r$ , and  $c = 8$ . Then  $a \mid b$  and  $a \mid c$ . Choose  $x = 1$  and  $y = -1$ . Then  $a$  divides  $bx + cy$ . However,  $bx + cy = (8 + r) - 8 = r$ . Thus  $2 \mid r$ .

We can also apply Theorem 2.1.14(iv) to conclude the result as follows: Let  $a = 2$ ,  $b = 8$ , and  $c = r$ . Now  $2 \mid 8$ , and so  $a \mid b$ . Also, it is given that  $a \mid (b + c)$ . Hence, by Theorem 2.1.14(iv)  $a \mid c$ , i.e.,  $2 \mid r$ .

## Greatest Common Divisors

Next we discuss greatest common divisors—another concept related to integers that we have worked with since our first algebra or pre-algebra course. We will apply the results obtained here in the last section of this chapter.

Consider the integers 18 and 24. Now  $2 \mid 18$  and  $2 \mid 24$ ;  $3 \mid 18$  and  $3 \mid 24$ ; and  $6 \mid 18$  and  $6 \mid 24$ . That is, the integers 18 and 24 are divisible by 2, 3, and 6. These integers 2, 3, and 6 are called the **common divisors** of 18 and 24. More formally, we have the following definitions.

---

**DEFINITION 2.1.17** ► A nonzero integer  $d$  is said to be a **common divisor** of integers  $a$  and  $b$  if  $d \mid a$  and  $d \mid b$ .

As shown above, the integers 2, 3, and 6 divide both 18 and 24. However, 6 is the largest positive integer that divides both 18 and 24. Such an integer is called the **greatest common divisor** and is of our interest.

---

**DEFINITION 2.1.18** ► A nonzero integer  $d$  is said to be a **greatest common divisor (gcd)** of  $a$  and  $b$

- (i) if  $d$  is a common divisor of  $a$  and  $b$ ; and
- (ii) if  $c$  is a common divisor of  $a$  and  $b$ , then  $c$  is a divisor of  $d$ .

Let  $d$  and  $d_1$  be two greatest common divisors of integers  $a$  and  $b$ . Then by Definition 2.1.18(ii), we find that  $d \mid d_1$  and  $d_1 \mid d$ . Hence, by Theorem 2.1.14(v),  $d = \pm d_1$ . So it follows that two different gcd's of  $a$  and  $b$  differ in their sign only. Of the two gcd's of  $a$  and  $b$  the positive one is denoted by  $\text{gcd}(a, b)$ .

For example, consider the integers 8 and 14. Now, the divisors of 8 are  $\pm 1, \pm 2, \pm 4$ , and  $\pm 8$ ; and the divisors of 14 are  $\pm 1, \pm 2, \pm 7$ , and  $\pm 14$ . Thus, the common divisors of 8 and 14 are  $\pm 1, \pm 2$ . From this it follows that  $\pm 2$  are the gcd's of 8 and 14. However, we write  $\text{gcd}(8, 14) = 2$ .

Therefore, to find  $\text{gcd}(a, b)$  it is enough to consider only positive divisors of these two integers.

Moreover, notice that  $2 = \text{gcd}(8, 14) = 2 \cdot 8 + (-1)14$ . Also notice that

$$6 = \text{gcd}(18, 24) = 1 \cdot 24 + (-1)18$$

and

$$2 = \text{gcd}(36, 50) = 7 \cdot 36 - 5 \cdot 50.$$

That is, the gcd of integers  $a$  and  $b$ , not both  $a$  and  $b$  zero, can be expressed as  $\text{gcd}(a, b) = sa + tb$  for some integers  $s$  and  $t$ .

The following theorem guarantees that for any integers  $a$  and  $b$  both not zero,  $\text{gcd}(a, b)$  always exists. From the theoretical point of view, we prove this theorem

using the well-ordering principle. From a practical and applications point of view, we will describe algorithms to find the gcd of nonzero integers as well as to find the integers  $s$  and  $t$  such that  $\gcd(a, b) = sa + tb$ .

**Theorem 2.1.19:** Let  $a$  and  $b$  be two integers such that not both are zero. Then  $\gcd(a, b)$  exists. Moreover, if  $d = \gcd(a, b)$ , then there exist integers  $s$  and  $t$  such that  $d = sa + tb$ .

**Proof:** (We are given the integers  $a$  and  $b$  such that not both are zero. Our objective is to show that  $\gcd(a, b)$  exists. As in the proof of some of the results in the earlier sections, we use the well-ordering principle to show the existence of  $\gcd(a, b)$ . Moreover, we also need to show the existence of integers  $s$  and  $t$  such that  $\gcd(a, b) = sa + tb$ . To apply the well-ordering principle, we need to construct a set, say  $S$ , of some nonnegative integers. So in this case, what should this set  $S$  be? Let us take a look at the  $\gcd(a, b)$ . From our convention,  $\gcd(a, b) > 0$ . Also we need to find integers  $s$  and  $t$  such that  $\gcd(a, b) = sa + tb$ , i.e.,  $sa + tb > 0$ . This suggests that we can start with the set  $S$  of all integers of the form  $ua + vb > 0$ ; i.e.,  $ua + vb \geq 1$ , where  $u$  and  $v$  are integers and then apply the well-ordering principle to  $S$ .)

Let us consider the following set  $S$ .

$$S = \{ua + vb \mid u, v \in \mathbb{Z} \text{ and } ua + vb \geq 1\}.$$

Because  $a$  and  $b$  are not both zero, it follows that  $a^2 + b^2 \geq 1$  and so,

$$aa + bb = a^2 + b^2 \in S.$$

This implies that  $S$  is a nonempty subset of the set of nonnegative integers. Therefore, by the well-ordering principle,  $S$  has a smallest element, say  $d$ . Thus, we have  $d \in S$  and  $1 \leq d \leq x$  for all  $x \in S$ .

Because  $d \in S$ ,  $d = sa + tb$  for some integers  $s$  and  $t$ .

Next we show that this  $d$  is the gcd of  $a$  and  $b$ . To do so, we verify the two properties of Definition 2.1.18.

Let  $c$  be a common divisor of  $a$  and  $b$ , i.e.,  $c \mid a$  and  $c \mid b$ . By Theorem 2.1.14(iii), we have  $c \mid (sa + tb)$ , i.e.,  $c \mid d$ . Thus, any common divisor of  $a$  and  $b$  is also a divisor of  $d$ .

Next, we show that  $d \mid a$  and  $d \mid b$ .

Consider  $d$  and  $a$ . Because  $d \geq 1$ , by Theorem 2.1.6 (division algorithm), there exist integers  $q$  and  $r$  such that

$$a = qd + r \tag{2.7}$$

and  $0 \leq r < d$ . This implies that

$$r = a - qd = a - q(sa + tb) = (1 - qs)a + (-qt)b = ua + vb,$$

where  $u = 1 - qs \in \mathbb{Z}$  and  $v = -qt \in \mathbb{Z}$ .

If  $r > 0$ , then  $r = ua - vb \in S$ , which contradicts the fact that  $d$  is the smallest integer in  $S$  as  $0 \leq r < d$ . Therefore,  $r = 0$ . This implies that  $a = qd$ , by (2.7), and so  $d \mid a$ . Similarly, we can show that  $d$  divides  $b$ . Consequently,  $d = \gcd(a, b)$ .

We have already shown that  $d = sa + tb$  for some integers  $s$  and  $t$ . ■

## Determining the Greatest Common Divisors

Theorem 2.1.19 proves that for any two integers  $a$  and  $b$ , not both zero,  $\gcd(a, b)$  exists. But how do we compute  $\gcd(a, b)$  in general? There is an efficient algorithm for computing  $\gcd(a, b)$ . Before describing it, however, first we prove the following lemma, which is needed by the algorithm.

**Lemma 2.1.20:** If  $a$  and  $b$  are positive integers such that  $a = qb + r$ ,  $0 \leq r < b$ , then

$$\gcd(a, b) = \gcd(b, r).$$

**Proof:** Let  $d = \gcd(a, b)$  and  $e = \gcd(b, r)$ . Because  $d = \gcd(a, b)$ , we have  $d$  divides  $a$  and  $d$  divides  $b$ . Now  $d$  divides  $b$  and so we have  $d$  divides  $qb$ .

Now  $r = a - qb$ ,  $d \mid a$ , and  $d \mid qb$ . Hence, by Theorem 2.1.14(iii), we have  $d$  divides  $r$ . We have thus proved that  $d$  is a common divisor of  $b$  and  $r$ . Because  $e = \gcd(b, r)$ , it follows that  $d$  divides  $e$ . Similarly, we can show that  $e$  divides  $d$ . Now  $d$  and  $e$  are positive integers and  $d \mid e$  and  $e \mid d$ . Hence, by Theorem 2.1.14(v),  $d = e$ . ■

## Euclidean Algorithm for Finding the GCD

We now describe the algorithm, called the *Euclidean algorithm*, for computing  $d = \gcd(a, b)$ . We will also describe how to express  $d$  in the form  $d = sa + nb$ .

Consider two positive integers  $a$  and  $b$ . By the division algorithm, there exist integers  $q_1$  and  $r_1$  such that

$$a = q_1 b + r_1, \quad 0 \leq r_1 < b.$$

Now consider  $b$  and  $r_1$ . Suppose  $r_1 \neq 0$ . Then by the division algorithm, there exist integers  $q_2$  and  $r_2$  such that

$$b = q_2 r_1 + r_2, \quad 0 \leq r_2 < r_1.$$

Next consider  $r_1$  and  $r_2$ . If  $r_2 \neq 0$ , by the division algorithm, there exist integers  $q_3$  and  $r_3$  such that

$$r_1 = q_3 r_2 + r_3, \quad 0 \leq r_3 < r_2.$$

We continue this process. Therefore, we obtain:

$$\begin{aligned} a &= q_1 b + r_1, & 0 \leq r_1 < b. \\ \text{If } r_1 \neq 0, \quad b &= q_2 r_1 + r_2, & 0 \leq r_2 < r_1. \\ \text{If } r_2 \neq 0, \quad r_1 &= q_3 r_2 + r_3, & 0 \leq r_3 < r_2. \\ \text{If } r_3 \neq 0, \quad r_2 &= q_4 r_3 + r_4, & 0 \leq r_4 < r_3. \\ \text{If } r_4 \neq 0, \quad r_3 &= q_5 r_4 + r_5, & 0 \leq r_5 < r_4. \\ &\vdots \end{aligned}$$

Now the remainders  $r_1, r_2, r_3, \dots$ , form the following decreasing sequence of nonnegative integers:

$$b > r_1 > r_2 > r_3 > \dots \geq 0.$$

Because  $b$  is a fixed positive integer, we must encounter the remainder 0 after

a finite number of steps. Therefore, the process terminates after some steps. Suppose the process terminates after  $(k + 1)$  steps. Then we have

$$\begin{aligned} r_{k-2} &= q_k r_{k-1} + r_k, \quad 0 < r_k < r_{k-1} \\ r_{k-1} &= q_{k+1} r_k + 0. \end{aligned}$$

Now the repeated application of Lemma 2.1.20 yields

$$\begin{aligned} \gcd(a, b) &= \gcd(b, r_1) \\ &= \gcd(r_1, r_2) \\ &= \gcd(r_2, r_3) \\ &\vdots \\ &= \gcd(r_{k-1}, r_k). \end{aligned}$$

Because  $r_{k-1} = q_{k+1} r_k$ , it follows that

$$\gcd(r_{k-1}, r_k) = r_k.$$

Hence,

$$\gcd(a, b) = r_k.$$

The following algorithm determines the gcd of two integers.

**ALGORITHM 2.2:** Euclidean algorithm to find the gcd.

*Input:* Integers  $x$  and  $y$ ,  $x > y \geq 0$   
*Output:* The gcd of  $x$  and  $y$

```

1. function GCD( $x, y$ )
2. begin
3.    $a := x;$ 
4.    $b := y;$ 
5.   while ( $b \neq 0$ ) do
6.     begin
7.        $r := a \bmod b;$ 
8.        $a := b;$ 
9.        $b := r;$ 
10.    end
11.    return  $a;$ 
12. end

```

**EXAMPLE 2.1.21**

Let  $x = 132$  and  $y = 108$ . We use the preceding algorithm to find the  $\gcd(132, 108)$ .

At Line 3,  $a = x = 132$  and at Line 4,  $b = y = 108$ . Next, we execute the **while** loop at Line 5.

**Iteration 1:**

Line 7:  $r = a \bmod b = 132 \bmod 108 = 24$ ;

Line 8:  $a = 108$ ;

Line 9:  $b = 24$ ;

Because  $b$  is not zero:

### Iteration 2:

Line 7:  $r = a \bmod b = 108 \bmod 24 = 12$ ;

Line 8:  $a = 24$ ;

Line 9:  $b = 12$ ;

Because  $b$  is not zero:

### Iteration 3:

Line 7:  $r = a \bmod b = 24 \bmod 12 = 0$ ;

Line 8:  $a = 12$ ;

Line 9:  $b = 0$ ;

Because  $b$  is 0, the **while** loop stops.

**return**  $a$ ; that is, **return** 12.

Thus,  $\gcd(132, 108) = 12$ .

Next we show how to find integers  $s$  and  $t$  such that the

$$\gcd(a, b) = sa + tb.$$

From the equations derived in the beginning of this section,

$$\begin{aligned} a &= q_1 b + r_1, & 0 \leq r_1 < b, \\ b &= q_2 r_1 + r_2, & 0 \leq r_2 < r_1, \\ r_1 &= q_3 r_2 + r_3, & 0 \leq r_3 < r_2, \\ r_2 &= q_4 r_3 + r_4, & 0 \leq r_4 < r_3, \\ r_3 &= q_5 r_4 + r_5, & 0 \leq r_5 < r_4, \\ &\vdots \end{aligned}$$

we have

$$\begin{aligned} r_1 &= a - q_1 b = s_1 a + t_1 b & s_1 = 1, & t_1 = -q_1 \\ r_2 &= b - q_2 r_1 = b - q_2(a - q_1 b) & s_2 = -q_2, & \\ &= -q_2 a + (q_1 q_2 + 1)b = s_2 a + t_2 b, & t_2 = (q_1 q_2 + 1) \\ r_3 &= r_1 - q_3 r_2 = a - q_1 b - q_3(s_2 a + t_2 b) & s_3 = 1 - q_3 s_2 = s_1 - q_3 s_2, & \\ &= (1 - q_3 s_2)a + (-q_1 - q_3 t_2)b = s_3 a + t_3 b & t_3 = -q_1 - q_3 t_2 = t_1 - q_3 t_2 \\ r_4 &= r_2 - q_4 r_3 = (s_2 a + t_2 b) - q_4(s_3 a + t_3 b) & s_4 = s_2 - q_4 s_3, & \\ &= (s_2 - q_4 s_3)a + (t_2 - q_4 t_3)b = s_4 a + t_4 b & t_4 = t_2 - q_4 t_3 \\ r_5 &= r_3 - q_5 r_4 = (s_3 a + t_3 b) - q_5(s_4 a + t_4 b) & s_5 = s_3 - q_5 s_4, & \\ &= (s_3 - q_5 s_4)a + (t_3 - q_5 t_4)b = s_5 a + t_5 b & t_5 = t_3 - q_5 t_4 \\ &\vdots \\ r_k &= r_{k-2} - q_k r_{k-1} & s = s_{k-2} - q_k s_{k-1}, & \\ &= (s_{k-2} a + t_{k-2} b) - q_k(s_{k-1} a + t_{k-1} b) & t = t_{k-2} - q_k t_{k-1}. & \\ &= (s_{k-2} - q_k s_{k-1})a + (t_{k-2} - q_k t_{k-1})b = sa + tb \end{aligned}$$

**REMARK 2.1.22** ▶ As one can see starting at  $s_3$  and  $t_3$ , a pattern is developing. Using this pattern, given the integers  $a$  and  $b$  and  $b \neq 0$ , we can determine integers  $s$  and  $t$  such that  $\gcd(a, b) = r_k = sa + tb$ . Notice that the algorithm simultaneously finds the  $\gcd(a, b)$ . We leave it as an exercise for the reader to design such an algorithm.

We note that for any two integers  $m, n$ , which are not both zero, we can show that

$$\gcd(m, n) = \gcd(m, -n) = \gcd(-m, n) = \gcd(-m, -n).$$

Also we note that

$$d = sm + tn = (-s)(-m) + tn = sm + (-t)(-n) = (-s)(-m) + (-t)(-n).$$

We illustrate the above procedure for finding the gcd and integers  $s$  and  $t$  by the following example.

### EXAMPLE 2.1.23

Consider the integers 376 and 140.

By repeated application of the division algorithm, we get

$$\begin{aligned} 376 &= 2 \cdot 140 + 96 \\ 140 &= 1 \cdot 96 + 44 \\ 96 &= 2 \cdot 44 + 8 \\ 44 &= 5 \cdot 8 + 4 \\ 8 &= 2 \cdot 4 + 0. \end{aligned}$$

Thus,  $\gcd(376, 140) = 4$  and

$$\begin{aligned} 96 &= 376 - 2 \cdot 140 \\ 44 &= 140 - 1 \cdot 96 = 140 - 1 \cdot (376 - 2 \cdot 140) = -1 \cdot 376 + 3 \cdot 140 \\ 8 &= 96 - 2 \cdot 44 = (376 - 2 \cdot 140) - 2 \cdot (-1 \cdot 376 + 3 \cdot 140) = 3 \cdot 376 + (-8) \cdot 140 \\ 4 &= 44 - 5 \cdot 8 = (-1 \cdot 376 + 3 \cdot 140) - 5 \cdot (3 \cdot 376 + (-8) \cdot 140) \\ &= -16 \cdot 376 + 43 \cdot 140. \end{aligned}$$

Hence,

$$\gcd(376, 140) = 4 = -16 \cdot 376 + 43 \cdot 140.$$

### Relatively Prime Integers

Consider the integers 5 and 8. It follows that  $\gcd(5, 8) = 1$ . Such a pair of integers are also of some interest. Let us describe some their properties.

**DEFINITION 2.1.24** ▶ Two integers  $a$  and  $b$  are said to be **relatively prime** if  $\gcd(a, b) = 1$ .

For example, 17 and 22 are relatively prime and 1 is relatively prime to every integer  $n$ .

The following theorem gives a criterion that can be used to determine if a pair of integers are relatively prime.

**Theorem 2.1.25:** Let  $a$  and  $b$  be two integers not both zero. Then  $a$  and  $b$  are relatively prime if and only if  $1 = ra + tb$  for some integers  $r$  and  $t$ .

**Proof:** Suppose  $a$  and  $b$  are relatively prime. Then  $\gcd(a, b) = 1$ . Hence, from Theorem 2.1.19, there exist integers  $r$  and  $t$  such that  $1 = ra + tb$ .

Conversely, assume that  $1 = ra + tb$  for some integers  $r$  and  $t$ . Let  $d = \gcd(a, b)$ . Then  $d \mid a$  and  $d \mid b$ . This implies that  $d \mid (ra + tb)$ , by Theorem 2.1.14(iii), and so  $d \mid 1$ . Because  $d > 0$  and  $d \mid 1$ , it follows that  $d = 1$ , and so  $a$  and  $b$  are relatively prime. ■

## Least Common Multiples

We close this section by reviewing another concept from an algebra or pre-algebra course.

---

**DEFINITION 2.1.26** ▶ Let  $a$  and  $b$  be two nonzero integers. A nonzero integer  $m$  is said to be a **least common multiple (lcm)** of  $a$  and  $b$

- (i) if  $m$  is a common multiple of  $a$  and  $b$ ; i.e.,  $a \mid m$  and  $b \mid m$ , and
- (ii) if  $c$  is a common multiple of  $a$  and  $b$ , then  $c$  is a multiple of  $m$ ; i.e., if  $a \mid c$  and  $b \mid c$ , then  $m \mid c$ .

Let  $m$  and  $m_1$  be two least common multiples of two nonzero integers  $a$  and  $b$ . Then by Definition 2.1.26(ii), we find that  $m \mid m_1$  and  $m_1 \mid m$ . Hence, by Theorem 2.1.14(v),  $m = \pm m_1$ . So it follows that two different least common multiples of  $a$  and  $b$  differ in their sign only. Of the two lcm's of  $a$  and  $b$  the positive one is denoted by  $\text{lcm}[a, b]$ . For example, 56 and  $-56$  are the lcm's of 8 and 14. But we write  $\text{lcm}[8, 14] = 56$ . In the following theorem, we establish a relation between gcd and lcm.

**Theorem 2.1.27:** Let  $a$  and  $b$  be two nonzero integers. Then

$$\gcd(a, b)\text{lcm}[a, b] = |ab|.$$

**Proof:** Let  $d = \gcd(a, b)$  and  $m = \text{lcm}[a, b]$ . Because  $d = \gcd(a, b)$ , we have  $d \mid a$ ,  $d \mid b$  and so  $\frac{a}{d}$  and  $\frac{b}{d}$  are integers. Similarly,  $m = \text{lcm}[a, b]$  implies  $a \mid m$ ,  $b \mid m$  and so  $\frac{m}{a}$  and  $\frac{m}{b}$  are integers.

Now  $a \mid ab$  and  $b \mid ab$ . Because  $m = \text{lcm}[a, b]$ , it follows that  $m \mid ab$  and so  $\frac{ab}{m} = k$ , for some integer  $k$ . This implies that

$$a = \left(\frac{m}{b}\right)k \quad \text{and} \quad b = \left(\frac{m}{a}\right)k.$$

From this and the fact that  $\frac{m}{a}$  and  $\frac{m}{b}$  are integers, it follows that  $k$  is a common divisor of  $a$  and  $b$ . However,  $d = \gcd(a, b)$  and so we must have

$$k \mid d. \tag{2.8}$$

Now  $\frac{a}{d}$  and  $\frac{b}{d}$  are integers and  $a \mid (a\frac{b}{d})$ ,  $b \mid (b\frac{a}{d})$ . Thus,  $\frac{ab}{d}$  is a common multiple of  $a$  and  $b$ . Because  $m = \text{lcm}[a, b]$ , we find that

$$m \mid \frac{ab}{d}.$$

This implies  $\frac{ab}{d} = mt$ , for some integer  $t$ . Thus,  $ab = dmt$ . Now

$$\begin{aligned} ab &= d(mt) \\ \Rightarrow \frac{ab}{m} &= dt \\ \Rightarrow k &= dt \quad \text{because } k = \frac{ab}{m} \\ \Rightarrow d &\mid k. \end{aligned} \tag{2.9}$$

Now  $d \mid k$  and  $k \mid d$ . Hence, by Theorem 2.1.14(v),  $d = \pm k = \pm \frac{ab}{m}$ , and so  $dm = |ab|$ . ■

**REMARK 2.1.28** ▶ Let  $a$  and  $b$  be two nonzero integers. By Theorem 2.1.19,  $\gcd(a, b)$  exists. It now follows from Theorem 2.1.27, that  $\operatorname{lcm}[a, b]$  also exists.

By Theorem 2.1.27, we can compute the lcm of two nonzero integers from the gcd and the product of the integers  $a$  and  $b$ . Consider the integers  $a = 24$  and  $b = -20$ . Now

$$\gcd(24, -20) \operatorname{lcm}[a, b] = |24 \cdot (-20)| = 480.$$

Because  $\gcd(24, -20) = 4$ , it follows that

$$4 \cdot \operatorname{lcm}[a, b] = 480,$$

and so  $\operatorname{lcm}[a, b] = 120$ .

## WORKED-OUT EXERCISES

**Exercise 1:** Let  $a, b, c, d$ , and  $e$  be consecutive integers. Show that 5 divides one of these.

**Solution:** Because  $a, b, c, d$ , and  $e$  are consecutive integers, there exists an integer  $n$  such that

$$a = n, \quad b = n + 1, \quad c = n + 2, \quad d = n + 3, \quad e = n + 4.$$

By the division algorithm, there exist integers  $q$  and  $r$  such that

$$n = 5q + r, \quad \text{where } 0 \leq r < 5.$$

If  $r = 0$ , then  $a = n = 5q$  and so  $5 \mid a$ .

If  $r = 1$ , then  $e = n + 4 = 5q + 5 = 5(q + 1)$  and so  $5 \mid e$ .

If  $r = 2$ , then  $d = n + 3 = 5q + 5 = 5(q + 1)$  and so  $5 \mid d$ .

If  $r = 3$ , then  $c = n + 2 = 5q + 5 = 5(q + 1)$  and so  $5 \mid c$ .

If  $r = 4$ , then  $b = n + 1 = 5q + 5 = 5(q + 1)$  and so  $5 \mid b$ .

Hence, 5 divides one of  $a, b, c, d$ , or  $e$ .

**Exercise 2:** Show that for any integer  $n$ ,  $n(n+1)(n+5)$  is a multiple of 3.

**Solution:** We have

$$\begin{aligned} n(n+1)(n+5) &= n(n+1)((n+2)+3) \\ &= n(n+1)(n+2) + 3n(n+1). \end{aligned}$$

Proceeding as in Worked-Out Exercise 1, we can show that the product of any three consecutive integers is divisible by 3. Hence, 3 divides  $n(n+1)(n+2)$ . Also 3 divides  $3n(n+1)$ . Therefore, by Theorem 2.1.14(iii), 3 divides  $n(n+1)(n+2) + 3n(n+1) = n(n+1)(n+5)$ . Hence,  $n(n+1)(n+5)$  is a multiple of 3.

**Exercise 3:** For any integer  $n$ , show that  $7n+1$  and  $15n+2$  are relatively prime.

**Solution:** Let

$$\gcd(7n+1, 15n+2) = d.$$

Then  $d \mid (7n+1)$  and  $d \mid (15n+2)$ . Now

$$15n+2 = 2(7n+1) + n.$$

By Theorem 2.1.14(iv), it follows that  $d \mid n$ . Now  $d \mid (7n+1)$  and  $d \mid n$ . Therefore, again by Theorem 2.1.14(iv), we find that  $d \mid 1$ . This implies that  $d = 1$ . Hence,  $\gcd(7n+1,$

$15n + 2) = 1$ , and therefore  $7n + 1$  and  $15n + 2$  are relatively prime.

**Exercise 4:** Let  $m, n, r$ , and  $t$  be integers such that  $(m - r)$  divides  $mn + rt$ . Prove that  $m - r$  divides  $mt + nr$ .

**Solution:** Let us write  $x = (mt + nr) - (mn + rt)$ . Then

$$\begin{aligned}x &= (mt + nr) - (mn + rt) \\&= (m - r)t - (m - r)n = (m - r)(t - n).\end{aligned}$$

This implies that  $m - r$  divides  $(mt + nr) - (mn + rt)$ . By the given condition,  $m - r$  divides  $mn + rt$ . Hence, by Theorem 2.1.14(iv),  $m - r$  divides  $mt + nr$ . (To apply Theorem 2.1.14(iv), we can take  $a = m - r$ ,  $b = -(mn + rt)$ , and  $c = mt + nr$ .)

**Exercise 5:** Let  $m$  and  $n$  be integers not both zero. Prove that

$$\gcd(km, kn) = k \cdot \gcd(m, n)$$

for any positive integer  $k$ .

**Solution:** Let  $\gcd(m, n) = d_1$  and  $\gcd(km, kn) = d_2$ . Because  $d_1 = \gcd(m, n)$ , there exist integers  $s$  and  $t$  such that  $d_1 = ms + nt$ . This implies

$$kd_1 = kms + knt.$$

Because  $\gcd(km, kn) = d_2$ , we have  $d_2$  divides  $km$  and  $kn$ . Thus, by Theorem 2.1.14(iii),  $d_2$  divides  $kms + knt$ ; i.e.,  $d_2$  divides  $kd_1$ .

On the other hand,  $d_1$  divides  $m$  and  $n$  and so  $kd_1$  divides  $km$  and  $kn$ . But  $d_2$  is the gcd of  $km$  and  $kn$ . Therefore,  $kd_1$  divides  $d_2$ . Now  $d_2 \mid kd_1$  and  $kd_1 \mid d_2$ . Thus, by Theorem 2.1.14(v),  $d_2 = kd_1$ . Hence,

$$\gcd(km, kn) = d_2 = kd_1 = k \cdot \gcd(m, n).$$

**Exercise 6:** Let  $a, b$ , and  $c$  be integers such that  $\gcd(a, b) = 1$  and  $\gcd(a, c) = 1$ . Prove that

$$\gcd(a, bc) = 1.$$

**Solution:** By the hypothesis,  $\gcd(a, b) = 1$  and  $\gcd(a, c) = 1$ . Therefore, by Theorem 2.1.25, there exist integers  $p, q, r$ , and  $s$  such that  $ap + bq = 1$  and  $ar + cs = 1$ . This implies that

$$(ap + bq)(ar + cs) = 1.$$

Simplify this to get

$$apar + apcs + bqr + bqcs = 1,$$

i.e.,

$$a(par + pcs + bqr) + bc(qs) = 1.$$

Let us write  $x = par + pcs + bqr$  and  $y = qs$ . Hence, there exist integers  $x$  and  $y$  such that  $ax + bcy = 1$ . This implies, by Theorem 2.1.25,  $\gcd(a, bc) = 1$ .

**Exercise 7:** If  $a, b, c$  are integers such that  $a \neq 0$  and  $\gcd(a, bc) = 1$ , then prove that  $\gcd(a, b) = 1 = \gcd(a, c)$ .

**Solution:** Suppose  $\gcd(a, bc) = 1$ . Then by Theorem 2.1.25, there exist integers  $s$  and  $t$  such that  $as + bct = 1$ . Hence,  $as + b(ct) = 1$ . This shows that  $\gcd(a, b) = 1$ , by Theorem 2.1.25. Similarly,  $\gcd(a, c) = 1$ .

**Exercise 8:** Let  $a$  and  $b$  be integers such that not both are zero. Show that  $\gcd(a^2, b^2) = (\gcd(a, b))^2$ .

**Solution:** Let  $d = \gcd(a, b)$ . Then  $\gcd(\frac{a}{d}, \frac{b}{d}) = 1$ . By Worked-Out Exercise 6 of this section,

$$\gcd\left(\frac{a^2}{d^2}, \frac{b^2}{d^2}\right) = 1.$$

Again by Worked-Out Exercise 6,

$$\gcd\left(\frac{a^2}{d^2}, \frac{b^2}{d^2}\right) = 1.$$

Now

$$\begin{aligned}\gcd(a^2, b^2) &= \gcd\left(d^2 \frac{a^2}{d^2}, d^2 \frac{b^2}{d^2}\right) \\&= d^2 \gcd\left(\frac{a^2}{d^2}, \frac{b^2}{d^2}\right) \quad \text{by Worked-Out Exercise 5 of this section} \\&= d^2 \quad \text{because } \gcd\left(\frac{a^2}{d^2}, \frac{b^2}{d^2}\right) = 1 \\&= (\gcd(a, b))^2.\end{aligned}$$

**Exercise 9:** Find all positive integers  $x$  and  $y$  such that  $\gcd(x, y) = 8$  and  $\text{lcm}[x, y] = 64$ .

**Solution:** Because  $\gcd(x, y) = 8$ ,  $8 \mid x$  and  $8 \mid y$ . Thus,

$$x = 8r, \quad y = 8t,$$

for some integers  $r$  and  $t$  such that  $\gcd(r, t) = 1$ . Also, because  $\text{lcm}[x, y] = 64$ ,  $x \mid 64$  and  $y \mid 64$ . Therefore,

$$xm = 64, \quad yn = 64,$$

for some integers  $m$  and  $n$ . Now  $64 = xm = 8rm$  and  $64 = yn = 8tn$ . This implies that  $rm = 8$  and  $tn = 8$ . From this, it follows that  $r = 8, t = 1$  or  $r = 1, t = 8$ . Hence,  $x = 64, y = 8$  or  $x = 8, y = 64$ .

**Exercise 10:** Find the gcd of 615 and 1080, and find integers  $s$  and  $t$  such that  $\gcd(615, 1080) = 615s + 1080t$ .

**Solution:** By the division algorithm, we get

$$\begin{aligned}1080 &= 1 \cdot 615 + 465 \\615 &= 1 \cdot 465 + 150 \\465 &= 3 \cdot 150 + 15 \\150 &= 10 \cdot 15.\end{aligned}$$

This implies that  $\gcd(615, 1080) = 15$ . Now,

$$\begin{aligned} 465 &= 1080 - 1 \cdot 615 \\ 150 &= 615 - 1 \cdot 465 = 615 - 1 \cdot (1080 - 615) \\ &= 2 \cdot 615 - 1 \cdot 1080 \\ 15 &= 465 - 3 \cdot 150 = 465 - 3(2 \cdot 615 - 1 \cdot 1080) \\ &= (1080 - 1 \cdot 615) - 3(2 \cdot 615 - 1 \cdot 1080) \\ &= 615 \cdot (-7) + 1080 \cdot (4). \end{aligned}$$

We can also determine  $s$  and  $t$  in the following manner.

$$\begin{aligned} 15 &= 465 - 3 \cdot 150 \\ &= 465 - 3 \cdot (615 - 1 \cdot 465) \\ &= 4 \cdot 465 - 3 \cdot 615 \\ &= 4 \cdot (1080 - 1 \cdot 615) - 3 \cdot 615 \\ &= 615(-7) + 1080(4). \end{aligned}$$

Hence, we can take  $s = -7$  and  $t = 4$ .

## SECTION REVIEW

### Key Terms

Pythagorean triple	additive identity	common divisor
well-ordering principle	multiplicative identity	greatest common divisor
the division algorithm	greater than	relatively prime
div	remainder	least common multiple
mod	divide	
quotient	divisor	

### Some Key Definitions

- Let  $a$  and  $b$  be integers. Then  $a$  is said to be greater than  $b$ , written  $a > b$ , if  $a - b$  is a positive integer. If  $a > b$ , then sometimes we also write  $b < a$ .
- Let  $a$  and  $b$  be two integers such that  $a \neq 0$ . If there exists an integer  $c$  such that  $b = ac$ , then  $a$  is said to divide  $b$  or  $a$  is said to be a divisor of  $b$  and we write  $a | b$ . A nonzero integer  $d$  is said to be a common divisor of integers  $a$  and  $b$  if  $d | a$  and  $d | b$ .
- A nonzero integer  $d$  is said to be a greatest common divisor of  $a$  and  $b$ 
  - if  $d$  is a common divisor of  $a$  and  $b$ ; and
  - if  $c$  is a common divisor of  $a$  and  $b$ , then  $c$  is a divisor of  $d$ .

### Some Key Results

- The well-ordering principle: Any nonempty subset of nonnegative integers has a least (i.e., smallest) element. That is, if  $S$  is a nonempty subset of the set of nonnegative integers, then there exists  $n \in S$  such that  $n \leq m$  for all  $m \in S$ .
- The division algorithm: Let  $a$  and  $b$  ( $\geq 1$ ) be integers. Then there exist unique integers  $q$  and  $r$  such that  $a = bq + r$ , where  $0 \leq r < b$ .
- Let  $a$  and  $b$  be two integers such that both are not zero. Then  $\gcd(a, b)$  exists. Moreover, if  $d = \gcd(a, b)$ , then there exist integers  $s$  and  $t$  such that  $d = sa + tb$ .

4. Let  $a$  and  $b$  be two integers such that both are not zero. Then  $a$  and  $b$  are relatively prime if and only if  $1 = ra + tb$  for some integers  $r$  and  $t$ .

## EXERCISES

---

1. Find the quotient  $q$  and the remainder  $r$  such that  $a = bq + r$ , where  $0 \leq r < |b|$ , in each of the following cases.
  - a.  $a = 600, b = 27$
  - b.  $a = -600, b = 27$
  - c.  $a = -600, b = -27$
  - d.  $a = 600, b = -27$
2. Find  $730 \bmod 4$  and  $-309 \bmod 7$ .
3. Find  $3092 \bmod 5$  and  $-7308 \bmod 11$ .
4. For any integer  $n$ , prove that  $n$  must be one of the forms  $8k, 8k+1, 8k+2, 8k+3, 8k+4, 8k+5, 8k+6$ , or  $8k+7$ , where  $k \in \mathbb{Z}$ . Hence, prove that the square of any odd integer is of the form  $8t+1$  for some integer  $t$ .
5. Prove that the product of four consecutive integers is divisible by 4.
6. Prove that for any two integers  $m$  and  $n$ , either both  $m+n$  and  $m-n$  are even or both  $m+n$  and  $m-n$  are odd.
7. For any integer  $n$ , prove that
  - a. 3 divides  $n^3 - n$ .
  - b. 3 divides one of the integers  $n, n+1$ , or  $2n+1$ .
  - c. 3 divides one of  $n, n+2$ , or  $n+4$ .
  - d. 3 divides one of  $n, 2n-1$ , or  $2n+1$ .
8. If  $a > 1$  is a positive integer and  $x$  is some integer such that  $a \mid (13x+7)$  and  $a \mid (39x+4)$ , then find  $a$ .
9. If  $a > 1$  is a positive integer and  $x$  is any integer such that  $a \mid (5x+4)$  and  $a \mid (65x+9)$  for some integer  $x$ , then find  $a$ .
10. Show that for any integer  $n$ ,  $5 \mid (n^5 - n)$  and  $3 \mid (n^5 - n)$ .
11. Prove Theorem 2.1.3.
12. Let  $a, b$ , and  $c$  be integers such that  $a \neq 0$ . If  $a \mid b$ , then prove that  $a \mid bc$ .
13. Let  $a, b$ , and  $c$  be integers such that  $a \neq 0$ . If  $a \mid b$  and  $a \mid c$ , then prove that  $a^2 \mid bc$ .
14. Let  $a, b$ , and  $c$  be integers such that  $a \neq 0$  and  $c \neq 0$ . Prove that  $a \mid b$  if and only if  $ac \mid bc$ .
15. Let  $a$  and  $b$  be integers such that  $a > 0$  and  $b > 0$ . If  $a \mid b$  and  $b \mid a$ , then prove that  $a = b$ .
16. Let  $a, b, c$ , and  $d$  be integers such that  $a \neq 0$  and  $b \neq 0$ . If  $a \mid c$  and  $b \mid d$ , then prove that  $ab \mid cd$ .
17. If  $m$  and  $n$  are two integers such that  $m \neq 0$  and  $m \mid n$ , then prove that  $m^k \mid n^k$  for any positive integer  $k$ .
18. Let  $m$  and  $n$  be integers such that not both are zero and  $\gcd(m, n) = d$ . Prove that the integers  $\frac{m}{d}$  and  $\frac{n}{d}$  are relatively prime.
19. Let  $a$  and  $b$  be integers such that not both are zero. If  $\gcd(a, b) = 1$ , then prove that  $\gcd(a+b, a-b) = 1$  or 2.
20. Let  $a$  and  $b$  be integers such that not both are zero. If  $\gcd(a, b) = 1$ , then prove that  $\gcd(a^2, b^2) = 1$ .
21. If  $a, b, c$  are pairwise relatively prime, then prove that  $\gcd(a, b)\gcd(a, c) = \gcd(a, bc)$ .
22. If  $n$  is odd, show that  $n(n^2 - 1)$  is a multiple of 24.
23. Let  $a$  and  $b$  be integers such that not both are zero. If  $\gcd(a, 4) = 2 = \gcd(b, 4)$ , then prove that  $\gcd(a+b, 4) = 4$ .
24. Let  $a$  and  $b$  be integers not both zero. Prove that  $\gcd(a, b) = \gcd(a, b+ac)$  for any integer  $c$ .
25. Let  $k$  be a positive integer. Find the  $\gcd(5k+3, 3k+2)$ .
26. If  $d$  is a positive integer such that  $d \mid (13n+6)$  and  $d \mid (12n+5)$  for some integer  $n$ , then prove that  $d = 1$  or 7.
27. For any two positive integers  $a$  and  $b$ , prove that  $(a+b) \cdot \text{lcm}[a, b] = b \cdot \text{lcm}[a, a+b]$ .
28. Prove that for any positive integer  $n$ , and for all positive integers  $a, b$ ,  $\text{lcm}[a, b]^n = \text{lcm}[a^n, b^n]$ .
29. If  $m, n, p, q$  are positive integers such that  $m \mid p$  and  $n \mid q$ , then prove that  $\gcd(m, n)$  divides  $\gcd(p, q)$ .
30. For any integer  $n$ , show that  $21n+4$  and  $14n+3$  are relatively prime.
31. For any integer  $n$ , show that  $3n+1$  and  $13n+4$  are relatively prime.
32. Find the gcd of the integers 4235 and 315 and find  $s$  and  $t$  such that  $\gcd(4235, 315) = 4235s + 315t$ .
33. Find the  $\gcd(2274, 174)$  and express it in the form  $2274s + 174t$ , where  $s$  and  $t$  are integers.
34. If  $n$  is a positive integer greater than 1, then prove that  $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$  is not an integer.
35. Let  $a, b$ , and  $c$  be positive integers. Prove that

$$\gcd(a^b - 1, a^c - 1) = a^{\gcd(b, c)} - 1.$$

## 2.2 REPRESENTATION OF INTEGERS IN COMPUTER

---

As we know, computers are electronic devices. Electrical signals are used inside the computer to process information. There are two types of electrical signals: analog and digital. *Analog signals* are continuous wave forms used to represent such things as sound. Audiotapes, for example, store data in analog signals. *Digital signals* represent information with a sequence of 0's and 1's. A 0 represents a low voltage, and

a 1 represents a high voltage. Digital signals are more reliable carriers of information than analog signals and can be copied from one device to another with exact precision. We might have noticed that when we make a copy of an audiotape, the sound quality of the copy is not as good as that of the original tape. This is because analog signals may not be copied with exact precision. Therefore, of the two electrical signals, the computer uses digital signals to achieve the greatest precision. Because digital signals are processed inside a computer, the language of a computer, called **machine language**, is a sequence of 0's and 1's. The digit 0 or 1 is called a **binary digit**, or **bit**. Sometimes a sequence of 0's and 1's is referred to as *binary code*.

The number system that we use in our daily lives is called the **decimal system**, or **base-10** system. The digits that are used to represent numbers in base 10 are 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9. However, in computer memory numbers are stored in machine language, i.e., as a sequence of 0's and 1's. The number system that a computer uses to store and manipulate numbers is called *binary* or **base 2**. For example, the number 9 in base 10 may be stored as 1001 in base 2, and the number 65 may be stored as 1000001 in base 2.

Other than base 2, two more number systems of interest to computer scientists are **base 8 (octal)** and **base 16 (hexadecimal)**. The digits that are used to represent numbers in base 8 are 0, 1, 2, 3, 4, 5, 6, and 7. The digits and letters that are used to represent numbers in base 16 are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F.

The main objective of this section is to discuss how to represent numbers in base 2 and how to perform arithmetic operations—addition and subtraction—on binary numbers. We will therefore, discuss how to convert numbers from base 10 to base 2 (as well as base 8 and base 16) and vice versa.

In fact, just as we can represent numbers in base 2, base 8, base 10, and base 16, we can also represent numbers in other bases, such as base 3 and base 20. The methods that we describe to convert a base-10 number to base 2 (and vice versa) are also applicable to other bases. Therefore, when describing such results we will focus on just base 2.

Consider the number 215. We can write

$$215 = 2 \cdot 10^2 + 1 \cdot 10 + 5.$$

Moreover,

$$215 = 1 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1$$

and

$$215 = 13 \cdot 16 + 7.$$

In general, we have the following result.

**Theorem 2.2.1:** Let  $k > 1$  be an integer. Then any positive integer  $n$  can be expressed uniquely as

$$n = a_m k^m + a_{m-1} k^{m-1} + a_{m-2} k^{m-2} + \dots + a_1 k^1 + a_0,$$

where  $m$  is a nonnegative integer  $a_m \neq 0$  and each  $0 \leq a_i \leq k - 1$  for  $i = 0, 1, 2, \dots, m$ .

**Proof:** By the division algorithm, there exist integers  $q_0$  and  $a_0$  such that

$$n = q_0 k + a_0,$$

where  $0 \leq a_0 \leq k - 1$ . Next, consider the integers  $q_0$  and  $k$ . If  $q_0 \neq 0$ , by the division algorithm, we obtain integers  $q_1$  and  $a_1$  such that

$$q_0 = q_1 k + a_1,$$

where  $0 \leq a_1 \leq k - 1$ . If  $q_1 \neq 0$ , we repeat the process with  $q_1$  and  $k$ . Thus, we obtain

$$\begin{aligned} n &= q_0 k + a_0, \quad \text{where } 0 \leq a_0 \leq k - 1, \\ q_0 &= q_1 k + a_1, \quad \text{where } 0 \leq a_1 \leq k - 1, \\ q_1 &= q_2 k + a_2, \quad \text{where } 0 \leq a_2 \leq k - 1, \\ &\vdots \end{aligned}$$

where  $n > q_0 > q_1 > q_2 > \dots$

Because there exist a finite number of positive integers smaller than  $n$ , the above process must terminate. So after a finite number of steps, we obtain

$$\begin{aligned} n &= q_0 k + a_0, \quad \text{where } 0 \leq a_0 \leq k - 1, q_0 \neq 0, \\ q_0 &= q_1 k + a_1, \quad \text{where } 0 \leq a_1 \leq k - 1, q_1 \neq 0, \\ q_1 &= q_2 k + a_2, \quad \text{where } 0 \leq a_2 \leq k - 1, q_2 \neq 0, \\ &\vdots \\ q_{m-2} &= q_{m-1} k + a_{m-1}, \quad \text{where } 0 \leq a_{m-1} \leq k - 1, q_{m-1} \neq 0, \\ q_{m-1} &= q_m k + a_m, \quad \text{where } 0 < a_m \leq k - 1, q_m = 0. \end{aligned}$$

Then

$$\begin{aligned} n &= q_0 k + a_0 \\ &= (q_1 k + a_1)k + a_0 \\ &= q_1 k^2 + a_1 k + a_0 \\ &= (q_2 k + a_2)k^2 + a_1 k + a_0 \\ &= q_2 k^3 + a_2 k^2 + a_1 k + a_0 \\ &\vdots \\ &= a_m k^m + a_{m-1} k^{m-1} + \cdots + a_2 k^2 + a_1 k + a_0, \end{aligned}$$

where  $0 \leq a_i \leq k - 1$ ,  $i = 0, 1, 2, \dots, m - 1$ , and  $0 < a_m \leq k - 1$ .

It is a simple exercise to show that the above representation is unique. ■

Notice that in the representation

$$n = a_m k^m + a_{m-1} k^{m-1} + a_{m-2} k^{m-2} + \cdots + a_1 k^1 + a_0,$$

if  $k = 10$ , then  $a_i \in \{0, 1, 2, 3, 4, \dots, 8, 9\}$ ,  $i = 0, 1, 2, \dots$ . Similarly, if  $k = 2$ , then each  $a_i$  is either 0 or 1.

### EXAMPLE 2.2.2

Let  $n = 35$  and  $k = 2$ .

$$\begin{aligned} 35 &= 17 \cdot 2 + 1; \quad q_0 = 17, \quad a_0 = 1, \\ 17 &= 8 \cdot 2 + 1; \quad q_1 = 8, \quad a_1 = 1, \\ 8 &= 4 \cdot 2 + 0; \quad q_2 = 4, \quad a_2 = 0, \\ 4 &= 2 \cdot 2 + 0; \quad q_3 = 2, \quad a_3 = 0, \\ 2 &= 1 \cdot 2 + 0; \quad q_4 = 1, \quad a_4 = 0, \\ 1 &= 0 \cdot 2 + 1; \quad q_5 = 0, \quad a_5 = 1 \quad \text{quotient } q_5 = 0, \text{ we stop} \end{aligned}$$

Hence,  $35 = 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2 + 1$ .

The preceding successive division by 2 can also be displayed as:

	Quotients	Remainders
2	35	$1 = a_0$
2	17	$1 = a_1$
2	8	$0 = a_2$
2	4	$0 = a_3$
2	2	$0 = a_4$
2	1	$1 = a_5$
	0	

We can also determine the binary representation of 35 as follows. First find the largest power of 2 in 35. Now  $2^5 = 32 < 35$  and  $2^6 = 64 > 35$ . So the largest power of 2 in 35 is 5. Next, subtract  $2^5$  from 35, i.e.,  $35 - 2^5 = 35 - 32 = 3$ . So we now work with the number 3. The largest power of 2 in 3 is 1. Subtract  $2^1$  from 3 to get  $3 - 2^1 = 1$ . Finally, the largest power of 2 in 1 is 0. We can now write  $35 = 32 + 2 + 1 = 2^5 + 2^1 + 1 = 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1$ .

---

**DEFINITION 2.2.3** ▶ Let  $n$  be a positive integer and  $k > 1$  be an integer. Then by Theorem 2.2.1,  $n$  can be expressed uniquely as

$$n = a_m k^m + a_{m-1} k^{m-1} + a_{m-2} k^{m-2} + \cdots + a_1 k + a_0, \quad (2.10)$$

where  $m$  is a nonnegative integer  $a_m \neq 0$  and  $0 \leq a_i \leq k-1$  for all  $i = 0, 1, 2, \dots, m$ .

The unique representation (2.10) of  $n$  is called the *representation of  $n$  in base  $k$*  (or *base  $k$  representation of  $n$* ), and we write this representation as

$$n = (a_m a_{m-1} a_{m-2} \cdots a_1 a_0)_k. \quad (2.11)$$

---

**REMARK 2.2.4** ▶ A word of caution. In (2.11),  $a_m a_{m-1} a_{m-2} \cdots a_1 a_0$  is not a product of integers, it is just a symbolic notation. For example,  $(100011)_2$  represents

$$1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1,$$

which is a base-2 representation of 35.

---

**REMARK 2.2.5** ▶ Sometimes we write binary numbers such as  $(100011)_2$  as  $100011_2$ . That is, we omit the parentheses.

---

**DEFINITION 2.2.6** ▶ Let  $n$  be a positive integer and  $k > 1$  be an integer.

- (i) If  $k = 10$ , the base-10 representation (2.11) of  $n$  is called the **decimal representation of  $n$** .
- (ii) If  $k = 2$ , the base-2 representation (2.11) of  $n$  is called the **binary representation of  $n$** . In this case, we also call  $(a_m a_{m-1} a_{m-2} \cdots a_1 a_0)_2$  a **binary number**.
- (iii) If  $k = 8$ , the base-8 representation (2.11) of  $n$  is called the **octal representation of  $n$** .

**EXAMPLE 2.2.7**

In Example 2.2.2, we showed that

$$35 = 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2 + 1.$$

Therefore, the binary representation of

$$35 = (100011)_2.$$

Because  $35 = 3 \cdot 10 + 5$ , the representation of 35 in base 10 is  $(35)_{10}$ . Therefore, we can write

$$(35)_{10} = (100011)_2.$$

Moreover, notice that  $35 = 4 \cdot 8^1 + 3 \cdot 8^0$ . Hence,  $(35)_{10} = (43)_8$ .

**EXAMPLE 2.2.8**

Let  $n = 215$ . Then  $215 = 2 \cdot 10^2 + 1 \cdot 10 + 5$ . Thus,  $215 = (215)_{10}$ . Also,

$$215 = 1 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1.$$

This implies that  $215 = (11010111)_2$ . Hence, we have

$$(215)_{10} = (11010111)_2.$$

Moreover, notice that  $215 = 3 \cdot 8^2 + 2 \cdot 8^1 + 7 \cdot 8^0$ . Hence,  $(215)_{10} = (327)_8$ .

From the preceding discussion, it follows that when we represent a number in the base  $k > 1$ , we need  $k$  symbols 0, 1, 2, 3, ...,  $k - 2$ ,  $k - 1$ . For example, for  $k = 10$ , we use the symbols or digits 0, 1, 2, 3, 4, ..., 8, and 9; and for  $k = 2$ , we use the symbols or digits 0 and 1.

**REMARK 2.2.9 ▶**

- (i) If  $n$  is a positive integer, then the decimal representation of  $n$  is itself, i.e.,  $n = n_{10}$ .
- (ii) For any base  $k > 1$ ,  $0 = 0_k$ .

Now we make the following convention: If  $k > 10$ , then we write  $A$  for 10,  $B$  for 11,  $C$  for 12,  $D$  for 13,  $E$  for 14,  $F$  for 15, and so on.

**DEFINITION 2.2.10 ▶**

Let  $n$  be a positive integer. The base-16 representation (2.11) of  $n$  is called the **hexadecimal representation** of  $n$ .

For example,  $(B10CA3)_{16}$  is a hexadecimal representation of a number  $n$ .

**EXAMPLE 2.2.11**

It can be shown that

$$\begin{aligned}(10)_{10} &= (1010)_2 = A_{16}, \\ (11)_{10} &= (1011)_2 = B_{16}, \\ (12)_{10} &= (1100)_2 = C_{16}, \\ (13)_{10} &= (1101)_2 = D_{16}, \\ (14)_{10} &= (1110)_2 = E_{16}, \\ (15)_{10} &= (1111)_2 = F_{16}.\end{aligned}$$

**EXAMPLE 2.2.12**

Let  $n = 215$ . Then  $215 = 13 \cdot 16 + 7 = D \cdot 16 + 7$ . (Notice that 13 in base 16 is represented as  $D$ ). Thus,  $215 = (D7)_{16}$ . Moreover, using the results of Example 2.2.8, we have

$$(215)_{10} = (11010111)_2 = (D7)_{16}.$$

**EXAMPLE 2.2.13**

Consider  $(B10CA3)_{16}$ . Let us find the decimal representation of this number. Now

$$\begin{aligned}(B10CA3)_{16} &= B \cdot 16^5 + 1 \cdot 16^4 + 0 \cdot 16^3 + C \cdot 16^2 + A \cdot 16^1 + 3 \\&= 11 \cdot 16^5 + 1 \cdot 16^4 + 0 \cdot 16^3 + 12 \cdot 16^2 + 10 \cdot 16^1 + 3 \\&= 11603107 \\&= (11603107)_{10}.\end{aligned}$$

**Algorithm: Decimal to Binary**

In this section, we design a recursive algorithm to convert a nonnegative integer in base 10 to base 2. First we define some terms.

Let  $x$  be a nonnegative integer. We call the remainder of  $x$  after division by 2 the *rightmost bit* of  $x$ . Thus, the rightmost bit of 33 is 1 because  $33 \bmod 2$  is 1, and the rightmost bit of 28 is 0 because  $28 \bmod 2$  is 0.

We first illustrate the algorithm to convert a nonnegative integer in base 10 to the equivalent number in binary format with the help of an example.

Suppose we want to find the binary representation of 35. First, we divide 35 by 2. The quotient is 17, and the remainder—that is, the rightmost bit of 35—is 1. Next, we divide 17 by 2. The quotient is 8, and the remainder—that is, the rightmost bit of 17—is 1. Next, we divide 8 by 2. The quotient is 4, and the remainder—that is, the rightmost bit of 8—is 0. We continue this process until the quotient becomes 0.

The rightmost bit of 35 cannot be printed until we have printed the rightmost bit of 17. The rightmost bit of 17 cannot be printed until we have printed the rightmost bit of 8, and so on. Thus, the binary representation of 35 is the binary representation of 17 (that is, the quotient of 35 after division by 2), followed by the rightmost bit of 35.

Thus, to convert an integer  $n$  in base 10 into the equivalent binary number, we first convert the quotient  $n \div 2$  into an equivalent binary number and then append the rightmost bit of  $n$  to the binary representation of  $n \div 2$ .

**ALGORITHM 2.3: Decimal to Binary.**

*Input:*  $n$ —a nonnegative integer

*Output:* binary representation of  $n$

```

1. procedure decimalToBinary( $n$ )
2. begin
3.   if  $n > 0$  then
4.     begin
5.       decimalToBinary( $n \bmod 2$ );
6.       print  $n \bmod 2$ ;
7.     end
8.   end

```

**REMARK 2.2.14** ► The preceding algorithm, which converts a base-10 number to base 2, is recursive. In Section Representing Strings into Computer Memory in Chapter 5, we will give a nonrecursive, i.e., an iterative, algorithm.

### Algorithm: Binary to Decimal

The following algorithm determines the decimal representation of a binary number. We assume that the binary representation is entered as an integer.

Consider the binary number  $(1001101)_2$ . Now

$$(1001101)_2 = 1 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0.$$

It follows that each bit in the binary number is multiplied by some power of 2. The result is then added. For example, in the binary number  $(1001101)_2$ , the first bit 1 is multiplied by 2 to the power of 6. We call this power of 2 the *weight of the bit*.

The weight of each bit in the binary number is assigned from right to left. The weight of the rightmost bit is 0. The weight of the bit immediately to the left of the rightmost bit is 1, the weight of the bit immediately to the left of it is 2, and so on. For the binary number  $(1001101)_2$ , the weight of each bit is as follows.

Weight	6	5	4	3	2	1	0
	1	0	0	1	1	0	1

To write an algorithm that converts a binary number into the equivalent decimal number, we note two things:

1. The weight of each bit in the binary number must be known, and
2. the weight is assigned from right to left.

Because we do not know in advance how many bits are in the binary number, we must process the bits from right to left. After processing a bit, we can add 1 to its weight, giving the weight of the bit immediately to the left of it. Also, each bit must be extracted from the binary number and multiplied by 2 to the power of its weight. To extract a bit, we can use the mod operator. Consider the following recursive algorithm.

#### ALGORITHM 2.4: Binary to Decimal.

*Input:*  $n$ —a binary number

$d$  is initialized to 0 before the procedure is called.

$w$  is initialized to 0 before the procedure is called.

*Output:*  $d$ —decimal representation of  $n$

1. **procedure** **binaryToDecimal**( $n$ ,  $d$ ,  $w$ )
2. **begin**
3.     **if**  $n > 0$  **then**
4.         **begin**
5.             bit :=  $n \bmod 10$ ; //extract the rightmost bit
6.              $d := d + \text{bit} * 2^w$ ; //update  $d$
7.              $n := n \bmod 10$ ; //remove the rightmost bit
8.              $w := w + 1$ ; //increment weight for the next bit

```

9.    binaryToDecimal(n, d, w);
10.   end
11. end

```

**REMARK 2.2.15** ► The preceding algorithm, which converts a base-2 number to base 10, is recursive. In Section Representing Strings into Computer Memory in Chapter 5, we will give a nonrecursive, i.e., an iterative, algorithm.

## Operations on Binary Numbers

As discussed in the preceding section, numbers in computer memory are represented as binary numbers. Therefore, in this section, we discuss how to add and subtract binary numbers. First, we describe the addition of binary numbers.

Let us start with the addition of  $0_2$  with  $0_2$ . It is easy to see that  $0_2 + 0_2 = 0_2$ . Similarly,  $1_2 + 0_2 = 1_2 = 0_2 + 1_2$ .

Let us consider the addition of  $1_2$  and  $1_2$ . Now  $1_2 = 1_{10}$  and  $1_{10} + 1_{10} = 2_{10} = 10_2$ . From this it follows that

$$\begin{array}{r} 1_2 \\ + \quad 1_2 \\ \hline 10_2 \end{array}$$

That is, when we add  $1_2$  and  $1_2$ , the sum is 0 and the carry is 1.

Now consider the sum

$$1_2 + 1_2 + 1_2 = (1_2 + 1_2) + 1_2 = 10_2 + 1_2.$$

Now  $10_2 = 2_{10}$ . Therefore,  $10_2 + 1_2 = 2_{10} + 1_{10} = 3_{10} = 11_2$ . Thus,

$$\begin{array}{r} 1_2 \\ + \quad 1_2 \\ \hline 11_2 \end{array}$$

From this, it also follows that

$$\begin{array}{r} 10_2 \\ + \quad 1_2 \\ \hline 11_2 \end{array}$$

Next, let us add the binary numbers  $x = 1011010_2$  and  $y = 10110_2$ . To accomplish this, we add the corresponding digits of  $x$  and  $y$  from right to left (as shown below). Whenever the corresponding digits of  $x$  and  $y$  are 1, the sum is 0 and the carry is 1, which is to be added with the bits left to these bits. Let us write these numbers as follows and demonstrate the additions.

$$\begin{array}{r} 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad & \leftarrow \text{carry row} \\ 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad = x \\ + \qquad \qquad \qquad \qquad \qquad \qquad \qquad = y \\ \hline 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \end{array}$$

Thus,  $1011010_2 + 10110_2 = 1110000_2$ .

Let  $x = 101_2$  and  $y = 11_2$ . Then  $x = 1 \cdot 2^2 + 0 \cdot 2^1 + 1$  and  $y = 1 \cdot 2^1 + 1 = 0 \cdot 2^2 + 1 \cdot 2^1 + 1$ . Now

$$\begin{aligned}
 & x + y \\
 &= (1+0) \cdot 2^2 + (0+1) \cdot 2^1 + (1+1) \\
 &= (1+0) \cdot 2^2 + (0+1+1) \cdot 2^1 + 0 && \text{because } 1+1=10=1 \cdot 2^1+0 \\
 &= (1+0+1) \cdot 2^2 + 0 \cdot 2^1 + 0 && \text{because } (0+1+1) \cdot 2^1 = (10) \cdot 2^1 \\
 &= 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 && \text{because } (1+0+1) \cdot 2^2 = (10) \cdot 2^2 \\
 &= (1000)_2.
 \end{aligned}$$

Hence in summary,

$$\begin{aligned}
 & x + y \\
 &= (1+0) \cdot 2^2 + (0+1) \cdot 2^1 + (1+1) \\
 &= (1+0) \cdot 2^2 + (0+1+1) \cdot 2^1 + 0 && \text{because } 1+1=10, \text{ we carry 1} \\
 &= (1+0+1) \cdot 2^2 + 0 \cdot 2^1 + 0 && \text{because } 1+1=10, \text{ we carry 1} \\
 &= 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 && \text{because } 1+1=10, \text{ we carry 1} \\
 &= (1000)_2.
 \end{aligned}$$

We now describe the general method to add two arbitrary binary numbers  $s = (a_n a_{n-1} a_{n-2} \cdots a_1 a_0)_2$  and  $t = (b_m b_{m-1} b_{m-2} \cdots b_1 b_0)_2$ . In the process, we also explain the term “carry.” Suppose  $n \geq m$ . That is, the number of bits in  $s$  is greater than or equal to the number of bits in  $t$ . We can make the number of bits in  $t$  the same as the number of bits in  $s$  by putting 0’s in the front. That is, we can write

$$b_m b_{m-1} b_{m-2} \cdots b_1 b_0 = b_n b_{n-1} \cdots b_{m+1} b_m b_{m-1} b_{m-2} \cdots b_1 b_0,$$

where  $b_n = b_{n-1} = \cdots = b_{m+1} = 0$ .

Hence, we consider the given binary numbers as

$$(a_n a_{n-1} a_{n-2} \cdots a_1 a_0)_2$$

and

$$(b_n b_{n-1} b_{n-2} \cdots b_1 b_0)_2.$$

Now

$$(a_n a_{n-1} a_{n-2} \cdots a_1 a_0)_2 = a_n 2^n + a_{n-1} 2^{n-1} + \cdots + a_1 2^1 + a_0$$

and

$$(b_n b_{n-1} b_{n-2} \cdots b_1 b_0)_2 = b_n 2^n + b_{n-1} 2^{n-1} + \cdots + b_1 2^1 + b_0.$$

Therefore, let us add  $a_n 2^n + a_{n-1} 2^{n-1} + a_{n-2} 2^{n-2} + \cdots + a_1 2^1 + a_0$  and  $b_n 2^n + b_{n-1} 2^{n-1} + b_{n-2} 2^{n-2} + \cdots + b_1 2^1 + b_0$ . Now

$$\begin{aligned}
 & (a_n 2^n + a_{n-1} 2^{n-1} + \cdots + a_1 2^1 + a_0) + (b_n 2^n + b_{n-1} 2^{n-1} + \cdots + b_1 2^1 + b_0) \\
 &= (a_n + b_n) 2^n + (a_{n-1} + b_{n-1}) 2^{n-1} + \cdots + (a_1 + b_1) 2^1 + (a_0 + b_0)
 \end{aligned}$$

Consider  $a_0$  and  $b_0$ . Note that each of  $a_0$  and  $b_0$  is either 0 or 1, and so by the division algorithm,

$$a_0 + b_0 = c_0 2 + r_0,$$

where each of  $c_0$  and  $r_0$  is either 0 or 1. Then

$$\begin{aligned} & (a_n + b_n)2^n + (a_{n-1} + b_{n-1})2^{n-1} + \cdots + (a_2 + b_2)2^2 + (a_1 + b_1)2^1 + (a_0 + b_0) \\ &= (a_n + b_n)2^n + (a_{n-1} + b_{n-1})2^{n-1} + \cdots + (a_2 + b_2)2^2 + (a_1 + b_1)2^1 + c_02^1 + r_0 \\ &= (a_n + b_n)2^n + (a_{n-1} + b_{n-1})2^{n-1} + \cdots + (a_2 + b_2)2^2 + (a_1 + b_1 + c_0)2^1 + r_0 \end{aligned}$$

We say that  $c_0$  is the carry.

Now consider the sum,  $a_1 + b_1 + c_0$ . Again by the division algorithm,

$$c_1 + b_1 + c_0 = c_12^1 + r_1,$$

where each of  $c_1$  and  $r_1$  is either 0 or 1. Thus,

$$\begin{aligned} & (a_n + b_n)2^n + (a_{n-1} + b_{n-1})2^{n-1} + \cdots + (a_2 + b_2)2^2 + (a_1 + b_1 + c_0)2^1 + r_0 \\ &= (a_n + b_n)2^n + (a_{n-1} + b_{n-1})2^{n-1} + \cdots + (a_2 + b_2)2^2 + (c_12^1 + r_1)2^1 + r_0 \\ &= (a_n + b_n)2^n + (a_{n-1} + b_{n-1})2^{n-1} + \cdots + (a_2 + b_2 + c_1)2^2 + r_12^1 + r_0 \end{aligned}$$

Here  $c_1$  is the carry.

We repeat the process and obtain

$$(a_n a_{n-1} a_{n-2} \cdots a_1 a_0)_2 + (b_n b_{n-1} b_{n-2} \cdots b_1 b_0)_2 = (r_{n+1} r_n r_{n-1} r_{n-2} \cdots r_1 r_0)_2.$$

### EXAMPLE 2.2.16

Let us add the numbers  $125_{10}$  and  $38_{10}$  using binary addition.

Now  $125_{10} = 1111101_2$  and  $38_{10} = 100110_2$ . Thus,

$$\begin{array}{r} 1 & 1 & 1 & 1 & 1 & 0 & 0 & \leftarrow \text{carry row} \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & = 125_{10} \\ + & & 1 & 0 & 0 & 1 & 1 & 0 \\ \hline 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & = 38_{10} \end{array}$$

Next, we discuss the subtraction of binary numbers.

In the preceding section, we discussed only the binary representation of positive integers. Moreover, for any base  $k > 1$ ,  $0 = 0_k$ . Therefore, next, we only discuss how to subtract the binary representation of a smaller number from the binary representation of a larger, nonnegative integer. Of course, if the two numbers are the same, then the subtraction gives the result zero.

Let us start with subtracting single-digit binary numbers. It follows that  $1_2 - 0_2 = 1_2$  and  $1_2 - 1_2 = 0_2$ .

Next, let us subtract  $1_2$  from  $10_2$ . Now  $10_2 - 1_2 = 2_{10} - 1_{10} = 1_{10} = 1_2$ . Thus, let us, write these numbers as

$$\begin{array}{r} 10_2 \\ - 1_2 \\ \hline 1_2 \end{array}$$

Moreover, note that

$$\begin{aligned} 10_2 - 1_2 &= (1 \cdot 2 + 0) - (0 \cdot 2 + 1) \\ &= (1 - 0) \cdot 2 + (0 - 1) \\ &= (1 - 0 - 1) \cdot 2 + 1 \cdot 2 - 1 && \text{We borrow 1 from the previous coefficient to do the subtraction.} \\ &= 0 \cdot 2 + 1 \\ &= (1)_2. \end{aligned}$$

Next, we subtract  $1_2$  from  $100_2$ . Let us write these numbers as

$$\begin{array}{r} 100_2 \\ - \quad 1_2 \\ \hline \end{array}$$

Following the rule of subtracting base-10 numbers, we subtract corresponding digits from right to left. If the top digit is smaller than the bottom digit, we borrow a 1 from the top digit that is left to it. Remember that when we borrow  $1_2$  from  $10_2$ , the leftover digit is  $1_2$ . To subtract  $1_2$  from  $0_2$ , we need to borrow a 1 from the digit left of  $0_2$ . However, the digit left of 0 is also 0. Therefore, we need to borrow a 1 for the digit left to the second 0. Then the third digit (from right become 0), the second digit from right now is  $10_2$ , and after we borrow a 1 from it, the second digit from right becomes 1. Thus, we have

$$\begin{array}{r} 1 \ 10 \leftarrow \text{digit(s) to be subtracted from} \\ 1 \ 1 \leftarrow \text{borrow row} \\ \cancel{1} \ 0 \ 0 \\ - \qquad \qquad \qquad 1 \\ \hline 1 \ 1 \end{array}$$

Hence,

$$\begin{array}{r} 100_2 \\ - \quad 1_2 \\ \hline 11_2 \end{array}$$

Moreover, note that

$$\begin{aligned} & (100)_2 - (1)_2 \\ &= (1 \cdot 2^2 + 0 \cdot 2 + 0) - (0 \cdot 2^2 + 0 \cdot 2 + 1) \\ &= (1 - 0) \cdot 2^2 + (0 - 0) \cdot 2 + (0 - 1) \\ &= (1 - 0) \cdot 2^2 + (0 - 0 - 1) \cdot 2 + 1 \cdot 2 + (0 - 1) && \text{We borrow 1 from the previous coefficient.} \\ &= (1 - 0) \cdot 2^2 + (0 - 1) \cdot 2 + 1 && \text{because } 1 \cdot 2 + (0 - 1) = 1 \\ &= (1 - 0 - 1) \cdot 2^2 + 1 \cdot 2^2 + (0 - 1) \cdot 2 + 1 && \text{We borrow 1 from the previous coefficient.} \\ &= (1 - 0 - 1) \cdot 2^2 + 1 \cdot 2 + 1 && \text{because } 1 \cdot 2^2 + (0 - 1) \cdot 2 = 1 \cdot 2 \\ &= 0 \cdot 2^2 + 1 \cdot 2 + 1 \\ &= (011)_2 = (11)_2. \end{aligned}$$

### EXAMPLE 2.2.17

Next let us subtract  $101_2$  from  $1010_2$ . Let us write these numbers as

$$\begin{array}{r} 10 \ 0 \ 10 \leftarrow \text{digit(s) to be subtracted from} \\ 1 \ \ \ \ 1 \leftarrow \text{borrow row} \\ \cancel{1} \ \emptyset \ \cancel{1} \ 0 = 1010_2 \\ - \quad 1 \ 0 \ 1 = 101_2 \\ \hline 1 \ 0 \ 1 \end{array}$$

$$\begin{array}{r} 1010_2 \\ - \quad 101_2 \\ \hline 101_2 \end{array}$$

## Two's Complements and Operations on Binary Numbers

In computer memory, integers are represented as binary numbers in fixed-length bit strings, such as 8, 16, 32, and 64. For the sake of discussion and to make our calculations easy and comprehensible, let us assume that integers are represented as 8-bit fixed-length strings. Then  $0_{10}$  is represented as  $0000000_2$ ,  $1_{10}$  is represented as  $00000001_2$ , and  $49_{10}$  is represented as  $00110001_2$ .

As remarked above, base-10 numbers in computer memory are represented as binary numbers. In the preceding section, we discussed how to add and subtract numbers in binary form. However, the subtraction of binary numbers is quite complicated. In this section, first we discuss a convenient way to represent negative numbers in binary form and then we discuss how to subtract numbers in binary form.

Suppose integers are represented as fixed-length strings of 8 bits. One way to distinguish between negative and nonnegative numbers is to reserve the leftmost bit to represent the sign of the number. If the leftmost bit is 0, then the remaining 7 bits represent a nonnegative number. If the leftmost bit is a 1, then the remaining 7 bits represent a negative number. However, this convention would lead to complicated methods of addition and subtraction as well as a nonunique representation ( $00000000_2$  and  $10000000_2$ ) of 0.

A convenient way to represent negative numbers is by using the two's complement, which is described next.

---

**DEFINITION 2.2.18** ▶ Let  $n$  be a fixed positive integer and let  $x$  be a positive integer that can be represented as an  $n$ -bit binary number. The binary representation of

$$2^n - x$$

is called the **two's complement** of  $x$  with respect to  $n$ .

**EXAMPLE 2.2.19**

Let  $n = 8$  and  $x = 45$ . Then  $2^8 - 45 = 256 - 45 = 211$ . Now,  $211_{10} = 11010011_2$ , which is the binary representation of the two's complement of 45 as an 8-bit string.

**EXAMPLE 2.2.20**

Let  $n = 16$  and  $x = 2546$ . Then  $2^{16} - 2546 = 65536 - 2546 = 62990$ . It can be shown that  $62990_{10} = 1111011000001110_2$ , which is the binary representation of two's complement of 2546.

In Example 2.2.19, we showed that the 8-bit binary representation of the two's complement of 45 is  $11010011_2$ . Let us write the 8-bit binary representation of 45. We have

$$45_{10} = 00101101_2.$$

In this binary representation, change all 1's to 0's and all 0's to 1's to get

$$11010010_2.$$

This is called the binary representation of the one's complement of 45. Now add the 8-bit representation of 1, which is  $00000001$ , to it. We have

$$\begin{array}{r} 11010010 \\ + \quad 00000001 \\ \hline 11010011 \end{array}$$

Notice that this is the 8-bit representation of the two's complement of  $45_{10}$ , as shown in Example 2.2.19. This suggests that in order to find the  $n$ -bit binary representation of the two's complement of a number  $x$  we do the following:

1. Find the  $n$ -bit binary representation of  $x$ .
2. Find the one's complement of the  $n$ -bit binary representation of  $x$ .
3. Add 1, i.e., the  $n$ -bit binary representation of 1.

The resulting  $n$ -bit string is the  $n$ -bit binary representation of the two's complement of  $x$ .

---

**DEFINITION 2.2.21** ▶ Let  $s$  be an  $n$ -bit binary string, where  $n \geq 1$ . The **one's complement** of  $s$  is the  $n$ -bit binary string obtained from  $s$  by changing 1's to 0's and 0's to 1's.

**EXAMPLE 2.2.22**

Let  $n = 8$  and  $x = 121$ . Now

1. The 8-bit binary representation of 121 is

$$121_{10} = 01111001_2.$$

2. In this representation, change 0's to 1's and 1's to 0's to get

$$10000110_2.$$

3. Add 00000001 to it:

$$\begin{array}{r} 10000110 \\ + 00000001 \\ \hline 10000111 \end{array}$$

Thus, the binary representation of the two's complement of  $121_{10}$  is the string  $10000111_2$ .

This preceding method of finding the two's complement follows from the following fact (we use 8-bit representations for illustration): The two's complement of  $x$  is given by:

$$2^8 - x.$$

Now,

$$2^8 - x = ((2^8 - 1) - x) + 1.$$

Also

$$2^8 - 1 = 255,$$

and as an 8-bit string,

$$255_{10} = 11111111_2.$$

Next, we determine the binary representation of

$$(2^8 - 1) - x.$$

Now, the 8-bit representation of  $2^8 - 1$  is 11111111. Therefore, to determine the binary representation of  $(2^8 - 1) - x$ , we subtract the 8-bit binary representation of  $x$  from the binary number 11111111<sub>2</sub>.

Suppose that the binary representation of  $x$  is  $a_7a_6a_5a_4a_3a_2a_1a_0$ , where each  $a_i$  is 0 or 1. Let us write the binary representations of  $(2^8 - 1) - x$  as follows.

$$\begin{array}{cccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ - & a_7 & a_6 & a_5 & a_4 & a_3 & a_2 & a_1 & a_0 \\ \hline b_7 & b_6 & b_5 & b_4 & b_3 & b_2 & b_1 & b_0 \end{array} = (2^8 - 1)_2 = 255_{10}$$

$$= x$$

$$= ((2^8 - 1) - x)_2$$

It follows that  $b_i = 1 - a_i$  for all  $i$ ,  $0 \leq i \leq 7$ . If  $a_i = 1$ , then  $b_i = 1 - a_i = 1 - 1 = 0$ . If  $a_i = 0$ , then  $b_i = 1 - a_i = 1 - 0 = 1$ . We therefore see that

$$b_i = \begin{cases} 1 & \text{if } a_i = 0, \\ 0 & \text{if } a_i = 1. \end{cases}$$

We can now conclude that to determine the binary representation of  $(2^8 - 1) - x$  in the 8-bit binary representation of  $x$ , we simply change 0's 1's and 1's to 0's.

Finally, to determine the binary representation of  $2^8 - x = ((2^8 - 1) - x) + 1$ , we add the 8-bit binary representation of  $1_{10}$ , which is  $00000001_8$ .

---

**REMARK 2.2.23** ► The preceding process for determining the two's complement of a positive integer  $x$  works for any  $n$ -bit representation of  $x$ , where  $n > 1$ , as stated in the next theorem.

**Theorem 2.2.24:** Let  $x$  be a positive integer that can be expressed as an  $n$ -bit binary number, say  $(x_{n-1}x_{n-2}\cdots x_1x_0)_2$ . Let  $(y_{n-1}y_{n-2}\cdots y_1y_0)_2$  be such that  $y_i = 1 - x_i$  for all  $i = 0, 1, \dots, n - 1$ . Then the binary representation of the two's complement of  $x$  is the number whose binary representation is given by

$$(y_{n-1}y_{n-2}\cdots y_1y_0)_2 + (00\ldots 01)_2,$$

where  $(00\ldots 01)_2$  is the  $n$ -bit representation of 1.

### EXAMPLE 2.2.25

Let  $x = 5563_{10}$  and  $n = 16$ . We determine the 16-bit binary representation of the two's complement of  $x$ .

1.  $5563_{10} = 0001010110111011_2$ .
2. Change 0's to 1's and 1's to 0's to get the string:  $1110101001000100_2$ .
3. Add  $0000000000000001_2$  to the string in (2):

$$\begin{array}{r} 1110101001000100 \\ + 0000000000000001 \\ \hline 1110101001000101 \end{array}$$

Hence, the 16-bit binary representation of the two's complement of  $5563_{10}$  is  $1110101001000101_2$ .

---

**REMARK 2.2.26** ► Let  $n \geq 1$  be an integer and let  $x$  be an integer that can be represented as an  $n$ -bit binary string. Then  $y = 2^n - x$  is the two's complement of  $x$ . The two's

complement of  $y$  is

$$2^n - y = 2^n - (2^n - x) = 2^n - 2^n + x = x.$$

This implies that the two's complement of the two's complement of  $x$  is  $x$ .

**REMARK 2.2.27** ▶ The  $n$ -bit binary representation of the two's complement of  $0_{10}$  is  $0_{10}$ . For example, suppose  $n = 8$ . The 8-bit binary representation of 0 is 00000000. Now change 1's to 0's and 0's to 1's to get 11111111. Next add 00000001. That is,

$$\begin{array}{r} 11111111 \\ + \quad 00000001 \\ \hline 100000000 \end{array}$$

Notice that the resulting string has 9 bits. However, because only 8 bits are used to store the integer, we discard the leftmost bit, which in this case is 1. Hence, the 8-bit binary representation of the two's complement of  $0_{10}$  is 00000000, which is  $0_{10}$ .

**DEFINITION 2.2.28** ▶ Let  $n \geq 1$  be an integer and let  $x$  be a nonnegative integer that can be expressed as an  $n$ -bit binary number. The first digit, i.e., first bit, of the  $n$ -bit binary string representing  $x$  is called the **leading bit** of  $x$ .

**EXAMPLE 2.2.29** Let  $n = 8$ . The leading bit of 00111010 is 0, and the leading bit of 10101100 is 1.

**Theorem 2.2.30:** Let  $n = 8$ . Let  $x$  be an integer such that  $1 \leq x \leq 127$  and let  $y$  be an integer such that  $128 \leq y \leq 255$ . The following assertions hold.

- (i)  $x$  can be expressed as an 8-bit binary string.
- (ii) The leading bit of  $x$  is 0.
- (iii)  $y$  can be expressed as an 8-bit binary string.
- (iv) The leading bit of  $y$  is 1.
- (v) The leading bit of the two's complement of  $x$  is 1.
- (vi) The leading bit of the two's complement of  $y$  is 0.

#### Proof:

- (i) It can be checked that  $127_{10} = 0111111_2$  and  $1_{10} = 00000001_2$ . Let  $x$  be an integer such that  $1 \leq x \leq 127$ . Now  $2^7 = 128$ . Suppose that

$$x = (a_n a_{n-1} \cdots a_1 a_0)_2$$

is the binary representation of  $x$ . Then

$$x = a_n \cdot 2^n + a_{n-1} \cdot 2^{n-1} + \cdots + a_1 \cdot 2 + a_0,$$

where  $a_n = 1$ . This implies that

$$x \geq a_n \cdot 2^n.$$

Suppose  $n \geq 7$ . Then  $x \geq 1 \cdot 2^7 = 128$ , which is a contradiction. Hence,  $n < 7$  or  $n \leq 6$ . Thus,

$$x = (a_6 a_5 a_4 a_3 a_2 a_1 a_0)_2.$$

This implies that we need at most 7 bits to represent  $x$  as a 7-bit binary string. If the number of bits in the binary representation of  $x$  are  $\leq 7$ , then we can add the necessary 0's to the left of the string to make it an 8-bit string. Hence,  $x$  can be expressed as an 8-bit binary string.

- (ii) By part (i),

$$x = (a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0)_2,$$

where  $a_7$  is the leading bit. This implies that

$$x = a_7 \cdot 2^7 + a_6 \cdot 2^6 + \cdots + a_1 \cdot 2 + a_0,$$

and so  $x \geq a_7 \cdot 2^7 = 128a_7$ . If  $a_7 = 1$ , then  $x \geq 128$ , a contradiction. Hence,  $a_7 = 0$  and so the leading bit of  $x$  is 0.

- (iii) The proof is similar to part (i).
- (iv) The proof is similar to part (ii).
- (v) Let  $x$  be such that

$$1 \leq x \leq 127 = 2^7 - 1.$$

Now

$$\begin{aligned} 1 &\leq x \leq 2^7 - 1 \\ \Rightarrow -1 &\geq -x \geq -(2^7 - 1) \\ \Rightarrow 2^8 - 1 &\geq 2^8 - x \geq 2^8 - (2^7 - 1) \\ \Rightarrow 255 &\geq 2^8 - x \geq 128, \end{aligned}$$

because  $2^8 - 1 = 256 - 1 = 255$  and  $2^8 - (2^7 - 1) = 256 - (128 - 1) = 129 \geq 128$ . Now  $2^8 - x$  is the two's complement of  $x$  and  $128 \leq 2^8 - x \leq 255$ . Hence, by part (iv), the leading bit of the two's complement of  $x$  is 1.

- (vi) The proof is similar to part (iv). ■

We leave the proof of the following theorem as an exercise. (See Exercise 36, p. 133.)

**Theorem 2.2.31:** Let  $n \geq 1$  be an integer and let  $x$  be a positive integer such that  $x$  can be expressed as an  $n$ -bit binary string.

- (i) If  $x = 2^{n-1}$ , then the leading bit of  $x$  and the leading bit of the two's complement of  $x$  is 1.
- (ii) If  $x = 0$ , the leading bit of  $x$  and the leading bit of the two's complement of  $x$  is 0.
- (iii) Let  $1 \leq x < 2^{n-1}$ . The leading bit of  $x$  is 0 and the leading bit of the two's complement of  $x$  is 1.
- (iv) Let  $2^{n-1} < x \leq 2^n - 1$ . The leading bit of  $x$  is 1 and the leading bit of the two's complement of  $x$  is 0.

Let  $x$  be an integer such that  $1 \leq x \leq 127$ . Then  $x$  can be expressed as an 8-bit string and the leading bit of  $x$  is 0. Consider  $y$  such that  $-127 \leq y \leq -1$ . Then

$y$  is a negative number and  $1 \leq |y| \leq 127$ . Suppose  $y$  is represented as the two's complement of  $|y|$ . By Theorem 2.2.30, the leading bit of  $|y|$  is 0. Again by Theorem 2.2.30, it follows that the leading bit of the two's complement of  $|y|$  is 1.

Suppose that negative numbers are represented as the two's complement of their absolute number. For example,  $-65$  is represented as the two's complement of  $65$ .

We can conclude that 8 bits can be used to represent integers in the range  $-128$  to  $127$ . Of course, negative numbers are represented as the two's complement of their absolute. For example,  $-65$  is represented as the two's complement of  $65$ . The following table shows the integers in the range  $-128$  to  $127$  and their binary representation.

Number	8-Bit Binary Representation	
127	01111111	
126	01111110	
:	:	
1	00000001	
0	00000000	
-1	11111111	$2^8 - 1$
:	:	:
-126	10000010	$2^8 - 126$
-127	10000001	$2^8 - 127$
-128	10000000	$2^8 - 128$

From this table it is clear that the leading bit of nonnegative integers is 0 and the leading bit of negative integers is 1.

---

**REMARK 2.2.32** ▶ Let  $n > 1$  be an integer. It can be shown that  $n$  bits are sufficient to represent any integer  $x$  such that  $-2^{n-1} \leq x \leq 2^{n-1} - 1$  as an  $n$ -bit string. Of course, negative numbers are represented as two's complements of their absolute values. For example, 16 bits are sufficient to represent any integer  $x$  such that  $-32768 = -2^{15} \leq x \leq 32767 = 2^{15} - 1$ , as a 16-bit string.

In the following examples, we show how numbers are added and subtracted using their binary representations, where negative numbers are represented as the two's complements of their absolute values.

### EXAMPLE 2.2.33

Let  $n = 8$ ,  $x = 67$ , and  $y = 38$ . We add  $x$  and  $y$  using their binary representation. Now

$$x = 67 = 01000011_2$$

and

$$y = 38 = 00100110_2.$$

Because both  $x$  and  $y$  are positive, we add their binary representation as in the

preceding section.

$$\begin{array}{r}
 & & 1 & 1 & & & & \\
 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & \leftarrow \text{carry row} \\
 + & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & = 67 \\
 \hline
 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & = 38
 \end{array}$$

Thus,  $01000011_2 + 00100110_2 = 01101001_2$ . Notice that  $01101001_2 = 105_{10}$ . Moreover,  $67 + 38 = 105$ .

### EXAMPLE 2.2.34

Let  $n = 8$ ,  $x = 67$ , and  $y = -38$ . We add  $x$  and  $y$  using their binary representation. Now

$$x = 67 = 01000011_2.$$

Because  $y < 0$ , we find the bit string of the two's complement of  $|y| = 38$ . Now,

$$38 = 00100110_2.$$

So

$$\begin{array}{r}
 \text{Change 1's to 0's} \\
 \text{and 0's to 1's} \\
 00100110 \xrightarrow{\quad\quad\quad} 11011001 \xrightarrow{\text{Add } 00000001} 11011010.
 \end{array}$$

Hence, the binary representation of  $y = -38$  is

$$11011010_2.$$

Next we add the bit strings of 67 and  $-38$ . We have:

$$\begin{array}{r}
 & & 1 & 1 & & & 1 & & \\
 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & \leftarrow \text{carry row} \\
 + & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & = -38 \\
 \hline
 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1
 \end{array}$$

Discard  
this bit

Notice that we get an extra bit, 1, to the left of the first bit, 0. This is called an overflow and this extra bit is discarded. Thus,  $01000011_2 + 11011010_2 = 00011101_2$ .

Also notice that  $00011101_2 = 29_{10}$ . Moreover,  $67 - 38 = 29$ .

### EXAMPLE 2.2.35

Let  $n = 8$ ,  $x = -67$ , and  $y = 38$ . We add  $x$  and  $y$  using their binary representation. Because  $x < 0$ , we find the binary representation of its two's complement. Now,

$$67 = 01000011_2.$$

We have

$$\begin{array}{r}
 \text{Change 1's to 0's} \\
 \text{and 0's to 1's} \\
 01000011 \xrightarrow{\quad\quad\quad} 10111100 \xrightarrow{\text{Add } 00000001} 10111101.
 \end{array}$$

Thus, the binary representation of  $x = -67$  is

$$10111101_2.$$

Now  $y > 0$ . The binary representation of  $y$  is

$$y = 38 = 00100110_2.$$

Next we add the bit strings of  $-67$  and  $38$ . We have:

$$\begin{array}{r} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\ + & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ \hline 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{array} \quad \leftarrow \text{carry row}$$

$$= -67$$

$$= 38$$

Thus,  $10111101_2 + 00100110_2 = 11100011_2$ . Now the leading bit of  $11100011$  is 1, so it is a negative number. So we find its two's complement (which will represent a positive integer). Now

$$11100011 \xrightarrow{\substack{\text{Change 1's to 0's} \\ \text{and 0's to 1's}}} 00011100 \xrightarrow{\text{Add 00000001}} 00011101.$$

Because  $00011101_2 = 29$ , the binary number  $11100011$  represents  $-29$ . Also, notice that  $-67 + 38 = -29$ .

### EXAMPLE 2.2.36

Let  $n = 8$ ,  $x = -67$ , and  $y = -38$ . We add  $x$  and  $y$  using their binary representation. Because  $x < 0$ , we find the binary representation of its two's complement. Now,

$$67 = 01000011_2.$$

We have

$$01000011 \xrightarrow{\substack{\text{Change 1's to 0's} \\ \text{and 0's to 1's}}} 10111100 \xrightarrow{\text{Add 00000001}} 10111101.$$

Thus, the binary representation of  $x = -67$  is

$$10111101_2.$$

Because  $y < 0$ , we find the bit string of the two's complement of  $|y| = 38$ . Now,

$$38 = 00100110_2.$$

So

$$00100110 \xrightarrow{\substack{\text{Change 1's to 0's} \\ \text{and 0's to 1's}}} 11011001 \xrightarrow{\text{Add 00000001}} 11011010.$$

Hence, the binary representation of  $y = -38$  is

$$11011010_2.$$

Next we add the bit strings of  $-67$  and  $-38$ . We have:

$$\begin{array}{r} 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\ + & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ \hline 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{array} \quad \leftarrow \text{carry row}$$

$$= -67$$

$$= -38$$

Discard  
this bit

Thus,  $10111101_2 + 11011010_2 = 10010111_2$ . Now the leading bit of  $10010111_2$  is 1, so it is a negative number. So we find its two's complement (which will represent a positive integer). Now

$$\begin{array}{r} \text{Change 1's to 0's} \\ \text{and 0's to 1's} \\ 10010111 \xrightarrow{\quad\quad\quad} 01101000 \xrightarrow{\text{Add } 00000001} 01101001. \end{array}$$

Since  $01101001_2 = 105$ , the binary number  $10010111$  represents  $-105$ . Also, notice that  $-67 - 38 = -105$ .

**REMARK 2.2.37** ► In the preceding examples, we used 8-bit representations of integers in the range  $-128$  to  $127$  and illustrated addition and subtraction using binary representations. However, we only used those numbers whose addition and/or subtraction is also in the range  $-128$  to  $127$ . Suppose that  $x = 75_{10}$  and  $y = 62_{10}$ . Now  $75_{10} = 01001011_2$  and  $62_{10} = 00111110_2$ . Therefore,

$$\begin{array}{r} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & \leftarrow \text{carry row} \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & = 75_{10} \\ + & 0 & 0 & 1 & 1 & 1 & 1 & 0 & = 62_{10} \\ \hline 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{array}$$

so  $75_{10} + 62_{10} = 01001011_2 + 00111110_2 = 10001001_2$ . However, because the leading bit of  $10001001_2$  is 1, in 8-bit representation, it represents a negative number. Also,  $75 + 62 = 137$ . Now  $137 > 127$ , so it cannot be correctly represented as an 8-bit string.

We therefore conclude that the way integers are represented in computer memory can affect their addition and subtraction.



## WORKED-OUT EXERCISES

**Exercise 1:** Convert  $(353)_{10}$  from decimal to binary.

**Solution:** We have

$$\begin{aligned} 353 &= 176 \cdot 2 + 1; & q_0 &= 176, & a_0 &= 1 \\ 176 &= 88 \cdot 2 + 0; & q_1 &= 88, & a_1 &= 0 \\ 88 &= 44 \cdot 2 + 0; & q_2 &= 44, & a_2 &= 0 \\ 44 &= 22 \cdot 2 + 0; & q_3 &= 22, & a_3 &= 0 \\ 22 &= 11 \cdot 2 + 0; & q_4 &= 11, & a_4 &= 0 \\ 11 &= 5 \cdot 2 + 1; & q_5 &= 5, & a_5 &= 1 \\ 5 &= 2 \cdot 2 + 1; & q_6 &= 2, & a_6 &= 1 \\ 2 &= 1 \cdot 2 + 0; & q_7 &= 1, & a_7 &= 0 \\ 1 &= 0 \cdot 2 + 1; & q_8 &= 0, & a_8 &= 1 \end{aligned}$$

Thus,

$$\begin{aligned} 353 &= 1 \cdot 2^8 + 0 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 \\ &\quad + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1. \end{aligned}$$

Hence,  $(353)_{10} = (101100001)_2$ .

**Exercise 2:** Convert  $(203)_{10}$  from decimal to binary.

**Solution:** We find the highest power of 2 in 203. This is  $2^7$ . Then  $203 = 2^7 + 75$ . Next, we find the highest power of 2 in 75. This is  $2^6$ . Hence,  $203 = 2^7 + 2^6 + 11 = 2^7 + 2^6 + 2^3 + 3 = 2^7 + 2^6 + 2^3 + 2 + 1 = (11001011)_2$ .

We can also find the binary representation as follows.

	Quotients	Remainders
2	203	$1 = a_0$
2	101	$1 = a_1$
2	50	$0 = a_2$
2	25	$1 = a_3$
2	12	$0 = a_4$
2	6	$0 = a_5$
2	3	$1 = a_6$
2	1	$1 = a_7$
	0	

**Exercise 3:** Convert  $(111011)_2$  to base-3 notation.

**Solution:** We have

$$\begin{aligned}(111011)_2 &= 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \\&= 59 \\&= (59)_{10}.\end{aligned}$$

Now,

$$\begin{aligned}59 &= 19 \cdot 3 + 2 \\19 &= 6 \cdot 3 + 1 \\6 &= 2 \cdot 3 + 0 \\2 &= 0 \cdot 3 + 2.\end{aligned}$$

Hence,  $(59)_{10} = (2012)_3$ .

**Exercise 4:** Convert  $(2FB5)_{16}$  from hexadecimal to binary.

**Solution:** We have

$$\begin{aligned}(2FB5)_{16} &= 2 \cdot 16^3 + F \cdot 16^2 + B \cdot 16^1 + 5 \cdot 16^0 \\&= 2 \cdot 16^3 + 15 \cdot 16^2 + 11 \cdot 16^1 + 5 \cdot 16^0 \\&\quad \text{because in hexadecimal } F = 15 \text{ and } B = 11 \\&= 2 \cdot 16^3 + 15 \cdot 16^2 + 11 \cdot 16^1 + 5 \cdot 1 \\&= 2 \cdot 2^{12} + 15 \cdot 2^8 + 11 \cdot 2^4 + 5 \cdot 2^0 \\&= (2+0) \cdot 2^{12} + (2^3 + 2^2 + 2^1 + 1) \cdot 2^8 \\&\quad + (2^3 + 2^1 + 1) \cdot 2^4 + (2^2 + 1) \cdot 2^0 \\&\quad \text{Express 2, 15, 11, and 5 in binary.} \\&= 2^{13} + 2^{11} + 2^{10} + 2^9 + 2^8 + 2^7 + 2^5 + 2^4 + 2^2 + 1 \\&= 1 \cdot 2^{13} + 0 \cdot 2^{12} + 1 \cdot 2^{11} + 1 \cdot 2^{10} + 1 \cdot 2^9 + 1 \cdot 2^8 \\&\quad + 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 \\&= 0 \cdot 2^1 + 1 \cdot 2^0 \\&= (10111110110101)_2.\end{aligned}$$

**Exercise 5:** Convert  $(10111110110011)_2$  from binary to hexadecimal.

**Solution:** We have

$$\begin{aligned}(10111110110011)_2 &= 1 \cdot 2^{13} + 0 \cdot 2^{12} + 1 \cdot 2^{11} + 1 \cdot 2^{10} + 1 \cdot 2^9 + 1 \cdot 2^8 + 1 \cdot 2^7 \\&\quad + 0 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 \\&= 2^{13} + 2^{11} + 2^{10} + 2^9 + 2^8 + 2^7 + 2^5 + 2^4 + 1 \cdot 2^1 + 1 \cdot 2^0 \\&= (2^4)^3 \cdot 2 + (2^4)^2 \cdot 2^3 + (2^4)^2 \cdot 2^2 + (2^4)^2 \cdot 2 \\&\quad + (2^4)^2 + 2^4 \cdot 2^3 + 2^4 \cdot 2 + 2^4 + 2^1 + 1\end{aligned}$$

$$\begin{aligned}&= 2 \cdot 16^3 + 8 \cdot 16^2 + 4 \cdot 16^2 + 2 \cdot 16^2 \\&\quad + 1 \cdot 16^2 + 8 \cdot 16 + 2 \cdot 16 + 16 + 2 + 1 \\&= 2 \cdot 16^3 + 15 \cdot 16^2 + 11 \cdot 16 + 3 \\&= (2FB3)_{16}.\end{aligned}$$

**Exercise 6:** Add  $x = 1111011_2$  and  $y = 11000_2$ .

**Solution:** We have

$$\begin{array}{r} 1 & 1 & 1 & 0 & 0 & 0 & \leftarrow \text{carry row} \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 = x \\ + & & & & & 1 & 0 \\ \hline 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 = y \end{array}$$

Thus,  $1111011_2 + 11000_2 = 10010011_2$ .

**Exercise 7:** Subtract  $10_2$  from  $100_2$ .

**Solution:** Now

$$\begin{array}{r} 10 & \leftarrow \text{digit(s) to be subtracted from} \\ 1 & \leftarrow \text{borrow row} \\ X & 0 & 0 \\ - & & 1 & 0 \\ \hline & 1 & 0 \end{array}$$

**Exercise 8:** Find the 8-bit binary representation of the two's complement of  $131_{10}$ .

**Solution:** Now  $131_{10} = 2^7 + 2 + 1 = (10000011)_2$ .

1. The 8-bit binary representation of 131 is

$$131_{10} = 10000011_2.$$

2. In this representation, change 0's to 1's and 1's to 0's to get

$$01111100_2.$$

3. Add 00000001 to it:

$$\begin{array}{r} 01111100 \\ + 00000001 \\ \hline 01111101 \end{array}$$

Thus, the 8-bit binary representation of the two's complement of  $131_{10}$  is the binary string  $01111101_2$ .

## SECTION REVIEW

### Key Terms

machine language

bit

base 10

binary digit

decimal system

base 2

base 8	decimal representation	two's complement
octal	binary representation	one's complement
base 16	octal representation	leading bit
hexadecimal	hexadecimal representation	

## Some Key Definitions

1. The language of a computer, called machine language, is a sequence of 0's and 1's.
2. Let  $n$  be a positive integer and  $k > 1$  be an integer.
  - (i) If  $k = 10$ , the base-10 representation of  $n$  is called the decimal representation of  $n$ .
  - (ii) If  $k = 2$ , the base-2 representation of  $n$  is called the binary representation of  $n$ . In this case, we also call  $(a_m a_{m-1} a_{m-2} \cdots a_1 a_0)_2$  a binary number.
  - (iii) If  $k = 8$ , the base-8 representation of  $n$  is called the octal representation of  $n$ .
  - (iv) Let  $n$  be a positive integer and  $k > 1$  be an integer. If  $k = 16$ , the base-16 representation of  $n$  is called the hexadecimal representation of  $n$ .
3. Let  $n$  be a fixed positive integer and let  $x$  be a positive integer that can be represented as an  $n$ -bit binary number. The binary representation of  $2^n - x$  is called the two's complement of  $x$  with respect to  $n$ .
4. Let  $n \geq 1$  be an integer and let  $x$  be a nonnegative integer that can be expressed as an  $n$ -bit binary number. The first digit, i.e., first bit, of the  $n$ -bit binary string representing  $x$  is called the leading bit of  $x$ .

## Some Key Results

1. Let  $k > 1$  be an integer. Then any positive integer  $n$  can be expressed uniquely as

$$n = a_m k^m + a_{m-1} k^{m-1} + a_{m-2} k^{m-2} + \cdots + a_1 k^1 + a_0,$$

where  $m$  is a nonnegative integer  $a_m \neq 0$  and each  $0 \leq a_i \leq k - 1$  for  $i = 0, 1, 2, \dots, m$ .

This unique representation of  $n$  is called the representation of  $n$  in base  $k$  (or base- $k$  representation of  $n$ ), and we write this representation as

$$n = (a_m a_{m-1} a_{m-2} \cdots a_1 a_0)_k.$$

2. Eight bits can be used to represent integers in the range  $-128$  to  $127$ . Of course, negative numbers are represented as the two's complement of their absolute.
3. Let  $n > 1$  be an integer. It can be shown that  $n$  bits are sufficient to represent any integer  $x$  such that  $-2^{n-1} \leq x \leq 2^{n-1} - 1$  is an  $n$ -bit string. Of course, negative numbers are represented as two's complements of their absolute values.

## EXERCISES

1. Convert  $(3199)_{10}$  from decimal to binary representation.
2. Convert  $(5554)_7$  from base-7 to decimal representation.
3. Convert  $(10010001)_2$  from binary to decimal representation.
4. Convert  $(CDEF)_{16}$  from hexadecimal to binary representation.
5. Convert  $(3DF9)_{16}$  from hexadecimal to binary representation.
6. Convert  $(2FB3)_{16}$  from hexadecimal to binary representation.
7. Convert  $(100111101001)_2$  from binary to hexadecimal representation.
8. Convert  $(10111100110011)_2$  from binary to hexadecimal representation.
9. Convert  $(3FB09)_{16}$  from hexadecimal to decimal representation.
10. Add  $1111010_2$  and  $1101001_2$ .
11. Add  $1010_2$  and  $1001001_2$ .
12. Add  $10111_2$  and  $11_2$ .
13. Subtract  $1_2$  from  $1000_2$ .
14. Subtract  $11_2$  from  $1000_2$ .
15. Subtract  $101_2$  from  $1100_2$ .

*In Exercises 16–34, assume that negative integers are represented as the two's complement of their absolute values.*

16. Find the 8-bit binary representation of the two's complement of  $119_{10}$ .
17. Find the 8-bit binary representation of the two's complement of  $100_{10}$ .
18. Find the integer whose 8-bit binary string is  $01100101_2$ . Is this a positive integer?
19. Find the integer whose 8-bit binary string is  $11101101_2$ . Is this a positive integer?
20. Find the 16-bit binary representation of the two's complement of  $7561_{10}$ .
21. Find the integer whose 16-bit binary string is  $0100100110110101_2$ . Is this a positive integer?
22. Find the integer whose 16-bit binary string is  $1110001110110001_2$ . Is this a positive integer?
23. Using 8-bit binary representation, evaluate  $99_{10} + 12_{10}$ .

24. Using 8-bit binary representation, evaluate  $55_{10} + 48_{10}$ .
25. Using 8-bit binary representation, evaluate  $55_{10} - 48_{10}$ .
26. Using 8-bit binary representation, evaluate  $-55_{10} + 78_{10}$ .
27. Using 8-bit binary representation, evaluate  $-38_{10} - 61_{10}$ .
28. Prove that  $736_{10}$  and  $945_{10}$  can be represented as 16-bit binary strings.
29. Using the 16-bit binary representation, evaluate the following.
 

a. $945_{10} + 736_{10}$	b. $945_{10} - 736_{10}$
c. $-945_{10} + 736_{10}$	d. $-945_{10} - 736_{10}$
30. Assume that integers are stored as 8-bit binary strings. Can the following addition be performed correctly? Justify your answer.

$$01111101_2 + 01000000_2$$

31. Assume that integers are stored as 8-bit binary strings. Can the following operation be performed correctly? Justify your answer.

$$11001111_2 + 11011100_2$$

32. Assume that integers are stored as 8-bit binary strings. Can the following operation be performed correctly? Justify your answer.

$$10001110_2 + 10110111_2$$

33. Assume that integers are stored as 16-bit binary strings. Can the following addition be performed correctly? Justify your answer.

$$0100110101010101_2 + 0100000101010000_2$$

34. Assume that integers are stored as 16-bit binary strings. Can the following operation be performed correctly? Justify your answer.

$$100011100010001_2 + 1100000101010111_2$$

35. Prove Theorem 2.2.24.
36. Prove Theorem 2.2.31.

## 2.3 MATHEMATICAL INDUCTION

Suppose we are asked to prove that for all positive integers  $n$ ,

$$1 + 2 + 3 + \cdots + n = \frac{n(n+1)}{2}. \quad (2.12)$$

Let  $P(n)$  denote the sentence

$$1 + 2 + 3 + \cdots + n = \frac{n(n+1)}{2}. \quad (2.13)$$

We are asked to prove that the statement  $\forall n P(n)$  is true in the domain of all positive integers.

If  $n = 1$ , then we find that

$$\frac{1(1+1)}{2} = \frac{2}{2} = 1.$$

Thus,  $P(1)$  is true.

Suppose  $n = 2$ . Then

$$\frac{2(2+1)}{2} = 1 + 2.$$

So we find that the statement is true for  $n = 2$ . Similarly, we can verify that  $P(n)$  is true when  $n = 3$  and 4.

Because there are an infinite number of positive integers, it is not possible to verify that  $\forall n P(n)$  is true by evaluating  $P(n)$  at each positive integer. However, we do know, at least intuitively, that  $P(n)$  is true for all positive integers. So how do we prove this fact?

Let us consider another problem. A local post office temporarily ran out of stamps except for 3- and 5-cent stamps. Everyone wants to mail their letters using stamps. So how can they mail their letters using only these two types of stamps? If someone needs a stamp of 4 cents, then it cannot be done. Similarly, for stamps of 7 cents it also cannot be done. However, for 8-cent stamps, one can use a 3-cent stamp and a 5-cent stamp. For  $n$ -cent stamps, where  $n > 8$ , one can use stamps as follows.

$$\begin{aligned} n = 9, \quad & 3 + 3 + 3 = 9 \\ n = 10, \quad & 5 + 5 = 10 \\ n = 11, \quad & 3 + 3 + 5 = 11 \\ n = 12, \quad & 3 + 3 + 3 + 3 = 12 \\ n = 13, \quad & 3 + 5 + 5 = 13 \\ n = 14, \quad & 3 + 3 + 3 + 5 = 14 \\ n = 15, \quad & 5 + 5 + 5 = 15 \text{ or } 3 + 3 + 3 + 3 + 3 = 15 \\ n = 16, \quad & 5 + 5 + 3 + 3 = 16 \end{aligned}$$

The natural question is how to justify that for the postage charges of  $n$  cents,  $n \geq 8$ , we can use a combination of 3- and 5-cent stamps.

Let  $P(n)$  denote the sentence: For any postage charges of  $n$  cents,  $n \geq 8$ , a combination of 3- and 5-cent stamps is sufficient. We prove in Example 2.3.3 that the statement  $\forall n P(n)$  is true in the domain of all positive integers  $n \geq 8$ .

There are many other such problems in which we are asked to prove that a statement is true for a set of values. In such cases, we employ a well-known technique based on the *principle of mathematical induction* or, simply, *induction*.

Before we formally define mathematical induction, let us illustrate how it works using some visual diagrams. Consider the diagrams of Figure 2.1.

Figure 2.1(a) consists of a row of *standing* dominos labeled  $1, 2, \dots, n, \dots$ . We assume that when a domino is knocked over, it also knocks the next domino. Suppose we want to show that if the first domino is knocked over, then all dominos are knocked over. Let  $P(n)$  denote the statement that the  $n$ th domino is knocked over. To apply the principle of mathematical induction, we do the following: Show that  $P(1)$  is true. Next assume that  $P(k)$  is true, i.e., the  $k$ th domino is knocked over, for some integer  $k \geq 1$ . Then prove that  $P(k + 1)$  is true. In other words, show that  $P(k) \rightarrow P(k + 1)$ , where  $k$  is an integer  $\geq 1$ . Then we can claim that all dominos are knocked over. In this case, suppose that the first domino is knocked over. Because it is given that the first domino is knocked over,  $P(1)$  is true. Now suppose that  $P(k)$  is true for some  $k \geq 1$ . By the assumption, because the  $k$ th

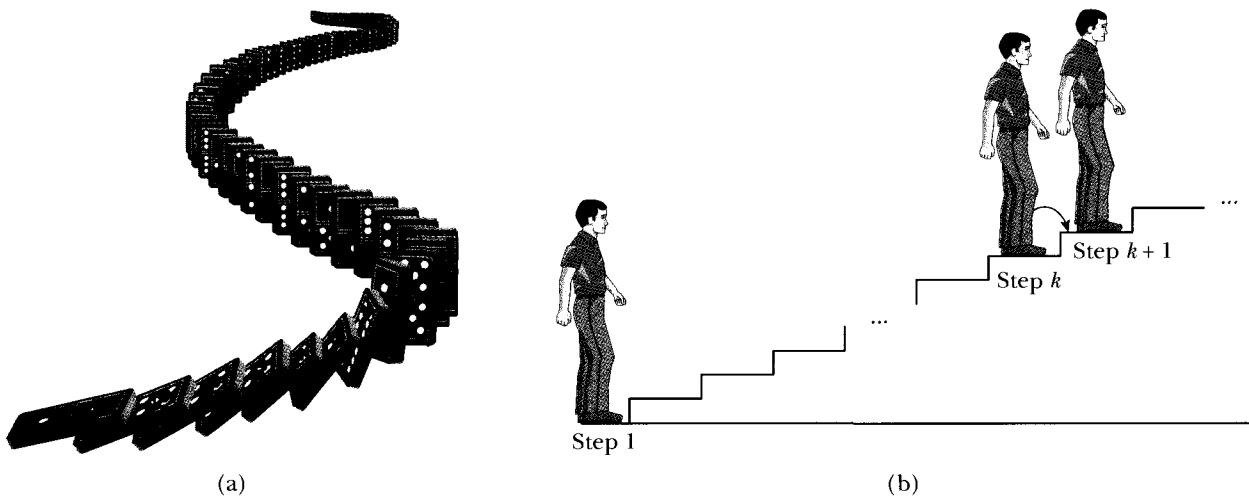


FIGURE 2.1

domino is knocked over, it also knocks the next domino, which is the  $(k + 1)$ th domino. Therefore,  $P(k + 1)$  is true, so  $P(k) \rightarrow P(k + 1)$ . Hence, we can claim that if the first domino is knocked over, then all dominos are knocked over.

Now consider the diagram of Figure 2.1(b), which consists of staircase steps labeled  $1, 2, \dots, n, \dots$ . We assume that when a staircase is climbed, the next staircase is also climbed. Suppose that we want to show that if the first staircase is climbed, then all staircases can be climbed. Let  $P(n)$  denote the statement that the  $n$ th staircase is climbed. Now it is given the first staircase is climbed, so  $P(1)$  is true. Now suppose that  $P(k)$  is true for some  $k \geq 1$ . By the assumption, because the  $k$ th staircase is climbed, the next staircase is also climbed, which is the  $(k + 1)$ th staircase. Therefore,  $P(k + 1)$  is true, so  $P(k) \rightarrow P(k + 1)$ . Hence, we can claim that if the first staircase is climbed, then all staircases are climbed.

There are two forms of the principle of mathematical induction. The first form is as follows.

**First Principle of Mathematical Induction:** Let  $P(n)$  be a sentence containing nonnegative integer  $n$ , and let  $n_0$  be a fixed nonnegative integer.

1. Suppose  $P(n_0)$  is true (i.e.,  $P(n)$  is true for  $n = n_0$ ).
2. Whenever  $k$  is an integer such that  $k \geq n_0$  and  $P(k)$  is true, then  $P(k + 1)$  is true.

Then  $P(n)$  is true for all integers  $n \geq n_0$ .

Typically, this form of the principle of mathematical induction is called the **first principle of mathematical induction**.

In the following example, we show how to use mathematical induction to prove (2.12), i.e.,

$$1 + 2 + 3 + \cdots + n = \frac{n(n + 1)}{2}$$

for all integers  $n \geq 1$ .

Let  $P(n)$  denote

$$P(n) : 1 + 2 + 3 + \cdots + n = \frac{n(n + 1)}{2}.$$

Because we want to prove that  $P(n)$  is true for all integers  $n$  such that  $n \geq 1$ , we start by verifying that  $P(n)$  is true when  $n = 1$ .

Let  $n = 1$ . Then

$$1 = \frac{1(1+1)}{2}.$$

Hence,  $P(1)$  is true.

Let  $k$  be an integer such that  $k \geq 1$ . Assume that  $P(k)$  is true, i.e.,

$$1 + 2 + 3 + \cdots + k = \frac{k(k+1)}{2}.$$

Next, we verify that  $P(k+1)$  is true, i.e.,

$$1 + 2 + 3 + \cdots + k + (k+1) = \frac{(k+1)(k+2)}{2}.$$

Let us evaluate the left side of this equation. We have

$$\begin{aligned} & 1 + 2 + 3 + \cdots + k + (k+1) \\ &= 1 + 2 + 3 + \cdots + k + (k+1) \\ &= \frac{k(k+1)}{2} + (k+1) && \text{because } P(k) \text{ is true} \\ &= (k+1)\left(\frac{k}{2} + 1\right) \\ &= \frac{(k+1)(k+2)}{2} \end{aligned}$$

This implies that  $P(k+1)$  is true. Hence, by induction it follows that  $P(n)$  is true for all integers  $n \geq 1$ .

Let us note the following: A proof of a mathematical statement by the principle of mathematical induction consists of three steps.

1. **Basis step:** To show that  $P(n_0)$  is true for a particular nonnegative integer.
2. **Inductive hypothesis:** To write the inductive hypothesis: Let  $k$  be an integer such that  $k \geq n_0$  and  $P(k)$  is true.
3. **Inductive step:** To show that  $P(k+1)$  is true.

---

**REMARK 2.3.1** ► The principle of mathematical induction is a very useful tool for proving various results. However, one should be very careful while using this principle. For example, consider the following statement:

$$5 \text{ divides } 5n + 3 \text{ for all positive integers } n.$$

Let us find the flaws in the following proof of this statement by induction.

Let  $P(n)$  denote the sentence:  $5n + 3$  is divisible by 5. We prove that, for all integers  $n \geq 1$ ,  $P(n)$  is true.

*Inductive hypothesis:* Let  $k$  be an integer such that  $k \geq 1$  and  $P(k)$  is true, i.e.,

$$5k + 3 \text{ is divisible by 5.}$$

*Inductive step:* We show that  $P(k+1)$  is true; i.e., we prove that  $5(k+1) + 3$  is divisible by 5.

Now,

$$5(k+1) + 3 = (5k+3) + 5.$$

Because 5 divides 5 and by the inductive hypothesis 5 divides  $(5k+3)$ , it follows that 5 divides  $5(k+1) + 3$ . Therefore,  $P(k+1)$  is true. Hence, by induction, 5 divides  $5n+3$  for all positive integers  $n$ .

This conclusion cannot be true. For example, 5 does not divide  $5 \cdot 1 + 3$ .

In the proof given in this remark, we did not check to see that the basis step, i.e.,  $P(1)$ , is true.

### EXAMPLE 2.3.2

In this example, we prove by mathematical induction that  $7^n + 5$  is divisible by 3 for all integers  $n \geq 0$ .

Let  $P(n)$  denote the sentence:  $7^n + 5$  is divisible by 3.

We prove that, for all integers  $n \geq 0$ ,  $P(n)$  is true.

*Basis step:* Let  $n = 0$ . Then  $7^n + 5 = 7^0 + 5 = 6$ , which is divisible by 3. Hence,  $P(0)$  is true.

*Inductive hypothesis:* Let  $k$  be an integer such that  $k \geq 0$  and  $P(k)$  is true, i.e.,

$$7^k + 5 \text{ is divisible by 3.}$$

*Inductive step:* We show that  $P(k+1)$  is true; i.e., we prove that  $7^{k+1} + 5$  is divisible by 3.

Now

$$7^{k+1} + 5 = 7 \cdot 7^k + 5 = 7(7^k + 5) - 30.$$

By the inductive hypothesis, 3 divides  $(7^k + 5)$  and we know that 3 divides 30. Therefore, it follows that 3 divides

$$7(7^k + 5) - 30 = 7^{k+1} + 5.$$

Thus, if  $k$  is any integer such that  $k \geq 0$  and  $P(k)$  is true, then  $P(k+1)$  is also true. Hence, by induction it follows that  $7^n + 5$  is divisible by 3 for all integers  $n \geq 0$ .

The preceding examples show how to use the first principle of induction to prove that certain statements are true. In those examples, in the inductive step we assume that the statement  $P(k)$  is true and use it to prove the inductive step that  $P(k+1)$  is true. Now if we assume that  $P(k)$  is true, then all the statements in between the basis step and the inductive step, that is,  $P(n_0+1), \dots, P(k-2)$ ,  $P(k-1)$ , must also be true, even though those statements were not needed to prove that  $P(k+1)$  is true. However, there are situations when these in-between statements are also needed to prove that  $P(k+1)$  is true. In such cases, the principle of induction, called the **second principle of mathematical induction**, or **strong principle of induction**, takes the following form.

**Second Principle of Mathematical Induction:** Let  $P(n)$  be a mathematical sentence about nonnegative integers  $n$  and let  $n_0$  be a fixed nonnegative integer.

1. Suppose  $P(n_0)$  is true.
2. If for any integer  $k \geq n_0$ ,  $P(n_0), P(n_0+1), P(n_0+2), \dots, P(k)$  are true imply that  $P(k+1)$  is true, then  $P(n)$  is true for all  $n \geq n_0$ .

In fact, we can prove that the first principle of mathematical induction holds if and only if the second principle of mathematical induction holds.

Let us now show some applications of this principle.

### EXAMPLE 2.3.3

In this example, we answer the second problem stated in the beginning of this section. That is, a local post office temporarily ran out of stamps except for 3- and 5-cent stamps. Using the second principle of mathematical induction, we show that to mail letters, any postage charges greater than or equal to 8 cents can be made by using 3- and 5-cent stamps.

Let  $P(n)$  denote the sentence: Any postage charges of  $n$  cents can be made by using 3- and 5-cent stamps. We prove that, for all integers  $n \geq 8$ ,  $P(n)$  is true.

*Basis step:* Let  $n = 8$ . Because  $3 + 5 = 8$ ,  $P(8)$  is true.

Now for  $n = 9$ ,  $3 + 3 + 3 = 9$  and so  $P(9)$  is true. For  $n = 10$ ,  $5 + 5 = 10$  and so  $P(10)$  is true. For  $n = 11$ ,  $3 + 3 + 5 = 11$  and so  $P(11)$  is true. For  $n = 12$ ,  $3 + 3 + 3 + 3 = 12$  and so  $P(12)$  is true.

*Inductive hypothesis:* Suppose  $P(n)$  is true for all  $n$  such that  $12 \leq n < k$ .

*Inductive step:* We show that  $P(k)$  is true; i.e., any postage charges of  $k$  cents can be made by using 3- and 5-cent stamps.

We have already verified that  $P(k)$  if  $k = 8, 9, 10, 11, 12$ . Therefore, we can consider  $k > 12$ .

From the induction hypothesis we find  $P(n)$  is true for  $n = k - 3$ . Therefore, we can make postage charges of  $k - 3$  cents by using 3- and 5-cent stamps. Now  $k = (k - 3) + 3$ , so a postage charge of  $k$  cents can be made by using 3- and 5-cent stamps.

Hence, by the second principle of mathematical induction, it follows that  $P(n)$  is true for all integers  $n$  such that  $10 \leq n$ .

Because the first principle of mathematical induction and second principle of mathematical induction are equivalent, we shall henceforth talk about induction only.

---

**REMARK 2.3.4** ▶ We should point out that either the principle of mathematical induction or the well-ordering principle can be proved as a theorem, given the other principle and properties of integers. In other words, we can say that these two principles are equivalent. (For example, in Worked-Out Exercise 10 at the end of this section, we show that the well-ordering principle implies the principle of mathematical induction.)

## Application: Loop Invariant (Program Correctness)

When a program is compiled—even if it does not produce any syntax errors—there is no guarantee that it will produce the correct results. This is especially true of large and complex programs. So how does one assure the reliability of the program? Proving that a program is correct is a major issue in the software industry. One way of verifying the correctness of a program is to run the program through a series of test cases. This technique may work fine as long as the number of test cases is small. However, even if the number of test cases is small, erroneous data can cause problems. Other techniques, such as mathematical tools, are available to prove the correctness of a program. In other words, programmers can write a proof that shows that the program is correct.

Throughout this section, we use the more general term *algorithm* instead of program.

Usually, there is more than one way to accomplish a task. For example, there are various ways to sort a list. Also, different programmers can implement the same

algorithm differently. For example, to implement a loop, one programmer might use a `for` loop while another programmer might use a `while` loop. Therefore, the user of an algorithm need not be concerned with how the algorithm is implemented, but he or she must know how to use the algorithm and what the algorithm does. These requirements are stated in the form of preconditions and postconditions. A **precondition** is an assertion (a set of statements) that remains true before the algorithm executes. A **postcondition** is an assertion that is true after the algorithm executes. In the algorithms presented until now, we have been specifying preconditions in the form of input and postconditions in the form of output. For example, for the algorithm that determines the largest of three numbers, the precondition is three numbers and the postcondition is the largest of three numbers.

As remarked in Chapter 1, in computer science, other than just proving theorems, mathematical logic is used to prove program correctness. As defined earlier, a program is a sequence of instructions whose objective is to accomplish something. The instruction can be a statement that is executed only once, such as an assignment statement; or a selection statement to make a decision; or a loop to execute certain statements over and over until certain conditions are met. Verifying the correctness of a statement that executes only once or the correctness of a statement that makes a decision is relatively easier than verifying the correctness of a loop. In this section, we illustrate how mathematical induction is used to prove the correctness of loops.

The syntax of the `while` loop is:

```
while booleanExpression do
    loopBody
```

The `booleanExpression` is evaluated. If the `booleanExpression` evaluates to `true`, the `loopBody` executes. After executing the `loopBody`, the `booleanExpression` is evaluated again. Then the `loopBody` continues to execute as long as the `booleanExpression` evaluates to `true`.

Let us take a look at the `booleanExpression` in the `while` statement. Because the `booleanExpression` is either `true` or `false`, it is a statement. Let  $q$  denote the `booleanExpression`.

It turns out that with a loop we can associate a predicate,  $P(n)$ . The predicate  $P(n)$  is such that

1.  $P(0)$  is true before the loop executes.
2. Now when the loop terminates,  $q$  is false and so  $\sim q$  is true. Therefore, if the loop executes, say  $N$  times, then  $P(N) \wedge \sim q$  is true after the loop terminates.
3. If  $P(k) \wedge q$  is true before the  $(k + 1)$ st iteration of the loop and  $k < n$ , then  $P(k + 1) \wedge q$  is true after the  $(k + 1)$ st iterations and  $k + 1 \leq n$ .

The predicate  $P(n)$  is called the **loop invariant** for the loop. In other words, a loop invariant is a set of statements that remains true each time the loop body is executed.

For example, in Chapter 1, the algorithm to determine the smallest element in a list  $L[1 \dots n]$  is given. The algorithm contains a `while` loop. The loop invariant for the `while` loop is:

$P(k) : x$  is smallest in  $L[1 \dots k]$  and  $k \leq n$ .

$q : k \leq n$ .

1. Before the loop executes,  $i = 1 \leq n$  and  $x = L[1]$ , which is the smallest in  $L[1 \dots 1]$ .

2. After the loop terminates,  $i \leq n$  is false and so  $\sim(i \leq n)$  is true; i.e.,  $i > n$  is true. Also,  $x$  is the smallest element in  $L[1 \dots n]$ .
3. Suppose that  $P(k) \wedge q$  is true. That is,  $x$  is smallest in  $L[1 \dots k]$  and  $i = k < n$ . Note that  $k < n$  so that the next iteration can take place. In the  $(k+1)$ st iteration,  $x$ , the smallest in  $L[1 \dots k]$ , is compared with  $L[k+1]$ . If  $x > L[k+1]$ , then  $x$  is assigned the value of  $L[k+1]$ . So  $P(k+1)$  is true. Moreover,  $i = k + 1 \leq n$  and so  $q$  is true. Thus,  $P(k+1) \wedge q$  is true.

Therefore, it follows that  $P(k)$  is a loop invariant.

**Notation 2.3.5:** Consider the following loop.

```
i := 0
x := 1;
while i < 5 do
begin
    i = i + 1;
    x := x + 2 * i;
end
```

Before the while loop,  $i$  is initialized to 0 and  $x$  is initialized to 1. The while loop executes 5 times. After each iteration of the loop,  $i$  is incremented by 1 and the value of  $x$  is updated by taking its previous value and adding twice the value of  $i$ . We denote by  $x_t$  and  $i_t$ , the values of  $x$  and  $i$ , respectively, after the  $t$ th iteration. For the preceding while loop, it follows that

$$\begin{array}{ll} i_0 = 0, & x_0 = 1 \\ i_1 = i_0 + 1 = 1, & x_1 = x_0 + 2i_1 = 1 + 2 = 3 \\ i_2 = i_1 + 1 = 2, & x_2 = x_1 + 2i_2 = 3 + 4 = 7 \\ i_3 = i_2 + 1 = 3, & x_3 = x_2 + 2i_3 = 7 + 6 = 13 \\ i_4 = i_3 + 1 = 4, & x_4 = x_3 + 2i_4 = 13 + 8 = 21 \\ i_5 = i_4 + 1 = 5, & x_5 = x_4 + 2i_5 = 21 + 10 = 31. \end{array}$$

For this table, it follows that after the loop terminates,  $i = 5$  and  $x = 31$ .

### EXAMPLE 2.3.6

Let  $n$  be a nonnegative integer. The *factorial* of  $n$ , written  $n!$ , is defined by

$$n! = \begin{cases} 1 & \text{if } n = 0, \\ n \cdot (n-1)! & \text{if } n > 0. \end{cases}$$

In other words, if  $n = 0$ ,  $n! = 1$ ; if  $n > 0$ , then  $n! = 1 \cdot 2 \cdots (n-1) \cdot n$ .

Consider the following algorithm that determines the factorial of a nonnegative integer.

### ALGORITHM 2.5: Compute $n!$ .

*Input:*  $n$ —nonnegative integer

*Output:*  $n!$ —the factorial of  $n$

1. **factorial**( $n$ )
2. **begin**
3.   *i* := 0;

```

4. fact := 1;
5. while (i < n) do
6. begin
7.   i := i + 1;
8.   fact := fact * i;
9. end
10. return fact;
11. end

```

We show that the loop invariant for this loop is

$$P(k) : \text{fact} = k!, \quad i = k, \quad \text{and } i \leq n.$$

*Basis step:* Before the loop,  $i_0 = 0$  and  $\text{fact}_0 = 1$ . Also  $n \geq 0$ , so  $i \leq n$ . Thus,  $P(0)$  is true.

*Inductive hypothesis:* Suppose  $P(k)$  is true. Thus,  $\text{fact}_k = k!$ ,  $i_k = k$ , and  $k < n$ . (We assume that  $k < n$  so that the next iteration occurs.)

*Inductive step:* In the  $(k+1)$ st iteration,

$$\begin{aligned} i_{k+1} &:= i_k + 1 = k + 1, \\ \text{fact}_{k+1} &:= \text{fact}_k * i_{k+1} = k! \cdot (k + 1) = (k + 1)! \end{aligned}$$

Also  $i = k < n$ , so  $i_{k+1} = k + 1 \leq n$ . Thus,  $P(k+1)$  is true.

When the loop terminates true,  $P(k)$  is true and  $i < n$  is false. Because  $i < n$  is false,  $i = n$  is true. Therefore, when the loop terminates,  $P(n) = P(i) : \text{fact} = i! = n!$ .

Now before the loop  $i$  is 0. Each time through the loop  $i$  is incremented by 1 and so after  $n - 1$  iterations,  $i = n$ ; i.e.,  $i < n$  is false and the loop terminates.

### EXAMPLE 2.3.7

Consider the following algorithm, which determines the largest element in a list of  $n$  elements.

**ALGORITHM 2.6:** Determine the largest element in a list of  $n$  elements.

*Input:*  $L[1 \dots n]$ —list of  $n$  elements  
 $n$ —number of elements in the list

*Output:*  $x$ —the largest element in  $L[1 \dots n]$

```

1. i := 1;
2. x := L[1];
3. while i < n do
4. begin
5.   i := i + 1;
6.   if x < L[i] then
7.     x := L[i];
8. end

```

At Line 2, we assume that the first element of the list is the largest element and assign its value to  $x$ . Thereafter,  $x$  is compared with each element in the list. Any time we find an element in the list larger than  $x$ , we copy the value of that element into  $x$ . The statement in Line 5 increments  $i$  and the statement in Line 6 compares if  $x$  is less than  $L[i]$ . It follows that after the first iteration of the loop,  $x$  is the largest of the first two elements. In fact, it follows that after  $k - 1$  iterations of the loop  $x$  is the largest of the first  $k$  elements of the loop (of course, we will prove it). After  $n - 1$  iterations of the loop,  $i = n$  and  $x$  is compared with all the elements of the list. This suggests that the loop invariant is:

$$P(k) : x \text{ is largest in } L[1 \dots k], \quad i = k + 1, \quad \text{and } i \leq n.$$

Let us prove this assertion.

*Basis step:* Before the loop executes,  $i = 1$  and  $x = L[1]$ . Also  $n$  is positive, so  $n \geq 1$ . Thus,  $i \leq n$ . Therefore,  $P(0) : x$  is largest in  $L[1 \dots 1]$ ,  $i = 1$ , and  $i < n$  is true.

*Inductive hypothesis:* Suppose that  $P(k)$  is true; i.e.,  $x$  = largest in  $L[1 \dots k]$ ,  $i = k + 1$ , and  $i < n - 1$ . (We assume that  $i < n - 1$ , so that the next iteration can take place.)

*Inductive step:*  $P(k + 1) : x$  is largest in  $L[1 \dots k + 1]$ ,  $i = k + 2$ , and  $i < n$ .

Because  $P(k)$  is true,  $x$  is largest in  $L[1 \dots k]$ ,  $i = k + 1$ , and  $i < n - 1$ . Consider the  $k$ th iteration of the loop.

$$i_{k+1} := i_k + 1 = k + 2$$

Evaluate  $x < L[i_{k+1}]$ . i.e., evaluate  $x < L[k + 1]$ . If this is true, then  $x_{k+1} := L[k + 1]$ ; i.e.,  $L[k + 1]$  is the new largest element in  $L[1 \dots k + 1]$  and its value is copied into  $x$ ; otherwise, the old value of  $x$ ,  $x_k$ , is the largest in  $L[1 \dots k + 1]$ . Also  $i_k = k + 1 < n - 1$ , so  $i_{k+1} = k + 2 < n$ . Hence,  $P(k + 1)$  is true.

Consequently,  $P(k)$  is a loop invariant.

Before the loop executes,  $i$  is 1. Each time through the loop  $i$  is incremented by 1. Therefore, after  $n - 1$  iterations,  $i = n$ , so  $i < n$  is false and the loop terminates.

When  $i = n$ ,  $P(n)$  is true, so  $x$  is largest in  $L[1 \dots n]$ .

## WORKED-OUT EXERCISES

**Exercise 1:** Show that

$$1^2 + 2^2 + 3^2 + \dots + n^2 = \frac{n(n + 1)(2n + 1)}{6},$$

for all integers  $n \geq 1$  by the principle of mathematical induction.

**Solution:** Let

$$P(n) : 1^2 + 2^2 + 3^2 + \dots + n^2 = \frac{n(n + 1)(2n + 1)}{6}.$$

Because we want to prove that  $P(n)$  is true for all integers  $n$  such that  $n \geq 1$ , we start by verifying that  $P(n)$  is true when  $n = 1$ .

*Basis step:* Let  $n = 1$ . Then

$$1 = \frac{1(1 + 1)(2 \cdot 1 + 1)}{6}.$$

Hence,  $P(1)$  is true.

*Inductive hypothesis:* Let  $k$  be an integer such that  $k \geq 1$ . Assume that  $P(k)$  is true; i.e.,

$$1^2 + 2^2 + 3^2 + \dots + k^2 = \frac{k(k + 1)(2k + 1)}{6}.$$

*Inductive step:* We verify that  $P(k + 1)$  is true; i.e.,

$$1^2 + 2^2 + 3^2 + \dots + k^2 + (k + 1)^2 = \frac{(k + 1)(k + 2)(2k + 3)}{6}.$$

Let us evaluate the left side of this equation. We have

$$\begin{aligned}
 & 1^2 + 2^2 + 3^2 + \cdots + k^2 + (k+1)^2 \\
 &= (1^2 + 2^2 + 3^2 + \cdots + k^2) + (k+1)^2 \\
 &= \frac{k(k+1)(2k+1)}{6} + (k+1)^2 \quad \text{because } P(k) \text{ is true} \\
 &= (k+1) \left( \frac{k(2k+1)}{6} + (k+1) \right) \\
 &= (k+1) \frac{k(2k+1) + 6(k+1)}{6} \\
 &= (k+1) \left( \frac{2k^2 + 7k + 6}{6} \right) \\
 &= \frac{(k+1)(k+2)(2k+3)}{6}.
 \end{aligned}$$

This implies that  $P(k+1)$  is true. Hence, by induction it follows that  $P(n)$  is true for all integers  $n \geq 1$ .

**Exercise 2:** Show that

$$1 + 2 + 2^2 + \cdots + 2^n = 2^{n+1} - 1 \quad \text{for all } n \geq 0.$$

**Solution:** Let

$$P(n) : 1 + 2 + 2^2 + \cdots + 2^n = 2^{n+1} - 1.$$

Because we want to prove that the statement  $\forall n P(n)$  is true in the domain of all nonnegative integers, as a basis step we verify that  $P(n)$  is true when  $n = 0$ .

*Basis step:* Let  $n = 0$ . Then

$$1 = 2^0 = 2 - 1 = 2^{0+1} - 1.$$

This shows that  $P(0)$  is true.

*Inductive hypothesis:* Let  $k$  be an integer such that  $k \geq 0$  and  $P(k)$  is true, i.e.,

$$1 + 2 + 2^2 + \cdots + 2^k = 2^{k+1} - 1.$$

*Inductive step:* Consider  $P(k+1)$ , i.e.,

$$1 + 2 + 2^2 + \cdots + 2^k + 2^{k+1} = 2^{k+2} - 1.$$

Let us evaluate the left side of this equation. We have

$$\begin{aligned}
 & 1 + 2 + 2^2 + \cdots + 2^{k+1} \\
 &= 1 + 2 + 2^2 + \cdots + 2^k + 2^{k+1} \\
 &= (1 + 2 + 2^2 + \cdots + 2^k) + 2^{k+1} \\
 &= (2^{k+1} - 1) + 2^{k+1} \quad \text{because } P(k) \text{ is true} \\
 &= 2 \cdot 2^{k+1} - 1 \\
 &= 2^{k+2} - 1.
 \end{aligned}$$

This implies that  $P(k+1)$  is true. Consequently, by induction

$$1 + 2 + 2^2 + \cdots + 2^n = 2^{n+1} - 1 \quad \text{for all } n \geq 0.$$

**Exercise 3:** Prove that  $5^n + 3$  is divisible by 4 for all integers  $n \geq 0$ .

**Solution:** Let  $P(n) : 5^n + 3$  is divisible by 4. We prove that the statement  $\forall n P(n)$  is true in the domain of all positive integers.

*Basis step:* Let  $n = 0$ . Then  $5^n + 3 = 5^0 + 3 = 4$ , which is divisible by 4. Thus  $P(4)$  is true.

*Inductive hypothesis:* Let  $k$  be an integer such that  $k \geq 0$  and  $5^k + 3$  is divisible by 4.

*Inductive step:* We show that  $5^{k+1} + 3$  is divisible by 4. Now

$$5^{k+1} + 3 = 5 \cdot 5^k + 3 = 5 \cdot 5^k + 15 - 12 = 5(5^k + 3) - 12.$$

By the inductive hypothesis, 4 divides  $(5^k + 3)$ . Also 4 divides 12. Therefore, by Theorem 2.1.14(iii), 4 divides

$$5(5^k + 3) - 12 = 5^{k+1} + 3.$$

Thus, if  $k$  is any integer such that  $k \geq 0$  and  $P(k)$  is true, then  $P(k+1)$  is also true. Hence, by induction it follows that  $5^n + 3$  is divisible by 4 for all integers  $n \geq 0$ .

**Exercise 4:** Prove that for any positive integer  $n$ , 7 divides  $3^{2n+1} + 2^{n+2}$ .

**Solution:** Let  $P(n) : 7$  divides  $3^{2n+1} + 2^{n+2}$ . We prove that the statement  $\forall n P(n)$  is true in the domain of all positive integers.

*Basis step:* Let  $n = 1$ . Then

$$3^{2 \cdot 1 + 1} + 2^{1+2} = 35,$$

which is divisible by 7. Hence,  $P(n)$  holds for  $n = 1$ .

*Inductive hypothesis:* Let  $k$  be a positive integer such that  $P(k)$  is true.

*Inductive step:* Let  $t = 3^{2(k+1)+1} + 2^{(k+1)+2}$ . Then

$$\begin{aligned}
 t &= 3^{2k+1+2} + 2^{k+1+2} \\
 &= 3^{2k+1} \cdot 3^2 + 2^{k+2} \cdot 2 \\
 &= 3^{2k+1} \cdot 9 + 2^{k+2} \cdot 2 \\
 &= 3^{2k+1} \cdot (7 + 2) + 2^{k+2} \cdot 2 \\
 &= 3^{2k+1} \cdot 7 + 3^{2k+1} \cdot 2 + 2^{k+2} \cdot 2 \\
 &= 3^{2k+1} \cdot 7 + 2(3^{2k+1} + 2^{k+2}).
 \end{aligned}$$

Now  $7 \mid (3^{2k+1} \cdot 7)$  and by the inductive hypothesis,  $7 \mid (3^{2k+1} + 2^{k+2})$ . Therefore, by Theorem 2.1.14(iii), we find that  $7 \mid t$ . This implies that  $P(k+1)$  is true. Hence, by induction,  $7 \mid (3^{2n+1} + 2^{n+2})$  for all positive integers  $n$ .

**Exercise 5:** Show that  $n^3 - 7n + 3$  is divisible by 3 for all positive integers  $n$ .

**Solution:** Let

$$P(n) : n^3 - 7n + 3 \text{ is divisible by 3}$$

We prove that the statement  $\forall n P(n)$  is true in the domain of all positive integers.

*Basis step:* For  $n = 1$ ,

$$1^3 - 7 \cdot 1 + 3 = -3,$$

shows that  $n^3 - 7n + 3$  is divisible by 3.

Hence,  $P(n)$  holds for  $n = 1$ .

*Inductive hypothesis:* Let  $k$  be a positive integer such that  $P(k)$  holds; i.e.,  $k^3 - 7k + 3$  is divisible by 3.

*Inductive step:* We show that  $P(k+1)$  holds; i.e.,  $(k+1)^3 - 7(k+1) + 3$  is divisible by 3. Now

$$\begin{aligned} & (k+1)^3 - 7(k+1) + 3 \\ &= k^3 + 3k^2 + 3k + 1 - 7k - 7 + 3 \\ &= (k^3 - 7k + 3) + 3(k^2 + k - 2). \end{aligned}$$

By the inductive hypothesis, 3 divides  $k^3 - 7k + 3$ . Also 3 divides  $3(k^2 + k - 2)$ . Therefore, by Theorem 2.1.14(iii), we find that 3 divides  $(k^3 - 7k + 3) + 3(k^2 + k - 2) = (k+1)^3 - 7(k+1) + 3$ . This implies that  $P(k+1)$  is true. Hence, by induction,  $n^3 - 7n + 3$  is divisible by 3 for all positive integers  $n$ .

**Exercise 6:** Prove that, for all integers  $n \geq 1$ ,

$$1 \cdot 2 + 2 \cdot 3 + 3 \cdot 4 + \cdots + n(n+1) = \frac{n(n+1)(n+3)}{3}.$$

**Solution:** Let  $P(n)$  denote:

$$P(n) : 1 \cdot 2 + 2 \cdot 3 + 3 \cdot 4 + \cdots + n(n+1)$$

$$= \frac{n(n+1)(n+2)}{3}.$$

We prove that the statement  $\forall n P(n)$  is true in the domain of all positive integers.

*Basis step:* For  $n = 1$ ,

$$1 \cdot 2 = \frac{1(1+1)(1+2)}{3}.$$

Thus,  $P(1)$  is true.

*Inductive hypothesis:* Let  $k$  be a positive integer such that  $P(k)$  holds.

*Inductive step:* We show that  $P(k+1)$  holds. Now

$$\begin{aligned} & 1 \cdot 2 + 2 \cdot 3 + 3 \cdot 4 + \cdots + k(k+1) + (k+1)(k+2) \\ &= \frac{k(k+1)(k+2)}{3} + (k+1)(k+2) \quad \text{because } P(k) \text{ holds} \\ &= (k+1)(k+2) \left( \frac{k}{3} + 1 \right) \\ &= \frac{(k+1)(k+2)(k+3)}{3}. \end{aligned}$$

Therefore,  $P(k+1)$  holds. Hence, by the induction  $P(n)$  holds for all positive integers  $n$ .

**Exercise 7:** Show that  $2n+1 < n^3$  for all integers  $n \geq 2$ .

**Solution:** Let  $P(n)$  be the sentence:  $2n+1 < n^3$ . We prove that the statement  $\forall n P(n)$  is true in the domain of all positive integers  $\geq 2$ .

*Basis step:* For  $n = 2$ ,  $2 \cdot 2 + 1 = 5 < 8 = 2^3$ . Hence,  $P(2)$  is true.

*Inductive hypothesis:* Suppose  $P(k)$  is true for some positive integer  $k \geq 2$ , i.e.,  $2k+1 < k^3$ .

*Inductive step:* We show  $P(k+1)$  is true; i.e., we show that  $2(k+1)+1 < (k+1)^3$ , or, equivalently,

$$2k+3 < (k+1)^3.$$

We have

$$\begin{aligned} 2k+3 &= 2k+1+2 \\ &< k^3+2 \quad \text{because } P(k) \text{ is true} \\ &< k^3+3k^2+3k+1 \quad \text{because } k \geq 2 \text{ and so} \\ &\quad 3k^2+3k+1 > 2 \\ &= (k+1)^3. \end{aligned}$$

Thus,  $P(k+1)$  holds. Hence, by induction,  $P(n)$  holds for all  $n \geq 2$ .

**Exercise 8:** Show that  $n^2 < 2^n$  for all integers  $n \geq 5$ .

**Solution:** Let  $P(n)$  be the statement:  $n^2 < 2^n$  for all integers  $n \geq 5$ .

*Basis step:* For  $n = 5$ ,  $5^2 = 25 < 32 = 2^5$ . Hence,  $P(5)$  holds.

*Inductive hypothesis:* Suppose  $k$  is a positive integer such that  $5 \leq k$  and  $P(k)$  is true.

*Inductive step:* We show that  $P(k+1)$  is true. Now

$$\begin{aligned} (k+1)^2 &= k^2 + 2k + 1 \\ &< 2^k + 2k + 1 \quad \text{by the induction hypothesis} \\ &< 2^k + 2^k \quad \text{by Exercise 10, at the end of} \\ &\quad \text{this section} \\ &= 2 \cdot 2^k \\ &= 2^{k+1}. \end{aligned}$$

So we find that  $P(k+1)$  is true. Hence, by induction  $P(n)$  holds for all  $n \geq 5$ .

**Exercise 9:** Let  $A$  be a set with  $n$  elements. Show that the number of subsets of  $A$  is  $2^n$ .

**Solution:** We prove it by induction on  $n$ . Let  $\mathcal{P}(A)$  denote the power set of  $A$ .

*Basis step:* Suppose  $n = 0$ . Then  $A$  is empty, so  $\mathcal{P}(A) = \{\emptyset\}$ . Thus, we find that the number of subsets of  $A$  is  $1 = 2^0$ .

*Inductive hypothesis:* Let  $k$  be a positive integer and for any set  $A$  with  $k$  elements, the number of subsets of  $A$  is  $2^k$ .

*Inductive step:* Let  $A$  be a set with  $k+1$  elements. Because  $k \geq 0$ ,  $A$  contains at least one element, say  $a$ . Let  $B = A - \{a\}$ . Then the number of elements of  $B$  is  $k$ . By the inductive hypothesis, the number of subsets of  $B$  is  $2^k$ . Let

$$\mathcal{T}_1 = \{D \in \mathcal{P}(A) \mid a \notin D\}.$$

Then, we have  $\mathcal{T}_1 = \mathcal{P}(B)$ , so  $|\mathcal{T}_1| = 2^k$ . Next, let

$$\mathcal{T}_2 = \{C \in \mathcal{P}(A) \mid a \in C\}.$$

Let  $C \in \mathcal{T}_2$ . Then  $a \in C$ , so  $C - \{a\} \in \mathcal{T}_1$ . Similarly, if  $D \in \mathcal{T}_1$ , then  $a \notin D$ , so  $D \cup \{a\} \in \mathcal{T}_2$ . From this we can conclude that  $|\mathcal{T}_2| = |\mathcal{T}_1| = 2^k$ .

Now, suppose  $E \in \mathcal{T}_1 \cap \mathcal{T}_2$ . Then  $E \in \mathcal{T}_1$  and  $E \in \mathcal{T}_2$ . Now  $E \in \mathcal{T}_1$  implies that  $a \notin E$  and  $E \in \mathcal{T}_2$  implies that  $a \in E$ . Thus, we have a contradiction. Hence, we must have  $\mathcal{T}_1 \cap \mathcal{T}_2 = \emptyset$ .

Let  $E \in \mathcal{P}(A)$ . Then either  $a \notin E$  or  $a \in E$ . If  $a \notin E$ , then  $E \in \mathcal{T}_1$  and if  $a \in E$ , then  $E \in \mathcal{T}_2$ . This implies that  $E \in \mathcal{T}_1 \cup \mathcal{T}_2$ . Hence,  $\mathcal{P}(A) \subseteq \mathcal{T}_1 \cup \mathcal{T}_2$ . However,  $\mathcal{T}_1 \cup \mathcal{T}_2 \subseteq \mathcal{P}(A)$ . We therefore have  $\mathcal{P}(A) = \mathcal{T}_1 \cup \mathcal{T}_2$ . Consequently, we have

$$\mathcal{P}(A) = \mathcal{T}_1 \cup \mathcal{T}_2 \quad \text{and} \quad \mathcal{T}_1 \cap \mathcal{T}_2 = \emptyset.$$

This implies that

$$|\mathcal{P}(A)| = |\mathcal{T}_1| + |\mathcal{T}_2| = 2^k + 2^k = 2 \cdot 2^k = 2^{k+1}.$$

This proves the inductive step. Hence, the result now follows by induction.

**Exercise 10:** Prove that the well-ordering principle implies the principle of mathematical induction.

**Solution:** To prove that the principle of mathematical induction follows from the well-ordering principle, we can prove the following: Let  $P(n)$  be a sentence involving the integer  $n$ . Suppose  $P(n)$  is true for some positive integer  $n_0$ . If  $k$  is a positive integer such that  $k \geq n_0$  and  $P(k)$  is true implies that  $P(k+1)$  is true. Then we prove that  $P(n)$  is true for all  $n \geq n_0$ .

**Proof:** Suppose  $P(n)$  is not true for all  $n \geq n_0$ . Then there exists a positive integer  $m > n_0$  such that  $P(m)$  is not true. Let

$$S = \{t \in \mathbb{N} \mid t > n_0 \text{ such that } P(t) \text{ is not true}\}.$$

Because  $m \in S$ ,  $S$  is a nonempty subset of the set of nonnegative integers. Hence, by the well-ordering principle, we find that  $S$  has a smallest element, say  $t_0$ . Because  $t_0 \in S$ ,  $P(t_0)$  is not true. Now  $t_0 > n_0$ , so  $t_0 - 1 \geq n_0$ . Because  $t_0 > t_0 - 1$  and  $t_0$  is the smallest element in  $S$ , it follows that  $t_0 - 1 \notin S$ . Then  $P(t_0 - 1)$  must be true. Therefore, from the hypothesis  $P((t_0 - 1) + 1)$  is true, i.e.,  $P(t_0)$  is true, which is a contradiction. This completes the proof.

**Exercise 11:** Consider the Fibonacci sequence  $f_1, f_2, f_3, \dots$ , where  $f_1 = 1, f_2 = 1$ , and

$$f_n = f_{n-1} + f_{n-2}$$

for  $n \geq 3$ . Then

$$f_3 = f_1 + f_2 = 1 + 1 = 2$$

$$f_4 = f_2 + f_3 = 1 + 2 = 3,$$

⋮

Show by the second principle of induction that

$$f_n \geq u^{n-2} \text{ for } n \geq 3, \quad \text{where } u = \frac{1 + \sqrt{5}}{2}.$$

**Solution:** Let

$$P(n) : f_n \geq u^{n-2}, \quad \text{where } u = \frac{1 + \sqrt{5}}{2}.$$

We prove that

$$\text{for all } n \geq 3, P(n)$$

is true.

We want to prove that  $P(n)$  is true for all  $n \geq 3$ . Therefore, as a basis step, we verify that  $P(3)$  is true.

*Basis step:* For  $n = 3$ , we have  $u = \frac{1 + \sqrt{5}}{2} < 2$  and  $f_3 = 2$ . Thus,

$$f_3 = 2 > u = u^{3-2}.$$

Hence,  $P(3)$  is true.

*Inductive hypothesis:* Assume that  $P(i)$  is true for all integers  $i$  such that  $3 \leq i \leq k$ , where  $k$  is a positive integer.

*Inductive step:* In this step, we show that  $P(k+1)$  is true.

Let  $u = \frac{1 + \sqrt{5}}{2}$ . Observe that  $u$  is a solution of  $x^2 - x - 1 = 0$ . This implies that  $u^2 = u + 1$ . Now

$$\begin{aligned} u^{k-1} &= u^2 \cdot u^{k-3} \\ &= (u + 1) \cdot u^{k-3} \quad \text{because } u^2 = u + 1 \\ &= u^{k-2} + u^{k-3} \\ &\leq f_k + f_{k-1}. \end{aligned} \quad \begin{array}{l} \text{By the inductive step,} \\ P(k) \text{ and } P(k-1) \text{ are true.} \end{array}$$

However,  $f_k + f_{k-1} = f_{k+1}$  for  $k \geq 3$ . Thus,  $f_{k+1} = f_k + f_{k-1} \geq u^{k-1}$ . Hence,  $P(k+1)$  is true.

Consequently, from the second principle of induction, it follows that  $f_n \geq u^{n-2}$  for all  $n \geq 3$ .

**Exercise 12:** Prove that the Euclidean algorithm, given in the preceding section to find the gcd of two integers  $x$  and  $y$  such that  $x > y \geq 0$ , is correct.

**Solution:** For easy reference, we rewrite the main body of the algorithm.

```

1. a := x;
2. b := y;
3. while b ≠ 0 do
4.   begin
5.     r := a mod b;
6.     a := b;
7.     b := r;
8.   end
9. return a; //gcd(x,y) = a.

```

Before the loop executes,  $a_0 = a = x$  and  $b_0 = b = y$ . Because  $0 \leq y < x$ , it follows that  $0 \leq b_0 < a_0$ . Also,  $\gcd(x, y) = \gcd(a, b) = \gcd(a_0, b_0)$ .

Consider the statement in Line 5. Because  $r$  is the remainder when  $a$  is divided by  $b$ ,  $0 \leq r < b$  and there exists an integer  $q$  such that  $a = bq + r$ . By Lemma 2.1.20,

$$\gcd(a, b) = \gcd(b, r).$$

Consider the  $k$ th iteration of the loop. In this iteration, we calculate

$$r_k = a_{k-1} \bmod b_{k-1},$$

$$a_k = b_{k-1}, \text{ and}$$

$$b_k = r_k,$$

where  $r_k$  is the value of  $r$  in this (and also after) this iteration. It follows that  $\gcd(a_{k-1}, b_{k-1}) = \gcd(b_{k-1}, r_k)$  and  $0 \leq b_k = r_k < b_{k-1} = a_k$ .

This suggests that the loop invariant is

$$P(k) : \gcd(a, b) = \gcd(x, y) \quad \text{and} \quad 0 \leq b < a.$$

Next we prove that this is indeed the loop invariant.

*Basis step:*  $P(0)$  : Before the loop executes,  $a_0 = a = x$  and  $b_0 = b = y$ , so  $\gcd(a_0, b_0) = \gcd(a, b) = \gcd(x, y)$  and also  $0 \leq b_0 < a_0$ . Hence,  $P(0)$  is true.

*Inductive hypothesis:* Suppose  $P(k)$  is true before the next iteration of the loop; i.e.,  $\gcd(a_k, b_k) = \gcd(x, y)$  and  $0 \leq b_k < a_k$ , where  $a_k$  and  $b_k$  are the values of  $a$  and  $b$ , respectively, after  $k$  iterations.

*Inductive step:* Consider  $P(k+1) : \gcd(a_{k+1}, b_{k+1}) = \gcd(x, y)$  and  $0 \leq b_{k+1} < a_{k+1}$ .

Let us determine,  $a_{k+1}$ ,  $b_{k+1}$ , and  $r_{k+1}$  during this  $(k+1)$ st iteration.

Because  $P(k)$  is true,  $\gcd(a_k, b_k) = \gcd(x, y)$  and  $0 \leq b_k < a_k$ .

Now  $r_{k+1} := a_k \bmod b_k$ . That is,  $r_{k+1}$  is the remainder when  $a_k$  is divided by  $b_k$ , so  $0 \leq r_{k+1} < b_k$ . Also,

$$\gcd(a_k, b_k) = \gcd(b_k, r_{k+1}).$$

Next,  $a_{k+1} = b_k < a_k$  and  $b_{k+1} = r_{k+1}$ . Hence,

$$\gcd(a_{k+1}, b_{k+1}) = \gcd(b_k, r_{k+1}) = \gcd(a_k, b_k) = \gcd(x, y).$$

Also,  $0 \leq b_{k+1} = r_{k+1} < b_k = a_{k+1}$ , i.e.,  $0 \leq b_{k+1} < a_{k+1}$ .

Consequently,  $P(k+1)$  is true.

We now show that the loop does terminate. This follows from the fact that  $b \geq 0$  and, for each  $k$ ,  $b_{k+1} = r_{k+1} < b_k$ . That is,  $0 \leq b_{k+1} < b_k < b_{k-1} < \dots < b_1 < b_0 = b$ . Consider the set  $A$  of these  $b_j$ 's. Because there are  $b+1$  integers from 0 through  $b$ , it follows that  $A$  is a finite set. Suppose that  $A$  has, say  $N$  elements. Then we must have  $b_N = 0$ . Because if  $b_N \neq 0$ ; i.e., the value of  $b$  after  $N$  iterations is not zero, then the while loop executes again producing  $b_{N+1}$ , so the set  $A$  will have  $N+1$  elements, a contradiction to our assumption. Thus,  $b_N = 0$ . Consequently, the loop does terminate.

Suppose the loop terminates after  $N$  iterations. Then  $P(N)$  is true, and  $b = b_N \neq 0$  is false. Because  $b \neq 0$  is false, it follows that  $b = 0$ . Now  $P(N)$  is true, so

$$\gcd(x, y) = \gcd(a_N, b_N) = \gcd(a_N, 0) = a_N = a,$$

the value of  $a$  after  $N$  iterations. Hence, the value of  $a$ , at Line 9, is the gcd of  $x$  and  $y$ .

It now follows that the loop correctly determines the gcd of  $x$  and  $y$ .

## SECTION REVIEW

### Key Terms

first principle of mathematical induction

basis step

inductive hypothesis

inductive step

second principle of mathematical induction

strong principle of induction

precondition

postcondition

loop invariant

### Some Key Results

1. First principle of mathematical induction: Let  $P(n)$  be a sentence containing nonnegative integer  $n$  and let  $n_0$  be a fixed nonnegative integer.
  - (i) Suppose  $P(n_0)$  is true (i.e.,  $P(n)$  is true for  $n = n_0$ ).
  - (ii) Whenever  $k$  is an integer such that  $k \geq n_0$  and  $P(k)$  is true, then  $P(k+1)$  is true.

Then  $P(n)$  is true for all integers  $n \geq n_0$ .

2. A proof of a mathematical statement by the principle of mathematical induction consists of three steps.

1. *Basis step:* To show that  $P(n_0)$  is true for particular nonnegative integer.
  2. *Inductive hypothesis:* To write the inductive hypothesis: Let  $k$  be an integer such that  $k \geq n_0$  and  $P(k)$  is true.
  3. *Inductive step:* To show that  $P(k+1)$  is true.
3. Let  $P(n)$  be a mathematical sentence about nonnegative integers  $n$  and let  $n_0$  be a fixed nonnegative integer.
- Suppose  $P(n_0)$  is true.
  - If for any integer  $k \geq n_0$ ,  $P(n_0), P(n_0 + 1), P(n_0 + 2), \dots, P(k)$  are true imply that  $P(k+1)$  is true, then  $P(n)$  is true for all  $n \geq n_0$ .

## EXERCISES

---

1. Use induction to prove that  $1 + 3 + 5 + \dots + (2n - 1) = n^2$  for all positive integers  $n$ .
2. Use induction to prove that

$$1 + 4 + 7 + \dots + (3n - 2) = \frac{n(3n - 1)}{2}$$

for all positive integers  $n$ .

3. Use induction to prove that

$$1^2 + 3^2 + 5^2 + \dots + (2n - 1)^2 = \frac{n(2n - 1)(2n + 1)}{3}$$

for all integers  $n \geq 1$ .

4. Use induction to prove that

$$1^3 + 2^3 + 3^3 + \dots + n^3 = \left(\frac{n(n+1)}{2}\right)^2$$

for all positive integers  $n$ .

5. Use induction to prove that

$$\frac{1}{1 \cdot 2} + \frac{1}{2 \cdot 3} + \frac{1}{3 \cdot 4} + \dots + \frac{1}{n(n+1)} = \frac{n}{n+1}$$

for all positive integers  $n$ .

6. Use induction to prove that

$$1^2 - 2^2 + 3^2 - \dots + (-1)^{n-1} n^2 = (-1)^{n-1} \frac{n(n+1)}{2}$$

for all positive integers  $n$ .

7. Use induction to prove that

- 3 divides  $2^{2n} - 1$  for all positive integers  $n$ .
- For all positive integers  $n$ ,  $3 \mid n(n^2 + 5)$ .
- $5^n - 1$  is divisible by 4 for all integers  $n \geq 0$ .
- For all positive integers  $n$ ,  $5 \mid 8^n - 3^n$ .

8. Use induction to prove that

- $12 \mid (13^n - 1)$  for all positive integers  $n$ .
- $8 \mid 12(3^n - 1)$  for all nonnegative integers  $n$ .

- $4 \mid (6 \cdot 7^n + 2 \cdot 3^n)$  for all nonnegative integers  $n$ .
- $8 \mid (7^n + 3^n - 2)$  for all positive integers  $n$ .
- $3 \mid (10^{n+1} + 10^n + 1)$  for all positive integers  $n$ .
- For all positive integers  $n$ ,  $11^n - 7^n$  is divisible by 4.

9. Use induction to prove that

$$1(1!) + 2(2!) + \dots + n(n!) = (n+1)! - 1$$

for all positive integers  $n \geq 1$ .

10. Show that  $2n + 1 < 2^n$  for all integers  $n \geq 3$ .

11. Prove that

- $3^n < n!$  for all integers  $n \geq 7$ .
- $n! \geq 2^n$  for all integers  $n \geq 4$ .

12. Prove that

- $n < 2^n$  for all integers  $n \geq 0$ .
- $n^2 \leq 2^n$  for all integers  $n \geq 4$ .

13. Prove that

$$\frac{1}{1^2} + \frac{1}{2^2} + \dots + \frac{1}{n^2} \leq 2 - \frac{1}{n}$$

for all integers  $n \geq 1$ .

14. Prove that  $n^2 < n!$  for all integers  $n \geq 4$ .

15. Prove that

$$2^{n+1} < 1 + (n+1)2^n$$

for all integers  $n \geq 1$ .

16. Show that  $9^n - 8n - 1$  is divisible by 64 for all integers  $n \geq 0$ .

17. If  $a$  and  $r$  are real numbers such that  $r \neq 1$ , then prove by induction on  $n$  that

$$a + ar + ar^2 + \dots + ar^n = \frac{ar^{n+1} - a}{r - 1}$$

for all integers  $n \geq 0$ .

18. If  $n$  straight lines are drawn on a plane such that no two are parallel and no three pass through the same point, then prove by induction that these  $n$  straight lines divide the plane into  $\frac{1}{2}(n^2 + n + 2)$  distinct regions.
19. By the second principle of mathematical induction, show that any postage charges of more than 17 cents can be made only using 4- and 7-cent stamps.
20. By the second principle of mathematical induction, show that any postage charges of greater than or equal to 5 cents can be made only using 2- and 5-cent stamps.
21. Let  $A, A_1, A_2, \dots, A_n$  be arbitrary sets,  $n \geq 1$ . Prove that
- $A \cap (\bigcup_{i=1}^n A_i) = \bigcup_{i=1}^n (A \cap A_i)$ .
  - $A \cup (\bigcap_{i=1}^n A_i) = \bigcap_{i=1}^n (A \cup A_i)$ .
22. Let  $A_1, A_2, \dots, A_n$  be arbitrary sets,  $n \geq 1$ . Prove that
- $(\bigcup_{i=1}^n A_i)' = \bigcap_{i=1}^n A'_i$ .
  - $(\bigcap_{i=1}^n A_i)' = \bigcup_{i=1}^n A'_i$ .
23. Let  $p, p_1, p_2, \dots, p_n$  be statement variables,  $n \geq 1$ . Prove that
- $p \vee (p_1 \wedge p_2 \wedge \dots \wedge p_n) = (p \vee p_1) \wedge (p \vee p_2) \wedge \dots \wedge (p \vee p_n)$ .
  - $p \wedge (p_1 \vee p_2 \vee \dots \vee p_n) = (p \wedge p_1) \vee (p \wedge p_2) \vee \dots \vee (p \wedge p_n)$ .
24. Let  $p_1, p_2, \dots, p_n$  be statement variables,  $n \geq 1$ . Prove that
- $\sim(p_1 \vee p_2 \vee \dots \vee p_n) = \sim p_1 \wedge \sim p_2 \wedge \dots \wedge \sim p_n$ .
  - $\sim(p_1 \wedge p_2 \wedge \dots \wedge p_n) = \sim p_1 \vee \sim p_2 \vee \dots \vee \sim p_n$ .

25. Determine what is the wrong in the following proofs by mathematical induction.

- a. We prove that in any set of  $n$  students, all students have the same age.

Let  $P(n)$  denote the sentence: In any set of  $n$  students, all students have the same age.

*Basis step:* If  $n = 1$ , then it is a set of only one student. Hence,  $P(1)$  is true.

*Inductive hypothesis:* Let  $k$  be a positive integer. Assume  $P(k)$  is true. This means that in any set of  $k$  students, all students have the same age.

*Inductive step:* Consider a set of  $k + 1$  students. We denote the students by  $A_1, A_2, A_3, A_4, \dots, A_k, A_{k+1}$ . Let

$$A = \{A_1, A_2, A_3, A_4, \dots, A_k\}$$

and

$$B = \{A_2, A_3, A_4, \dots, A_k, A_{k+1}\}.$$

Now  $A$  and  $B$  are both sets of  $k$  students. Hence, by the induction hypothesis, all the students of set  $A$  have the same age and, similarly, all the students of set  $B$  have the same age. Because  $A_2, A_3, A_4, \dots, A_k$  belong to both sets, we conclude that all the students  $A_1, A_2, A_3, A_4, \dots, A_k, A_{k+1}$  have the same age.

- b. We show that for any nonzero integer  $a$ ,  $a^n = 1$  for all nonnegative integers  $n$ .

Let  $P(n)$  denote:  $a^n = 1$  for all nonzero integers  $a$ .

*Basis step:* If  $n = 0$ , then  $a^0 = 1$ , so  $P(0)$  is true.

*Inductive hypothesis:* Suppose  $P(0), P(1), P(2), \dots, P(k)$  are true.

*Inductive step:*  $a^{k+1} = \frac{a^k a^k}{a^{k-1}} = \frac{1 \cdot 1}{1} = 1$ . Hence, by the second principle of mathematical induction, the result follows.

26. Show that the following algorithm correctly determines the sum of the elements of a list. What is the loop invariant?

**ALGORITHM 2.7:** Determine the sum of the elements of a list.

*Input:*  $L[1 \dots n]$ —list of  $n$  elements  
*n*—number of elements in the list

*Output:*  $sum$ —the sum of the elements of  $L[1 \dots n]$

```

1. i := 0;
2. sum := 0;
3. while i < n do
4.   begin
5.     i := i + 1;
6.     sum := sum + L[i];
7.   end
8. return sum;
```

27. Show that the following algorithm correctly determines  $2^n$  and  $(n + 1)!$ , for any nonnegative integer  $n$  such that  $2^n \leq (n + 1)!$ . What is the loop invariant?

**ALGORITHM 2.8:** Compute  $2^n$  and  $(n + 1)!$ .

*Input:*  $n$ —a nonnegative integer

*Output:*  $x$ —such that  $x = 2^n$   
 $y$ —such that  $y = (n + 1)!$

```

1. x := 1;
2. y := 1;
3. i := 0
4. while i < n do
5.   begin
6.     i := i + 1;
7.     x := x * 2;
8.     y := y * (i + 1);
9.   end
```

28. Prove that the division algorithm, as given in this chapter, correctly determines the quotient and remainder.

## 2.4 PRIME NUMBERS

In the school of Pythagoras, the ancient Greek mathematicians distinguished between prime integers and composite integers. They noticed that among the positive integers, 1, 2, 3, 4, 5, 6, ..., some integers have only two positive divisors and the others (except 1) have more than two. The first type of positive integers are called prime integers, and the second type of positive integers are called composite integers. Euclid, a famous Greek mathematician, first proved that there are infinitely many primes. Different questions regarding primes have interested mathematicians from ancient to modern times. For practical applications, it has become necessary to find efficient algorithms for determining whether or not a positive integer is prime. In fact, an active area of research in mathematics and computer science is the search for efficient algorithms for testing whether or not large integers are primes. Prime numbers are used in the construction of code-words, as we will illustrate in Chapter 6. Mathematicians and computer scientists alike are interested in discovering larger and larger primes.

---

**DEFINITION 2.4.1** ► An integer  $p > 1$  is called a **prime number**, or **prime**, if the only positive divisors of  $p$  are 1 and  $p$ . An integer  $q > 1$  that is not prime is called **composite**.

The integers 2, 3, 5, 7, and 11 are prime numbers, and the integers 4, 6, 8, and 9 are composite. Among all the even integers only 2 is prime; all the others are composite.

For any positive integer  $n > 1$ , the integers 1 and  $n$  are called the **trivial positive divisors** of  $n$ . Therefore, an integer  $n > 1$  is a prime integer if and only if  $n$  has only trivial positive divisors, and an integer  $n > 1$  is a composite integer if and only if  $n$  has a nontrivial positive divisor.

The following theorem gives a criterion that can be used to determine whether a positive integer is prime.

**Theorem 2.4.2:** An integer  $p > 1$  is prime if and only if for all integers  $a$  and  $b$ ,  $p$  divides  $ab$  implies either  $p$  divides  $a$  or  $p$  divides  $b$ .

**Proof:** The statement of this theorem is a biimplication. Here, we need to prove two implications. First, we prove the implication: If an integer  $p > 1$  is prime, then for all integers  $a$  and  $b$ ,  $p$  divides  $ab$  implies either  $p$  divides  $a$  or  $p$  divides  $b$ . After this, we prove the implication: If an integer  $p > 1$  is such that for all integers  $a$  and  $b$ ,  $p$  divides  $ab$  implies  $p$  divides  $a$  or  $p$  divides  $b$ , then  $p$  is prime.

Suppose  $p$  is a prime and  $a, b$  are integers such that  $p \mid ab$ . Consider  $p$  and  $a$ . There are two choices: Either  $p \mid a$  or  $p$  does not divide  $a$ . If  $p$  divides  $a$ , then the result is true.

Assume that  $p \nmid a$ . Because  $p$  is prime, the only positive divisors  $p$  are 1 and  $p$ . It follows that

$$\gcd(p, a) = 1.$$

Hence, by Theorem 2.1.19, there exist integers  $r$  and  $t$  such that

$$1 = rp + ta.$$

Multiplying both sides by  $b$ , we get

$$b = brp + tab.$$

Now  $p$  divides  $brp$  and  $p$  divides  $t(ab)$ . Hence, by Theorem 2.1.14(iii),  $p$  divides  $brp + tab$ , i.e.,  $p \mid b$ .

We now prove the second implication: Suppose that  $p > 1$  is an integer such that for all integers  $a$  and  $b$ ,  $p$  divides  $ab$  implies either  $p$  divides  $a$  or  $p$  divides  $b$ . We show that the only positive divisors of  $p$  are 1 and  $p$ .

Let  $q$  be a positive divisor of  $p$ . Then  $q < p$ . Now  $q \mid p$  implies

$$p = qr$$

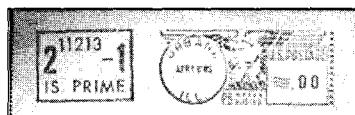
for some integer  $r$ . Also,  $p$  divides  $p$ , so  $p$  divides  $qr$  as  $p = qr$ . Because  $p$  divides  $qr$ , from our assumption, either  $p$  divides  $q$  or  $p$  divides  $r$ . However,  $0 < q < p$ , so  $p$  does not divide  $q$ . Therefore,  $p$  divides  $r$ . This implies  $r = pt$  for some integer  $t$ . Hence, we have

$$p = qr = qpt.$$

This implies  $qt = 1$ , which in turn implies that  $q = 1$ . So we conclude that 1 and  $p$  are the only positive divisors of  $p$ . Hence,  $p$  is prime. ■

#### REMARK 2.4.3 ▶

Typically, after proving one of the implications of a biimplication, to prove the second implication, we usually use the word conversely. For example, in the proof of Theorem 2.4.2, we could have started the proof of the second implication by writing: Conversely, assume that  $p > 1$  is an integer such that for all integers  $a$  and  $b$ ,  $p$  divides  $ab$  implies either  $p$  divides  $a$  or  $p$  divides  $b$ . Most often this is how the proof of a biimplication is structured. That is, first prove one of the implications and then start the proof of the second implication with the word conversely.



Prime numbers have been a mathematical mystery for over 2,000 years. No formula or efficient algorithm has been devised to determine all prime numbers.

Significant results involving primes date back to 300 B.C., when Euclid proved that there are an infinite number of primes. Euclid also proved the Fundamental Theorem of Arithmetic, which states that every integer can be written as a product of primes in a unique way. In 200 B.C., the Greek Eratosthenes created an algorithm for calculating primes called the Sieve of Eratosthenes. This method involves listing all numbers up to  $n$  and then crossing out all multiples of primes up to the square root of  $n$ , the remaining numbers being prime.

The properties of the Fibonacci numbers, a sequence first presented

#### The Search for Prime

in 1200 A.D., are of great interest to mathematicians and computer scientists studying primes. As early as the 1500s, scholars studied the large number of primes that exist based on the formula  $2^n - 1$ , where  $n$  is prime. It was noted, in 1536, that  $2^{11} - 1$  is not prime. However, this formula (called the Mersenne numbers) is still used today for finding candidates for the largest prime number.

During the 1600s, Fermat studied primes extensively and developed Fermat's Little Theorem to help understand the properties of prime numbers. This theorem stated that  $a^p \equiv a \pmod{p}$ , where  $p$  is prime and  $a$  is an integer. He also determined that many numbers of the form  $2^n + 1$  are prime if  $n$  is a power of 2. Euler, working in the 1700's, showed that this was not always true, citing the case of  $2^{32} + 1$ . Euler also determined that the series of the reciprocal of prime numbers is a divergent

series—this says that there are more prime numbers than many people intuitively believe. In 1896, Vallee Poussin proved that the density of primes is  $1/\log n$ , which is known as the Prime Number Theorem.

In the search for the largest prime, the Mersenne numbers have been the main focus of testing. In 1750, Euler verified that  $2^{31} - 1$  was prime. By 1876, Francois Edouard Anatole Lucas verified that  $2^{127} - 1$  was prime. Using the first modern supercomputer (Cray I), David Slowinski found the 27th Mersenne prime in 1979 containing 13,395 digits. Finally, using the Internet, anyone can use the spare processing time on their personal computer to search for primes based on downloadable algorithms. To date, the largest known prime, Mersenne 39, was found in December 2001 by Michael Cameron, George Woltman, and Scott Kurowski; it contains 4,053,946 digits.

**EXAMPLE 2.4.4**

Consider the integer 12. Now 12 divides  $120 = 30 \cdot 4$  but  $12 \nmid 30$  and  $12 \nmid 4$ . Hence, 12 is not prime.

Theorem 2.4.2 is restricted to only two integers  $a$  and  $b$ . It can be generalized as follows.

**Corollary 2.4.5:** If a prime number  $p$  divides  $a_1 a_2 \cdots a_n$ ,  $n > 1$ , then  $p$  divides one of the integers  $a_1, a_2, \dots, a_n$ .

**Proof:** We prove this result by induction on  $n$ .

*Basis step:* Let  $n = 2$ . Suppose  $p$  divides  $a_1 a_2$ . Then, by Theorem 2.4.2,  $p$  divides  $a_1$  or  $a_2$ . Hence, the result is true for  $n = 2$ .

*Inductive hypothesis:* Let  $k \geq 2$  be an integer. If  $p$  divides  $a_1 a_2 \cdots a_k$ , then  $p$  divides one of the integers  $a_1, a_2, \dots, a_k$ .

*Inductive step:* Let  $k + 1 \geq 2$  and suppose  $p$  divides  $a_1 a_2 \cdots a_{k+1}$ . Let us write

$$a_1 a_2 \cdots a_{k+1} = a_1 a_2 \cdots a_k a_{k+1} = ba_{k+1},$$

where  $b = a_1 a_2 \cdots a_k$ . Then  $p \mid ba_{k+1}$ . By Theorem 2.4.2,  $p \mid b$  or  $p \mid a_{k+1}$ . If  $p \mid a_{k+1}$ , then the result is true. So suppose  $p \mid b$ ; that is,  $p \mid a_1 a_2 \cdots a_k$ . By the induction hypothesis,  $p$  divides one of the integers  $a_1, a_2, \dots, a_k$ . Consequently,  $p$  divides one of the integers  $a_1, a_2, \dots, a_{k+1}$ . This proves the inductive step.

Hence, by induction, if  $p$  divides  $a_1 a_2 \cdots a_n$ , then  $p$  divides one of the integers  $a_1, a_2, \dots, a_n$  for all  $n \geq 2$ . ■

Consider now the integers 20, 21, 23, 35. Observe that  $5 \mid 20$ ,  $7 \mid 21$ ,  $23 \mid 23$ ,  $5 \mid 35$ . So we find that each of these integers has a prime factor. We prove this fact for any integer  $n \geq 2$ .

**Theorem 2.4.6:** Every integer  $n \geq 2$  has a prime factor.

**Proof:** Let  $P(n)$  denote the sentence

$$P(n) : \text{If } n \text{ is an integer } \geq 2, \text{ then } n \text{ has a prime factor.}$$

We prove that  $\forall n P(n)$  is true in the domain  $D$  of all integers  $\geq 2$ . We prove this by second principle of mathematical induction.

*Basis step:* If  $n = 2$ , then 2 is a prime factor of  $n$ . Hence, the result holds for  $n = 2$ .

*Inductive hypothesis:* Suppose  $k$  is a positive integer such that  $k \geq 2$  and each of statements  $P(2), P(3), P(4), \dots, P(k)$  is true; i.e., each of the integers 2, 3, 4, ...,  $k$  has a prime factor.

*Inductive step:* Consider now the integer  $k + 1 \geq 2$ . We show that  $P(k + 1)$  is true. If  $k + 1$  is prime, then  $k + 1$  is a prime factor of  $k + 1$ .

Suppose  $k + 1$  is composite. Then there exist integers  $r$  and  $s$  such that

$$k + 1 = rs,$$

where  $1 < r < k + 1$  and  $1 < s < k + 1$ . Then, by the inductive hypothesis,  $r$  has a prime factor which is also a prime factor of  $k + 1$ . Thus,  $k + 1$  has a prime factor. We can now conclude by induction that every integer  $n \geq 2$  has a prime factor. ■

**R E M A R K 2.4.7** ► Theorem 2.4.6 can also be proved by using the well-ordering principle as follows. Suppose there exists a positive integer  $m > 1$  such that  $m$  has no prime factor. Let

$$S = \{n \in \mathbb{N} \mid n \text{ has no prime factor}\}.$$

Clearly,  $m \in S$ . Hence,  $S$  is a nonempty subset of the set of nonnegative integers. Then, by the well-ordering principle,  $S$  has a least element, say  $t$ . Now  $t$  is not a prime integer, otherwise  $t \notin S$ . Therefore,  $t$  is composite. So there exists a positive integer  $p$  such that  $1 < p < t$  and  $p \mid t$ . Because any prime factor of  $p$  is also a prime factor of  $t$ , it follows that  $p$  has no prime factor. Then  $p \in S$ . This contradicts the fact that  $t$  is the least element in  $S$ . Hence, every positive integer  $> 1$  has a prime factor.

The next theorem, which is more than 2,000 years old, shows that the number of primes is infinite. It was proved by Euclid around 300 B.C., and is still to this date considered one of the most elegant proofs in mathematics. It uses the method of *reductio ad absurdum* (i.e., the so-called *method of negation*).

### Theorem 2.4.8: There are infinitely many primes.

**Proof:** Suppose that there are only a finite number of distinct primes. Let the number of primes be  $n$ . Moreover, let  $p_1 = 2, p_2 = 3, p_3 = 5, p_4 = 7, \dots, p_n$  be the list of all prime integers. Construct the integer

$$m = p_1 p_2 p_3 \cdots p_n + 1.$$

Obviously,  $m > 1$ . By Theorem 2.4.6, we see that  $m$  must have a prime factor, say  $p$ . Then  $p$  must be one of  $p_i, i = 1, 2, \dots, n$ . This implies that

$$p \mid (p_1 p_2 p_3 \cdots p_n).$$



**Euclid**

(ca. 325–265 B.C.)

Very little is known about the life of the Greek mathematician

Euclid. It is believed that he was schooled in Plato's renowned Academy

#### Historical Notes

and that he spent the better part of his life developing his mathematical theories in Alexandria under Ptolemy I.

*The Elements*, a collection of 13 volumes concerning various aspects of geometry and mathematics, is credited to Euclid, though there are some who

speculate that the creation of *The Elements* was the work of a collaborative. The clarity of the theorems and postulates within *The Elements* established the rigor with which mathematics would be practiced for centuries.

Let us write  $b = p_1 p_2 p_3 \cdots p_n$ . Now  $p \mid m$  and  $p \mid b$ . Therefore, by Theorem 2.1.14(iii), we have

$$p \mid (m - b).$$

However,  $m - b = p_1 p_2 p_3 \cdots p_n + 1 - p_1 p_2 p_3 \cdots p_n = 1$ , so  $p \nmid 1$ , which is a contradiction as  $p \geq 2$ . Hence, there are infinitely many primes. ■

**REMARK 2.4.9** ▶ Theorem 2.4.8 fascinates mathematicians, and they have devised several different proofs of this theorem. Here we give two other proofs.

*Second proof of Theorem 2.4.8.* Suppose that there are only a finite number of primes. Let  $p$  be the largest one. Let

$$n = p(p - 1)(p - 2) \cdots 3 \cdot 2 + 1.$$

Obviously,  $n > 1$ . By Theorem 2.4.6, we see that  $n$  must have a prime factor, say  $q$ . Because  $q$  does not divide  $p(p - 1)(p - 2) \cdots 3 \cdot 2$ ,  $q > p$ . This contradicts our assumption that  $p$  is the largest prime. Hence, there are infinitely many primes.

*Third proof of Theorem 2.4.8.* Suppose that there are only a finite number of distinct primes. Let the number of primes be  $n$ . Moreover, let  $p_1 = 2, p_2 = 3, p_3 = 5, p_4 = 7, \dots$ , and  $p_n$  be the list of all prime integers. It follows that  $n > 4$ . Let  $x = p_1 p_2 p_3 \cdots p_r$  and  $y = p_{r+1} p_{r+2} \cdots p_n$ , where  $1 < r < n$ . Suppose

$$m = x + y = p_1 p_2 p_3 \cdots p_r + p_{r+1} p_{r+2} \cdots p_n.$$

Obviously,  $m > 1$ . By Theorem 2.4.6, we see that  $m$  must have a prime factor, say  $q$ . Suppose that  $q$  is one of  $p_1, p_2, p_3, \dots$ , or  $p_r$ . Then  $q \mid x$ . Now  $q \mid m$ ,  $q \mid x$ , and  $m = x + y$ , so  $q \mid y$ . This implies that  $q$  is one of  $p_{r+1}, p_{r+2}, \dots, p_n$ , which is a contradiction. Thus,  $q$  cannot be one of  $p_1, p_2, p_3, \dots$ , or  $p_r$ .

Similarly, we can show that  $q$  cannot be any one of  $p_{r+1}, p_{r+2}, \dots$ , or  $p_n$ . Thus,  $q$  is a prime different from the primes  $p_1, p_2, p_3, \dots$ , and  $p_n$ , which is a contradiction. Hence, there are infinitely many primes.

Now, given a particular integer, how can we test whether it is prime? Also, if it is a composite integer, how can we determine nontrivial divisors of  $n$ ? We now proceed to answer all these questions.

**Theorem 2.4.10:** If  $n$  is a composite integer, then  $n$  has a prime factor not exceeding  $\sqrt{n}$ .

**Proof:** Suppose that  $n$  is composite. Then there exist integers  $r$  and  $t$  such that  $1 < r \leq t < n$  and  $n = rt$ . We must have  $r \leq \sqrt{n}$ . For if  $r > \sqrt{n}$ , then  $t \geq r > \sqrt{n}$  and thus

$$n = rt > \sqrt{n}\sqrt{n} = n,$$

which leads to a contradiction. From Theorem 2.4.6, we find that  $r$  has a prime factor, say  $p$ . Now  $p$  is a factor of  $r$  and  $r$  is a factor of  $n$ . Therefore,  $p$  is a prime factor of  $n$ . Moreover,  $p \leq r \leq \sqrt{n}$ . ■

Given a particular integer, how can we determine whether it is prime or composite? We now give an algorithm for making this determination.

**Algorithm to test whether an integer  $n > 1$  is a prime:**

- Step 1.** Check whether  $n$  is 2. If  $n$  is 2,  $n$  is prime; if not go to step 2.
- Step 2.** Check whether 2 divides  $n$ . If 2 divides  $n$ , then  $n$  is not a prime. If 2 does not divide  $n$ , then go to step 3.
- Step 3.** Find all odd primes  $p \leq \sqrt{n}$ . If there is no such odd prime, then  $n$  is prime. Otherwise, go to step 4.
- Step 4.** Check whether  $p$  divides  $n$ , where  $p$  is a prime obtained in step 3. If  $p$  divides  $n$ , then  $n$  is not a prime. If  $p$  does not divide  $n$  for any prime  $p$  obtained in step 3, then  $n$  is a prime.

**EXAMPLE 2.4.11**

Consider the integer 133. Observe that 2 does not divide 133. We now find all odd primes  $p$  such that  $p^2 \leq 133$ . These primes are 3, 5, 7, and 11. Now none of 3, 5, 7, and 11 divide 133. Hence, 133 is a prime.

**EXAMPLE 2.4.12**

Consider the integer 287. The primes 2, 3, 5, 7, 11, and 13 are the only primes  $p$  such that  $p^2 \leq 287$ . None of 2, 3, and 5 divide 287. However, 7 divides 287. Hence, 287 is a composite integer.

We now show how to find all primes less than or equal to a fixed positive integer  $n > 1$ . Let us take  $n = 100$ . First, we find all primes  $p$  such that  $p^2 \leq 100$ . These primes are 2, 3, 5, and 7. So to find all primes less than or equal to 100 we need only find those numbers that are not divisible by 2, 3, 5 and 7. For this we go through the following steps.

- Step 1.** List all integers from 2 to 100.
- Step 2.** Cross out all multiples of 2 that are greater than 2 and less than 100.
- Step 3.** Cross out those integers remaining in the list that are multiples of 3, other than 3.
- Step 4.** Cross out those integers remaining in the list that are multiples of 5, other than 5.
- Step 5.** Cross out those integers remaining in the list that are multiples of 7, other than 7.

All the remaining integers in the list must be prime.<sup>1</sup>

Table 2.1 shows the result of this process. The multiples of 2 are crossed out by /, the multiples of 3 are crossed out by –, the multiples of 5 are crossed out by ×, and multiples of 7 are crossed out by \.

The remaining integers are 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, and 97. These are the prime numbers less than 100.

<sup>1</sup>This process of finding all primes less than a fixed number goes back to antiquity and is known as the Sieve of Eratosthenes.

**Table 2.1** Prime numbers between 1 and 100

	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

**ALGORITHM 2.9:** Determine the list of primes  $p$  such that  $2 \leq p \leq n$ .

*Input:*  $n$ —an integer such that  $n > 1$

*Output:*  $L$ —list of all primes  $p$  such that  $2 \leq p \leq n$

```

1. procedure primes(n, L)
2. begin
3.   for i := 2 to n do
4.     if i is prime then
5.       add i to L;
6. end

```

Note that the statement in Line 4 tests whether  $i$  is prime. Testing whether  $i$  is prime can be accomplished by using the algorithm that comes before Example 2.4.11.

The preceding algorithm can determine all primes less than or equal to a fixed positive integer  $n > 1$ . However, for large integers, this algorithm is not efficient. (There are several efficient algorithms in the literature, but their discussion is beyond the scope of this book.)

Consider the integer 24. We can write  $24 = 2^3 \cdot 3$ . That is, 24 can be written as a product of prime powers. Similarly,  $49,500 = 2^2 \cdot 3^2 \cdot 5^3 \cdot 11$ . In the next theorem, called the **Fundamental Theorem of Arithmetic**, we prove that any positive integer can be written as a product of prime powers.

**Theorem 2.4.13: Fundamental Theorem of Arithmetic.** Every integer  $n \geq 2$  can be expressed uniquely as a product of (one or more) primes, up to the order of the factors. More precisely, any integer  $n \geq 2$  can be expressed as  $n = p_1 p_2 \cdots p_r$  where  $p_1, p_2, \dots, p_r$  are primes. Moreover, if  $n = p_1 p_2 \cdots p_r$  and  $n = q_1 q_2 \cdots q_s$  are two factorizations of  $n$  as a product of primes, then  $r = s$  and the  $q_j$  can be relabeled so that  $p_i = q_i$  for all  $i = 1, 2, \dots, r$ .

**Proof:** We divide the proof into two parts—existence and uniqueness.

*Existence:* Next we prove the existence part.

Let  $P(n)$  denote the statement

$$P(n) : n \text{ can be expressed as a product of primes.}$$

We prove that

$$\forall n P(n) \text{ is true in the domain of all integers } \geq 2.$$

To do so we use the second principle of induction.

*Basis step:* If  $n = 2$ , then  $P(2)$  is true because  $2 = 2$  and 2 is a prime.

*Inductive hypothesis:* Assume that  $k$  is a positive integer,  $k \geq 2$ , and each of the statements  $P(2), P(3), P(4), \dots, P(k)$  is true; i.e., each of the integers  $2, 3, 4, \dots, k - 1, k$  can be expressed as a product of primes.

*Inductive step:* We now show that  $P(k + 1)$  is true; i.e., the integer  $k + 1$  can be expressed as a product of primes.

If  $k + 1$  is prime, then  $P(k + 1)$  is true as we can write  $k + 1 = k + 1$ . Suppose that  $k + 1$  is composite. Then there exist integers  $r$  and  $s$  such that

$$k + 1 = rs,$$

where  $2 \leq r \leq k$  and  $2 \leq s \leq k$ . By the inductive hypothesis, both  $r$  and  $s$  can be expressed as a product of primes. Let

$$r = a_1 a_2 \cdots a_l,$$

and

$$s = b_1 b_2 \cdots b_m,$$

where  $a_1, a_2, \dots, a_l, b_1, b_2, \dots, b_m$  are primes. Thus, we have

$$k + 1 = rs = a_1 a_2 \cdots a_l b_1 b_2 \cdots b_m,$$

which is a product of primes. Hence,  $P(k + 1)$  is true. We can now conclude from induction that any integer  $n \geq 2$  can be expressed as a product of primes.

*Uniqueness:* Next we prove the uniqueness part.

We also prove this by induction.

Let  $Q(n)$  denote the statement: If  $n = p_1 p_2 \cdots p_r$  and  $n = q_1 q_2 \cdots q_s$  are two factorizations of  $n$  as a product of primes, then  $r = s$  and the  $q_j$  can be relabeled so that  $p_i = q_i$  for all  $i = 1, 2, \dots, r$ .

We prove that

$$\forall n Q(n) \text{ is true in the domain of all integers } \geq 2.$$

To do so, we again use the second principle of induction.

*Basis step:* If  $n = 2$ , then  $Q(2)$  is true because  $2 = 2$ .

*Inductive hypothesis:* Assume that  $k$  is a positive integer,  $k \geq 2$ , and each of the statements  $Q(2), Q(3), Q(4), \dots, Q(k)$  is true; i.e., each of the integers  $2, 3, 4, \dots, k - 1, k$  can be expressed as a product of primes uniquely.

*Inductive step:* We now show that  $Q(k + 1)$  is true. If  $k + 1$  is a prime integer, we are done.

Assume that  $k + 1$  is not a prime integer and can be expressed as a product of primes in two ways; say

$$k + 1 = p_1 p_2 \cdots p_t = q_1 q_2 \cdots q_r. \quad (2.14)$$

Now  $p_1$  divides  $k + 1$ . This implies that  $p_1$  divides  $q_1 q_2 \cdots q_r$ . By Corollary 2.4.5,  $p_1$  divides one of  $q_1, q_2, \dots, q_r$ , say  $p_1$  divides  $q_k$ . Now  $p_1$  and  $q_k$  are both primes and  $p_1 \mid q_k$ . Thus, we must have  $p_1 = q_k$ . We cancel this common factor from the equality (2.14) and obtain

$$p_2 p_3 \cdots p_t = q_1 q_2 \cdots q_{k-1} q_{k+1} \cdots q_r. \quad (2.15)$$

Let

$$m = p_2 p_3 \cdots p_t = q_1 q_2 \cdots q_{k-1} q_{k+1} \cdots q_r. \quad (2.16)$$

Now  $m \in \{2, 3, \dots, k\}$ . Thus, by the induction hypothesis,  $P(m)$  is true. Hence, the above two factorizations of  $m$  are the same (up to the order of factors) and therefore the two factorizations of  $k + 1$  are the same. This completes the induction. ■

## Factoring a Positive Integer

Let  $n$  be an integer such that  $n > 1$ . Suppose  $n$  is composite. We would like to factor  $n$ ; i.e., write  $n = ab$ , where  $a$  and  $b$  are integers such that  $1 < a < n$  and  $1 < b < n$ .

By Theorem 2.4.13, every integer  $n > 1$  admits a unique prime factorization. However, how do we find the factorization? We next describe a way to find such a factorization.

Let  $n > 1$  be an integer. By Theorem 2.4.6,  $n$  has a prime factor, say  $p_1$ . Then  $p_1 \mid n$  and so

$$n = p_1 n_1,$$

for some integer  $n_1$ . By Theorem 2.4.10,  $p_1 < \sqrt{n}$ .

If  $n_1 > 1$ , then again by Theorem 2.4.6,  $n_1$  has a prime factor, say  $p_2$ . Then  $p_2 \mid n_1$  and so

$$n_1 = p_2 n_2,$$

for some integer  $n_2$ . By Theorem 2.4.10,  $p_2 < \sqrt{n_1}$ . Also, we can now write

$$n = p_1 p_2 n_2.$$

If  $n_2 > 1$ , then again by Theorem 2.4.6,  $n_2$  has a prime factor, say  $p_3$ . Then  $p_3 \mid n_2$ , and so

$$n_2 = p_3 n_3,$$

for some integer  $n_3$ . By Theorem 2.4.10,  $p_3 < \sqrt{n_2}$ . We can now write

$$n = p_1 p_2 p_3 n_3.$$

We can continue this process. Because there are only a finite number of positive integers less than  $n$ , after a finite number of steps this process must come to an end. Thus, we can write

$$n = p_1 p_2 \cdots p_r, \quad (2.17)$$

where  $p_1, p_2, \dots, p_r$  are primes.

Collecting the same primes in (2.17), we obtain

$$n = p_1^{r_1} p_2^{r_2} \cdots p_k^{r_k}, \quad (2.18)$$

where  $r_i > 0$ ,  $i = 1, 2, \dots, k$ , each  $p_i$  is a prime integer, and  $p_i \neq p_j$  for  $i \neq j$ .

We call the representation of  $n$  in (2.18) the **standard factorization** of  $n$ .

For example, the standard factorization 7875 of is  $7875 = 3^2 \cdot 5^3 \cdot 7$ .

Let us see how efficient this method of factorizing  $n$  is.

Let  $d$  be a positive integer such that  $d$  is a divisor of  $n$ . We can express  $d$  in the standard form as

$$d = p_1^{t_1} p_2^{t_2} \cdots p_k^{t_k}, \quad (2.19)$$

where  $0 \leq t_i \leq r_i$ . From this, it also follows that the number of positive divisors of  $n$  is

$$(r_1 + 1)(r_2 + 1)(r_3 + 1) \cdots (r_k + 1).$$

For example, the number of positive divisors of  $7875 = 3^2 \cdot 5^3 \cdot 7$  is

$$(2 + 1)(3 + 1)(1 + 1) = 24.$$

Therefore, if the number of positive divisors is very large, the preceding method of factorizing  $n$  could be very time-consuming.

### Fermat's Factorization Method

We now describe another method of factorizing a positive integer. It is known as Fermat's Factorization Method.

Let  $n > 1$  be an integer. If  $n$  is an even integer, then we can express

$$n = 2^r m$$

such that  $m$  is odd.

Suppose that  $n$  is an odd integer. In Fermat's Factorization Method, we find two integers  $x$  and  $y$  such that

$$n = x^2 - y^2,$$

so

$$n = (x + y)(x - y).$$

Therefore, we begin by determining possible integers  $x$  and  $y$  such that  $x^2 - n = y^2$ . For this:

- Determine the smallest integer  $k$ , such that  $k^2 \geq n$ .



#### Historical Notes

##### Pierre de Fermat (1601–1665)

Fermat was born to a wealthy aristocratic family and was schooled at the University of Toulouse where he began his first serious mathematical work and initiated relationships with noted mathematicians of his day. After completing his studies at Toulouse, he pursued and obtained a degree in law in Orléans. Fermat then worked as a member of

the lower parliament while continuing his mathematical research and correspondence rather as a hobby than as a profession. He quickly rose through the political system to a high position in the criminal court. His rapid ascension was due in equal parts to seniority and the lethal effects of the plague that was ravishing France.

Best known for Fermat's Last Theorem, Fermat sparked controversy for centuries as mathematicians painstakingly tried to prove his theorem. He also managed to frustrate and raise the ire of mathematicians of his day. Descartes in particular found Fermat's work deficient and without merit. However, the course of time and mathematical discoveries have revealed the true merit of Fermat's work.

2. Look successively at the numbers

$$k^2 - n, (k+1)^2 - n, (k+2)^2 - n, (k+3)^2 - n, \dots$$

until a number  $x \geq \sqrt{n}$  is found making  $x^2 - n$  a square.

For example, consider 5073. We find that

$$71 < \sqrt{5073} < 72.$$

Now

$$\begin{aligned} 72^2 - 5073 &= 111, \\ 73^2 - 5073 &= 256 = 16^2. \end{aligned}$$

We can take  $x = 73$  and  $y = 16$ . Then

$$5073 = 73^2 - 16^2 = (73 + 16)(73 - 16) = 89 \cdot 57.$$

It also follows that 5073 is not a prime number.

**ALGORITHM 2.10:** Fermat's Factorization Algorithm to factor an odd composite positive integer.

*Input:*  $n$ —an odd composite positive integer

*Output:* Positive integers  $a$  and  $b$  such that  $n = ab$

```

1. procedure FermatFactorization( $n$ ,  $a$ ,  $b$ )
2. begin
3.    $x := \lceil \sqrt{n} \rceil;$ 
4.    $y := 0;$ 
5.   while  $x^2 - y^2 > n$  do
6.     begin
7.        $y := y + 1;$ 
8.       if  $x^2 - y^2 < n$  then
9.         begin
10.           $x := x + 1;$ 
11.           $y := 1;$ 
12.        end
13.      else
14.        if  $x^2 - y^2 = n$  then
15.          begin
16.             $a := x - y;$ 
17.             $b := x + y;$ 
18.          end
19.        end
20.      end

```

In this algorithm, the expression  $\lceil n \rceil$  denotes the smallest positive integer greater than or equal to  $n$ .

We began this section by saying that many mathematicians are trying to find larger and larger primes. The largest prime known today has a special form,  $2^p - 1$ , where  $p$  is a prime integer. These types of primes are called *Mersenne primes* after a seventeenth-century French monk who studied the positive integers of the form  $2^m - 1$ , where  $m > 1$ . For any positive integer  $m > 1$ ,  $M_m = 2^m - 1$  is called the  $m$ th Mersenne number, and if  $p$  is prime and  $M_p = 2^p - 1$  is also prime, then  $M_p$  is called a Mersenne prime. For example,  $M_2 = 2^2 - 1 = 3$ ,  $M_3 = 2^3 - 1 = 7$ ,  $M_5 = 2^5 - 1 = 31$ , and  $M_7 = 2^7 - 1 = 127$ , whereas  $M_{11} = 2^{11} - 1 = 2047 = 23 \cdot 89$  is composite.

We conclude the section with the following information.

- In 1992, David Slowinski and Paul Gage verified using a computer that  $2^{756839}-1$  is prime.
- In 1994, David Slowinski and Paul Gage verified using a computer that  $2^{859433}-1$  is prime.
- In 1996, Joel Armengaud, a computer programmer, discovered by computer that  $2^{1398269}-1$  is prime.
- In 1997, Gordon Spence of Hampshire, England, discovered by computer that  $2^{2976221}-1$  is prime.
- In 1999, Nayan Hajratwala discovered by the Lucas-Lehmer test that  $2^{6972593}-1$  is prime.

---

**REMARK 2.4.14** ► Determining if a given number is prime has fascinated mathematicians for several centuries and, more recently, computer scientists for several decades. The first recorded algorithm for this was given by Eratosthenes ca. 250 B.C. However, his algorithm is very inefficient on large numbers. Since then several efforts have been made to design an efficient (in other words, polynomial-time) algorithm. In 2002, the first deterministic polynomial-time algorithm that determines whether an input number is prime or composite was found by an Indian professor, Manindra Agrawal, and two graduate students, Neeraj Kayal and Nitin Saxena.



## WORKED-OUT EXERCISES

---

**Exercise 1:** Show that for any integer  $n > 4$ ,  $n^4 + 64$  is a composite integer.

**Solution:** We have  $n^4 + 64 = (n^2)^2 + (8)^2 + 16n^2 - 16n^2 = (n^2 + 8)^2 - (4n)^2 = (n^2 + 8 + 4n)(n^2 + 8 - 4n) = (n^2 + 8 + 4n)((n - 4)n + 8)$ .

Because  $n > 4$ ,  $n - 4 > 0$ , so  $((n - 4)n + 8) > 1$ . Again,  $(n^2 + 8 + 4n) > 1$ . Hence, for any integer  $n > 4$ ,  $n^4 + 64$  is a composite integer.

**Exercise 2:** Determine which of the following integers are prime.

- (a) 293 (b) 9823

**Solution:**

- We first find all primes  $p$  such that  $p^2 \leq 293$ . These primes are 2, 3, 5, 7, 11, 13, and 17. Now none of these primes divide 293. Hence, 293 is a prime.
- We consider primes  $p$  such that  $p^2 \leq 9823$ . These primes are 2, 3, 5, 7, 11, 13, 17, ... . None of 2, 3, 5, 7 divide 9823. However, 11 divides 9823. Hence, 9823 is not a prime.

**Exercise 3:** Find all prime divisors of  $60!$ , where  $60!$  denotes the factorial of 60, i.e., the product all integers from 1 to 60.

**Solution:** Because  $60!$  is the product of all integers from 1 to 60, the prime divisors of  $60!$  are those primes that are less than 60. Hence, 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, and 59 are the only prime divisors of  $60!$ .

**Exercise 4:** Let  $p$  be a prime such that  $p \mid a^9$ ,  $p \mid (a^2 + b^2)$ . Prove that  $p \mid b$ .

**Solution:** Because  $p \mid a^9$ , by Corollary 2.4.5,  $p \mid a$ . Now  $p \mid a$  and  $p \mid (a^2 + b^2)$ . Therefore, by Theorem 2.1.14(iii),  $p \mid b^2$ . This implies that  $p \mid b$ , because  $p$  is a prime integer.

**Exercise 5:** Let  $p$  be a prime integer such that  $\gcd(a, p^3) = p$  and  $\gcd(b, p^4) = p$ . Find  $\gcd(ab, p^7)$ .

**Solution:** By the given condition,  $\gcd(a, p^3) = p$ . Therefore,  $p \mid a$ . Also,  $p^2 \nmid a$ . (For if  $p^2 \mid a$ , then  $\gcd(a, p^3) \geq p^2 > p$ , which is a contradiction.) Now  $a$  can be written as a product of prime powers. Because  $p \mid a$  and  $p^2 \nmid a$ , it follows that  $p$  appears as a factor in the prime factorization of  $a$ , but  $p^k$ , where  $k \geq 2$ , does not appear in that prime factorization.

In a similar manner,  $\gcd(b, p^4) = p$  implies that  $p \mid b$  and  $p^2 \nmid b$ . As before, it follows that  $p$  appears as a factor in the prime factorization of  $b$ , but  $p^k$ , where  $k \geq 2$ , does not appear in that prime factorization.

It now follows that  $p^2 \mid ab$  and  $p^3 \nmid ab$ . Hence,  $\gcd(ab, p^7) = p^2$ .

**Exercise 6:** Show that every odd prime is the form  $4n + 1$  or  $4n + 3$ . Also, show that the number of primes of the form  $4n + 3$  is infinite.

**Solution:** Any integer is one of the forms  $4n$ ,  $4n + 1$ ,  $4n + 2$ , or  $4n + 3$ . Now  $4n$  cannot be prime for any integer  $n$  and  $4n + 2 = 2(2n + 1)$  is prime only for  $n = 0$ , which is even. Hence, every odd prime is the form  $4n + 1$  or  $4n + 3$ . Next, we show that the number of primes of the form  $4n + 3$  is infinite.

Notice that 3, 7, 11, and 19 are primes of the form  $4n + 3$ . Suppose there exist only a finite number of primes of the form  $4n + 3$ . Let  $p_1, p_2, p_3, \dots$ , and  $p_k$  be the complete list of all primes of the form  $4n + 3$ . Consider the integer

$$m = 4p_1p_2 \cdots p_k - 1.$$

Observe that none of the primes  $p_1, p_2, p_3, \dots, p_k$  and 2 divide  $m$ . Because  $m > 1$ , we find that  $m$  has an odd prime divisor.

Now any odd prime is either of the form  $4n + 1$  or of the form  $4n + 3$ . Because the product of any two integers of the form  $4n + 1$  is again an integer of the form  $4n + 1$ , it follows that all the prime divisors of  $m$  cannot be of the form  $4n + 1$ . Hence,  $m$  has a prime divisor of the form  $4n + 3$ , which must be one of  $p_1, p_2, p_3, \dots, p_k$ . This contradicts the fact that none of  $p_i$  divides  $m$ . Hence, the number of primes of the form  $4n + 3$  is infinite.

**Exercise 7:** Justify that for any positive integer  $n$ ,  $f(n) = n^2 + n + 1$  may not always be prime.

**Solution:** Because  $f(41) = 41^2 + 41 + 1 = 41(41 + 1 + 1)$ , it follows that  $f(41)$  is not prime.

**Exercise 8:** Let  $n$  be a positive integer such that  $n^3 - 1$  is prime. Prove that  $n = 2$ .

**Solution:** We can write

$$n^3 - 1 = (n - 1)(n^2 + n + 1).$$

Because  $n^3 - 1$  is prime, either  $n - 1 = 1$  or  $n^2 + n + 1 = 1$ . Now  $n \geq 1$ , so  $n^2 + n + 1 > 1$ , i.e.,  $n^2 + n + 1 \neq 1$ . Thus, we must have  $n - 1 = 1$ . This implies that  $n = 2$ .

**Exercise 9:** For any positive integer  $n$ , if  $2^n - 1$  is a prime, then prove that  $n$  is also prime.

**Solution:** Suppose that  $2^n - 1$  is prime, but  $n$  is not prime. If  $n = 1$ , then  $2 - 1 = 1$ , which is not prime, a contradiction. If  $n > 1$  and not prime, then there exist integers  $m$  and  $k$  such that  $1 < m < n$ ,  $1 < k < n$ , and  $n = mk$ .

Now,

$$\begin{aligned} 2^n - 1 &= 2^{mk} - 1 \\ &= (2^m - 1)(2^{m(k-1)} + 2^{m(k-2)} + \cdots + 2^m + 1). \end{aligned}$$

Because  $m > 1$ , we find that  $2^m - 1 > 1$ . Also,  $k > 1$ , and so we have

$$2^{m(k-1)} + 2^{m(k-2)} + \cdots + 2^m + 1 > 1.$$

Thus,  $2^n - 1$  is expressed as the product of two integers both of which are greater than one. Hence,  $2^n - 1$  is not prime. This is a contradiction to our assumption. Consequently, if  $2^n - 1$  is a prime, then  $n$  is also prime.

## SECTION REVIEW

### Key Terms

prime number  
prime  
composite

Fundamental Theorem of Arithmetic  
trivial positive divisor  
standard factorization

## Key Definition

- An integer  $p > 1$  is called a prime number, or prime, if the only positive divisors of  $p$  are 1 and  $p$ . An integer  $q > 1$  that is not prime is called composite.

## Some Key Results

- An integer  $p > 1$  is prime if and only if for all integers  $a$  and  $b$ ,  $p$  divides  $ab$  implies either  $p$  divides  $a$  or  $p$  divides  $b$ .
- Every integer  $n \geq 2$  has a prime factor.
- There are infinitely many primes.
- If  $n$  is a composite integer, then  $n$  has a prime factor not exceeding  $\sqrt{n}$ .
- Fundamental Theorem of Arithmetic: Every integer  $n \geq 2$  can be expressed uniquely as a product of (one or more) primes, up to the order of the factors. More precisely, any integer  $n \geq 2$  can be expressed as  $n = p_1 p_2 \cdots p_r$ , where  $p_1, p_2, \dots, p_r$  are primes. Moreover, if  $n = p_1 p_2 \cdots p_r$  and  $n = q_1 q_2 \cdots q_s$  are two factorizations of  $n$  as a product of primes, then  $r = s$  and the  $q_i$  can be relabeled so that  $p_i = q_i$  for all  $i = 1, 2, \dots, r$ .

## EXERCISES

---

- Determine which of the following integers are primes.
  - 391
  - 1999
  - 2033
- Express 873, 675, and 1617 as a product of primes.
- Find all possible divisors of 2502, 399, and 2177.
- Find all prime numbers  $p$  such that  $100 \leq p \leq 140$ .
- Let  $p$  be a prime such that  $p \mid a^6$ ,  $p \mid a^3 + b^7$ . Prove that  $p \mid b$ .
- If  $p$  is a prime integer such that  $p = n^2 - 9$  for some integer  $n$ , then show that  $p = 7$ .
- Let  $p$  be a prime integer such that  $\gcd(a, p^4) = p$  and  $\gcd(b, p^3) = p$ . Find the  $\gcd(ab, p^7)$ .
- If  $n$  is a positive integer such that  $n^3 + 1$  is prime, then show that  $n = 1$ .
- Find all prime factors of 90!
- If  $a > 0$  and  $n \geq 2$  are integers such that  $a^n - 1$  is prime, then show that  $a = 2$ .
- Let  $p_n$  denote the  $n$ th prime. Prove that  $p_{n+1} \leq p_1 p_2 \cdots p_n + 1$  and hence show that  $p_n \leq 2^{2^{n-1}}$
- If  $p$  is a prime, prove that there exist no positive integers  $m$  and  $n$  such that  $m^2 = pn^2$ .
- Let  $k$  be a positive integer. Prove that the following integers

$$(k+1)! + 2, (k+1)! + 3, \dots,$$

$$(k+1)! + k, (k+1)! + (k+1)$$

are  $k$  consecutive composite integers.

- Find five consecutive composite integers.
- Prove that any prime  $p$  of the form  $7k + 1$  is also of the form  $14t + 1$ .
- Prove that there are infinitely many primes of the form  $4n - 1$ .
- Prove that there are infinitely many primes of the form  $3n + 2$ .
- Let  $a, b$  be two integers such that  $3 \mid (a^2 + b^2)$ . Then prove that  $3 \mid a$  and  $3 \mid b$ .
- Prove that the only prime of the form  $n^2 - 4$  is 5.
- Prove that the only prime of the form  $n^3 - 1$  is 7.
- Factor the numbers 8067 and 9,970,716 by Fermat's method.
- If  $p_n$  is the  $n$ th prime, then prove that

$$\frac{1}{p_1} + \frac{1}{p_2} + \cdots + \frac{1}{p_n}$$

is not an integer.

## 2.5 LINEAR DIOPHANTINE EQUATIONS

Trina wants to order a new line of clothing for the store. She wants to order shirts and sweaters costing \$20 for each shirt and \$23 for each sweater. She has a total of \$745 to invest. Trina is interested in knowing how many ways the order can be placed. Suppose that Trina orders  $x$  number of shirts and  $y$  number of sweaters. Then the problem can be stated as: Find all positive integers  $x$  and  $y$  such that

$$20x + 23y = 745.$$

We discussed a similar problem, in Section 2.3 (Mathematical Induction), which was stated as follows: A local post office temporarily ran out of stamps except for 3- and 5-cent stamps. Using the second principle of mathematical induction, we showed that, to mail letters, any postage charges greater than or equal to 8 cents can be made by using 3- and 5-cent stamps.

Suppose Ron wants to use 80-cent stamps. How many ways can Ron do this? This problem (see Worked-Out Exercise 5, at the end of this section) can be stated as follows: Find all positive integers  $x$  and  $y$  such that

$$3x + 5y = 80,$$

where  $x$  is the number of 3-cent stamps and  $y$  is the number of 5-cent stamps.

The mathematician Diophantus of Alexandria initiated the study of such equations in addition to carrying out extensive studies of problems relating to indeterminate equations. It is customary to apply the term Diophantine to any equation with integer coefficients that is to be solved in integers.

A **Diophantine equation** is an algebraic equation in one or more unknowns with integer coefficients, for which integer solutions are sought. Such an equation may have no solution, a finite number of solutions, or an infinite number of solutions. The famous equation

$$x^n + y^n = z^n$$

of Fermat's Last Theorem is also a Diophantine equation.

In this section, we consider linear Diophantine equations only and discuss the necessary and sufficient condition for such an equation to admit integral solutions. We will then apply the results obtained to problems such as the one stated above.

**DEFINITION 2.5.1** ► A linear equation of the form  $ax + by = c$ , where  $a, b, c$  are integers and  $x, y$  are variables such that the solutions are restricted to integers, is called a **linear Diophantine equation in two variables**.



### Historical Notes

#### Diophantus

(ca. 200–284 A.D.)

Much of what is known about Diophantus' life is taken from a cryptic riddle called The Greek Anthology. It implies that Diophantus married at 26 years of age and had a

son who died four years before his own death at 84, though there is speculation that the puzzle is completely fictitious. What is somewhat more certain is that he was a Hellenized Babylonian living in Alexandria during the Silver Age.

Diophantus is best known for his collection of writings called *Arithmetica*, of which 6 of the 13 original volumes

still exist today. The book considers 130 mathematical problems and their solutions—specifically, positive rational solutions. Today, Diophantine equations only allow for linear solutions.

Let

$$ax + by = c$$

be a linear Diophantine equation. If  $(x_0, y_0)$  is a pair of integers such that  $ax_0 + by_0 = c$ , then  $(x_0, y_0)$  is said to be an *integral solution* of this equation.

Consider the Diophantine equation

$$14x + 12y = 33.$$

Suppose this equation has an integral solution  $(x_0, y_0)$ . Then

$$14x_0 + 12y_0 = 33.$$

Now 2 divides  $14x_0 + 12y_0$ . Hence 2 divides 33, which is not true. So we find that a Diophantine equation may not automatically have an integral solution. The next theorem tells us when a Diophantine equation has an integral solution.

### Theorem 2.5.2: The linear Diophantine equation

$$ax + by = c \quad (2.20)$$

with  $a \neq 0, b \neq 0$  has a solution if and only if  $d$  divides  $c$ , where  $d = \gcd(a, b)$ . Moreover, if  $x = x_0, y = y_0$  is a particular solution of this equation, then all solutions of this equation are given by

$$x = x_0 + \frac{b}{d}n, \quad y = y_0 - \frac{a}{d}n,$$

where  $n$  is any integer.

**Proof:** Let  $d = \gcd(a, b)$ .

Suppose  $(x_0, y_0)$  is a solution of  $ax + by = c$ . Then

$$ax_0 + by_0 = c.$$

Because  $d = \gcd(a, b)$ , we have  $d$  divides  $a$  and  $b$ , and so by Theorem 2.1.14(iii),  $d$  divides  $ax_0 + by_0$ . This in turn implies that  $d$  divides  $c$ .

Conversely, suppose  $d$  divides  $c$ . Then  $c = dk$  for some integer  $k$ . Because  $d = \gcd(a, b)$ , by Theorem 2.1.19, there exist integers  $r$  and  $t$  such that  $ar + bt = d$ . Now

$$\begin{aligned} ar + bt &= d \\ \Rightarrow ark + btk &= dk \\ \Rightarrow ark + btk &= c \quad \text{because } c = dk \\ \Rightarrow a(rk) + b(tk) &= c \end{aligned}$$

Now  $rk$  and  $tk$  are integers. Therefore,  $x = rk$  and  $y = tk$  is a solution of the Diophantine equation  $ax + by = c$ .

Suppose now that  $(x_0, y_0)$  is a particular solution of  $ax + by = c$  and  $(x', y')$  is another solution of  $ax + by = c$ . Then

$$ax' + by' = ax_0 + by_0.$$

Dividing both sides by  $d$ , we get

$$\frac{a}{d}x' + \frac{b}{d}y' = \frac{a}{d}x_0 + \frac{b}{d}y_0,$$

which is equivalent to

$$\frac{a}{d}(x' - x_0) = \frac{b}{d}(y_0 - y'). \quad (2.21)$$

Now  $d = \gcd(a, b)$ , so we have

$$\gcd\left(\frac{a}{d}, \frac{b}{d}\right) = 1.$$

From (2.21), we have  $\frac{a}{d}$  divides  $\frac{b}{d}(y_0 - y')$ . Because  $\gcd(\frac{a}{d}, \frac{b}{d}) = 1$ , we must have  $\frac{a}{d}$  divides  $y_0 - y'$ . Therefore, there exists an integer  $k$  such that

$$y_0 - y' = \frac{a}{d}k.$$

This implies that

$$y' = y_0 - \frac{a}{d}k.$$

Substitute  $y_0 - y' = \frac{a}{d}k$  in (2.21) to get

$$\frac{a}{d}(x' - x_0) = \frac{b}{d} \cdot \frac{a}{d}k.$$

Because  $\frac{a}{d} \neq 0$ , canceling  $\frac{a}{d}$  from both sides we get

$$x' - x_0 = \frac{b}{d}k,$$

or

$$x' = x_0 + \frac{b}{d}k.$$

Therefore, for the solution  $(x', y')$  of the Diophantine equation  $ax + by = c$ , there exists an integer  $k$  such that

$$x' = x_0 + \frac{b}{d}k \quad \text{and} \quad y' = y_0 - \frac{a}{d}k.$$

In fact, for any integer  $n$ ,

$$\left(x_0 + \frac{b}{d}n, \quad y_0 - \frac{a}{d}n\right)$$

is a solution of  $ax + by = c$  because

$$\begin{aligned} a\left(x_0 + \frac{b}{d}n\right) + b\left(y_0 - \frac{a}{d}n\right) &= (ax_0 + by_0) + \frac{ab}{d}n - \frac{ab}{d}n \\ &= ax_0 + by_0 = c. \end{aligned}$$

Consequently, if  $(x_0, y_0)$  is a solution of (2.20), then all solutions are given by

$$x = x_0 + \frac{b}{d}n, \quad y = y_0 - \frac{a}{d}n,$$

where  $n \in \mathbb{Z}$ . ■

### EXAMPLE 2.5.3

In this example, we solve the problem posed at the beginning of this section. That is, find the integral solution of the following linear Diophantine equation.

$$20x + 23y = 745. \quad (2.22)$$

Now  $\gcd(20, 23) = 1$  and by using the division algorithm, we have

$$\begin{aligned} 23 &= 1 \cdot 20 + 3, \\ 20 &= 6 \cdot 3 + 2, \\ 3 &= 1 \cdot 2 + 1, \\ 2 &= 2 \cdot 1 + 0. \end{aligned}$$

Hence,

$$\begin{aligned} 1 &= 3 - 1 \cdot 2 \\ &= 3 - 1(20 - 6 \cdot 3) \\ &= 3 - 1 \cdot 20 + 6 \cdot 3 \\ &= 7 \cdot 3 - 1 \cdot 20 \\ &= 7 \cdot (23 - 1 \cdot 20) - 1 \cdot 20 \\ &= 7 \cdot 23 - 8 \cdot 20. \end{aligned}$$

Thus,  $1 = 7 \cdot 23 - 8 \cdot 20$ . Multiply both sides of this equation by 745 to get

$$745 = 20(-5960) + 23(5215).$$

This implies that  $x = -5960$  and  $y = 5215$  is an integral solution of (2.22). Hence, all integral solutions of (2.22) are

$$x = -5960 + 23n, \quad y = 5215 - 20n \quad \text{for any integer } n.$$

Because we must have  $x > 0, y > 0$ , we find that

$$-5960 + 23n > 0 \quad \text{and} \quad 5215 - 20n > 0.$$

Therefore,

$$\frac{5960}{23} < n < \frac{5215}{20},$$

i.e.,

$$259\frac{3}{23} < n < 260\frac{15}{20}.$$

Thus,  $n = 260$ . Because there is only one choice for  $n$ , it follows that there is only one way the order can be placed. Moreover, for  $n = 260$ ,

$$x = -5960 + 23 \cdot 260 = 20 \quad \text{and} \quad y = 5215 - 20 \cdot 260 = 15.$$

Hence, the number of shirts is 20 and the number of sweaters is 15.

#### EXAMPLE 2.5.4

In this example, we determine all solutions of the equation

$$8x + 14y = 58. \tag{2.23}$$

Here  $a = 8$ ,  $b = 14$ , and  $c = 58$ . Now  $\gcd(a, b) = \gcd(8, 14) = 2$  and  $2 \mid 58$ . Hence, by Theorem 2.5.2, (2.23) has a solution. Now

$$2 = 2 \cdot 8 + (-1) \cdot 14.$$

Multiply both sides with 29 to get

$$58 = 8 \cdot 58 + 14 \cdot (-29).$$

This implies that  $x_0 = 58$  and  $y_0 = -29$  is a solution of (2.23).

Again by Theorem 2.5.2, all integral solutions are given by

$$x = 58 + \frac{14}{2}n \quad \text{and} \quad y = -29 - \frac{8}{2}n \quad \text{for all integers } n,$$

i.e.,

$$x = 58 + 7n \quad \text{and} \quad y = -29 - 4n \quad \text{for all integers } n.$$

The following algorithm determines the integral solutions, if any, of a linear Diophantine equation.

**ALGORITHM 2.11:** Determine the integral solutions of a linear Diophantine equation.

*Input:* Integers  $a, b$ , and  $c$   
*Output:*  $(x, y)$ , specifying integral solutions of the linear Diophantine equation  $ax + by = c$

```

1. procedure integralSolutions(a,b,c)
2. begin
3.   d := gcd(a,b);
4.   if c mod d = 0 then //d divides c
5.     begin
6.       determine integers s and t such that d = sa + tb
7.       k := c div d;
8.       x0 := s * k;
9.       y0 := t * k;
10.      print (x0 +  $\frac{b}{d}n$ , y0 -  $\frac{a}{d}n$ )
11.    end
12.   else
13.     print "The equation has no integral solutions."
14.   end
```

Suppose  $x_0, y_0, a, b$ , and  $d = \gcd(a, b)$  are as in Example 2.5.4. Then the print statement in Line 10 outputs  $(58 + 7n, -29 - 4n)$ .

## WORKED-OUT EXERCISES

**Exercise 1:** Find all the integral solutions of the equation  $15x + 14y = 7$ .

**Solution:** Now  $\gcd(15, 14) = 1$  and 1 divides 7. Therefore,  $15x + 14y = 7$  has an integral solution. Now

$$1 = 15 \cdot 1 + 14(-1).$$

We multiply this equality by 7 to get

$$15 \cdot 7 + 14(-7) = 7.$$

This implies that  $x = 7$  and  $y = -7$  is an integral solution of the given equation. All integral solutions of the given equations are given by

$$x = 7 + \frac{14}{1}n \quad \text{and} \quad y = -7 - \frac{15}{1}n \quad \text{for any integer } n,$$

i.e.,

$$x = 7 + 14n, \quad y = -7 - 15n$$

for any integer  $n$ .

**Exercise 2:** Which of the following linear equations cannot be solved in integers?

- (a)  $28x + 16y = 6$       (b)  $51x + 18y = 12$   
 (c)  $21x + 35y = 45$

**Solution:**

- (a)  $\gcd(28, 16) = 4$  and 4 does not divide 6. Hence, this equation has no integral solution.  
 (b)  $\gcd(51, 18) = 3$  and 3 divides 12. This implies that the given equation has solutions in integers.  
 (c)  $\gcd(21, 35) = 7$  and 7 does not divide 45. Hence, the given equation has no integral solution.

**Exercise 3:** Find all positive integral solutions of  $19x + 37y = 500$ .

**Solution:** The  $\gcd(19, 37) = 1$  and 1 divides 500. Hence,  $19x + 37y = 500$  has integral solutions.

By the division algorithm,

$$37 = 19 \cdot 1 + 18$$

$$18 = 1 \cdot 18 + 1.$$

Thus,

$$18 = 37 - 19.$$

$$1 = 19 - 18 = 19 - (37 - 19) = 19 \cdot 2 + 37 \cdot (-1).$$

This implies that

$$500 = 19 \cdot (1000) + 37 \cdot (-500).$$

Thus, an integral solution of the given equation is  $x = 1000, y = -500$ . All integral solutions of the given equation are given by

$$x = 1000 + 37n, \quad y = -500 - 19n,$$

for all integers  $n$ .

When the solutions are positive, then

$$1000 + 37n > 0 \quad \text{and} \quad -500 - 19n > 0.$$

The integer  $n$  must satisfy

$$\begin{aligned} -\frac{1000}{37} &< n < -\frac{500}{19} \\ -27.027 &< n < -26.31. \end{aligned}$$

Therefore,  $n = -27$ . The equation  $19x + 37y = 500$  has exactly one positive integral solution and this is

$$x = 1000 + 37 \cdot (-27) = 1,$$

$$y = -500 - 19 \cdot (-27) = 13,$$

That is, the positive solution of the given Diophantine equation is  $(1, 13)$ .

**Exercise 4:** Steve has \$1000 to buy certain bags and boxes filled with surprise gifts. If each bag costs \$25 and each box costs \$35, how many ways can the items be bought? Also, for each choice, determine the number of each item bought.

**Solution:** Suppose Steve bought  $x$  bags and  $y$  boxes. Then

$$25x + 35y = 1000. \quad (2.24)$$

Now  $\gcd(25, 35) = 5$ . Because  $5 \mid 25$  and  $5 \mid 35$ , the preceding equation has a solution.

By using the division algorithm, we have

$$\begin{aligned} 35 &= 1 \cdot 25 + 10, \\ 25 &= 2 \cdot 10 + 5, \\ 10 &= 2 \cdot 5 + 0. \end{aligned}$$

Hence,

$$\begin{aligned} 5 &= 25 - 2 \cdot 10 \\ &= 25 - 2(35 - 1 \cdot 25) \\ &= 25 - 2 \cdot 35 + 2 \cdot 25 \\ &= 3 \cdot 25 - 2 \cdot 35. \end{aligned}$$

Multiplying both sides by 200, we get

$$1000 = 600 \cdot 25 - 400 \cdot 35.$$

This implies that  $x_0 = 600$  and  $y_0 = -400$  is an integral solution of (2.24). Hence, all integral solutions of (2.24) are

$$\begin{aligned} x &= 600 + \frac{35}{5}n = 600 + 7n, \\ y &= -400 - \frac{25}{5}n = -400 - 5n, \end{aligned} \quad (2.25)$$

for any integer  $n$ .

Because we must have  $x > 0, y > 0$ , we find that

$$600 + 7n > 0 \quad \text{and} \quad -400 - 5n > 0.$$

This implies that

$$600 > -7n \quad \text{and} \quad -5n > 400$$

or

$$\frac{600}{7} > -n \quad \text{and} \quad -n > 80.$$

Let us write  $m = -n$ . Then

$$\frac{600}{7} > m \quad \text{and} \quad m > 80.$$

Therefore,

$$80 < m < \frac{600}{7}.$$

The integers greater than 80 and less than  $\frac{600}{7}$  are 81, 82, 83, 84, and 85. Thus, there are five ways Steve can buy the items.

Substitute  $n = -m$  in (2.25) to get

$$x = 600 - 7m, \quad y = -400 + 5m,$$

where  $m = 81, 82, 83, 84$ , or  $85$ . For each choice of  $m$ , the item can be bought as follows.

$m$	81	82	83	84	85
Number of bags: $x$	33	26	19	12	5
Number of boxes: $y$	5	10	15	20	25

**Exercise 5:** A local post office temporarily ran out of stamps except for 3- and 5-cent stamps. Ron wants to mail a letter that needs 80 cents' worth of stamps. How many ways can Ron use 3- and 5-cent stamps to pay the 80-cents postage charge?

**Solution:** This problem is equivalent to finding all the positive integral solutions of  $3x + 5y = 80$ .

The  $\gcd(3, 5) = 1$  and 1 divides 80. Therefore,  $3x + 5y = 80$  has an integral solution.

Now,

$$1 = 3 \cdot 2 + 5(-1).$$

This implies that

$$80 = 3(160) + 5(-80).$$

Thus, an integral solution of the given equation is  $x = 160$ ,  $y = -80$ . All integral solutions of the given equation are given by

$$x = 160 + 5n, \quad y = -80 - 3n,$$

for all integers  $n$ .

When the solutions are positive, then

$$160 + 5n > 0 \quad \text{and} \quad -80 - 3n > 0.$$

This implies  $n$  must satisfy

$$-\frac{160}{5} < n < -\frac{80}{3}$$

or

$$-32 < n < -26.66.$$

Because  $n$  is an integer, we must have  $n = -31, -30, -29, -28$ , and  $-27$ . Therefore, the equation  $3x + 5y = 80$  has exactly five positive integral solutions and these are

$$\begin{array}{ll} x = 160 + 5 \cdot (-31) = 5, & y = -80 - 3 \cdot (-31) = 13, \\ x = 160 + 5 \cdot (-30) = 10, & y = -80 - 3 \cdot (-30) = 10, \\ x = 160 + 5 \cdot (-29) = 15, & y = -80 - 3 \cdot (-29) = 7, \\ x = 160 + 5 \cdot (-28) = 20, & y = -80 - 3 \cdot (-28) = 4, \\ x = 160 + 5 \cdot (-27) = 25, & y = -80 - 3 \cdot (-27) = 1. \end{array}$$

That is, the positive integral solutions are  $(5, 13)$ ,  $(10, 10)$ ,  $(15, 7)$ ,  $(20, 4)$ , and  $(25, 1)$ . To pay for the 80-cents postage charges, Ron can use the available stamps as follows

Number of 3-cent stamps	Number of 5-cent stamps
5	13
10	10
15	7
20	4
25	1.

## SECTION REVIEW

### Key Term

Diophantine equation

### Key Definition

1. A linear equation of the form  $ax + by = c$ , where  $a, b, c$  are integers and  $x, y$  are variables such that the solutions are restricted to integers, is called a linear Diophantine equation in two variables.

### Key Result

1. The linear Diophantine equation

$$ax + by = c$$

with  $a \neq 0, b \neq 0$  has an integral solution if and only if  $d$  divides  $c$ , where  $d = \gcd(a, b)$ . Moreover, if  $x = x_0, y = y_0$  is a particular integral solution of this equation, then all integral solutions of this equation are given by

$$x = x_0 + \frac{b}{d}n, \quad y = y_0 - \frac{a}{d}n,$$

where  $n$  is any integer.

## EXERCISES

1. Which of the following Diophantine equations cannot be solved?
  - a.  $154x + 260y = 5$
  - b.  $108x + 30y = 7$
  - c.  $45x + 14y = 1$
  - d.  $621x + 736y = 46$
2. Find all integral solutions of the Diophantine equation  $158x - 47y = 9$ .
3. Find all integral solutions of the Diophantine equation  $29x - 19y = 114$ .
4. Find all integral solutions of the Diophantine equation  $101x + 12y = 1$ .
5. Find all integral solutions of the Diophantine equation  $25x + 65y = 50$ .
6. Find all integral solutions of the Diophantine equation  $3x + 4y = 9$ .
7. Find all integral solutions of the Diophantine equation  $7x - 5y = 100$ .
8. Find all positive solutions, if any, of the Diophantine equation  $9x + 7y = 200$ .
9. Find all positive integral solutions, if any, of the Diophantine equation  $17x + 15y = 145$ .
10. Find all positive integral solutions, if any, of the Diophantine equation  $61x + 56y = 7643$ .
11. Find all positive integral solutions, if any, of the Diophantine equation  $2x + 3y = 50$ .
12. Find all positive integral solutions, if any, of the Diophantine equation  $120x + 41y = 11$ .
13. Find all positive integral solutions, if any, of the Diophantine equation  $5x + 7y = 100$ .
14. Find all positive integral solutions, if any, of the Diophantine equation  $14x + 21y = 400$ .
15. John wants to buy \$50 worth of pens and pencils. If each pen costs \$2 and each pencil costs \$3, how many ways can he place his order?
16. Minnie found a new job in Anchorage, Alaska. She needs to buy winter coats and business suits and has only \$1800 to spend. If a winter coat costs \$630 and a business suit costs \$270, how many suits and coats can she buy?
17. A fruit-seller orders \$1000 worth of mangoes and oranges. If one basket of mangoes costs \$20 and one basket of oranges costs \$172, how many baskets of each type does he order?
18. A certain housing complex contains rental apartments of two types: A and B. The monthly rent of each apartment of type A is \$1230 and that of type B is \$870. When all the apartments are rented, the total income is \$87,300 per month. Find the number of apartments of each type.
19. A local post office temporarily ran out of stamps except for 4- and 7-cent stamps. Bob wants to mail a letter that needs 220 cents' worth of postage charges. In how many ways Bob can arrange 4- and 7-cent stamps for this purpose?

## ► PROGRAMMING EXERCISES

1. Write a program to implement the Euclidean algorithm to determine the gcd of positive integers. Let  $a$  and  $b$  be positive integers. Extend the program to determine the integers  $s$  and  $t$  such  $\gcd(a, b) = sa + tb$ .
2. Write a program to implement the algorithm `decimalToBinary` as given in this chapter.
3. In this chapter, we described the algorithm `decimalToBinary` to convert a decimal number into the equivalent binary number. As remarked, two more number systems, octal (base 8) and hexadecimal

(base 16), are of interest to computer scientists. Recall that the digits in the octal number system are 0, 1, 2, 3, 4, 5, 6, and 7; the digits in the hexadecimal number system are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F.

The algorithm to convert a positive decimal number into an equivalent number in octal (or hexadecimal) is the same as discussed for binary numbers. Here we divide the decimal number by 8 (for octal) and by 16 (for hexadecimal). In fact, the algorithm for converting a decimal number to base 2, or 8, or 16 can be extended to any arbitrary base. Suppose we want to convert a decimal number  $n$  into an equivalent

number in base  $b$ , where  $b$  is between 2 and 36. We then divide the decimal number  $n$  by  $b$  as in the algorithm for converting decimal to binary.

Note that the digits in, say base 20, are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G, H, I, and J.

Write a program that converts a number in decimal to a given base  $b$ , where  $b$  is between 2 and 36. The program should prompt the user to enter the number in decimal and in the desired base.

Test your program on the following data.

9098 and base 20

692 and base 8

753 and base 16

4. Write a program to implement the algorithm `binaryToDecimal` as given in this chapter.
5. Write a program to convert a number in a given base,  $b > 1$ , into an equivalent number in base 10.

6. Write a program that performs addition and subtraction of binary numbers. Represent negative numbers as the two's complement of their absolute value. The program should prompt the user to specify the numbers of bits used to store the binary number.
7. Write a program to determine whether a positive integer is prime.
8. Write a program to determine the prime factorization of an integer.
9. Write a program to implement Fermat's factorization algorithm.
10. Write a program to find the solutions, if any, of a linear Diophantine equation in two variables.

## Relations and Posets

The objectives of this chapter are to:

- Learn about relations and their basic properties
- Explore equivalence relations
- Become aware of closures
- Learn about posets
- Explore how relations are used in the design of relational databases

Quite often we come across expressions describing how certain objects or elements are related. For example, in Chapter 1, we said that two logical expressions are equivalent if they have the same truth table. Here, we can consider that two logical expressions are related if they are equivalent. In Chapter 2, we said that two integers are relatively prime if their gcd is 1. Here, we can consider two integers related if they are relatively prime. Let us also consider the following sentences.

1. Robin *is a brother of* Ron.
2. Shelly *is taller than* Juli.
3. 5 *divides* 100.
4. {1, 3, 4} *is a subset of* {1, 2, 3, 4, 5, 6}.
5. The straight line  $l_1$  *is perpendicular to* the straight line  $l_2$ .

Each of these sentences expresses a relationship between people, numbers, sets, or geometric objects. Intuitively, we can say that the word *relation* expresses an association between two objects. Notice that the relation *is a brother of* expresses an association between Robin and Ron. Similarly, the relation *is a subset of*

expresses an association between the sets  $\{1, 3, 4\}$  and  $\{1, 2, 3, 4, 5, 6\}$ .

In fact, relations are a natural way to associate objects of various groups or sets. In this chapter, first we develop various concepts about relations and then we present an application of relations to the design of a relational database.

## 3.1 RELATIONS

---

As previously stated, relations are a natural way to associate objects of various sets. For example, we can take the set, say  $A$ , of people in a town and the set, say  $B$ , of businesses in that town. We can say that an element  $a$  of  $A$  is related to an element  $b$  of  $B$ , if  $a$  is an employee of  $b$ . As another example, we can take the set  $F$  of farmers and the set  $V$  of vegetables. We can relate the elements of  $F$  to the elements of  $V$  by defining that an element  $a$  of  $F$  is related to an element  $b$  of  $V$  if  $a$  grows  $b$ . The obvious question is how to represent relations in mathematics.

Let us consider the second example; i.e., the set  $F$  of farmers and the set  $V$  of vegetables. To be specific, suppose that

$$F = \{\text{Michael, Tara, Anita, Ajay}\}$$

and

$$V = \{\text{beans, tomatoes, peppers, spinach}\}.$$

Suppose that Michael grows beans and peppers, Tara grows tomatoes and peppers, Anita grows tomatoes, and Ajay grows peppers. Rather than writing long sentences, we can make the ordered pair  $(a, b)$  if  $a$  grows  $b$ , where  $a \in F$  and  $b \in V$ . Thus, we have the ordered pairs  $(\text{Michael, beans})$ ,  $(\text{Michael, peppers})$ ,  $(\text{Tara, tomatoes})$ ,  $(\text{Tara, peppers})$ ,  $(\text{Anita, tomatoes})$ ,  $(\text{Ajay, peppers})$ . Next, we can collect these pairs and form the set, say  $R$ , where

$$R = \{(\text{Michael, beans}), (\text{Michael, peppers}), (\text{Tara, tomatoes}), (\text{Tara, peppers}), (\text{Anita, tomatoes}), (\text{Ajay, peppers})\}.$$

From this we see that  $R$  is nothing but a subset of the set  $F \times V$ . This suggests that we can define relations as the sets of ordered pairs, that is, subsets of the Cartesian cross products. Indeed, this is how we define relations in mathematics. More formally, we have the following definition.

---

**DEFINITION 3.1.1** ► A **binary relation**,<sup>1</sup> or simply a **relation**,  $R$  from a set  $A$  into a set  $B$  is a subset of  $A \times B$ .

---

<sup>1</sup>Since this relation is derived out of ordered pairs of elements, it is called a binary relation. If one defines a relation of ordered triplets instead, one may speak of a *ternary relation*. Indeed,  $n$ -ary relations (for any positive integral value of  $n$ ) on a set are perfectly definable, but since we shall have no occasion to deal with them, we drop any adjective to the term *relation* and henceforth speak of a *relation* simply in the sense of a binary relation.

Let  $R$  be a relation from  $A$  into  $B$ , i.e.,  $R \subseteq A \times B$ . If  $(a, b) \in R$ , we say that,  $a$  is  **$R$ -related** (or **related**, if the relation under consideration is understood) to  $b$  and write  $a R b$  (or,  $R(a) = b$ ). If  $(a, b) \notin R$ , i.e., if  $a$  is not  $R$ -related to  $b$ , we denote it by  $a \not R b$ .

### EXAMPLE 3.1.2

Let

$$A = \{\text{New Delhi, Ottawa, London, Paris, Washington}\}$$

$$B = \{\text{Canada, England, India, France, United States}\}.$$

Let  $x \in A$  and  $y \in B$ . Define the relation between  $x$  and  $y$  by “ $x$  is the capital of  $y$ .”

Using this relation, we can make the following ordered pairs.

(New Delhi, India), (Ottawa, Canada), (London, England),

(Paris, France), (Washington, United States)

If we denote

$$R = \{( \text{New Delhi, India}), (\text{Ottawa, Canada}), (\text{London, England}),$$

$$(\text{Paris, France}), (\text{Washington, United States})\},$$

then we find that  $R$  is a subset of  $A \times B$ , so  $R$  is a relation from  $A$  into  $B$ .

Let  $A$  and  $B$  be sets. Because  $\emptyset \subseteq A \times B$  and  $A \times B \subseteq A \times B$ , it follows that  $\emptyset$  and  $A \times B$  are relations from  $A$  into  $B$ , called the **empty relation** and the **universal relation**, respectively. Moreover, for any relation  $R$  from  $A$  into  $B$ , we have  $\emptyset \subseteq R \subseteq A \times B$ .

There are various ways we can express a relation. For example, because a relation  $R$  from a set  $A$  into a set  $B$  is a subset of the set  $A \times B$ , to describe  $R$  we can use the roster form or the set-builder form. Moreover, if the sets  $A$  and  $B$  are finite, we can draw diagrams to describe  $R$ . Some of these methods are described in this chapter. First, however, let us consider some examples where we use the roster and/or set-builder form.

In the following example, we use the roster form to represent a relation.

### EXAMPLE 3.1.3

Let  $A = \{1, 2, 3, 4\}$  and  $B = \{p, q, r\}$ . Let

$$R = \{(1, q), (2, r), (3, q), (4, p)\}.$$

Then  $R$  is a subset of  $A \times B$ , so  $R$  is a relation from  $A$  into  $B$ . Notice that  $1 R q$  but  $3 \not R p$ .

In each of the following examples, we use set-builder form to represent a relation  $R$ .

### EXAMPLE 3.1.4

Let  $A$  denote the set of states in the United States and  $B = \mathbb{N}$ . Let

$$R = \{(x, n) \mid x \in A, n \in B, \text{and } n \text{ denote the}$$

number of people of the state }  $x$  in 2005\}.

Then  $R \subseteq A \times B$ , so  $R$  is a relation from  $A$  into  $B$ . If the number of people in Nebraska in 2005 is, say 1,700,000, then  $(\text{Nebraska}, 1700000) \in R$ .

In the preceding examples, the sets  $A$  and  $B$  are such that  $A \neq B$ . However, to define a relation from a set into another set we only want the relation to be a subset of the Cartesian product. Therefore, these two sets may be the same. In other

words, we can consider a relation  $R$  from a set, say  $A$ , into itself, i.e.,  $R \subseteq A \times A$ . Relations from a set into itself are of special interest. For example, to describe which city is directly connected to another city in the United States, we can define a relation from the set of cities to itself. We will also see that such relations can be described using visual diagrams.

If  $R$  is a relation from a set  $A$  into itself, then we simply say that  $R$  is a *relation* on  $A$ .

### EXAMPLE 3.1.5

Let  $S = \{1, 2, 3, 4, 5\}$ . Let  $R$  be defined by for all  $a, b \in S$ ,  $a R b$  if and only if  $a < b$ . Using the set-builder notation, we can write  $R$  as

$$R = \{(a, b) \mid a, b \in S \text{ and } a < b\}.$$

Then  $R \subseteq S \times S$ , so it is a relation on  $S$ . Now  $1 < 2$ , so  $(1, 2) \in R$ , i.e.,  $1 R 2$ . However,  $3 \not< 2$ , so  $(3, 2) \notin R$ , i.e.,  $3 \not R 2$ .

### EXAMPLE 3.1.6

Let  $S = \{1, 2, 3, 4, 5, 6\}$ . Let  $R$  be defined by for all  $a, b \in S$ ,  $a R b$  if and only if  $a, b$  are relatively prime, i.e.,  $\gcd(a, b) = 1$ . Using set-builder notation, we can write  $R$  as

$$R = \{(a, b) \mid a, b \in S \text{ and } \gcd(a, b) = 1\}.$$

Then  $R \subseteq S \times S$ , so it is a relation on  $S$ . Notice that  $2 R 3$  but  $2 \not R 4$ .

### EXAMPLE 3.1.7

Let  $U$  be a set. Let  $R$  be the set defined as:

$$R = \{(A, B) \mid A, B \in \mathcal{P}(U) \text{ and } A \subseteq B\},$$

where  $\mathcal{P}(U)$  is the power set of  $U$ . In other words, for subsets  $A$  and  $B$  of  $U$ , if  $A$  is a subset of  $B$ , then  $(A, B) \in R$ .

Let  $A$  be a subset of  $U$ . Then  $\emptyset \subseteq A$ , so  $(\emptyset, A) \in R$ . Similarly,  $A \subseteq A$ , so  $(A, A) \in R$ . Also,  $A \subseteq U$ , so  $(A, U) \in R$ . It follows that for all  $A \in \mathcal{P}(U)$ ,  $\emptyset R A$ ,  $A R A$ , and  $A R U$ .

### EXAMPLE 3.1.8

Consider the set  $\mathbb{R}$  of real numbers. A relation on  $\mathbb{R}$  is a subset of  $\mathbb{R} \times \mathbb{R}$ . Let

$$R = \{(x, y) \in \mathbb{R} \times \mathbb{R} \mid x^2 + y^2 = 1\}.$$

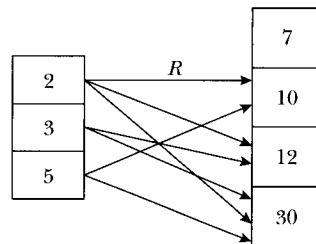
This is a relation on  $\mathbb{R}$ . Notice that  $0 R 1$ , but  $1 \not R 4$ . Note that  $R$  is the set of points on the unit circle, i.e., the set of all points on the circle with center  $(0, 0)$  and radius 1.

As remarked previously, if the sets  $A$  and  $B$  are finite, then we can draw a visual diagram describing the relation  $R$  from  $A$  into  $B$ . The diagram that we describe next is called an **arrow diagram**. Let us explain it with the help of an example.

Let  $A = \{2, 3, 5\}$  and  $B = \{7, 10, 12, 30\}$ . We define the relation  $R$  from  $A$  into  $B$  as follows: For all  $a \in A$  and  $b \in B$ ,  $a R b$  if and only if  $a$  divides  $b$ . Because  $2 | 10$ ,  $2 | 12$ , and  $2 | 30$ , we have  $(2, 10)$ ,  $(2, 12)$ , and  $(2, 30) \in R$ . We can show that

$$R = \{(2, 10), (2, 12), (2, 30), (3, 12), (3, 30), (5, 10), (5, 30)\} \subseteq A \times B.$$

To draw the arrow diagram of  $R$ , we do the following: We write the elements of  $A$  in one column and the elements of  $B$  in another column. We next draw an arrow from an element, say  $a$ , of  $A$  to an element, say  $b$ , of  $B$ , if  $(a, b) \in R$ . Following this convention, we obtain the diagram shown in Figure 3.1.



**FIGURE 3.1** Arrow diagram of the relation  $R$

The symbol  $\longrightarrow$  (called an arrow) represents the relation  $R$ .

Another interesting way of describing a relation defined on a finite set into itself using pictorial representation is by means of a directed graph. In fact, the directed graph representation is a particular type of arrow diagram representation. Next, we describe the directed graph representation of  $R$ .

Let  $R$  be a relation on a finite set  $A$ . We can describe  $R$  pictorially as follows: For each element of  $A$ , we draw a small or big dot and label the dot by the corresponding element of  $A$ . We next draw an arrow from a dot labeled, say  $a$ , to another dot labeled, say  $b$ , if  $a R b$ . The resulting pictorial representation of  $R$  is called the **directed graph representation** of the relation  $R$ . In the directed graph representation of  $R$ , each dot is called a **vertex**.

If a vertex is labeled, say  $a$ , then we also call it vertex  $a$ . An arc from a vertex labeled, say  $a$ , to another vertex, say  $b$ , is called a **directed edge**, or **directed arc**, from  $a$  to  $b$ . The picture thus obtained is also called a **directed graph**, or **digraph**, of  $R$ . (Digraphs are discussed in greater detail in Chapter 10.)

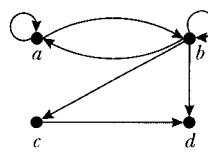
Formally, we define the ordered pair  $(A, R)$  a *directed graph*, or *digraph*, of the relation  $R$ , where each element of  $A$  is a called a *vertex of the digraph*. For vertices  $a$  and  $b$ , if  $a R b$ , we say that  $a$  is **adjacent to  $b$**  and  $b$  is **adjacent from  $a$** .

In the digraph of a relation  $R$ , there is a directed edge or arc from a vertex  $a$  to a vertex  $b$  if and only if  $a R b$ .

Let  $A = \{a, b, c, d\}$ . Let  $R$  be the relation defined by the following set.

$$R = \{(a, a), (a, b), (b, b), (b, c), (b, a), (b, d), (c, d)\}$$

Let us construct a digraph of the relation. To do so, we first draw four dots and label them  $a$ ,  $b$ ,  $c$ , and  $d$ , respectively. Because  $(a, a) \in R$ , we draw an arc from  $a$  to  $a$ ; because  $(a, b) \in R$ , we draw an arc from  $a$  to  $b$ . Similarly, we draw arcs from  $b$  to  $b$ ,  $b$  to  $c$ ,  $b$  to  $a$ ,  $b$  to  $d$ , and  $c$  to  $d$  (see Figure 3.2). The graph in Figure 3.2 is a digraph of the relation  $R$ .



**FIGURE 3.2**  
Digraph of the relation  $R$

How we draw the digraph is not important. For example, we could have put  $a$  and  $c$  in the upper row and  $b$  and  $d$  in the lower row. Of course, we will draw the edges accordingly. The main point is to show the relationship between the vertices.

If the vertices  $a$  and  $b$  are  $R$ -related, i.e.,  $a R b$ , then there is a directed edge from  $a$  to  $b$ , and conversely, if there is a directed edge from  $a$  to  $b$ , then we understand that  $a$  is in the relation  $R$  to  $b$ . Notice that for an element  $a \in A$  such that  $(a, a) \in R$ , we have drawn a directed edge from  $a$  to  $a$ . Such a directed edge is called a **loop** at vertex  $a$ .

## The Domain and Range of the Relation

Let  $R$  be a relation from a set  $A$  into a set  $B$ . Then  $R \subseteq A \times B$ . The elements of the relation  $R$  tell which element of  $A$  is  $R$ -related to which element of  $B$ . We can collect the elements of  $A$  that are related to the elements of  $B$  into a set. Similarly, the elements of  $B$  to which the elements of  $A$  are related can be collected into another set. Therefore, there are two sets (they may be empty) that naturally arise from a relation. More formally, we have the following definition.

---

**DEFINITION 3.1.9** ▶ Let  $R$  be a relation from a set  $A$  into a set  $B$ . Then the **domain** of  $R$ , denoted by  $\mathcal{D}(R)$ , is the set

$$\mathcal{D}(R) = \{a \mid a \in A \text{ and there exists } b \in B \text{ such that } (a, b) \in R\}.$$

The **range**, or **image**, of  $R$ , denoted by  $\mathcal{I}(R)$ , or  $\text{Im}(R)$ , is the set

$$\text{Im}(R) = \{b \mid b \in B \text{ and there exists } a \in A \text{ such that } (a, b) \in R\}.$$

Let  $R$  be a relation from a set  $A$  into a set  $B$ . From Definition 3.1.9, it follows that

1.  $\mathcal{D}(R)$  is the set of all elements of  $A$  that are related to some elements of  $B$ .
2.  $\text{Im}(R)$  is the set of all those elements of  $B$  that have some elements of  $A$  related to them.

### EXAMPLE 3.1.10

Let  $A = \{4, 5, 6, 11\}$  and  $B = \{20, 23, 24, 28, 31\}$ . Let us define a relation  $R$  from  $A$  into  $B$  for all  $a, b \in R$ :

$$a R b \quad \text{if and only if} \quad a \text{ divides } b.$$

Now  $4 \mid 20$ ,  $4 \mid 24$ , and  $4 \mid 28$ . Thus,  $4 R 20$ ,  $4 R 24$ , and  $4 R 28$ . In fact, it can be checked that

$$R = \{(4, 20), (4, 24), (4, 28), (5, 20), (6, 24)\}.$$

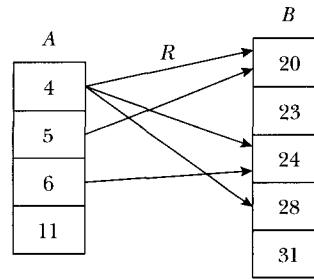
We can now conclude that

$$\mathcal{D}(R) = \{4, 5, 6\}$$

and

$$\text{Im}(R) = \{20, 24, 28\}.$$

The arrow diagram of  $R$  is as shown in Figure 3.3.

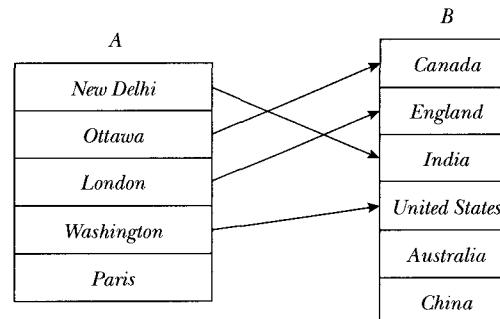


**FIGURE 3.3** Arrow diagram of the relation  $R$  from  $A$  into  $B$

From Figure 3.3, we can see which members are in the domain and which members are in the image set. For example, there is no arrow originating from element 11 of  $A$ , so 11 is not the domain of the relation. Also, there is no arrow ending at elements 23 and 31 of  $B$ . Hence, 23 and 31 are not in the image set.

#### EXAMPLE 3.1.11

Let us consider the following relation (shown in Figure 3.4), where  $\rightarrow$  represents the relation “is a capital of” from set  $A$  into set  $B$ .



**FIGURE 3.4** Arrow diagram of the relation between cities and countries

The domain of this relation is the set {New Delhi, Ottawa, London, Washington} and the image is the set {Canada, England, India, United States}.

Consider the relation  $R$  from set  $A$  into set  $B$  as shown in Figure 3.3. If we reverse the directions of the arrows, we get a relation from set  $B$  into set  $A$ . The relation obtained by reversing the arrows is the inverse relation of the relation  $R$ .

---

**DEFINITION 3.1.12** ▶ Let  $R$  be a relation from a set  $A$  into a set  $B$ . The **inverse** of  $R$ , denoted by  $R^{-1}$ , is the relation from  $B$  into  $A$ , which consists of those ordered pairs that, when reversed, belong to  $R$ , i.e.,

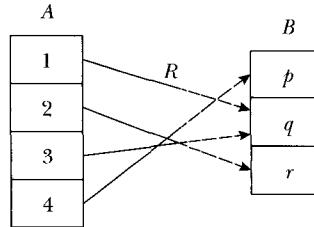
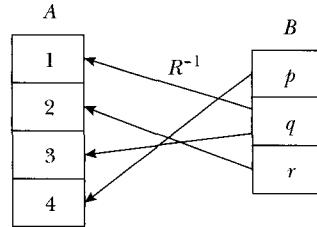
$$R^{-1} = \{(b, a) \mid (a, b) \in R\}.$$

#### EXAMPLE 3.1.13

Let  $A$ ,  $B$ , and  $R$  be as defined in Example 3.1.3. Then

$$R^{-1} = \{(q, 1), (r, 2), (q, 3), (p, 4)\}.$$

The arrow diagram for the relation  $R$  of Example 3.1.3, is shown in Figure 3.5.

FIGURE 3.5 Arrow diagram of  $R$ FIGURE 3.6 Arrow diagram of  $R^{-1}$ 

To find  $R^{-1}$ , just reverse the directions of the arrows, as shown in Figure 3.6. The arrows are from the elements of set  $B$  to the elements of set  $A$ . We also have

$$\begin{aligned}\mathcal{D}(R) &= \{1, 2, 3, 4\} = \text{Im}(R^{-1}), \\ \text{Im}(R) &= \{p, q, r\} = \mathcal{D}(R^{-1}).\end{aligned}$$

We leave the proof of the following theorem, which is illustrated in Example 3.1.13, as an exercise.

**Theorem 3.1.14:** Let  $R$  be a relation from a set  $A$  into a set  $B$ . Then  $\mathcal{D}(R) = \text{Im}(R^{-1})$  and  $\text{Im}(R) = \mathcal{D}(R^{-1})$ . Furthermore,  $(R^{-1})^{-1} = R$ .

### Constructing New Relations from Existing Relations

New relations can be constructed from existing relations. For example, let  $R$  and  $S$  be relations from a set  $A$  into a set  $B$ . Then  $R \subseteq A \times B$  and  $S \subseteq A \times B$ . Because  $R$  and  $S$  are sets, we can construct the sets

$$R \cap S, \quad R \cup S, \quad R - S, \quad \text{and} \quad (A \times B) - R$$

in a natural way. Now  $R \cap S \subseteq A \times B$ ,  $R \cup S \subseteq A \times B$ ,  $R - S \subseteq A \times B$ , and  $(A \times B) - R \subseteq A \times B$ . Therefore,  $R \cap S$ ,  $R \cup S$ ,  $R - S$ , and  $(A \times B) - R$  are relations from  $A$  into  $B$ . In all these relations, the domain and range of the relations under consideration are subsets of  $A$  and  $B$ , respectively.

#### EXAMPLE 3.1.15

Let  $A = \{a, b, c\}$  and  $B = \{1, 2, 3, 4\}$ . Let  $R$  and  $S$  be relations from  $A$  into  $B$  defined as follows:

$$R = \{(a, 2), (a, 4), (b, 2), (b, 3), (c, 1), (c, 4)\}$$

and

$$S = \{(a, 3), (b, 1), (b, 2), (b, 4), (c, 4), (c, 1)\}.$$

Then

$$R \cap S = \{(b, 2), (c, 1), (c, 4)\},$$

$$R \cup S = \{(a, 2), (a, 3), (a, 4), (b, 1), (b, 2), (b, 3), (b, 4), (c, 1), (c, 4)\},$$

$$R - S = \{(a, 2), (a, 4), (b, 3)\}, \quad \text{and}$$

$$(A \times B) - R = \{(a, 1), (a, 3), (b, 1), (b, 4), (c, 2), (c, 3)\}.$$

**EXAMPLE 3.1.16**

Let  $A = \{2, 3, 5, 6, 7\}$  and  $B = \{3, 4, 10, 12, 14, 15\}$ . Let  $R$  and  $S$  be relations from  $A$  into  $B$  defined by: For all  $a \in A$ ,  $b \in B$ ,

$$a R b \quad \text{if and only if} \quad a | b,$$

and

$$a S b \quad \text{if and only if} \quad a \geq b.$$

Then we have

$$\begin{aligned} R = & \{(2, 4), (2, 10), (2, 12), (2, 14), (3, 3), (3, 12), \\ & (3, 15), (5, 10), (5, 15), (6, 12), (7, 14)\} \end{aligned}$$

and

$$S = \{(3, 3), (5, 3), (5, 4), (6, 3), (6, 4), (7, 3), (7, 4)\}.$$

This implies that

$$R \cap S = \{(3, 3)\}$$

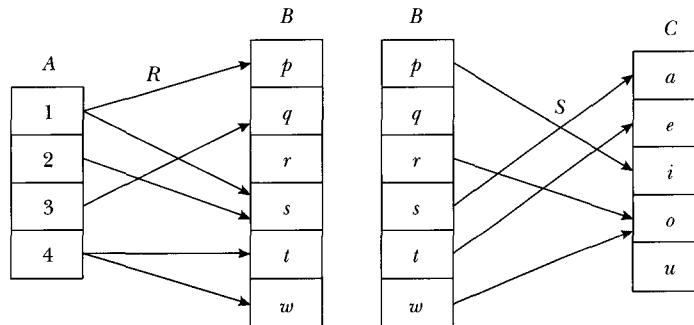
and

$$\begin{aligned} R \cup S = & \{(2, 4), (2, 10), (2, 12), (2, 14), (3, 3), (3, 12), (3, 15), (5, 3), (5, 4), \\ & (5, 10), (5, 15), (6, 3), (6, 4), (6, 12), (7, 3), (7, 4), (7, 14)\}. \end{aligned}$$

Notice that here if  $a | b$  and  $a \geq b$ , then we must have  $a = b$ .

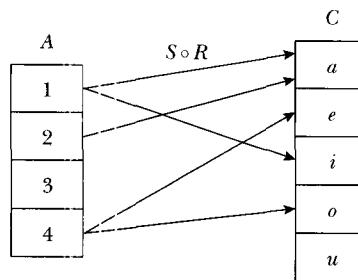
Now, given a relation  $R$  from a set  $A$  into a set  $B$  and a relation  $S$  from set  $B$  into a set  $C$ , there is a relation from set  $A$  into set  $C$  that arises in a natural way as follows: Let us denote the new relation by  $T$ . Suppose  $(a, b) \in R$  and  $(b, c) \in S$ . Then we make  $(a, c) \in T$ . Every element of  $T$  is constructed in this way. That is,  $(a, c) \in T$  for some  $a \in A$  and  $c \in C$  if and only if there exists  $b \in B$  such that  $(a, b) \in R$  and  $(b, c) \in S$ . This relation  $T$  is called the *composition* of  $R$  and  $S$  and is denoted by  $S \circ R$ . Note that to form the composition of  $R$  and  $S$ , the domain of  $S$  and the range of  $R$  must be subsets of the same set.

We explain this with the help of an arrow diagram. Consider the relations  $R$  and  $S$  as given in Figure 3.7.



**FIGURE 3.7** Arrow diagrams of  $R$  and  $S$

The composition  $S \circ R$  is given by Figure 3.8.

FIGURE 3.8 Arrow diagram of  $S \circ R$ 

More formally, we have the following definition.

**DEFINITION 3.1.17** ▶ Let  $R$  be a relation from a set  $A$  into a set  $B$  and let  $S$  be a relation from set  $B$  into a set  $C$ . The **composition** of  $R$  and  $S$ , denoted by  $S \circ R$ , is the relation<sup>2</sup> from  $A$  into  $C$ , defined by, for all  $a \in A$ ,  $c \in C$ ,  $a (S \circ R) c$  if there exists some  $b \in B$  such that  $a R b$  and  $b S c$ .

Let  $R$  be a relation on a set  $A$ . We give the following recursive definition of  $R^n$ ,  $n \in \mathbb{N}$  as:

$$\begin{aligned} R^1 &= R \\ R^n &= R \circ R^{n-1}, \quad \text{if } n > 1. \end{aligned}$$

**REMARK 3.1.18** ▶ Let  $R$  be a relation on a set  $A$ . By induction, we can show that  $R \circ R^{n-1} = R^{n-1} \circ R$  for all integers  $n > 1$ .

### EXAMPLE 3.1.19

Let  $A = \{a, b, c, d, e\}$  and  $R$  be a relation defined on  $A$  by

$$R = \{(a, c), (b, a), (b, b), (b, e), (c, d), (d, c), (d, d), (e, a), (e, c)\}.$$

Let us determine  $R^2$ . Note that  $(x, y) \in R^2$  if and only if there exists  $z \in A$  such that  $(x, y) \in R$  and  $(y, z) \in R$ . We have

$$\begin{array}{lll} a R c, \quad c R d \Rightarrow a R^2 d, & c R d, \quad d R d \Rightarrow c R^2 d, \\ b R a, \quad a R c \Rightarrow b R^2 c, & d R c, \quad c R d \Rightarrow d R^2 d, \\ b R b, \quad b R a \Rightarrow b R^2 a, & d R d, \quad d R c \Rightarrow d R^2 c, \\ b R b, \quad b R b \Rightarrow b R^2 b, & e R a, \quad a R c \Rightarrow e R^2 c, \\ b R b, \quad b R e \Rightarrow b R^2 e, & e R c, \quad c R d \Rightarrow e R^2 d. \\ c R d, \quad d R c \Rightarrow c R^2 c, & \end{array}$$

Hence,

$$R^2 = \{(a, d), (b, a), (b, b), (b, c), (b, e), (c, c), (c, d), (d, c), (d, d), (e, c), (e, d)\}.$$

Observe that  $a R^2 d$ ,  $d R^2 c \Rightarrow a R^3 c$ . Similarly,  $a R^2 d$ ,  $d R d \Rightarrow a R^3 d$ ;  $b R^2 e$ ,  $e R c \Rightarrow b R^3 c$ . As in the case of  $R^2$ , we can show that

$$\begin{aligned} R^3 &= \{(a, c), (a, d), (b, a), (b, b), (b, c), (b, d), (b, e), \\ &\quad (c, c), (c, d), (d, c), (d, d), (e, c), (e, d)\}. \end{aligned}$$

<sup>2</sup>To denote composition of  $R$  and  $S$ , we used  $S \circ R$  rather than  $R \circ S$ . The reason behind the apparent reversal of order in notation will be better appreciated after the introduction of composition of functions in the next chapter.

**EXAMPLE 3.1.20**

Let  $R$  be the relation on  $\mathbb{Z}$  defined by, for all  $a, b \in \mathbb{Z}$ ,

$$a R b \text{ if and only if } a < b.$$

Now,  $1 R 2$  and  $2 R 3$ , so  $1 R^2 3$ . Also, note that  $1 < 2$ , but  $(1, 2) \notin R^2$ . We can show that, for all  $a, b \in \mathbb{Z}$ ,

$$a R^2 b \text{ if and only if there exists } x \in \mathbb{Z} \text{ such that } a < x < b.$$

We leave the proof of the following theorem as an exercise.

**Theorem 3.1.21:** Let  $A, B, C$ , and  $D$  be sets. Let  $R$  be a relation from  $A$  into  $B$ ,  $S$  be a relation from  $B$  into  $C$ , and  $T$  be a relation from  $C$  into  $D$ . Then

$$T \circ (S \circ R) = (T \circ S) \circ R.$$

That is, the composition of relations is associative.

## Equivalence Relations

The relations that we have discussed until now are not required to satisfy any conditions. We now discuss relations that satisfy certain conditions. Later in the book, we will see the applications of the concepts developed next.

**DEFINITION 3.1.22** ▶ Let  $A$  be a set and let  $R$  be a relation on  $A$ . Then  $R$  is called

- (i) **reflexive**, if for all  $a \in A$ ,  $a R a$ ;
- (ii) **symmetric**, if for all  $a, b \in A$ , whenever  $a R b$  holds,  $b R a$  must also hold;
- (iii) **transitive**, if for all  $a, b, c \in A$ , whenever  $a R b$  and  $b R c$  hold,  $a R c$  must also hold.

Let  $A$  be a set and let  $R$  be a relation on  $A$ . Observe that

- $R$  is *not reflexive*, if there exists an  $a \in A$  such that  $(a, a) \notin R$ ;
- $R$  is *not symmetric*, if there exists  $a, b \in A$  such that  $(a, b) \in R$  but  $(b, a) \notin R$ ;
- $R$  is *not transitive*, if there exist  $a, b, c \in A$  such that  $(a, b) \in R$ ,  $(b, c) \in R$  but  $(a, c) \notin R$ .

The concept presented in the next definition is of paramount importance in mathematics and computer science.

**DEFINITION 3.1.23** ▶ A relation  $R$  on a set  $A$  is called an **equivalence relation** if  $R$  is reflexive, symmetric, and transitive.

**EXAMPLE 3.1.24**

Let  $A = \{a, b, c, d\}$  and  $R = \{(a, a), (b, b), (c, c), (d, d), (a, b), (b, a)\}$ . We can show that  $R$  is an equivalence relation on  $A$ .

**EXAMPLE 3.1.25**

Perhaps one of the most natural examples of an equivalence relation is the equality relation on the set of all real numbers. To be specific, let  $R$  be the relation on  $\mathbb{R}$  defined by  $a R b$  if and only if  $a = b$  for all  $a, b \in \mathbb{R}$ . Then  $R$  is an equivalence relation called the **equality relation** on  $\mathbb{R}$ .

If  $R$  is a relation on a finite set  $A$ , then we can effectively use the digraph of  $R$  to determine if  $R$  is reflexive, symmetric, or transitive. In fact, we can show that

1.  $R$  is reflexive if and only if there is a loop at each vertex of the digraph  $(A, R)$ .
2.  $R$  is symmetric if and only if in the digraph of  $R$  if there is a directed edge from one vertex  $a$  to another vertex  $b$ , then there must exist a directed edge from vertex  $b$  to vertex  $a$ .
3.  $R$  is a transitive relation if and only if in the digraph of  $R$  if there is a directed edge from one vertex  $a$  to another vertex  $b$ , and if there exists a directed edge from vertex  $b$  to vertex  $c$ , then there must exist a directed edge from vertex  $a$  to vertex  $c$ .

Consider the following example.

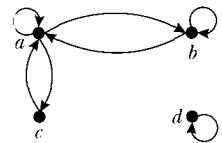
**EXAMPLE 3.1.26**

Let  $A = \{a, b, c, d\}$ . Consider the relation

$$R = \{(a, a), (a, b), (a, c), (b, b), (b, a), (c, a), (d, d)\}$$

on  $A$ .

Let us study the behavior of  $R$  from its digraph as shown in Figure 3.9.



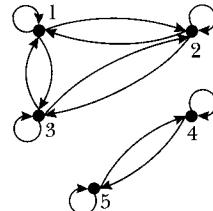
**FIGURE 3.9** Digraph of  $R$

1. The relation  $R$  is not reflexive, because there is no loop at vertex  $c$ .
2. In the digraph if there is a directed edge from one vertex  $x$  to another vertex  $y$ , then there exists a directed edge from vertex  $y$  to vertex  $x$ . For example, there is a directed edge from  $a$  to  $b$ , and we also find that there is a directed edge from  $b$  to  $a$ ; there is a directed edge from  $a$  to  $c$ , and we also find that there is a directed edge from  $c$  to  $a$ . Hence, the given relation is a symmetric relation.
3. The relation is not transitive because there is a directed edge from vertex  $c$  to vertex  $a$  and there exists a directed edge from vertex  $a$  to vertex  $c$ , but there is no directed edge from vertex  $c$  to vertex  $c$ .

Let us consider another example, to clarify these points.

**EXAMPLE 3.1.27**

Let  $A = \{1, 2, 3, 4, 5\}$  and let  $R$  be a relation on  $A$  such that the digraph of  $R$  is as shown in Figure 3.10.

FIGURE 3.10 Digraph of  $R$ 

In the digraph:

1. Every vertex has a loop. Hence,  $R$  is reflexive.
2. We see that if there is a directed edge from one vertex,  $a$ , to another vertex,  $b$ , then there exists a directed edge from vertex  $b$  to vertex  $a$ . Hence,  $R$  is symmetric.
3. We also see that if there is a directed edge from vertex  $a$  to vertex  $b$  and there is a directed edge from vertex  $b$  to vertex  $c$ , then there exists a directed edge from vertex  $a$  to vertex  $c$ . Hence, the relation  $R$  is a transitive relation.

It is important to understand that the reflexivity, symmetry, and transitivity of a relation are *independent* of each other; i.e., not one of these properties implies the others. This fact is illustrated in the following examples.

### EXAMPLE 3.1.28

Let  $R$  be a relation on  $\mathbb{Z}$  defined by: For all  $a, b \in \mathbb{Z}$ ,  $a R b$  if and only if  $ab \geq 0$ .

1. Because for all  $a \in \mathbb{Z}$ ,  $a^2 \geq 0$ , we have  $a R a$ , for all  $a \in \mathbb{Z}$ , so  $R$  is reflexive.
2. Let  $a, b \in \mathbb{Z}$  be such that  $a R b$ . Then,  $ab \geq 0$ . Now,  $ba = ab \geq 0$ , so  $b R a$ . Therefore,  $R$  is symmetric.
3. Now,  $R$  is not transitive. Indeed,  $-2 R 0$  as  $-2 \cdot 0 = 0$  and  $0 R 7$  as  $0 \cdot 7 = 0$ . However,  $-2 \cdot 7 = -14 < 0$ , so  $-2 R 7$ .

Hence,  $R$  is reflexive and symmetric, but not transitive.

### EXAMPLE 3.1.29

Let  $R$  be a relation on  $\mathbb{N}$  such that  $a R b$  if and only if  $a | b$  (i.e.,  $a$  divides  $b$ ) in  $\mathbb{N}$ . Here  $3 R 6$  as  $3 | 6$ , whereas  $6 \nmid 3$ , so  $6 R 3$ . Therefore,  $R$  is not symmetric. We can check that  $R$  is both reflexive and transitive.

### EXAMPLE 3.1.30

Let  $R$  be a relation on  $\mathbb{Z}$  such that

$$a R b \quad \text{if and only if} \quad ab > 0,$$

for all  $a, b \in \mathbb{Z}$ .

1. Now,  $0 \in \mathbb{Z}$  and  $0 \cdot 0 = 0 \not> 0$ . This implies that  $0 R 0$ , so  $R$  is not reflexive.
2. Suppose  $a, b \in \mathbb{Z}$  and  $a R b$ . Then  $ab > 0$ , which implies that  $ba = ab > 0$ , so  $b R a$ . Therefore,  $R$  is symmetric.
3. Let  $a, b, c \in \mathbb{Z}$  and  $a R b$  and  $b R c$ . Thus, we have  $ab > 0$  and  $bc > 0$ . Now  $ab > 0$  implies that either  $a > 0$  and  $b > 0$  or  $a < 0$  and  $b < 0$ .

Suppose that  $a > 0$ ,  $b > 0$ . Now  $bc > 0$  together with  $b > 0$  implies that  $c > 0$ . Thus we have  $a > 0$  and  $c > 0$ , so  $ac > 0$ . Therefore,  $a R c$ .

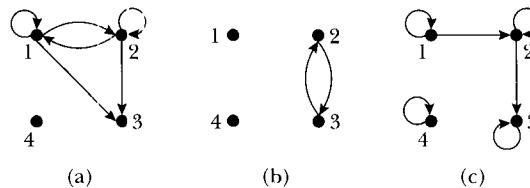
Now suppose that  $a < 0$  and  $b < 0$ . Because  $b < 0$  and  $bc > 0$ , we must have  $c < 0$ . Thus, in this case, we have  $a < 0$  and  $c < 0$ ; i.e., both  $a$  and  $c$  are negative, which implies that  $ac > 0$ . Therefore,  $ac > 0$ , so  $a R c$ .

Thus, in either case we have  $a R c$ . Hence,  $R$  is transitive.

To illustrate that a relation may satisfy one or more of the three properties—reflexive, symmetric, and transitive—the sets in Examples 3.1.28–3.1.30 are infinite sets. Let us consider some examples in which the set  $A$  is a finite set. In this case we can use the digraph of the relation to determine which property is satisfied by the relation.

### EXAMPLE 3.1.31

Let  $A = \{1, 2, 3, 4\}$  and let us consider the digraphs of Figure 3.11, with vertex set  $A$ .



**FIGURE 3.11** Digraphs of relations

1. Let the digraph in Figure 3.11(a) represent the relation  $R_1$  on  $A$ . In this digraph, there is no loop at vertex 3, so  $R_1$  is not reflexive. Next, there is a directed edge from 1 to 3, but no directed edge from 3 to 1. Hence,  $R_1$  is not symmetric. We can check that  $R_1$  is transitive. Notice that  $R_1 = \{(1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 3)\}$ .
2. Let the digraph in Figure 3.11(b) represent the relation  $R_2$  on  $A$ . Because  $(1, 1) \notin R_2$ , it follows that  $R_2$  is not reflexive. Clearly,  $R_2$  is symmetric. Next,  $(2, 3) \in R_2$  and  $(3, 2) \in R_2$ , but  $(2, 2) \notin R_2$ . Hence,  $R_2$  is not transitive. Notice that  $R_2 = \{(2, 3), (3, 2)\}$ .
3. Let the digraph in Figure 3.11(c) represent the relation  $R_3$  on  $A$ . Here  $(a, a) \in R_3$  for all  $a \in A$ . Thus,  $R_3$  is reflexive. Because  $(1, 2) \in R_3$  and  $(2, 1) \notin R_3$ , it follows that  $R_3$  is not symmetric. Moreover,  $(1, 2) \in R_3$  and  $(2, 3) \in R_3$ , but  $(1, 3) \notin R_3$ . Hence,  $R_3$  is not transitive. Notice that  $R_3 = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 2), (2, 3)\}$ .

The relation in the next example was initially studied by Gauss. We will see the application of this relation in Chapter 6. (In fact, all of Chapter 6 is devoted to the study of this relation, its properties, and its applications in various things we use daily.)

### EXAMPLE 3.1.32

Let  $m$  be a fixed positive integer. Define the relation  $R$  on  $\mathbb{Z}$  as follows: For all  $a, b \in \mathbb{Z}$ ,

$$a R b \quad \text{if and only if} \quad m | (a - b); \quad \text{i.e., } a - b = mk \quad \text{for some } k \in \mathbb{Z}.$$

Let us show that  $R$  is an equivalence relation on  $\mathbb{Z}$ .

*Reflexive:* Let  $a \in \mathbb{Z}$ . Now  $a - a = m0$ , so  $a R a$ . It follows that  $R$  is reflexive.

*Symmetric:* Let  $a, b \in \mathbb{Z}$  and  $a R b$ . Then there exists  $k \in \mathbb{Z}$  such that

$$a - b = mk.$$

This implies

$$b - a = (-k)m,$$

so  $m | (b - a)$ , i.e.,  $b R a$ . Hence,  $R$  is symmetric.

*Transitive:* Let  $a, b, c \in \mathbb{Z}$  and suppose  $a R b$  and  $b R c$ . Then there exist  $p, q \in \mathbb{Z}$  such that

$$a - b = pm$$

and

$$b - c = qm.$$

We have

$$a - c = (a - b) + (b - c) = pm + qm = (p + q)m = km,$$

where  $k = p + q \in \mathbb{Z}$ . This implies that  $a R c$ . It follows that  $R$  is transitive.

Consequently,  $R$  is an equivalence relation on  $\mathbb{Z}$ .

The equivalence relation  $R$  is called **congruence modulo  $m$** .

To prove that a relation is an equivalence relation, we must show that the relation is reflexive, symmetric, and transitive. So far, we have done this by using the definitions. The next theorem gives some criteria that can be used to determine whether a relation is an equivalence relation.

**Theorem 3.1.33:** Let  $R$  be a relation on a set  $A$ . Then  $R$  is an equivalence relation on  $A$  if and only if

- (i)  $\delta_A \subseteq R$ , where  $\delta_A = \{(a, a) \mid a \in A\}$ ,
- (ii)  $R = R^{-1}$ , and
- (iii)  $R \circ R \subseteq R$ .

**Proof:** Suppose  $R$  is an equivalence relation. Then  $R$  is reflexive, symmetric, and transitive.

(i) Because  $R$  is reflexive, we have  $(a, a) \in R$  for all  $a \in A$ . This implies that  $\delta_A \subseteq R$ .

(ii) To show  $R = R^{-1}$ , as usual, we show that  $R \subseteq R^{-1}$  and  $R^{-1} \subseteq R$ .

Let  $(a, b) \in R$ . Because  $R$  is symmetric,  $(b, a) \in R$ . Now  $(b, a) \in R$  implies  $(a, b) \in R^{-1}$  by the definition of the inverse relation. Thus,  $R \subseteq R^{-1}$ . Next, let  $(a, b) \in R^{-1}$ . This implies that  $(b, a) \in R$  by the definition of the inverse relation. Now  $(b, a) \in R$ , and  $R$  is symmetric, so  $(a, b) \in R$ . Therefore,  $R^{-1} \subseteq R$ . Hence,  $R = R^{-1}$ . This proves part (ii).

(iii) Let  $(a, b) \in R \circ R$ . By the definition of the composition of relations, there exists  $c \in A$  such that  $(a, c) \in R$  and  $(c, b) \in R$ . The transitive property of  $R$  together with  $(a, c) \in R$ ,  $(c, b) \in R$  implies that  $(a, b) \in R$ . Thus,  $R \circ R \subseteq R$ . Hence, part (iii) holds.

Conversely, suppose that parts (i), (ii), and (iii) hold. By part (i),  $\delta_A \subseteq R$ , so  $(a, a) \in R$  for all  $a \in A$ . Thus,  $R$  is reflexive. Next, let  $(a, b) \in R$ . Then because  $R = R^{-1}$ , we have  $(a, b) \in R^{-1}$ . This implies that  $(b, a) \in R$  by the definition of the inverse of a relation. Hence,  $R$  is symmetric. Finally, to show that  $R$  is transitive, let  $(a, b), (b, c) \in R$ . There exists an element  $b \in A$  such that  $a R b$  and  $b R c$ . This implies that  $(a, c) \in R \circ R \subseteq R$ . Thus,  $R$  is transitive.

Consequently,  $R$  is an equivalence relation. ■

## Equivalence Classes and Partitions

Consider the equivalence relation given in Example 3.1.27. For this relation  $R$ , consider the subsets  $A_1 = \{1, 2, 3\}$  and  $A_2 = \{4, 5\}$ . We can check that any two elements of  $A_1$  are related to each other. Similarly, any two elements of  $A_2$  are related to each other. No element of  $A_1$  is related to any element of  $A_2$ . Moreover,  $A = A_1 \cup A_2$  and  $A_1 \cap A_2 = \emptyset$ . The subsets  $A_1$  and  $A_2$  are usually called equivalence classes of the relation  $R$ . More formally, we have the following definition.

---

**DEFINITION 3.1.34** ▶ Let  $R$  be an equivalence relation on a set  $X$ . For all  $x \in X$ , let  $[x]$  denote the set

$$[x] = \{y \in X \mid y R x\}.$$

The subset  $[x]$  of  $X$  is called the **equivalence class** ( **$R$ -class**, or  **$R$ -equivalence class**) of the equivalence relation  $R$  determined by  $x$ .

The relation

$$R = \{(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (1, 4), (4, 1), (2, 3), (3, 2)\}$$

on the set  $A = \{1, 2, 3, 4, 5\}$  is an equivalence relation. The equivalence class  $[1]$  is the subset of those elements of  $A$  that are related to 1. Because only  $1 R 1$  and  $4 R 1$ , we have  $[1] = \{1, 4\}$ .

The following theorem furnishes some fundamental properties of equivalence classes.

**Theorem 3.1.35:** Let  $R$  be an equivalence relation on a set  $A$ . Then,

- (i) for all  $a \in A$ ,  $[a] \neq \emptyset$ ;
- (ii) if  $b \in [a]$ , then  $[a] = [b]$ , where  $a, b \in A$ ;
- (iii) for all  $a, b \in A$ , either  $[a] = [b]$  or  $[a] \cap [b] = \emptyset$ ;
- (iv)  $A$  is the union of all equivalence classes with respect to  $R$ , i.e.,

$$A = \bigcup_{a \in A} [a].$$

**Proof:**

- (i) Let  $a \in A$ . Because  $R$  is reflexive, we have  $a R a$ . Therefore, by the definition of an equivalence class,  $a \in [a]$ , so  $[a] \neq \emptyset$ .
- (ii) Let  $b \in [a]$ . Then  $b R a$ , and because  $R$  is symmetric, we have  $a R b$ . Now let  $x \in [a]$ . Then  $x R a$ . Thus, we have  $x R a$  and  $a R b$ , so by transitivity, we have  $x R b$ . This implies that  $x \in [b]$ . Hence,  $[a] \subseteq [b]$ . Similarly, we can show that  $[b] \subseteq [a]$ . Consequently,  $[a] = [b]$ .

- (iii) Let  $a, b \in A$  and suppose  $[a] \cap [b] \neq \emptyset$ . Then there exists  $x \in [a] \cap [b]$ . Thus, we have  $x \in [a]$  and  $x \in [b]$ . Because  $x \in [a]$ ,  $[a] = [x]$  by part (ii). Similarly,  $x \in [b]$ , so by part (ii),  $[b] = [x]$ . Hence, we have  $[a] = [x] = [b]$ . Consequently, if  $[a] \cap [b] \neq \emptyset$ , then  $[a] = [b]$ .
- (iv) Let  $a \in A$ . Then  $a \in [a] \subseteq \bigcup_{a \in A} [a]$ . Hence,  $A \subseteq \bigcup_{a \in A} [a]$ . Because  $[a] \subseteq A$  for all  $a \in A$ , we have  $\bigcup_{a \in A} [a] \subseteq A$ . Consequently,  $A = \bigcup_{a \in A} [a]$ . ■

We now introduce the idea of a *partition* of a set. Let  $S$  be a set of 15 pencils, each of which is either red, blue, green, or yellow. Suppose that the elements of  $S$  are listed as follows:

$$S = \{r_1, b_2, b_3, r_4, y_5, y_6, g_7, b_8, y_9, y_{10}, b_{11}, r_{12}, b_{13}, g_{14}, g_{15}\},$$

where  $r$  represents a red pencil,  $b$  represents a blue pencil,  $g$  represents a green pencil, and  $y$  represents a yellow pencil. Let  $R$ ,  $G$ ,  $B$ , and  $Y$  denote the set of all red pencils, the set of all green pencils, the set of all blue pencils, and the set of all yellow pencils, respectively. Then  $R = \{r_1, r_4, r_{12}\}$ ,  $B = \{b_2, b_3, b_8, b_{11}, b_{13}\}$ ,  $G = \{g_7, g_{14}, g_{15}\}$ , and  $Y = \{y_5, y_6, y_9, y_{10}\}$ . None of these subsets is empty; the union  $R \cup G \cup B \cup Y$  of these subsets is the set  $S$ ; and these subsets are pairwise disjoint, i.e.,  $R \cap G = R \cap B = R \cap Y = G \cap B = G \cap Y = B \cap Y = \emptyset$ . In other words, the subsets  $R$ ,  $G$ ,  $B$ , and  $Y$  partition the set  $S$  into nonoverlapping regions.

Let us consider another example.

### EXAMPLE 3.1.36

Consider the relation

$$R = \{(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (1, 4), (4, 1), (2, 3), (3, 2)\}$$

on the set  $S = \{1, 2, 3, 4, 5\}$ . Let us draw the digraph of  $R$ , which is shown in Figure 3.12.

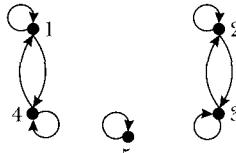


FIGURE 3.12 Digraph of  $R$

We find that the digraph is divided into three distinct blocks. From these blocks we can form the subsets  $\{1, 4\}$ ,  $\{2, 3\}$ , and  $\{5\}$ . These subsets are pairwise disjoint, and their union is  $S$ .

Thus, we see that a partition of a nonempty set  $S$  is a division of  $S$  into nonintersecting nonempty subsets.

More formally, we have the following definition.

**DEFINITION 3.1.37** ▶ Let  $S$  be a nonempty set and let  $\mathcal{P}$  be a collection of nonempty subsets of  $S$ . Then  $\mathcal{P}$  is called a **partition** of  $S$ , if the following properties hold:

- (i) For all  $A_i, A_j \in \mathcal{P}$ , either  $A_i = A_j$  or  $A_i \cap A_j = \emptyset$ ,
- (ii)  $S = \bigcup_{A_i \in \mathcal{P}} A_i$ .

Let  $\mathcal{P}$  be a partition of a nonempty set  $S$  and let  $A \in \mathcal{P}$ . Then  $A$  is called a **block** of the partition  $\mathcal{P}$ .

**EXAMPLE 3.1.38**

Let  $S = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$ . Let  $S_1 = \{1, 3\}$ ,  $S_2 = \{2, 5, 7\}$ ,  $S_3 = \{4, 6\}$ ,  $S_4 = \{8\}$ ,  $S_5 = \{9, 10\}$ , and  $S_6 = \{11, 12\}$ . Then  $S_1, S_2, S_3, S_4, S_5$ , and  $S_6$  are nonempty and

$$S := S_1 \cup S_2 \cup S_3 \cup S_4 \cup S_5 \cup S_6.$$

Also notice that the sets  $S_1, S_2, S_3, S_4, S_5$ , and  $S_6$  are pairwise disjoint. It now follows that  $\mathcal{P} = \{S_1, S_2, S_3, S_4, S_5, S_6\}$  is a partition of  $S$ .

The following are examples of some other partitions of  $S$ :

$$\mathcal{Q} = \{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}, \{7\}, \{8\}, \{9\}, \{10\}, \{11\}, \{12\}\},$$

$$\mathcal{R} = \{\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}\},$$

$$\mathcal{S} = \{\{1, 3, 5, 7, 9\}, \{2, 4, 6, 8, 10\}, \{11, 12\}\}.$$

From Example 3.1.38, it follows that partition of a set is not unique. The following example shows that we can also partition infinite sets and that we may have more than one partition of an infinite set.

**EXAMPLE 3.1.39**

1. Consider  $\mathbb{Z}$ , the set of integers. Let  $A$  be the set of all even integers and let  $B$  be the set of all odd integers. Then  $A$  and  $B$  are nonempty,  $A \cap B = \emptyset$ , and  $\mathbb{Z} = A \cup B$ . It follows that  $\{A, B\}$  is a partition of  $\mathbb{Z}$ .

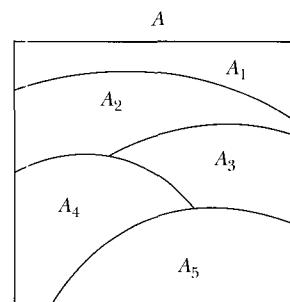
Also, notice that if  $C = \{x \in \mathbb{Z} \mid x \leq 0\}$ , then the set  $\{\mathbb{N}, C\}$  is another partition of  $\mathbb{Z}$ .

2. Consider the set  $\mathbb{N}$ . Let  $n \in \mathbb{N}$  and consider the set  $A_n = \{n\}$ ; that is, the set  $A_n$  consists of only one element. For example,  $A_1 = \{1\}$ ,  $A_2 = \{2\}$ , and so on. We can show that the sets  $A_1, A_2, A_3, \dots$  are pairwise disjoint and their union is  $\mathbb{N}$ . Thus, the set  $\mathcal{P} = \{A_1, A_2, A_3, \dots\}$  is a partition of  $\mathbb{N}$ . Notice that the partition  $\mathcal{P}$  consists of an infinite number of sets.

**EXAMPLE 3.1.40**

Let  $A$  denote the set of the lowercase English alphabet. Let  $B$  be the set of lowercase consonants and  $C$  be the set of lowercase vowels. Then  $B$  and  $C$  are nonempty,  $B \cap C = \emptyset$ , and  $A = B \cup C$ . Thus,  $\{B, C\}$  is a partition of  $A$ .

Let  $A$  be a set and let  $\{A_1, A_2, A_3, A_4, A_5\}$  be a partition of  $A$ . Corresponding to this partition, we can draw a Venn diagram, as shown in Figure 3.13.



**FIGURE 3.13** Partition of  $A$

Let  $A$  be a nonempty set and  $R$  be an equivalence relation on  $A$ . Let  $\mathcal{P} = \{[a] \mid a \in A\}$ ; i.e.,  $\mathcal{P}$  is the set of all equivalence classes of  $A$  with respect to the relation  $R$ . By Theorem 3.1.35, we have

- (i) for all  $a \in A$ ,  $[a] \neq \emptyset$ ;
- (ii) for all  $a, b \in A$ , either  $[a] = [b]$  or  $[a] \cap [b] = \emptyset$ ;
- (iii)  $A = \bigcup_{[a] \in \mathcal{P}} [a]$ .

It now follows that  $\mathcal{P}$  is a partition of the set  $A$ . We record this result in the following theorem.

**Theorem 3.1.41:** Let  $R$  be an equivalence relation on a set  $A$ . Then,  $\mathcal{P} = \{[a] \mid a \in A\}$ ; the set of all equivalence classes of  $R$  is a partition of  $A$ .

The following example further clarifies Theorem 3.1.41.

### EXAMPLE 3.1.42

Consider the set  $\mathbb{Z}$  and the relation  $R$  on  $\mathbb{Z}$  as defined in Example 3.1.32, where  $m$  is a fixed positive integer. To be specific, let  $m = 3$ . Then for any two integers  $n$  and  $t$ ,  $n R t$  if and only if 3 divides  $n - t$ . For this equivalence relation, the equivalence class  $[0]$  of 0 is

$$[0] = \{\dots, -6, -3, 0, 3, 6, 9, 12, \dots\}.$$

Similarly, the equivalence class  $[1]$  of 1 is

$$[1] = \{\dots, -8, -5, -2, 1, 4, 7, 10, \dots\},$$

and the equivalence class  $[2]$  of 2 is

$$[2] = \{\dots, -7, -4, -1, 2, 5, 8, 11, \dots\}.$$

We can verify that  $\mathbb{Z} = [0] \cup [1] \cup [2]$ .

By Theorem 3.1.41, given an equivalence relation on a nonempty set  $A$ , we can construct a partition of  $A$  consisting of the equivalence classes on  $A$ . Turning the matter around, we now prove that, corresponding to any given partition of a set, one can associate an equivalence relation.

**Theorem 3.1.43:** Let  $\mathcal{P}$  be a partition of a given set  $S$ . Define a relation  $R$  on  $S$  as follows: For all  $a, b \in S$ ,

$$a R b \quad \text{if and only if} \quad \text{there exists } B \in \mathcal{P} \text{ such that } a, b \in B.$$

Then  $R$  is an equivalence relation on  $S$ . Moreover, the corresponding equivalence classes are precisely the elements of  $\mathcal{P}$ .

**Proof:** Because  $\mathcal{P}$  is a partition of  $S$ , we have

$$S = \bigcup_{B \in \mathcal{P}} B.$$

Next, we show that  $R$  is an equivalence relation. (First notice that the elements  $a, b \in S$  are  $R$ -related if and only if  $a$  and  $b$  are in the same element of  $\mathcal{P}$ . Moreover, if  $B \in \mathcal{P}$ , then any two elements of  $B$  are related.)

*Reflexive:* Let  $a \in S$ . This implies that  $a \in \bigcup_{B \in \mathcal{P}} B$ , so  $a \in B$  for some  $B \in \mathcal{P}$ . Because  $a, a \in B$ , it follows that  $a R a$ . Hence,  $R$  is reflexive.

*Symmetric:* Let  $a R b$ . Then there exists  $B \in \mathcal{P}$  such that  $a, b \in B$ . This implies that  $b, a \in B$ , so  $b R a$ . Hence,  $R$  is symmetric.

*Transitive:* Let  $a, b, c \in A$  and  $a R b, b R c$ . Then there exist  $B, C \in \mathcal{P}$  such that  $a, b \in B$  and  $b, c \in C$ . Thus, we have  $b \in B \cap C$ , which implies that  $B \cap C \neq \emptyset$ . Because  $\mathcal{P}$  is a partition of  $S$ ,  $B \cap C \neq \emptyset$  implies that  $B = C$ . Thus,  $c \in C = B$ . Hence,  $a, c \in B$ , so  $a R c$  holds. This shows that  $R$  is transitive.

Consequently,  $R$  is an equivalence relation.

Finally, we show that, the equivalence classes of  $R$  are precisely the elements of  $\mathcal{P}$ . Let  $a \in S$ . Because  $S = \bigcup_{B \in \mathcal{P}} B$ , there exists  $B \in \mathcal{P}$  such that  $a \in B$ . We claim that  $[a] = B$ , where  $[a]$  is the equivalence class containing  $a$ . To prove this, let  $x \in [a]$ . Then  $x R a$ , so  $x \in B$  as  $a \in B$ . Thus,  $[a] \subseteq B$ . Again, because  $a \in B$ , we have  $b R a$  for all  $b \in B$ , so  $b \in [a]$  for all  $b \in B$ . Hence,  $B \subseteq [a]$ . It now follows that  $[a] = B$ . Next, observe that if  $C \in \mathcal{P}$ , then  $C = [u]$  for all  $u \in C$ . Hence, the equivalence classes with respect to  $R$  are precisely the elements of  $\mathcal{P}$ . ■

The relation  $R$  described in Theorem 3.1.43 is called the **equivalence relation on  $S$  induced by the partition  $\mathcal{P}$** .

Consider the partition

$$S = S_1 \cup S_2 \cup S_3 \cup S_4 \cup S_5 \cup S_6.$$

on the set

$$S = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\},$$

where  $S_1 = \{1, 3\}$ ,  $S_2 = \{2, 5, 7\}$ ,  $S_3 = \{4, 6\}$ ,  $S_4 = \{8\}$ ,  $S_5 = \{9, 10\}$ , and  $S_6 = \{11, 12\}$ . To find the equivalence relation  $R$  induced by the partition  $\{S_1, S_2, S_3, S_4, S_5, S_6\}$ , we do the following. Consider the set  $S_1$  of the partition, and make any two elements of  $S_1$  related to each other. Then  $(1, 1), (3, 3), (1, 3), (3, 1) \in R$ . We do the same with other sets of the partition. So the equivalence relation  $R$  on  $S$  induced by this partition is

$$\begin{aligned} R = & \{(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (6, 6), (7, 7), (8, 8), (9, 9), (10, 10), \\ & (11, 11), (12, 12), (1, 3), (3, 1), (2, 5), (5, 2), (2, 7), (7, 2), (5, 7), \\ & (7, 5), (4, 6), (6, 4), (9, 10), (10, 9), (11, 12), (12, 11)\}. \end{aligned}$$

## Closures

Let  $A$  be the set

$$A = \{1, 2, 3, 4\}.$$

Let  $R$  be a relation on  $A$  defined by

$$R = \{(1, 1), (3, 3), (1, 3), (2, 3), (3, 2), (4, 2)\}.$$

Because  $(2, 2) \notin R$ ,  $R$  is not reflexive. Consider the set  $S = R \cup \{(2, 2), (4, 4)\}$ , i.e.,

$$S = \{(1, 1), (2, 2), (3, 3), (1, 3), (2, 3), (3, 2), (4, 2), (4, 4)\}.$$

(Notice that  $(4, 4) \notin R$ ). Then  $S$  is a relation on  $A$ ,  $S$  is reflexive, and  $R \subseteq S$ . Moreover, observe that  $S$  is the smallest such relation on  $A$ . That is, if  $T$  is a reflexive relation on  $A$  and  $R \subseteq T$ , then we must have  $S \subseteq T$ .

In fact, what we did was add enough elements to the elements of  $R$  to make the resulting relation reflexive. Such a relation  $S$  is called the *reflexive closure* of  $R$ . More formally, we have the following definition.

**DEFINITION 3.1.44** ▶ Let  $R$  be a relation on a nonempty subset  $A$ . A relation  $S$  on  $A$  is called the **reflexive closure** of  $R$  if

- (i)  $S$  is reflexive;
- (ii)  $R \subseteq S$ ; and
- (iii) if  $T$  is a reflexive relation on  $A$  such that  $R \subseteq T$ , then  $S \subseteq T$ .

From the definition of the reflexive closure, it follows that if  $R$  is reflexive, then the reflexive closure of  $R$  is  $R$  itself.

Before giving a necessary and sufficient condition for a relation to be the reflexive closure of a relation, let us observe one more thing from the preceding relation  $R$ .

We have  $(1, 3) \in R$ , but  $(3, 1) \notin R$ . Thus,  $R$  is not symmetric. We also have  $(4, 2) \in R$ , but  $(2, 4) \notin R$ . Therefore, if we consider the relation

$$S' = \{(1, 1), (3, 3), (1, 3), (3, 1), (2, 3), (3, 2), (4, 2), (2, 4)\},$$

then we have

- (i)  $S'$  is a symmetric relation on  $A$ , and
- (ii)  $R \subseteq S'$ .

Moreover, we can also prove that if  $T$  is any symmetric relation on  $A$  such that  $R \subseteq T$ , then  $S' \subseteq T$ . In other words,  $S'$  is the smallest symmetric relation on  $A$  such that  $R \subseteq S'$ . Such a relation  $S'$  is called the **symmetric closure** of  $R$ . More formally, we have the following definition.

**DEFINITION 3.1.45** ▶ Let  $R$  be a relation on a nonempty set  $A$ . A relation  $S$  on  $A$  is called the **symmetric closure** of  $R$  if

- (i)  $S$  is symmetric;
- (ii)  $R \subseteq S$ ; and
- (iii) if  $T$  is a symmetric relation on  $A$  such that  $R \subseteq T$ , then  $S \subseteq T$ .

From the definition of the symmetric closure, it follows that if  $R$  is symmetric, then the symmetric closure of  $R$  is  $R$  itself.

The following theorem not only gives a necessary and sufficient condition for a relation to be the reflexive (symmetric) closure, it also shows how to construct the reflexive (symmetric) closure.

**Theorem 3.1.46:** Let  $R$  be a relation on a nonempty set  $A$ .

- (i) A relation  $S$  on  $A$  is the **reflexive closure** of  $R$  if and only if  $S = R \cup \delta_A$ , where

$$\delta_A = \{(a, a) \mid a \in A\}.$$

- (ii) A relation  $S$  on  $A$  is the **symmetric closure** of  $R$  if and only if  $S = R \cup R^{-1}$ .

**Proof:**

- (i) Let us first observe the following: Because for all  $a \in A$ ,

$$(a, a) \in \delta_A \subseteq R \cup \delta_A,$$

it follows that  $R \cup \delta_A$  is reflexive. Moreover,  $R \subseteq R \cup \delta_A$ .

Let  $S$  be the reflexive closure of  $R$ . Then  $S$  is the smallest reflexive relation on  $A$  such that  $R \subseteq S$ . Next, we show that  $S = R \cup \delta_A$ . To do so, we show that  $S \subseteq R \cup \delta_A$  and  $R \cup \delta_A \subseteq S$ .

Because  $S$  is reflexive,  $(a, a) \in S$  for all  $a \in A$ , so  $\delta_A \subseteq S$ . Also,  $R \subseteq S$ . Thus,  $R \cup \delta_A \subseteq S$ . On the other hand,  $R \cup \delta_A$  is a reflexive relation on  $A$  and  $R \subseteq R \cup \delta_A$ . Therefore, by the definition of the reflexive closure, we must have  $S \subseteq R \cup \delta_A$ . Consequently,  $S = R \cup \delta_A$ .

Conversely, suppose that  $S = R \cup \delta_A$ . We have already shown that  $R \cup \delta_A$  is reflexive and  $R \subseteq R \cup \delta_A$ . Let  $T$  be a reflexive relation on  $A$  such that  $R \subseteq T$ . Because  $T$  is reflexive,  $(a, a) \in T$  for all  $a \in A$ , so  $\delta_A \subseteq T$ . Thus,  $S = R \cup \delta_A \subseteq T$ . That is,  $S$  is the smallest reflexive relation on  $A$  such that  $R \subseteq S$ . Hence  $S$ , that is,  $R \cup \delta_A$ , is the reflexive closure of  $R$ .

- (ii) Let us first observe the following: Let  $(a, b) \in R \cup R^{-1}$ . Then  $(a, b) \in R$  or  $(a, b) \in R^{-1}$ . Suppose that  $(a, b) \in R$ . Then, by the definition of the inverse of a relation, we have  $(b, a) \in R^{-1}$ . Therefore, we have  $(a, b), (b, a) \in R \cup R^{-1}$ . Now, suppose that  $(a, b) \in R^{-1}$ . Then, again by the inverse of a relation, we have  $(b, a) \in R$ . Hence,  $(a, b), (b, a) \in R \cup R^{-1}$ . Moreover, we have  $R \subseteq R \cup R^{-1}$ . We have thus proved that  $R \cup R^{-1}$  is a symmetric relation on  $A$  such that  $R \subseteq R \cup R^{-1}$ .

Let  $S$  be the symmetric closure of  $R$ . We show that  $S = R \cup R^{-1}$ .

Because  $S$  is the symmetric closure of  $R$ ,  $R \subseteq S$ . Next, we show that  $R^{-1} \subseteq S$ . Let  $(a, b) \in R^{-1}$ . Then, by the definition of the inverse of a relation,  $(b, a) \in R$ , so  $(b, a) \in S$  as  $R \subseteq S$ . Now, because  $S$  is symmetric and  $(b, a) \in S$ , we must have  $(a, b) \in S$ . Thus,  $R^{-1} \subseteq S$ . It now follows that  $R \cup R^{-1} \subseteq S$ .

On the other hand,  $S$  is the symmetric closure of  $R$ , and  $R \cup R^{-1}$  is a symmetric relation on  $A$  such that  $R \subseteq R \cup R^{-1}$ . Therefore, by the definition of the symmetric closure, we must have  $S \subseteq R \cup R^{-1}$ . Consequently,  $S = R \cup R^{-1}$ .

Conversely, suppose that  $S = R \cup R^{-1}$ . We have already proved that  $R \cup R^{-1}$  is a symmetric relation on  $A$  such that  $R \subseteq R \cup R^{-1}$ . We only need to prove that this is the smallest such relation.

Let  $T$  be a symmetric relation on  $A$  such that  $R \subseteq T$ . Next, we show that  $R^{-1} \subseteq T$ . Let  $(a, b) \in R^{-1}$ , then  $(b, a) \in R \subseteq T$ . Now, because  $T$  is symmetric and  $(b, a) \in T$ , we must have  $(a, b) \in T$ . Thus,  $R^{-1} \subseteq T$ . It now follows that  $S = R \cup R^{-1} \subseteq T$ . Hence,  $S$  is the symmetric closure of  $R$ . ■

The following corollary immediately follows Theorem 3.1.46.

**Corollary 3.1.47:** Let  $R$  be a relation on a nonempty subset  $A$ .

- (i) If  $R$  is reflexive, then the reflexive closure of  $R$  is  $R$ .
- (ii) If  $R$  is symmetric, then the symmetric closure of  $R$  is  $R$ .

### EXAMPLE 3.1.48

Let us consider a relation  $R$  on a set  $A = \{a, b, c\}$ , given by  $R = \{(a, a), (a, b), (b, c)\}$ . Then the reflexive closure of  $R$  is

$$\begin{aligned} R \cup \delta_A &= R \cup \{(a, a), (b, b), (c, c)\} \\ &= \{(a, a), (a, b), (b, c), (b, b), (c, c)\}. \end{aligned}$$

The symmetric closure of  $R$  is

$$\begin{aligned} R \cup R^{-1} &= R \cup \{(a, a), (b, a), (c, b)\} \\ &= \{(a, a), (a, b), (b, c), (b, a), (c, b)\}. \end{aligned}$$

Let us again consider the relation given at the beginning of this section; i.e., let  $A$  be the set

$$A = \{1, 2, 3, 4\}$$

and  $R$  be a relation on  $A$  defined by

$$R = \{(1, 1), (3, 3), (1, 3), (2, 3), (3, 2), (4, 2)\}.$$

Notice that  $(4, 2), (2, 3) \in R$ , but  $(4, 3) \notin R$ . Therefore,  $R$  is not transitive. To the elements of the set  $R$ , we want to add only the elements that are necessary to make the resulting set a transitive relation. For example, consider the set

$$S = R \cup \{(1, 2), (2, 2), (4, 3)\}.$$

Then  $S$  is the smallest transitive relation on  $A$  such that  $R \subseteq S$ . Such a relation  $S$  is called the transitive closure of  $R$ . More formally, we have the following definition.

---

**DEFINITION 3.1.49** ▶ Let  $R$  be a relation on a nonempty subset  $A$ . A relation  $S$  on  $A$  is called the **transitive closure** of  $R$  if

- (i)  $S$  is transitive;
- (ii)  $R \subseteq S$ ; and
- (iii) if  $T$  is a transitive relation on  $A$  such that  $R \subseteq T$ , then  $S \subseteq T$ .

Determining the reflexive and symmetric closure of a relation is relatively easier than determining the transitive closure. Our objective is to establish enough results to lead to an algorithm, described later in this chapter, to determine the transitive closure. We begin with the following theorem.

**Theorem 3.1.50:** Let  $R$  be a relation on a nonempty set  $A$ . Let

$$R^\infty = R \cup R^2 \cup R^3 \cup \dots = \bigcup_{n=1}^{\infty} R^n.$$

Then  $R^\infty$  is the transitive closure of  $R$ .

**Proof:** From the definition of  $R^\infty$ , we have  $R \subseteq R^\infty$ . Next, we show that  $R^\infty$  is transitive. Let  $(a, b), (b, c) \in R^\infty = \bigcup_{n=1}^{\infty} R^n$ . Then there exist positive integers  $m$  and  $n$  such that  $(a, b) \in R^m$  and  $(b, c) \in R^n$ . This implies that

$$(a, c) \in R^n \circ R^m$$

by the definition of the composition of relations. Because  $R^n \circ R^m = R^{n+m} \subseteq R^\infty$ , we have  $(a, c) \in R^\infty$ . Hence,  $R^\infty$  is transitive.

Let  $T$  be a transitive relation on  $A$  such that  $R \subseteq T$ . Next, by induction we prove that  $R^n \subseteq T$  for all  $n \geq 1$ .

*Basis step:* Let  $n = 1$ . Then  $R^1 = R \subseteq T$ . Thus, the result is true for  $n = 1$ .

*Inductive hypothesis:* Let  $k$  be an integer,  $k \geq 1$ , such that  $R^k \subseteq T$ .

*Inductive step:* Consider  $R^{k+1}$ , where  $k$  is an integer such that  $k \geq 1$ . By the definition

$$R^{k+1} = R \circ R^k.$$

We show that  $R^{k+1} \subseteq T$ . Let  $(a, b) \in R^{k+1}$ , i.e.,  $(a, b) \in R \circ R^k$ . By the definition of composition, there exists  $c \in A$  such that  $(a, c) \in R^k$  and  $(c, b) \in R$ . By the inductive hypothesis, we have  $R^k \subseteq T$ . Also, from our assumption,  $R \subseteq T$ . It now follows that  $(a, c), (c, b) \in T$ . Because  $T$  is transitive, we have  $(a, b) \in T$ . Consequently,  $R^{k+1} \subseteq T$ .

Hence, by induction,  $R^n \subseteq T$  for all  $n \geq 1$ .

Because  $R^n \subseteq T$  for all  $n \geq 1$ , we have

$$R^\infty = R \cup R^2 \cup R^3 \cup \dots \cup R^n \cup \dots \subseteq T.$$

We have thus proved the following:

- (i)  $R \subseteq R^\infty$ .
- (ii)  $R^\infty$  is transitive.
- (iii) For any transitive relation  $T$ , if  $R \subseteq T$ , then  $R^\infty \subseteq T$ .

By the definition of the transitive closure, we can conclude that  $R^\infty$  is the transitive closure of  $R$ . ■

Next we introduce the concepts of directed walks and paths in a relation.

**DEFINITION 3.1.51** ▶ Let  $R$  be a relation on a set  $A$ . Let  $a, b \in A$ .

- (i) We say that there is a **directed walk**, or **walk**, from  $a$  to  $b$  in the relation  $R$  if either  $(a, b) \in R$  or there exists  $a_1, a_2, \dots, a_k \in A$  such that

$$(a, a_1), (a_1, a_2), \dots, (a_k, b) \in R,$$

i.e.,

$$a R a_1, a_1 R a_2, \dots, a_k R b.$$

The elements  $a, a_1, a_2, \dots, a_k, b$  are called the **vertices of the walk**. Vertex  $a$  is called the **initial vertex** and vertex  $b$  is called the **terminal vertex**.

Moreover, vertices  $a_1, a_2, \dots, a_k$  are called the **internal vertices** of the walk.

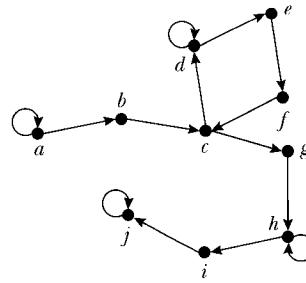
- (ii) A directed walk from a vertex  $a$  to a vertex  $b$  is called a **path** from  $a$  to  $b$  if all the vertices of the walk except possibly  $a$  and  $b$  are distinct.

**REMARK 3.1.52** ▶ Let  $R$  be a relation on a finite set  $A$ . Then we can draw the digraph of the relation. Therefore, when we study a relation  $R$  via the digraph, we talk about the directed walk or path in the digraph.

### EXAMPLE 3.1.53

Let  $A = \{a, b, c, d, e, f, g, h, i, j\}$ . Let  $R$  be a relation on  $A$  such that the digraph of  $R$  is as shown in Figure 3.14.

Then  $a, b, c, d, e, f, c, g$  is a directed walk in  $R$  as  $a R b, b R c, c R d, d R e, e R f, f R c, c R g$ . Similarly,  $a, b, c, e, g$  is also a directed walk in  $R$ . In the walk  $a, b, c, d, e, f, c, g$ , the internal vertices are  $b, c, d, e, f$ , and  $c$ , which are not distinct as  $c$  repeats.



**FIGURE 3.14** Digraph, walk, and paths

Hence, this walk is not a path. In the walk  $a, b, c, g$ , the internal vertices are  $b$  and  $c$ , which are distinct. Therefore, the walk  $a, b, c, g$  is a path.

The next lemma shows that if there is a directed walk from a vertex  $a$  to a vertex  $b$ , then using the directed walk we can construct a path from  $a$  to  $b$  such that the internal vertices of the path are from the set of internal vertices of the directed walk. This, in fact, is the key to constructing an algorithm to determine the transitive closure of a relation of a finite set.

**Lemma 3.1.54:** Let  $R$  be a relation on a set  $A$ . Let  $a, b \in A$ . Suppose that  $a$  and  $b$  are distinct and there is a directed walk with internal vertices  $a_1, a_2, \dots, a_k \in A$  from  $a$  to  $b$  in  $R$ . Then there exists a directed path with internal vertices  $x_1, x_2, \dots, x_q \in \{a_1, a_2, \dots, a_k\}$  from  $a$  to  $b$  in  $R$ .

**Proof:** Suppose that there is a directed walk with internal vertices  $a_1, a_2, \dots, a_k \in A$  from  $a$  to  $b$  in  $R$ . If the vertices  $a_1, a_2, \dots, a_k$  are distinct, then the result is true. Suppose that the vertices  $a_1, a_2, \dots, a_k$  are not all distinct. Then we prove the result by induction on  $k$ .

*Basis step:* If  $k = 1$ , then there is a path with internal vertex  $a_1 \in A$  from  $a$  to  $b$  and the result is true.

*Inductive hypothesis:* Suppose that the result is true for any path with  $t$  internal vertices. That is, for all  $a, b \in A$  if there is a directed walk with internal vertices  $a_1, a_2, \dots, a_t \in A$  from  $a$  to  $b$  in  $R$ , then there exists a directed path with internal vertices  $y_1, y_2, \dots, y_l \in \{a_1, a_2, \dots, a_t\}$  from  $a$  to  $b$  in  $R$ .

*Inductive step:* Suppose that there is a directed walk with internal vertices

$$a_1, a_2, \dots, a_{t+1} \in A$$

from  $a$  to  $b$ . Then

$$(a, a_1), (a_1, a_2), \dots, (a_t, a_{t+1}), (a_{t+1}, b) \in R.$$

Consider the vertices  $a_1, a_2, \dots, a_t$ . Because  $(a, a_1), (a_1, a_2), \dots, (a_t, a_{t+1}) \in R$ , we have a directed walk with  $t$  internal vertices  $a_1, a_2, \dots, a_t$  from  $a$  to  $a_{t+1}$ . By the inductive hypothesis, there exists a directed path with vertices  $x_1, x_2, \dots, x_r \in \{a_1, a_2, \dots, a_t\}$  from  $a$  to  $a_{t+1}$ . Then,

$$(a, x_1), (x_1, x_2), \dots, (x_r, a_{t+1}) \in R.$$

Now

$$(a, x_1), (x_1, x_2), \dots, (x_r, a_{t+1}), (a_{t+1}, b) \in R.$$

This implies that we have a directed walk with internal vertices  $x_1, x_2, \dots, x_r, a_{t+1}$  from  $a$  to  $b$ .

There are two cases: Either  $a_{t+1} \neq x_i$  for all  $i$ ,  $i = 1, 2, \dots, r$  or  $a_{t+1} = x_i$  for some  $i$ .

First, suppose  $a_{t+1} \neq x_i$  for all  $i$ ,  $i = 1, 2, \dots, r$ , then the elements  $x_1, x_2, \dots, x_r, a_{t+1}$  are distinct, so we have a directed path. Therefore, in this case the result is true.

Suppose that  $a_{t+1} = x_i$  for some  $i$ ,  $1 \leq i \leq r$ . Then we claim that  $a_{t+1}$  cannot be equal to any other  $x_j$ .

For suppose there exists  $j$ ,  $1 \leq j \leq r$ ,  $j \neq i$  such that  $a_{t+1} = x_j$ . Then we have  $i \neq j$  and  $x_i = x_j$ , which is a contradiction as  $x_j$ 's are distinct. Hence,  $a_{t+1} \neq x_j$  for all  $j$ ,  $1 \leq j \leq r$ ,  $j \neq i$ .

Now

$$(a, x_1), (x_1, x_2), \dots, (x_{i-1}, x_i), (x_i, x_{i+1}), \dots, (x_r, a_{t+1}), (a_{t+1}, b) \in R,$$

i.e.,

$$(a, x_1), (x_1, x_2), \dots, (x_{i-1}, a_{t+1}), (a_{t+1}, x_{i+1}), \dots, (x_r, a_{t+1}), (a_{t+1}, b) \in R.$$

From this, we can omit the edges  $(a_{t+1}, x_{i+1}), \dots, (x_r, a_{t+1})$  and consider

$$(a, x_1), (x_1, x_2), \dots, (x_{i-1}, a_{t+1}), (a_{t+1}, b) \in R.$$

This implies that we can take the directed walk with internal vertices  $x_1, x_2, \dots, x_{i-1}, a_{t+1}$  from  $a$  to  $b$ . Now the elements  $x_1, x_2, \dots, x_{i-1}, a_{t+1}$  are distinct, so this is a directed path. Moreover,

$$x_1, x_2, \dots, x_{i-1}, a_{t+1} \in \{a_1, a_2, \dots, a_t, a_{t+1}\}.$$

We have thus proved the result for  $t + 1$ .

The result now follows by induction. ■

### EXAMPLE 3.1.55

Let  $A$  and  $R$  be as in Example 3.1.53. Consider  $a$  and  $h$ . Then  $a$  and  $h$  are distinct. Now  $a, b, c, d, e, f, g, h$  is a directed walk in  $R$ . In this walk, the internal vertices  $b, c, d, e, f, g$  are not distinct. Consider the walk  $a, b, c, g, h$ . Now  $b, c, g \in \{b, c, d, e, f, g\}$  and  $a, b, c, g, h$  is a path.

**Lemma 3.1.56:** Let  $A = \{a_1, a_2, \dots, a_n\}$  be a finite set,  $n \geq 1$ . Let  $R$  be a relation on  $A$ . Let  $a, b \in A$ . Suppose that  $(a, b) \in R^m$ ,  $m \geq 1$ . Then there exists  $k$ ,  $1 \leq k \leq n$ , such that  $(a, b) \in R^k$ .

**Proof:** Suppose that  $(a, b) \in R^k$ . If  $k \leq n$ , then the result is true. Suppose that  $k > n$ .

Because  $(a, b) \in R^k$ , there exist  $a'_1, a'_2, \dots, a'_{k-1} \in A$  such that

$$(a, a'_1), (a'_1, a'_2), \dots, (a'_{k-1}, b) \in R.$$

Thus, we have a directed walk with internal vertices  $a'_1, a'_2, \dots, a'_{k-1}$  from  $a$  to  $b$

in  $R$ . By Lemma 3.1.54, there exists a path with internal vertices  $x_1, x_2, \dots, x_t \in \{a'_1, a'_2, \dots, a'_{k-1}\} \subseteq A$  from  $a$  to  $b$  in  $R$ . Because the vertices  $x_1, x_2, \dots, x_t \in A$  are distinct and  $A$  has  $n$  elements, we must have  $t \leq n$ .

Consider the path  $a, x_1, x_2, \dots, x_t, b$ . If  $a \neq x_i$  for all  $i$  or  $b \neq x_i$  for all  $i$ , then we must have  $t \leq n - 1$ .

Suppose  $a = x_i$  for some  $i$ . Then we can take the path  $a, x_{i+1}, \dots, x_t, b$ , so the internal vertices are  $\leq n - 1$ . Similarly, if  $b = x_i$  for some  $i$ , then we can take the path  $a, x_1, x_2, \dots, x_{i-1}, b$ , so we have a path with internal vertices  $\leq n - 1$ .

Thus, there exists  $x_1, x_2, \dots, x_t \in A$ , such that  $(a, x_1), (x_1, x_2), \dots, (x_t, b) \in R$ , and  $1 \leq t \leq n - 1$ . This implies that  $(a, b) \in R^{t+1}$ . Let  $k = t + 1$ . Then  $1 \leq k \leq n$  and  $(a, b) \in R^k$ . This completes the proof. ■

The following tells us how to find the transitive closure of a relation on a finite set.

**Theorem 3.1.57:** Let  $A = \{a_1, a_2, \dots, a_n\}$  be a finite set,  $n \geq 1$ . Let  $R$  be a relation on  $A$ . The transitive closure  $R^\infty$  of  $R$  is given by

$$R^\infty = R \cup R^2 \cup \dots \cup R^n.$$

**Proof:** By the definition of  $R^\infty$ , we have  $R \cup R^2 \cup \dots \cup R^n \subseteq R^\infty$ . We only need to show that  $R^\infty \subseteq R \cup R^2 \cup \dots \cup R^n$ . The result then will follow from the equality of sets.

Let  $(a, b) \in R^\infty$ . Then  $(a, b) \in R^k$  for some  $k \geq 1$ . By Lemma 3.1.56, there exists  $t$ ,  $1 \leq t \leq n$ , such that  $(a, b) \in R^t$ . Because  $R^t \subseteq R \cup R^2 \cup \dots \cup R^n$ , we must have  $(a, b) \in R \cup R^2 \cup \dots \cup R^n$ . It now follows that  $R^\infty \subseteq R \cup R^2 \cup \dots \cup R^n$ .

Consequently,  $R^\infty = R \cup R^2 \cup \dots \cup R^n$ . ■

The following example uses Theorem 3.1.57 to find the transitive closure of a relation on a finite set.

### EXAMPLE 3.1.58

Let  $A$  be the set

$$A = \{1, 2, 3, 4\}$$

and  $R$  be a relation on  $A$  defined by

$$R = \{(1, 1), (3, 3), (1, 3), (2, 3), (3, 2), (4, 2)\}.$$

Then, as noted previously,  $R$  is not transitive. For example,  $(4, 2), (2, 3) \in R$ , but  $(4, 3) \notin R$ . Let us determine the transitive closure of  $R$ . By Theorem 3.1.57,

$$R^\infty = R \cup R^2 \cup R^3 \cup R^4.$$

Now

$$R^2 = \{(1, 1), (1, 2), (1, 3), (2, 2), (2, 3), (3, 2), (3, 3), (4, 3)\},$$

$$R^3 = \{(1, 1), (1, 2), (1, 3), (2, 2), (2, 3), (3, 2), (3, 3), (4, 2), (4, 3)\},$$

and

$$R^4 = \{(1, 1), (1, 2), (1, 3), (2, 2), (2, 3), (3, 2), (3, 3), (4, 2), (4, 3)\}.$$

Hence,

$$R^\infty = \{(1, 1), (1, 2), (1, 3), (2, 2), (2, 3), (3, 2), (3, 3), (4, 2), (4, 3)\}.$$

In Chapter 4, Matrices and Closures of Relations, we will describe an algorithmic way to determine the transitive closure of a relation on a finite set.

## WORKED-OUT EXERCISES

**Exercise 1:** Find the domain and range of the relation  $R$ , where  $R$  is as defined in (a) and (b).

- (a)  $A = \{1, 2, 3, 4, 5\}$ ,  $B = \{1, 2, 3, 10\}$ ,  $a R b$  if and only if  $2a = b$
- (b)  $A = \{1, 2, 3, 4\} = B$ ,  $a R b$  if and only if  $a + b = 5$

**Solution:**

- (a)  $R = \{(1, 2), (5, 10)\}$ . Hence, the domain of  $R$ ,  $D(R) = \{1, 5\}$  and the range of  $R$ ,  $\text{Im}(R) = \{2, 10\}$ .
- (b)  $R = \{(1, 4), (2, 3)\}$ . Hence, the domain of  $R$ ,  $D(R) = \{1, 2\}$  and the range of  $R$ ,  $\text{Im}(R) = \{3, 4\}$ .

**Exercise 2:** Find three distinct binary relations from  $A = \{a, b, c\}$  into  $B = \{0, 2\}$ .

**Solution:**  $\alpha = \{(a, 2), (b, 0)\}$ ,  $\beta = \{(a, 2), (b, 2), (a, 0)\}$ ,  $\delta = \{(c, 0), (c, 2)\}$  are three distinct binary relations. (Note that other examples of distinct binary relations from  $A$  into  $B$  also exist.)

**Exercise 3:** Find the number of relations on a set of  $n$  elements.

**Solution:** Let  $A$  be a set of  $n$  elements. Then the number of elements of  $A \times A$  is  $n^2$ , so the number of subsets of  $A \times A$  is  $2^{n^2}$ . Because any subset of  $A \times A$  is a relation on  $A$  and any relation on  $A$  is a subset of  $A \times A$ , it follows that the number of relations on  $A$  is  $2^{n^2}$ .

**Exercise 4:** Let  $A = \{1, 2, 3, 4, 5\}$  and let a relation on  $A$  be defined by

$$R = \{(1, 1), (1, 2), (2, 3), (3, 4), (3, 5), (4, 5)\}.$$

Compute  $R^2$  and  $R^3$ .

**Solution:** Now  $(a, b) \in R^2$  if and only if there exists  $c \in A$  such that  $(a, c) \in R$  and  $(c, b) \in R$ . We have

- $(1, 1) \in R$  and  $(1, 1) \in R \Rightarrow (1, 1) \in R^2$ ,
- $(1, 1) \in R$  and  $(1, 2) \in R \Rightarrow (1, 2) \in R^2$ ,
- $(1, 2) \in R$  and  $(2, 3) \in R \Rightarrow (1, 3) \in R^2$ ,
- $(2, 3) \in R$  and  $(3, 4) \in R \Rightarrow (2, 4) \in R^2$ ,
- $(2, 3) \in R$  and  $(3, 5) \in R \Rightarrow (2, 5) \in R^2$ ,
- $(3, 4) \in R$  and  $(4, 5) \in R \Rightarrow (3, 5) \in R^2$ .

Hence,

$$R^2 = \{(1, 1), (1, 2), (1, 3), (2, 4), (2, 5), (3, 5)\}.$$

Again,  $(a, b) \in R^3$  if and only if there exists  $c \in A$  such that  $(a, c) \in R^2$  and  $(c, b) \in R$ . We have

- $(1, 1) \in R^2$  and  $(1, 1) \in R \Rightarrow (1, 1) \in R^3$ ,
- $(1, 1) \in R^2$  and  $(1, 2) \in R \Rightarrow (1, 2) \in R^3$ ,
- $(1, 2) \in R^2$  and  $(2, 3) \in R \Rightarrow (1, 3) \in R^3$ ,
- $(2, 4) \in R^2$  and  $(4, 5) \in R \Rightarrow (2, 5) \in R^3$ ,
- $(1, 3) \in R^2$  and  $(3, 5) \in R \Rightarrow (1, 5) \in R^3$ ,
- $(1, 3) \in R^2$  and  $(3, 4) \in R \Rightarrow (1, 4) \in R^3$ .

Hence,

$$R^3 = \{(1, 1), (1, 2), (1, 3), (1, 4), (2, 5), (1, 5)\}.$$

**Exercise 5:** For each of the following relations on  $A = \{1, 2, 3, 4\}$ , determine whether it is reflexive, symmetric, or transitive.

- (a)  $R = \{(1, 4), (4, 1)\}$
- (b)  $R = \{(1, 1)\}$
- (c)  $R = \{(1, 1), (2, 2), (3, 3), (4, 4), (2, 3), (3, 2)\}$
- (d)  $R = \{(1, 3), (1, 4)\}$ .

**Solution:**

- (a) Because  $(1, 1) \notin R$ ,  $R$  is not reflexive. In  $R$ ,  $(a, b) \in R \Rightarrow (b, a) \in R$ . Hence,  $R$  is symmetric. Now  $(4, 1) \in R$  and  $(1, 4) \in R$ , but  $(4, 4) \notin R$ . Hence,  $R$  is not transitive.
- (b) Because  $(2, 2) \notin R$ ,  $R$  is not reflexive. In  $R$ ,  $(a, b) \in R \Rightarrow (b, a) \in R$ . Hence,  $R$  is symmetric. Similarly,  $R$  is transitive.
- (c) Because  $(a, a) \in R$  for all  $a \in A$ , it follows that  $R$  is reflexive. Similarly,  $R$  is symmetric and transitive.
- (d) Because  $(1, 1) \notin R$ ,  $R$  is not reflexive. Also  $(1, 3) \in R$  and  $(3, 1) \notin R$ . Hence,  $R$  is not symmetric. Now  $R$  is transitive because there are no two pairs of the form  $(a, b), (b, c)$  in  $R$ . If there are pairs  $(a, b), (b, c)$  in  $R$ , then for transitivity we have to check whether  $(a, c) \in R$ .

**Exercise 6:** Which of the relations in Worked-Out Exercise 5 are equivalence relations? Describe the partition of  $A$  corresponding to these equivalence relations.

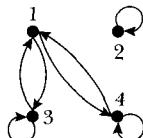
**Solution:** Relation (c) of Worked-Out Exercise 5 is an equivalence relation. The partition of  $\{1, 2, 3, 4\}$  corresponding to this equivalence relation is  $\{\{1\}, \{2, 3\}, \{4\}\}$ .

**Exercise 7:** Draw a directed graph representation of the following relations on the set  $\{1, 2, 3, 4\}$ . Decide whether the relation is reflexive, symmetric, or transitive.

- (a)  $R_1 = \{(1, 3), (1, 4), (2, 2), (3, 1), (3, 3), (4, 1), (4, 4)\}$
- (b)  $R_2 = \{(1, 1), (2, 2), (3, 3), (4, 4)\}$
- (c)  $R_3 = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 2), (2, 3), (1, 3), (3, 2)\}$

**Solution:**

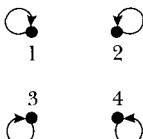
- (a) We draw a directed graph with four vertices 1, 2, 3, 4. We draw an arc from a vertex  $x$  to a vertex  $y$  if and only if  $(x, y) \in R_1$ . Then we obtain the directed graph of the given relation, as shown in Figure 3.15.



**FIGURE 3.15**  
Digraph of  $R_1$

Now:

- (1) The relation  $R_1$  is not reflexive, because there is no loop at the vertex 1.
  - (2) In the directed graph, if there is a directed edge from one vertex  $x$  to another vertex  $y$ , then there exists a directed edge from vertex  $y$  to vertex  $x$ . For example, there is a directed edge from 1 to 3 and there is also a directed edge from 3 to 1; there is a directed edge from 1 to 4 and there is also a directed edge from 4 to 1. Hence, the given relation is a symmetric relation.
  - (3) The relation is not transitive because there is a directed edge from vertex 1 to vertex 3 and there is a directed edge from vertex 3 to vertex 1, but there is no directed edge from vertex 1 to vertex 1.
- (b) We draw a directed graph with four vertices 1, 2, 3, 4. We draw an arc from a vertex  $x$  to a vertex  $y$  if and only if  $(x, y) \in R_2$ . Then we obtain the digraph of the given relation, as shown in Figure 3.16.



**FIGURE 3.16**  
Digraph of  $R_2$

Now:

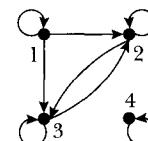
- (1) The relation  $R_2$  is reflexive, because there is a loop on each of the vertices.
- (2) We know that the relation  $R_2$  is a symmetric relation if and only if in its digraph if there is a directed edge from one vertex  $x$  to another vertex  $y$ , then there must exist a directed edge from vertex  $y$  to vertex  $x$ . Now in this digraph of the given relation  $R_2$ , we find that there are no directed edges between two different vertices. Therefore, we need not check the above condition for the symmetric property of  $R_2$ . Hence, the given relation is a symmetric relation.
- (3) It is known that a relation  $R_2$  on a finite set is transitive if and only if in the digraph of  $R_2$  if there is a directed edge from vertex  $a$  to vertex  $b$  and there is

a directed edge from vertex  $b$  to vertex  $c$ , then there must exist a directed edge from vertex  $a$  to vertex  $c$ . Now in the digraph of the given relation  $R_2$ , we find that there are no directed edges between two different vertices. Thus, we need not check the above condition for transitivity of  $R_2$ . Hence, the given relation is transitive.

- (c) We draw a directed graph with four vertices 1, 2, 3, 4. We draw an arc from a vertex  $x$  to a vertex  $y$  if and only if  $(x, y) \in R_3$ , where

$$R_3 = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 2), (2, 3), (1, 3), (3, 2)\}.$$

We obtain the digraph representation of  $R_3$ , as shown in Figure 3.17.



**FIGURE 3.17**  
Digraph of  $R_3$

Now:

- (1) The relation  $R_3$  is reflexive, because there is a loop on each of the vertices.
- (2) In the directed graph if there is a directed edge from vertex 1 to vertex 2, but there is no directed edge from vertex 2 to vertex 1. Hence,  $R_3$  is not symmetric.
- (3) We find that in the digraph of  $R_3$  if there is a directed edge from one vertex  $a$  to another vertex  $b$  and if there is a directed edge from vertex  $b$  to vertex  $c$ , then there exists a directed edge from vertex  $a$  to vertex  $c$ . For example, there is a directed edge from 1 to 3 and there is also a directed edge from 3 to 2, and we see that there is a directed edge from 1 to 2. Also, there is a directed edge from 1 to 2 and there is a directed edge from 2 to 3, and we see that there is a directed edge from 1 to 3. Hence,  $R_3$  is a transitive relation.

**Exercise 8:** In each of the following cases, determine whether the relation  $R$  is an equivalence relation on the set  $\mathbb{Z}$  of all integers. If it is an equivalence relation, then find the corresponding partition on the set.

- (a)  $R = \{(a, b) \in \mathbb{Z} \times \mathbb{Z} \mid |a - b| \leq 4\}$
- (b)  $R = \{(a, b) \in \mathbb{Z} \times \mathbb{Z} \mid a - b \text{ is a multiple of } 7\}$

**Solution:**

- (a) In this case, the relation  $R$  is not transitive. Indeed, we see that  $6 R 5$  and  $5 R 1$  hold, which we can easily check from the given definition of  $R$ , but  $6 R 1$  as  $|6 - 1| = 5 \not\leq 4$ . Consequently,  $R$  is not a transitive relation and hence not an equivalence relation.
- (b) Let us consider any integer  $a$ . Then  $a - a = 0 = 0 \cdot 7$ , so  $a R a$ . This implies that  $R$  is reflexive. Now let

$a, b \in \mathbb{Z}$  and  $a R b$ . Then

$$\begin{aligned} a - b &= 7n \quad \text{for some } n \in \mathbb{Z} \\ \Rightarrow b - a &= 7(-n) \\ \Rightarrow b &R a. \end{aligned}$$

This shows that  $R$  is symmetric.

For transitivity, let  $a, b, c \in \mathbb{Z}$  such that  $a R b$  and  $b R c$ . Then

$$\begin{aligned} a - b &= 7n_1 \quad \text{and} \quad b - c = 7n_2 \quad \text{for some } n_1, n_2 \in \mathbb{Z}. \\ \Rightarrow a - b + b - c &= 7(n_1 + n_2) \\ \Rightarrow a - c &= 7(n_1 + n_2), n_1 + n_2 \in \mathbb{Z} \\ \Rightarrow a &R c. \end{aligned}$$

Hence,  $R$  is transitive. Consequently,  $R$  is an equivalence relation on  $S$ .

To determine the partition of  $\mathbb{Z}$  corresponding to  $R$ , we need to determine the equivalence classes with respect to  $R$ . Now, let  $m \in \mathbb{Z}$ . Then  $x \in [m]$  if and only if  $x = 7q + m$  for some  $q \in \mathbb{Z}$ . Consequently, for all  $m = 0, 1, 2, \dots, 6$ ,  $[m] = [7q + m]$  for all  $q \in \mathbb{Z}$ . Hence, there are seven equivalence classes, which are

$$\begin{aligned} [0] &= [7] = [14] = \dots = [-7] = [-14] = \dots \\ [1] &= [8] = [15] = \dots = [-6] = [-13] = \dots \\ [2] &= [9] = [16] = \dots = [-5] = [-12] = \dots \\ [3] &= [10] = [17] = \dots = [-4] = [-11] = \dots \\ [4] &= [11] = [18] = \dots = [-3] = [-10] = \dots \\ [5] &= [12] = [19] = \dots = [-2] = [-9] = \dots \\ [6] &= [13] = [20] = \dots = [-1] = [-8] = \dots \end{aligned}$$

**Exercise 9:** On the set  $\mathbb{Z}$  of all integers, define the relation  $R$  by

$$R = \{(a, b) \in \mathbb{Z} \times \mathbb{Z} \mid a^2 - b^2 \text{ is divisible by } 5\}.$$

Show that  $R$  is an equivalence relation. Find the equivalence classes of this equivalence relation on  $\mathbb{Z}$ .

**Solution:** Let  $a \in \mathbb{Z}$ . Then  $a^2 - a^2 = 0$  is divisible by 5. Therefore,  $a R a$  for all  $a \in R$ . This implies that  $R$  is reflexive. Let  $a, b \in \mathbb{Z}$  such that  $a R b$ . Then  $5 \mid (a^2 - b^2)$ , so  $5 = n(a^2 - b^2)$  for some integer  $n$ . This implies that  $5 = (-n)(b^2 - a^2)$  and  $-n \in \mathbb{Z}$ . We therefore have  $b R a$ . Thus,  $R$  is symmetric.

For transitivity, let  $a, b, c \in \mathbb{Z}$  and  $a R b$  and  $b R c$ . Then  $5 \mid (a^2 - b^2)$  and  $5 \mid (b^2 - c^2)$ . Thus,  $5 \mid (a^2 - b^2) + (b^2 - c^2)$  (by Theorem 2.1.14(iii)), i.e.,  $5 \mid a^2 - c^2$ . Hence,  $a R c$ . Consequently,  $R$  is an equivalence relation.

Next, we determine the equivalence classes. Let  $a \in \mathbb{Z}$ . Then, by the division algorithm,  $a = 5k + r$  for some  $k, r \in \mathbb{Z}$ , where  $0 \leq r < 5$ . This implies that  $a^2 = 25k^2 + 10kr + r^2$ .

If  $r = 0$ , then  $r^2 = 0$ ; if  $r = 1$ , then  $r^2 = 1^2$ ; if  $r = 2$ , then  $r^2 = 2^2$ ; if  $r = 3$ , then  $r^2 = 9 = 5 \cdot 1 + 4 = 5 \cdot 1 + 2^2$ ; and if  $r = 4$ , then  $r^2 = 16 = 5 \cdot 3 + 1 = 5 \cdot 3 + 1^2$ .

From this we can conclude that for any  $a \in \mathbb{Z}$ , either  $(a, 0) \in R$ , or  $(a, 1) \in R$ , or  $(a, 2) \in R$ . Hence, there are only three equivalence classes, which are  $[0], [1]$ , and  $[2]$ .

**Exercise 10:** The following relations are defined on the set of real numbers  $\mathbb{R}$ . Determine whether these relations are reflexive, symmetric, or transitive.

- (a)  $a R b$  if and only if  $|a - b| > 0$
- (b)  $a R b$  if and only if  $1 + ab > 0$
- (c)  $a R b$  if and only if  $|a| \leq b$

**Solution:**

- (a)  $R$  is not reflexive, because for any  $a \in \mathbb{R}$ ,  $a - a = 0$ , so  $|a - a| \neq 0$ , i.e.,  $a \not R a$ .

For any  $a, b \in \mathbb{R}$ , we have  $|a - b| = |b - a|$ , so we have that if  $|a - b| > 0$ , then  $|b - a| > 0$ . In other words,  $a R b$  implies  $b R a$ . Therefore,  $R$  is symmetric.  $R$  is not transitive. Indeed, consider  $1, 0 \in \mathbb{R}$ . Then  $|1 - 0| = |0 - 1| = 1 > 0$  shows that  $1 R 0$  and  $0 R 1$ , but  $1 \not R 1$  as  $|1 - 1| = 0 \neq 0$ .

- (b) Because for all  $a \in \mathbb{R}$ ,  $a^2 > 0$  we have  $1 + a^2 > 0$ , so  $a R a$ , for all  $a \in \mathbb{R}$ , so  $R$  is reflexive. Again, for all  $a, b \in \mathbb{R}$ , if  $1 + ab > 0$ , then  $1 + ba > 0$  as  $ab = ba$ . This implies that if  $a R b$ , then  $b R a$ . So  $R$  is symmetric. However,  $R$  is not transitive. In fact, let us consider,  $3, -\frac{1}{9}, -6 \in \mathbb{R}$ . Then,

$$1 + 3 \left( -\frac{1}{9} \right) = 1 - \frac{1}{3} = \frac{2}{3} > 0$$

shows that  $3 R (-\frac{1}{9})$  and

$$1 + \left( -\frac{1}{9} \right) (-6) = 1 + \frac{2}{3} = \frac{5}{3} > 0$$

shows that  $(-\frac{1}{9}) R (-6)$ . But as

$$1 + 3(-6) = 1 - 18 = -17 \neq 0,$$

we can conclude that  $3 \not R (-6)$ . Hence,  $R$  is not a transitive relation.

- (c) Let us consider  $-2 \in \mathbb{R}$ . Then  $|-2| = 2 \not\leq -2$ . Therefore,  $(-2) R (-2)$ , showing that  $R$  is not reflexive.  $R$  is not symmetric, either. Indeed,  $|-2| = 2 \leq 5$ , so  $-2 R 5$ . But  $|5| = 5 \not\leq -2$ . Therefore,  $5 \not R (-2)$ , so  $R$  is not symmetric. Now, let  $p, q, r \in \mathbb{R}$  such that  $p R q$  and  $q R r$ . Then  $|p| \leq q$  and  $|q| \leq r$ . Now because  $q \geq |p| \geq 0$ , we have  $|q| = q$ . Therefore,  $|p| \leq q \leq r$  gives  $p R r$ , so  $R$  is transitive.

**Exercise 11:** Let  $S$  be a finite set with  $n$  elements. Prove that the number of reflexive relations that can be defined on  $S$  is  $2^{(n^2-n)}$ .

**Solution:** Here  $|S| = n$ , so  $|S \times S| = n^2$ . We know that any relation  $R$  on  $S$  is a subset of  $S \times S$ . Now if  $R$  is a reflexive relation on  $S$ , then it must contain all the elements  $(a, a)$  for all  $a \in S$ . Let

$$\delta_S = \{(a, a) \mid a \in S\}.$$

We have  $|S \times S - \delta_S| = n^2 - n$ , so the number of subsets of  $S \times S - \delta_S$  is  $2^{(n^2-n)}$ . Hence, the number of relations on  $S$  that do not contain any element of  $\delta_S$  is  $2^{(n^2-n)}$ . Now, a relation on  $S$  is reflexive if and only if it is of the form  $R \cup \delta_S$ , where  $R$  is any  $2^{(n^2-n)}$  subsets of  $S \times S - \delta_S$ . Hence, the number of reflexive relations on  $S$  is  $2^{(n^2-n)}$ .

**Exercise 12:** Find all the equivalence relations on the set  $A = \{1, 2, 3\}$ .

**Solution:** If  $R$  is an equivalence relation on the set  $A$ , then we know that the equivalence classes with respect to  $R$  will give a partition on  $A$ . Conversely, for any partition  $\mathcal{P}$  of the set  $A$  there exists an equivalence relation, say  $R$ , on  $A$  such that the  $R$ -classes are precisely the members of  $\mathcal{P}$ . Consequently, the number of equivalence relations on a finite set is equal to the number of different partitions of it. Now there are exactly five partitions of  $A$  and these are

$$\begin{aligned} &\{\{1\}, \{2\}, \{3\}\}, \\ &\{\{1\}, \{2, 3\}\}, \\ &\{\{2\}, \{1, 3\}\}, \\ &\{\{3\}, \{1, 2\}\}, \end{aligned}$$

and

$$\{\{1, 2, 3\}\}.$$

So it follows that there are five equivalence relations on  $A$ , which are

$$\begin{aligned} R_1 &= \{(1, 1), (2, 2), (3, 3)\}, \\ R_2 &= \{(1, 1), (2, 2), (3, 3), (2, 3), (3, 2)\}, \\ R_3 &= \{(1, 1), (2, 2), (3, 3), (1, 3), (3, 1)\}, \\ R_4 &= \{(1, 1), (2, 2), (3, 3), (1, 2), (2, 1)\}, \end{aligned}$$

and

$$R_5 = A \times A.$$

**Exercise 13:** Justify the following statements or give counterexamples to disprove them.

- (a) The intersection of two equivalence relations is again an equivalence relation.
- (b) The union of two equivalence relations is again an equivalence relation.
- (c) If  $R_1$  and  $R_2$  are two symmetric relations on a set, then so is  $R_1 \circ R_2$ .
- (d) If  $R$  is a reflexive and transitive relation, then  $R \circ R$  is transitive.

### Solution:

- (a) True: Let  $R$  and  $L$  be two equivalence relations on the set  $A$ . Because both  $R$  and  $L$  are reflexive, we have for all  $a \in A$ ,  $(a, a) \in R$  and  $(a, a) \in L$ . Thus, for all  $a \in A$ ,  $(a, a) \in R \cap L$  and, consequently,  $R \cap L$  is reflexive.

Now, let  $a, b \in A$  such that  $(a, b) \in R \cap L$ . Then  $(a, b) \in R$  and  $(a, b) \in L$ . From this it follows that  $(b, a) \in R$  and  $(b, a) \in L$  (as both  $R$  and  $L$  are symmetric), so  $(b, a) \in R \cap L$ . Hence,  $R \cap L$  is symmetric.

Again, let  $a, b, c \in A$ ,  $(a, b) \in R \cap L$ , and  $(b, c) \in R \cap L$ . Then  $(a, b), (b, c) \in R$  and  $(a, b), (b, c) \in L$ . By the transitivity of  $R$  and  $L$ , we have  $(a, c) \in R$  as well as  $(a, c) \in L$ , so  $(a, c) \in R \cap L$ . Thus,  $R \cap L$  is transitive. Consequently,  $R \cap L$  is an equivalence relation on  $A$ .

- (b) False: Let  $A = \{1, 2, 3\}$ . We consider two equivalence relations on  $A$  given by

$$R = \{(1, 1), (2, 2), (3, 3), (2, 3), (3, 2)\}$$

and

$$L = \{(1, 1), (2, 2), (3, 3), (3, 1), (1, 3)\}.$$

Here

$$R \cup L = \{(1, 1), (2, 2), (3, 3), (2, 3), (3, 2), (3, 1), (1, 3)\},$$

which is not an equivalence relation as it lacks transitivity. Indeed,  $(2, 3), (3, 1) \in R \cup L$  but  $(2, 1) \notin R \cup L$ .

- (c) False: Let  $A = \{1, 2, 3\}$  and let  $R_1 = \{(2, 3), (3, 2)\}$  and  $R_2 = \{(1, 2), (2, 1)\}$ . Then  $R_1$  and  $R_2$  are symmetric relations on  $A$ . Now,  $R_1 \circ R_2 = \{(1, 3)\}$ , which is not symmetric.

- (d) True: Let  $R$  be a reflexive and transitive relation on  $A$ . Suppose for some  $a, b, c \in A$ ,  $(a, b), (b, c) \in R \circ R$ . Then,  $(a, x), (x, b), (b, y), (y, c) \in R$  for some  $x, y \in A$ . Now, by the transitivity of  $R$ ,  $(a, b) \in R$  and  $(b, c) \in R$ . Again by transitivity,  $(a, b) \in R$  and  $(b, c) \in R$  implies  $(a, c) \in R$ . By reflexivity, we have  $(a, a) \in R$ . Now,  $(a, a), (a, c) \in R$ , so by the definition of the composition of relations,  $(a, c) \in R \circ R$ . Hence,  $R \circ R$  is transitive.

## SECTION REVIEW

### Key Terms

binary relation	directed graph representation	digraph
relation	vertex	adjacent to
$R$ -related	directed edge	adjacent from
related	directed arc	loop
empty relation	arrow diagram	domain
universal relation	directed graph	range

image	equivalence class	transitive closure
inverse	$R$ -class	directed walk
composition	$R$ -equivalence class	walk
reflexive	partition	vertices of the walk
symmetric	block	initial vertex
transitive	equivalence relation induced by the partition	terminal vertex
equivalence relation	reflexive closure	internal vertices
equality relation	symmetric closure	path
congruence modulo $m$		

## Some Key Definitions

1. A binary relation, or simply a relation,  $R$  from a set  $A$  into a set  $B$  is a subset of  $A \times B$ .
2. If  $R$  is a relation from a set  $A$  into itself, then we simply say that  $R$  is a relation on  $A$ .
3. Let  $R$  be a relation from a set  $A$  into a set  $B$ . Then the domain of  $R$ , denoted by  $\mathcal{D}(R)$ , is the set

$$\mathcal{D}(R) = \{a \mid a \in A \text{ and there exists } b \in B \text{ such that } (a, b) \in R\}.$$

The range, or image, of  $R$ , denoted by  $\mathcal{I}(R)$ , or  $\text{Im}(R)$  is the set

$$\text{Im}(R) = \{b \mid b \in B \text{ and there exists } a \in A \text{ such that } (a, b) \in R\}.$$

4. Let  $R$  be a relation from a set  $A$  into a set  $B$ . The inverse of  $R$ , denoted by  $R^{-1}$ , is the relation from  $B$  into  $A$ , which consists of those ordered pairs that, when reversed, belong to  $R$ , i.e.,  $R^{-1} = \{(b, a) \mid (a, b) \in R\}$ .
5. Let  $R$  be a relation from a set  $A$  into a set  $B$  and  $S$  be a relation from  $B$  into a set  $C$ . The composition of  $R$  and  $S$ , denoted by  $S \circ R$ , is the relation from  $A$  into  $C$ , defined by  $a(S \circ R)c$  if there exists some  $b \in B$  such that  $aRb$  and  $bSc$  for all  $a \in A, c \in C$ .
6. Let  $A$  be a set and  $R$  be a relation on  $A$ . Then  $R$  is called
  - (i) reflexive, if for all  $a \in A$ ,  $aR a$ ;
  - (ii) symmetric, if for all  $a, b \in A$ , whenever  $aRb$  holds,  $bRa$  must also hold;
  - (iii) transitive, if for all  $a, b, c \in A$ , whenever  $aRb$  and  $bRc$  hold,  $aRc$  must also hold.
7. A relation  $R$  on a set  $A$  is called an equivalence relation if  $R$  is reflexive, symmetric, and transitive.
8. Let  $R$  be an equivalence relation on a set  $X$ . For all  $x \in X$ , let  $[x]$  denote the set  $[x] = \{y \in X \mid yR x\}$ . The subset  $[x]$  of  $X$  is called the equivalence class ( $R$ -class or  $R$ -equivalence class) of the equivalence relation  $R$  determined by  $x$ .
9. Let  $S$  be a nonempty set and let  $\mathcal{P}$  be a collection of nonempty subsets of  $S$ . Then  $\mathcal{P}$  is called a partition of  $S$ , if the following properties hold:
  - (i) For all  $A_i, A_j \in \mathcal{P}$ , either  $A_i = A_j$  or  $A_i \cap A_j = \emptyset$ ;
  - (ii)  $S = \bigcup_{A_i \in \mathcal{P}} A_i$ .

## Some Key Results

1. Let  $R$  be a relation on a set  $A$ . Then  $R$  is an equivalence relation on  $A$  if and only if
  - (i)  $\delta_A \subseteq R$ , where  $\delta_A = \{(a, a) \mid a \in A\}$ ;
  - (ii)  $R = R^{-1}$ ; and
  - (iii)  $R \circ R \subseteq R$ .
2. Let  $R$  be an equivalence relation on the set  $A$ . Then,
  - (i) for all  $a \in A$ ,  $[a] \neq \emptyset$ ;
  - (ii) if  $b \in [a]$ , then  $[a] = [b]$ , where  $a, b \in A$ ;
  - (iii) for all  $a, b \in A$ , either  $[a] = [b]$  or  $[a] \cap [b] = \emptyset$ ;
  - (iv)  $A$  is the union of all equivalence classes with respect to  $R$ , i.e.,  $A = \bigcup_{a \in A} [a]$ .
3. Let  $R$  be a relation on a nonempty set  $A$ .
  - (i) A relation  $S$  on  $A$  is the reflexive closure of  $R$  if and only if  $S = R \cup \delta_A$ , where  $\delta_A = \{(a, a) \mid a \in A\}$ .
  - (ii) A relation  $S$  on  $A$  is the symmetric closure of  $R$  if and only if  $S = R \cup R^{-1}$ .
4. Let  $R$  be a relation on a nonempty subset  $A$ .
  - (i) If  $R$  is reflexive, then the reflexive closure of  $R$  is  $R$ .
  - (ii) If  $R$  is symmetric, the symmetric closure of  $R$  is  $R$ .
5. Let  $R$  be a relation on a nonempty set  $A$ . Let  $R^\infty = R \cup R^2 \cup R^3 \cup \dots = \bigcup_{n=1}^{\infty} R^n$ . Then  $R^\infty$  is the transitive closure of  $R$ .
6. Let  $A = \{a_1, a_2, \dots, a_n\}$  be a finite set,  $n \geq 1$ . Let  $R$  be a relation on  $A$ . Let  $a, b \in A$ . Suppose that  $(a, b) \in R^m$ ,  $m \geq 1$ . Then there exists  $k$ ,  $1 \leq k \leq n$  such that  $(a, b) \in R^k$ .
7. Let  $A = \{a_1, a_2, \dots, a_n\}$  be a finite set,  $n \geq 1$ . Let  $R$  be a relation on  $A$ . The transitive closure  $R^\infty$  of  $R$  is given by  $R^\infty = R \cup R^2 \cup \dots \cup R^n$ .

## EXERCISES

---

1. Write the following relations as sets of ordered pairs.
  - a.  $R_1$  is from  $A = \{2, 5, 7, 8\}$  into  $B = \{1, 3, 4, 10, 16\}$  defined by  $a R_1 b$  if and only if  $a$  divides  $b$ .
  - b.  $R_2$  is from  $A = \{2, 6, 8, 10\}$  into  $B = \{1, 9, 11\}$  defined by  $a R_2 b$  if and only if  $a$  divides  $b$ .
  - c.  $R_3$  is from  $A = \{1, 2, 3, 4\}$  into  $B = \{5, 6, 10\}$  defined by  $a R_3 b$  if and only if  $\gcd(a, b) = 1$ .
  - d.  $R_4$  is from  $A = \{2, 5, 7, 18\}$  into  $B = \{2, 3, 4, 5, 10\}$  defined by  $a R_4 b$  if and only if  $a \leq b$ .
  - e.  $R_5$  is from  $A = \{2, 6, 8, 10, 12\}$  into  $B = \{1, 9, 11\}$  defined by  $a R_5 b$  if and only if  $a = b + 1$ .
2. Find the domains and ranges of the binary relations of Exercise 1.
3. Let  $A = \{1, 2, 3, 4, 5\}$  and let  $R$  be a relation on  $A$  defined by

$$R = \{(1, 3), (2, 1), (2, 2), (2, 5), (3, 4), (4, 3), (4, 4), (5, 1), (5, 3)\}.$$

Compute  $R^2, R^3, R^{-1}, R \circ R^{-1}, R^{-1} \circ R, R \cup R^2 \cup R^3$ .

4. Let  $A = \{2, 3, 4\}$  and let  $R$  be a relation on  $A$  defined by

$$R = \{(2, 3), (3, 2), (4, 2), (2, 4)\}.$$

Compute  $R^2, R^3, R^4, R^{-1}, R \circ R^{-1}, R^{-1} \circ R, R \cup R^2 \cup R^3 \cup R^4$ .

5. Let  $A = \{1, 2, 3, 4\}$  and let  $R$  be a relation on  $A$  defined by  $a R b$  if and only if  $a < b$  for all  $a, b \in A$ . Find  $R^2$  and  $R^3$ .
6. For each of the following relations on  $A = \{1, 2, 3, 4\}$ , decide whether it is reflexive, symmetric, or transitive.

- a.  $R = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 2), (2, 1)\}$   
b.  $R = \{(1, 1), (2, 2), (3, 3), (1, 2), (2, 1)\}$   
c.  $R = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 2)\}$   
d.  $R = \{(1, 2), (2, 1)\}$   
e.  $R = \{(1, 1), (2, 2)\}$   
f.  $R = \{(1, 1), (2, 2), (3, 3), (4, 4)\}$   
g.  $R = \{(1, 1), (1, 2), (2, 1)\}$   
h.  $R = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 3)\}$   
i.  $R = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 2), (2, 1), (2, 3), (3, 4)\}$   
j.  $a R b$  if and only if  $a$  divides  $b$  for all  $a, b \in A$
7. Which of the relations in Exercise 6 are equivalence relations? If a relation is an equivalence relation, then find the corresponding partitions of the set  $A$ .
8. Find all of the reflexive relations on the set  $A = \{a, b\}$ .
9. Find the number of reflexive relations on a set of three elements.
10. Find the number of symmetric relations on a set of three elements.
11. For the following partitions  $\mathcal{P} = \{\{a\}, \{b, c\}, \{d, e\}\}$ , write the corresponding equivalence relation on the set  $A = \{a, b, c, d, e\}$ .
12. Let  $X$  be a nonempty set. Prove that the following conditions are equivalent:
- $R$  is an equivalence relation on  $X$ .
  - $R$  is a reflexive relation on  $X$ , and for all  $x, y, z \in X$ , if  $x R y$  and  $y R z$ , then  $z R x$ .
  - $R$  is a reflexive relation on  $X$ , and for all  $x, y, z \in X$ , if  $x R y$  and  $x R z$ , then  $y R z$ .
13. Let  $A = \{n \in \mathbb{Z} \mid 1 \leq n \leq 20\}$ . Define a relation  $R$  on  $A$  by  $a R b$  if and only if 5 divides  $a - b$  for all  $a, b \in A$ . Show that  $R$  is an equivalence relation on  $A$ . Find all the equivalence classes.
14. Let  $R_1$  and  $R_2$  be two equivalence relations on a set  $A$  such that  $R_1 \circ R_2 = R_2 \circ R_1$ . Prove that  $R_1 \circ R_2$  is also an equivalence relation.
15. Given a relation  $R = \{(1, 2), (2, 3)\}$  on the set of all natural numbers, add a minimum number of ordered pairs of natural numbers so that the resulting relation is symmetric and transitive.
16. Let  $R$  be a relation defined on the set  $\mathbb{Z}$  as  $R = \{(x, y) \in \mathbb{Z} \times \mathbb{Z} \mid x + 2y = 31\}$ . Find the domain and range of  $R$ . Show that this relation is neither reflexive, nor symmetric, nor transitive.
17. Determine which of the following relations  $R$  are equivalence relations on the set of integers.
- $a R b$  if and only if  $a^2 = b^2$
  - $a R b$  if and only if  $a \leq |b|$
  - $a R b$  if and only if  $a - b$  is an even integer
  - $a R b$  if and only if  $a \geq b$
  - $a R b$  if and only if  $|a| = |b|$
  - $a R b$  if and only if  $b = a^r$  for some positive integer  $r$
18. Determine which of the following relations  $R$  on  $S$  is an equivalence relation.
- $S = \mathbb{Q}, R = \{(a, b) \in \mathbb{Q} \times \mathbb{Q} \mid a - b \text{ is an integer}\}$
- b.  $S =$  the set of all lines in the Euclidean plane  $\mathbb{R} \times \mathbb{R}$ ,  $l R m$  if and only if  $l$  is perpendicular to  $m$  for all lines  $l, m$
- c.  $S =$  the set of all lines in the Euclidean plane  $\mathbb{R} \times \mathbb{R}$ ,  $l R m$  if and only if  $l$  is parallel to  $m$  for all lines  $l, m$ . (Assume that a line is parallel to itself.)
- d.  $S = \mathbb{Z} \times \mathbb{Z}$ ,  $(a, b) R (c, d)$  if and only if  $a + d = b + c$
- e.  $S = \mathbb{Z} \times (\mathbb{Z} - \{0\})$ ,  $(a, b) R (c, d)$  if and only if  $ad = bc$
- f.  $S = \mathbb{Z}$ ,  $n R m$  if and only if  $n + m$  is even
- g.  $S = \mathbb{Z}$ ,  $n R m$  if and only if  $n - m = 0$  or 5
- h.  $S = \mathbb{Z}$ ,  $n R m$  if and only if  $n - m = 0$ , or 5, or -5
- i.  $S = \mathbb{C}$ ,  $x R y$  if and only if  $x^2 + y^2 = 1$
- j.  $S = \mathbb{Z}$ ,  $x R y$  if and only if  $|x| = |y|$
19. Determine which of the following relations are (i) reflexive, (ii) symmetric, (iii) transitive.
- The relation  $R = \{(1, 1), (2, 2), (3, 4), (4, 3), (4, 4)\}$  on  $A = \{1, 2, 3, 4\}$
  - The relation ‘|’ of divisibility on the set of natural numbers
20. Draw a digraph of the following relation on the set  $\{1, 2, 3, 4, 5\}$ . Determine whether it is reflexive, symmetric, or transitive.
- $R = \{(2, 2), (3, 3), (5, 5), (1, 3), (1, 4), (4, 1), (1, 5), (4, 5), (5, 2)\}$
  - $R = \{(1, 1), (2, 2), (3, 3), (4, 4), (5, 5)\}$
  - $R = \{(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (1, 5), (2, 3), (5, 1), (3, 2)\}$
21. The digraphs in Figure 3.18 represent relations on a set  $S$ . For each digraph, write the set and the corresponding relation as a set of ordered pairs. Moreover, determine whether the relation is reflexive, symmetric, or transitive. Are any of the relations equivalence relations? If yes, which ones?

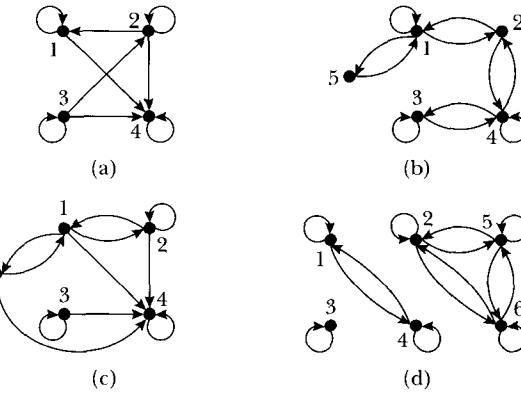


FIGURE 3.18 Various Digraphs

22. On the set  $\mathbb{Z}$ , define an equivalence relation  $R$  by  $a R b$  if and only if  $a^2 - b^2$  is divisible by 3. Show that  $R$  is an equivalence relation. Find the corresponding partition on  $\mathbb{Z}$ .
23. Prove Theorem 3.1.14.

24. Let  $S$  be a finite set with  $n$  elements. Prove that the number of symmetric relations on  $S$  is  $2^{n(n+1)/2}$  and the number of relations that are both reflexive and symmetric is  $2^{n(n-1)/2}$ .
25. Find the reflexive closures of the following relations  $R$  on  $A = \{1, 2, 3, 4\}$ .
- $R = \{(1, 1), (2, 2), (3, 3), (1, 2), (2, 1)\}$
  - $R = \{(1, 3), (1, 2), (2, 1)\}$
26. Find the symmetric closures of the following relations  $R$  on  $A = \{1, 2, 3, 4\}$ .
- $R = \{(1, 1), (2, 2), (3, 3), (1, 3), (2, 1)\}$
  - $R = \{(1, 3), (1, 2), (2, 1)\}$
27. Find the transitive closures of the following relations  $R$  on  $A = \{1, 2, 3, 4\}$ .
- $R = \{(2, 2), (3, 3), (1, 3), (2, 1)\}$
  - $R = \{(1, 3), (3, 2), (2, 1)\}$
28. Prove that a relation  $R$  on a set  $S$  is transitive if and only if  $R^n \subseteq R$  for all  $n = 1, 2, 3, \dots$ .
29. If the following assertions are true, prove them; otherwise give counterexamples to disprove them.
- If  $R$  is a reflexive relation on a set  $A$ , then so is  $R^{-1}$ .
  - If  $R$  is a transitive relation on a set  $A$ , then so is  $R^{-1}$ .
  - If  $R_1$  and  $R_2$  are transitive relations on a set  $A$ , then so is  $R_1 \circ R_2$ .
  - If a relation is symmetric and transitive, then it is reflexive.
  - Every relation must either be symmetric or antisymmetric.
  - Let  $R$  be a reflexive and transitive relation on a set  $A$ . Then  $R \cap R^{-1}$  is an equivalence relation.

## 3.2 PARTIALLY ORDERED SETS

In the first section of this chapter, we defined binary relations and studied their basic properties. More specifically, we discussed reflexive, symmetric, and transitive relations. In this section, we consider binary relations, which are reflexive and transitive and satisfy a new property, called the antisymmetric property. We begin with the following definition.

**DEFINITION 3.2.1** ► A relation  $R$  on a set  $S$  is called **antisymmetric** if for all  $a, b \in S$ ,  $a R b$  and  $b R a$ , then  $a = b$ .

On the set of all integers, the usual “less than or equal to,”  $\leq$  relation is an antisymmetric relation, because if  $a$  and  $b$  are integers such that  $a \leq b$  and  $b \leq a$ , then  $a = b$ .

If  $T$  is the set of subsets of a set  $A$ , then the inclusion relation  $\subseteq$  is an antisymmetric relation, because for subsets  $X$  and  $Y$  of  $A$  such that  $X \subseteq Y$  and  $Y \subseteq X$ , we have  $X = Y$ .

**DEFINITION 3.2.2** ► A relation  $R$  on a set  $A$  is called a **partial order** on  $A$  if  $R$  is reflexive, antisymmetric, and transitive. In other words, if  $R$  satisfies the following conditions:

- $a R a$  for all  $a \in A$  (i.e.,  $R$  is reflexive).
- For all  $a, b \in A$  if  $a R b$  and  $b R a$ , then  $a = b$  (i.e.,  $R$  is antisymmetric).
- For all  $a, b, c \in A$ , if  $a R b$  and  $b R c$ , then  $a R c$  (i.e.,  $R$  is transitive).

A set  $A$  together with a partial order relation  $R$  is called a **partially ordered set**, or simply **poset**, and we denote this poset by  $(A, R)$ .

Let  $(A, R)$  be a poset. If there is no confusion about the partial order, we may refer to this poset simply by  $A$ .

### EXAMPLE 3.2.3

The set  $\mathbb{Z}$ , together with the usual “less than or equal to,”  $\leq$  relation is a poset.

Note that the relation  $<$  (only less than) is not a partial order relation on  $\mathbb{Z}$  because the relation  $<$  is not reflexive as  $1 \not< 1$ .

**EXAMPLE 3.2.4**

Consider  $\mathbb{N}$ , the set of all natural numbers, and the divisibility relation,  $R$ , on  $\mathbb{N}$ . That is, for all  $a, b \in \mathbb{N}$ ,  $a R b$  if  $a | b$  (i.e.,  $a R b$  if there exists a positive integer  $c$  such that  $b = ac$ ).

We show that  $R$  is a partial order relation on  $\mathbb{N}$ . That is, we show that the divisibility relation is reflexive, antisymmetric, and transitive. For simplicity, we write  $R$  as  $|$ .

*Reflexive:* Let  $a \in \mathbb{N}$ . Because  $a = 1a$ , we have  $a | a$ .

*Antisymmetric:* Let  $a | b$  and  $b | a$ . Then  $b = ad$  and  $a = bc$  for some positive integers  $c$  and  $d$ . Therefore,  $a = bc = adc$ , so  $1 = cd$ . Because  $c$  and  $d$  are positive integers and  $cd = 1$ , it follows that  $c = d = 1$ . Hence,  $a = b$ .

*Transitive:* Let  $a | b$  and  $b | c$  in  $\mathbb{N}$ . Then  $b = an$  and  $c = bm$  for some positive integers  $m$  and  $n$ . This implies that  $c = bm = anm$ , and because  $m$  and  $n$  are positive integers,  $nm$  is a positive integer. Thus,  $a | c$  in  $\mathbb{N}$ .

Consequently, the divisibility relation is a partial order on  $\mathbb{N}$ , and hence  $(\mathbb{N}, |)$  is a poset.

---

**REMARK 3.2.5** ▶ Though the divisibility relation is a partial order relation on the set of all positive integers, it is not so on the set of all nonzero integers. For example,  $4 = (-1)(-4)$  and  $-4 = (-1)4$  imply that  $4 | -4$  and  $-4 | 4$ , but  $4 \neq -4$ .

**EXAMPLE 3.2.6**

Let  $S$  be a set and  $\mathcal{T}$  be the set of some subsets of  $S$ . Let  $R$  be a relation on  $\mathcal{T}$  given by  $R = \{(A, B) \in \mathcal{T} \times \mathcal{T} \mid A \subseteq B\}$ . We show that  $R$  is a partial order on  $\mathcal{T}$ .

Because  $A \subseteq A$  for all  $A \in \mathcal{T}$ , we find that the relation  $R$  is reflexive.

To show that  $R$  is antisymmetric, let  $(A, B), (B, A) \in R$ . Then, by the definition of  $R$ ,  $A \subseteq B$  and  $B \subseteq A$ , so  $A = B$ . Thus,  $R$  is antisymmetric.

To show that  $R$  is transitive, let  $(A, B), (B, C) \in R$ . Then  $A \subseteq B$  and  $B \subseteq C$ , so  $A \subseteq C$ . Hence,  $R$  is transitive. Consequently,  $R$  is a partial order on  $\mathcal{T}$ .

Let  $R$  be a partial order on a set  $S$ . Then  $R^{-1}$  is also a partial order relation on  $S$ . Therefore, if  $(S, R)$  is a poset, then  $(S, R^{-1})$  is a poset. The poset  $(S, R^{-1})$  is called the **dual** of  $(S, R)$ .

For example, the relation  $\geq$  (the usual “greater than or equal to”) on the set  $\mathbb{Z}$  is the inverse relation of  $\leq$ , and hence the poset  $(\mathbb{Z}, \geq)$  is the dual of the poset  $(\mathbb{Z}, \leq)$ .

**Notation 3.2.7:** Let  $R$  be a partial order on a set  $A$ ; i.e.,  $(A, R)$  is a poset. We usually denote  $R$  by  $\leq_A$ . If the set  $A$  is understood, then we write  $\leq_A$  as  $\leq$ . If  $A$  is a partially ordered set with a partial order  $\leq$ , then we denote this by  $(A, \leq_A)$  or  $(A, \leq)$ .

Let  $(A, \leq)$  be a poset and  $a, b \in A$ . If  $a \leq b$  and  $a \neq b$ , then we write  $a < b$ . Note that here  $\leq$  means any relation, not the usual less than or equal to. Similarly,  $<$  means related but not equal.

---

**DEFINITION 3.2.8** ▶ Let  $(S, \leq)$  be a poset and  $a, b \in S$ . If either  $a \leq b$  or  $b \leq a$ , then we say that  $a$  and  $b$  are **comparable**. The poset  $(S, \leq)$  is called a **linearly ordered set**, or a **totally ordered set**, or a **chain**, if for all  $a, b \in S$  either  $a \leq b$  or  $b \leq a$ .

Thus, a linearly ordered set, or a totally ordered set, or a chain, is a poset in which any two elements are comparable.

**EXAMPLE 3.2.9**

- (i) Consider the poset  $(\mathbb{Z}, \leq)$  of Example 3.2.3. For any two integers  $a$  and  $b$ ,  $a < b$ , or  $a = b$ , or  $a > b$ . Thus, any two integers with respect to the partial order  $\leq$  are comparable. Hence,  $(\mathbb{Z}, \leq)$  is a chain.
- (ii) Consider the poset  $(\mathbb{N}, \leq)$  of Example 3.2.4. Notice that here the relation  $\leq$  is the divisibility relation. That is,  $a \leq b$  means  $a | b$ . Now, 3 does not divide 5 and 5 does not divide 3. Therefore, 3 and 5 are not comparable. Hence,  $(\mathbb{N}, \leq)$  is not a chain.
- (iii) Let  $A$  be a set with more than one element. Consider  $\mathcal{P}(A)$ , the power set of  $A$ , together with the set inclusion relation. Then, as in Example 3.2.6,  $(\mathcal{P}(A), \leq)$  is a poset. (Notice that here  $\leq$  means  $\subseteq$ ). Let  $a$  and  $b$  be distinct elements of  $A$ . Then  $\{a\}$  is not a subset of  $\{b\}$  and  $\{b\}$  is not subset of  $\{a\}$ ; i.e.,  $\{a\}$  and  $\{b\}$  are not comparable. It follows that  $(\mathcal{P}(A), \leq)$  is not a chain.

**Lexicographic Order**

Let  $(A, \leq)$  and  $(B, \leq)$  be two posets. Define a relation  $R$  on the set  $A \times B$  by  $(a, b) R (c, d)$  if  $a \leq c$  and  $b \leq d$  for all  $(a, b), (c, d) \in A \times B$ . This relation  $R$  is a partial order and it is called the **product partial order**.

There is another partial order relation, denoted by  $\preceq$ , on  $A \times B$ , which is defined as follows:

$$(a, b) \preceq (c, d) \text{ if and only if } a < c \text{ or } a = c \text{ and } b \leq d.$$

This partial order is called **lexicographic order**.

We can extend lexicographic order from the Cartesian product of two sets to, say  $n$  sets, as follows. Let  $A_1, A_2, \dots, A_n$ ,  $n \geq 1$ , be partially ordered sets; i.e.,  $(A_i, \leq)$  is a poset for all  $i = 1, 2, \dots, n$ . Define the relation  $\preceq$  on  $A_1 \times A_2 \times \dots \times A_{n-1} \times A_n$  as follows: Let  $(a_1, a_2, \dots, a_n), (b_1, b_2, \dots, b_n) \in A_1 \times A_2 \times \dots \times A_{n-1} \times A_n$ . Then

$$(a_1, a_2, \dots, a_n) \preceq (b_1, b_2, \dots, b_n)$$

if and only if

$$\begin{aligned} a_1 &< b_1 \quad \text{or} \\ a_1 &= b_1 \quad \text{and} \quad a_2 < b_2 \quad \text{or} \\ a_1 &= b_1, a_2 = b_2 \quad \text{and} \quad a_3 < b_3 \quad \text{or} \\ &\vdots \\ a_1 &= b_1, a_2 = b_2, a_3 = b_3, \dots, a_{n-1} = b_{n-1} \quad \text{and} \quad a_n < b_n \quad \text{or} \\ a_1 &= b_1, a_2 = b_2, a_3 = b_3, \dots, a_{n-1} = b_{n-1} \quad \text{and} \quad a_n = b_n. \end{aligned}$$

**EXAMPLE 3.2.10**

Consider the poset  $\mathbb{R}$  of all real numbers under partial order  $\leq$  (usual “less than or equal to”) relation. Then  $(\mathbb{R} \times \mathbb{R}, \preceq)$  is a poset under the lexicographic order relation.

In this poset,

$$\begin{aligned} (2, 8) &\preceq (3, 0), \quad \text{because } 2 < 3, \\ (5, 1) &\preceq (5, 3), \quad \text{because } 5 = 5, \text{ but } 1 < 3, \\ (5, 3) &\not\preceq (1, 0), \quad \text{because } 5 > 1. \end{aligned}$$

We show that this poset is a linearly ordered set.

Let  $(a, b)$  and  $(c, d)$  be two elements of  $\mathbb{R} \times \mathbb{R}$ . Now for the real numbers  $a$  and  $c$ , either  $a < c$  or  $a = c$  or  $c < a$ . Therefore,

- if  $a < c$ , then  $(a, b) \preceq (c, d)$ ,
- if  $a = c$  and  $b < d$ , then  $(a, b) \preceq (c, d)$ ,
- if  $a = c$  and  $b > d$ , then  $(c, d) \preceq (a, b)$ ,
- if  $a = c$  and  $b = d$ , then  $(a, b) = (c, d)$ ,
- if  $a > c$ , then  $(c, d) \preceq (a, b)$ .

Thus, we find that  $(\mathbb{R} \times \mathbb{R}, \preceq)$  is a linearly ordered set.

Let us look at the ordering that is used to arrange the words in an English dictionary. Let  $A$  be the set of all 26 letters,  $a, b, c, d, e, \dots, x, y, z$ . We define an ordering on  $A$  as follows:  $a$  is the first element,  $b$  is the second element,  $c$  is the third element,  $\dots$ ,  $x$  is the 24th element,  $y$  is the 25th element, and  $z$  is the 26th element.

Let  $a_i, a_j$  denote the  $i$ th and  $j$ th elements, respectively, where  $i, j \in \{1, 2, 3, \dots, 24, 25, 26\}$ . Define  $a_i \leq a_j$  if and only if  $i \leq j$ . Then  $A$  is a poset under this relation.

We denote the Cartesian products  $A \times A \times A \times \dots \times A$  of  $n$   $A$ 's by  $A^n$ . Then  $(A^n, \preceq)$  is a poset under the lexicographic order relation  $\preceq$ .

Consider two words  $a_1 a_2 \dots a_n$  and  $b_1 b_2 \dots b_m$  over  $A$ . Let  $r$  be the minimum of  $m$  and  $n$ . Define the relation  $R$  on the set of all English words on  $A$  as follows:

$$a_1 a_2 \dots a_n R b_1 b_2 \dots b_m$$

if and only if

- $(a_1, a_2, \dots, a_r) \neq (b_1, b_2, \dots, b_r)$  and  $(a_1, a_2, \dots, a_r) \preceq (b_1, b_2, \dots, b_r)$  in the poset  $(A^r, \preceq)$  or
- $(a_1, a_2, \dots, a_r) = (b_1, b_2, \dots, b_r)$  and  $m > n$ .

One can verify that  $R$  is a partial order relation. This relation is called **dictionary order**, and it also is denoted by  $\preceq$ .

### EXAMPLE 3.2.11

In the dictionary, the word *mango* comes before *money* because of the letters in the second position:  $a < o$ . Likewise, *grass* comes before *grit*; here we have to compare the first three letters, we see that the first two letters are the same and in the third position,  $a < i$ . To compare the words *earth* and *earthquake*, we have to compare the first five letters. Because the first five letters are the same, our next step is to compare the length of the words. As the word *earthquake* is longer than word *earth*, it follows that *earth* comes before *earthquake* in the dictionary.

## Digraphs of Posets

Because any partial order is also a relation, we can give a digraph representation of partial order.

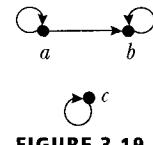
### EXAMPLE 3.2.12

On the set  $S = \{a, b, c\}$ , consider the relation

$$R = \{(a, a), (b, b), (c, c), (a, b)\}.$$

The digraph of  $R$  is shown in Figure 3.19.

From the directed graph it follows that the given relation is reflexive and transitive. This relation is also antisymmetric because there is a directed edge



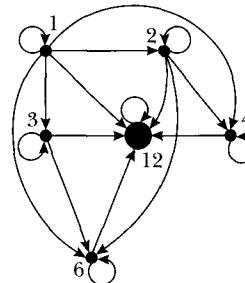
**FIGURE 3.19**  
Digraph of  $R$

from  $a$  to  $b$ , but there is no directed edge from  $b$  to  $a$ . Again, in the graph we notice that there are two distinct vertices  $a$  and  $c$  such that there are no directed edges from  $a$  to  $c$  and from  $c$  to  $a$ .

In a digraph of a partial order, one can see that if there is a directed edge from a vertex  $a$  to a different vertex  $b$ , then there is no directed edge from  $b$  to  $a$ .

### EXAMPLE 3.2.13

Let  $S = \{1, 2, 3, 4, 6, 12\}$ . Consider the divisibility relation on  $S$ , which is a partial order. A digraph of this poset is as shown in Figure 3.20.



**FIGURE 3.20** Digraph  
of a relation on  $S$

A digraph representation of a partial order suggests the following theorem.

**Theorem 3.2.14:** A digraph of a partial order relation  $R$  cannot contain a closed directed path other than loops. (A path  $a_1, a_2, \dots, a_n$  in the digraph is **closed** if  $a_1 R a_2, a_2 R a_3, \dots, a_n R a_1$ .)

**Proof:** Let  $a_1, a_2, \dots, a_n, n \neq 1$ , be a closed directed path of distinct vertices  $a_1, a_2, \dots, a_n$ . Then  $a_1 R a_2, a_2 R a_3, \dots, a_n R a_1$ . Now  $R$  is transitive, and  $a_1 R a_2, a_2 R a_3, \dots, a_{n-1} R a_n$ . Therefore,  $a_1 R a_n$ . Also, we have  $a_n R a_1$ . Now,  $a_1 R a_n$  and  $a_n R a_1$ , and so by the antisymmetric property of  $R$ , it follows that  $a_1 = a_n$ . This contradicts our assumption that  $a_1, a_2, \dots, a_n$  are distinct. Consequently, there is no directed closed path other than loops. ■

By Theorem 3.2.14, it follows that if a digraph of a relation contains a closed path other than loops, then the corresponding relation is not a partial order.

### EXAMPLE 3.2.15

On the set  $S = \{a, b, c\}$  consider the relation

$$R = \{(a, a), (b, b), (c, c), (a, b), (b, c), (c, a)\}.$$

The digraph of this relation is given in Figure 3.21.

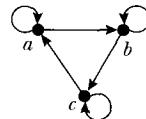


FIGURE 3.21

Digraph of  $R$ 

In this digraph, we see that  $a, b, c, a$  form a closed path. Hence, the given relation is not a partial order relation.

### Hasse Diagram

Another visual device used in the study of posets is the Hasse diagram (named after Helmut Hasse, a twentieth-century German number theorist). Before we discuss how to draw Hasse diagrams, however, we need to define a few terms.

Let  $(S, \leq)$  be a poset and  $x, y \in S$ . We say that  $y$  **covers**  $x$ , if  $x \leq y$ ,  $x \neq y$ , and there are no elements  $z \in S$  such that  $x < z < y$ .

We draw a diagram using the elements of  $S$  as follows: We represent the elements of  $S$  in the diagram by the elements themselves such that if  $x \leq y$ , then  $y$  is placed above  $x$ . We connect  $x$  with  $y$  by a line segment if and only if  $y$  covers  $x$ . The resulting diagram is called the **Hasse diagram** of  $(S, \leq)$ .

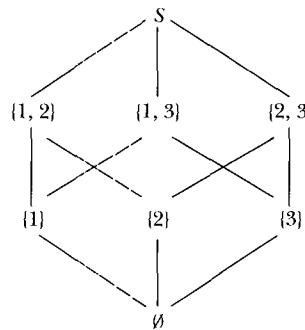
The following example illustrates how to draw Hasse diagrams.

#### EXAMPLE 3.2.16

Let  $S = \{1, 2, 3\}$ . Then

$$\mathcal{P}(S) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}, S\}.$$

Now  $(\mathcal{P}(S), \leq)$  is a poset, where  $\leq$  denotes the set inclusion relation. The poset diagram of  $(\mathcal{P}(S), \leq)$  is shown in Figure 3.22.

FIGURE 3.22 Hasse diagram of  $(\mathcal{P}(S), \leq)$ 

**Helmut Hasse**  
(1898–1979)

Helmut Hasse was born in Germany and attended the Fichtegymnasium in Berlin for two years. At the age of 15, he volunteered for naval service during World War I. Upon leaving the navy, he entered the University of Göttingen.

#### Historical Notes

In 1920, Hasse studied p-adic numbers under the tutelage of Hensel and discovered the Hasse principle as part of his dissertation. This principle states that the representability of a number by a given form and whether two forms are equivalent can be decided using only local information. Beginning in 1922, Hasse began working at the University of Kiel in the area of field theory.

During World War II, Hasse worked for the navy on ballistics but was denied membership in the Nazi Party because of his Jewish ancestry. He finished his career at Hamburg (1950–1966), writing a textbook containing an introduction to algebraic number theory.

A Hasse diagram of a poset  $(S, \leq)$  of finite elements can also be obtained from the digraph of the poset  $(S, \leq)$ . To do this, we use the following steps.

1. In the digraph, we place vertex  $a$  above vertex  $b$  if  $a$  covers  $b$  in the poset  $(S, \leq)$ .
2. We delete all loops from the digraph. (Because the relation is reflexive, there is a loop at each vertex, so it is not necessary to show the loop).
3. We delete all the directed edges that are implied by the transitive property. For example, suppose  $a \leq b, b \leq c$ . Then  $a \leq c$ , so we omit the edges from  $a$  to  $c$ .
4. We omit the arrow signs from the directed edges (because we draw the directed edges following the condition stated in step (1).)

In the following example, we show how we can construct the Hasse diagram of the poset of Example 3.2.16 from the digraph of the partial order.

### EXAMPLE 3.2.17

Let  $S = \{1, 2, 3\}$ . Then

$$\mathcal{P}(S) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}, S\}.$$

Now  $(\mathcal{P}(S), \leq)$  is a poset, where  $\leq$  denotes the set inclusion relation.

Let us draw the digraph of this inclusion relation (see Figure 3.23). Place the vertex  $A$  above vertex  $B$  if  $B \subset A$ . Now follow steps (2), (3), and (4). Thus, we obtain the Hasse diagram as shown in Figure 3.22.

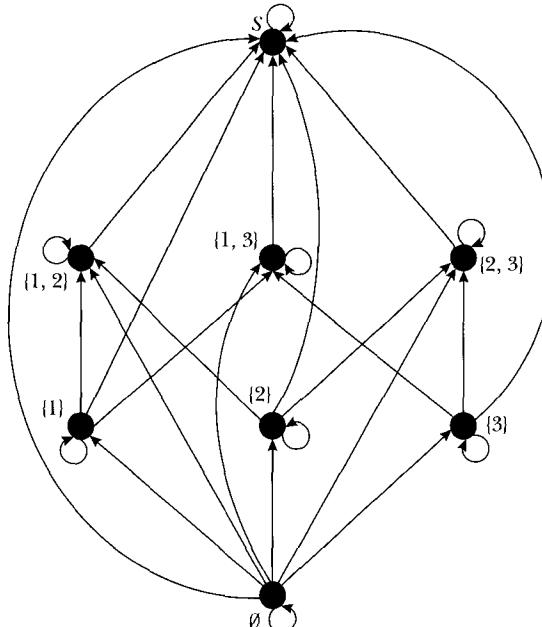


FIGURE 3.23 Digraph of  $(\mathcal{P}(S), \leq)$

## Minimal and Maximal Elements

Let us now define some special elements in a poset.

**DEFINITION 3.2.18** ▶ Let  $(S, \leq)$  be a poset. An element  $a \in S$  is called

- (i) a **minimal element** if there is no element  $b \in S$  such that  $b < a$ ,
- (ii) a **maximal element** if there is no element  $b \in S$  such that  $a < b$ ,
- (iii) a **greatest element** if  $b \leq a$  for all  $b \in S$ ,
- (iv) a **least element** if  $a \leq b$  for all  $b \in S$ .

### EXAMPLE 3.2.19

In this example, we consider the poset  $(S, \leq)$ , where

$$S = \{2, 4, 5, 10, 15, 20\}$$

and the partial order  $\leq$  is the divisibility relation.

In this poset, there is no element  $b \in S$  such that  $b \neq 5$  and  $b$  divides 5. (That is, 5 is not divisible by any other element of  $S$  except 5). Hence, 5 is a minimal element. Similarly, 2 is a minimal element.

Now, 10 is not a minimal element because  $2 \in S$  and 2 divides 10. That is, there exists an element  $b \in S$  such that  $b < 10$ . Similarly, 4, 15, and 20 are not minimal elements.

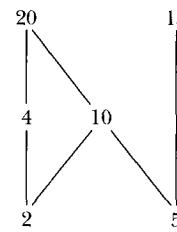
We find that 2 and 5 are the only minimal elements of this poset. Notice that 2 does not divide 5. Therefore, it is not true that  $2 \leq b$ , for all  $b \in S$ , and so 2 is not a least element in  $(S, \leq)$ . Similarly, 5 is not a least element. Actually, we can show that this poset has no least element.

There is no element  $b \in S$  such that  $b \neq 15$ ,  $b > 15$ , and 15 divides  $b$ . That is, there is no element  $b \in S$  such that  $15 < b$ . Thus, 15 is a maximal element. Similarly, 20 is a maximal element.

Notice that 10 is not a maximal element because  $20 \in S$  and 10 divides 20. That is, there exists an element  $b \in S$  such that  $10 < b$ . Similarly, 4 is not a maximal element.

We find that 20 and 15 are the only maximal elements of this poset.

We also notice that 10 does not divide 15, hence it is not true that  $b \leq 15$ , for all  $b \in S$ , and so 15 is not a greatest element in  $(S, \leq)$ . Actually, we can show that this poset has no greatest element. Let us draw the Hasse diagram of this poset. (See Figure 3.24.) The Hasse diagram shows the maximal and minimal elements.



**FIGURE 3.24**  
Hasse diagram

The following lemma ensures that every finite poset contains a minimal element. In this connection, we point out that a poset with an infinite number of elements may not have a minimal element. For example, the poset  $(\mathbb{Z}, \leq)$  of all integers under the usual ‘less than or equal to’ relation has no minimal and maximal elements.

**Lemma 3.2.20:** Let  $(S, \leq)$  be a poset such that  $S$  is a finite nonempty set. Then this poset has a minimal element.

**Proof:** Let  $a_1$  be an element of  $S$ . If  $a_1$  is a minimal element, then we are done. Suppose  $a_1$  is not a minimal element. Then there exists  $a_2 \in S$  such that  $a_2 < a_1$ . If  $a_2$  is a minimal element, then we are done, otherwise there exists  $a_3 \in S$  such that  $a_3 < a_2$ . If  $a_3$  is not a minimal element, we repeat this process. Now  $a_3 < a_2 < a_1$  shows that  $a_3, a_2, a_1$  are distinct elements. Because  $S$  is finite, after a finite number of steps we get an element  $a_n \in S$  such that  $a_n$  is a minimal element. ■

**REMARK 3.2.21** ▶ Let  $(S, \leq)$  be a poset such that  $S$  is a finite nonempty set. Then  $S$  has minimal and maximal elements, but  $S$  may not have the least and the greatest elements.

**DEFINITION 3.2.22** ▶ Let  $S$  be a set and let  $\leq_1$  and  $\leq_2$  be two partial order relations on  $S$ . The relation  $\leq_2$  is said to be **compatible** with the relation  $\leq_1$  if  $a \leq_1 b$  implies  $a \leq_2 b$ .

It is interesting to note that given a finite nonempty set, say  $S$ , we can define a linear order on it as follows.

Because  $S$  is nonempty,  $S$  has at least one element. Choose an element from  $S$ , and call it the first element,  $a_1$ . Let  $S_1 = S - \{a_1\}$ . If  $S_1$  is not empty, then from  $S_1$  choose an element  $a_2$ . Let  $S_2 = S - \{a_1, a_2\}$ . If  $S_2$  is not empty, then from  $S_2$  choose an element  $a_3$ . Let  $S_3 = S - \{a_1, a_2, a_3\}$ . If  $S_3$  is not empty, continue the process. Because  $S$  is a finite set, this process must stop after a finite number of steps. Hence, there exists a positive integer  $n$  such that  $S_n = S - \{a_1, a_2, \dots, a_n\}$  is empty, where  $a_n$  is an element of  $S_{n-1} = S - \{a_1, a_2, \dots, a_{n-1}\}$ . We now define a partial order  $\leq_1$  on  $S$  by  $a_1 \leq_1 a_2 \leq_1 \dots \leq_1 a_n$ . This means that  $a_i \leq_1 a_j$  if and only if either  $i = j$  or  $i < j$ , where  $i, j \in \{1, 2, \dots, n\}$ . It follows that this is a linear order.

Now suppose that not only  $S$  is a finite nonempty set, but  $S$  also has a partial order  $\leq$ . Can we define a linear order  $\leq_1$  on  $S$  that is compatible with the partial order  $\leq$ ? The following theorem proves that there exists such a linear order.

**Theorem 3.2.23:** Let  $(S, \leq)$  be a finite poset. There exists a linear order  $\leq_1$  on  $S$  which is compatible with the relation  $\leq$ .

**Proof:** Because  $(S, \leq)$  is a finite poset, by Lemma 3.2.20 there exists a minimal element, say  $a_1$ , in  $S$ . Let  $S_1 = S - \{a_1\}$ . If  $S_1$  is not empty, then  $S_1$  is also a poset under the partial order relation induced by the given partial order relation on  $S$ ; i.e., for all  $a, b \in S_1$ ,  $a \leq b$  in  $S_1$  if and only if  $a \leq b$  in  $S$ . By Lemma 3.2.20,  $S_1$  has a minimal element, say  $a_2$ .

Let  $S_2 = S - \{a_1, a_2\}$ . If  $S_2$  is not empty, then  $S_2$  is also a poset under the partial order relation induced by the given partial order on  $S$ . By Lemma 3.2.20,  $S_2$  has a minimal element, say  $a_3$ . Let  $S_3 = S - \{a_1, a_2, a_3\}$ . If  $S_3$  is not empty, repeat this process.

Because  $S$  is a finite set, the above process must stop after a finite number of steps. Hence, there exists a positive integer  $n$  such that  $S_n = S - \{a_1, a_2, \dots, a_n\}$  is empty, where  $a_n$  is a minimal element in  $S_{n-1} = S - \{a_1, a_2, \dots, a_{n-1}\}$ .

We now define a partial order  $\leq_1$  on  $S$  by considering  $a_1 \leq_1 a_2 \leq_1 \dots \leq_1 a_n$ . It follows that this is a linear order. We now show that  $\leq_1$  is compatible with  $\leq$ .

Let  $a, b \in S$  be such that  $a < b$ . Because  $S = \{a_1, a_2, \dots, a_n\}$ , there exist  $i$  and  $j$  such that  $a = a_i$  and  $b = a_j$ . Now  $a < b$  implies that  $i < j$ . Hence, we find that  $a_i \leq_1 \dots \leq_1 a_j$ . By the transitive property, we can conclude that  $a_i \leq_1 a_j$ . Therefore, it follows that the linear order  $\leq_1$  on  $S$  is compatible with the relation  $\leq$ . ■

The following example clarifies the construction of the compatible relation in Theorem 3.2.23.

**EXAMPLE 3.2.24**

Consider the poset  $(S, \leq)$  of the Example 3.2.19. In this poset,  $S = \{2, 4, 5, 10, 15, 20\}$  and the partial order relation is the divisibility relation.

As shown before, 5 is a minimal element of this poset. Let  $a_1 = 5$  and  $S_1 = S - \{5\} = \{2, 4, 10, 15, 20\}$ . Then  $S_1$  is also a poset under the divisibility relation. Also,  $S_1$  has a minimal element  $a_2 = 2$ . Let

$$S_2 = S - \{2, 5\} = \{4, 10, 15, 20\}.$$

Now,  $S_2$  has a minimal element  $a_3 = 4$ . Let

$$S_3 = S - \{2, 5, 4\} = \{10, 15, 20\}.$$

$S_3$  has a minimal element  $a_4 = 10$ . Let

$$S_4 = S - \{2, 5, 4, 10\} = \{15, 20\}.$$

$a_5 = 15$  is a minimal element of  $\{15, 20\}$ . Let

$$S_5 = S - \{2, 5, 4, 10, 15\} = \{20\}.$$

Finally,  $a_6 = 20$  is a minimal element of  $\{20\}$ . We now define a partial order  $\leq_1$  on  $S$  by  $5 \leq_1 2 \leq_1 4 \leq_1 10 \leq_1 15 \leq_1 20$ . It follows that this is a linear order.

We now show that  $\leq_1$  is compatible with  $\leq$ .

Now,  $5 \leq 15$  because 5 divides 15. In the relation  $\leq_1$ , we have

$$5 \leq_1 2 \leq_1 4 \leq_1 10 \leq_1 15.$$

By the transitivity of  $\leq_1$ , we have  $5 \leq_1 15$ . Similarly, we can verify that the compatibility holds for other elements. So it follows that the linear order  $\leq_1$  on  $S$  is compatible with the relation  $\leq$ .

We also note that the relation  $\leq_1$  on  $S$  is not the only linear order that is compatible with the relation  $\leq$ . In the construction of the linear order  $\leq_1$ , we started with the minimal element with 5. Now, 2 is another minimal element, so we can use it as the first element and construct the linear order

$$2 \leq_2 5 \leq_2 4 \leq_2 10 \leq_2 15 \leq_2 20.$$

We can also construct the linear order

$$2 \leq_3 5 \leq_3 4 \leq_3 15 \leq_3 10 \leq_3 20.$$

Both linear orders  $\leq_2$  and  $\leq_3$  are compatible with the relation  $\leq$ .

---

**DEFINITION 3.2.25** ▶ The process of constructing a linear order  $\leq_1$  on a poset  $(S, \leq)$  that is compatible with the partial order relation  $\leq$  is called **topological ordering**.

---

**REMARK 3.2.26** ▶ We discuss topological ordering in Chapter 10 Graph Theory.

## Lattices

**DEFINITION 3.2.27** ▶ Let  $(S, \leq)$  be a poset and let  $\{a, b\}$  be a subset of  $S$ . An element  $c \in S$  is called an **upper bound** of  $\{a, b\}$  if  $a \leq c$  and  $b \leq c$ .

An element  $d \in S$  is called a **least upper bound (lub)** of  $\{a, b\}$  if

- (i)  $d$  is an upper bound of  $\{a, b\}$ ; and
- (ii) if  $c \in S$  is an upper bound of  $\{a, b\}$ , then  $d \leq c$ .

### EXAMPLE 3.2.28

- (i) Consider the set  $\mathbb{N}$  together with the divisibility relation of Example 3.2.4. For all  $a, b \in \mathbb{N}$ ,  $a \leq b$  if and only if  $a$  divides  $b$ .

Consider the subset  $\{12, 8\}$ . We see that 24, 48, and 72 are all common multiples of 12 and 8. Hence,  $12 \leq 24$  and  $8 \leq 24$ ;  $12 \leq 48$  and  $8 \leq 48$ ; and  $12 \leq 72$  and  $8 \leq 72$ . Therefore, 24, 48, and 72 are upper bounds of  $\{12, 8\}$ . However, 24 is the least upper bound of  $\{12, 8\}$ . Notice that  $24 \notin \{12, 8\}$ .

- (ii) Consider the set  $\mathbb{Z}$ , together with the usual “less than or equal to,”  $\leq$ , relation of Example 3.2.3. Consider the subset  $\{5, 7\}$ . We see that  $7, 8, 9, \dots$  are all upper bounds of  $\{5, 7\}$ . However, 7 is the least upper bound of  $\{5, 7\}$ . Notice that  $7 \in \{5, 7\}$ .
- (iii) Let  $S = \{1, 2, 3\}$ . Let  $\leq$  denote the set inclusion relation. Then  $(\mathcal{P}(S), \leq)$  is a poset. Let  $A = \{1, 2\}$  and  $B = \{1, 3\}$ . Then  $A \cup B = \{1, 2, 3\}$  is a least upper bound of  $\{A, B\}$ . Notice that  $\{1, 2, 3\} = A \cup B \notin \{A, B\}$ .

The following theorem shows that the lub of a subset, if it exists, is unique.

**Theorem 3.2.29:** In a poset  $(S, \leq)$ , if a subset  $\{a, b\}$  of  $S$  has a lub, then this lub is unique.

**Proof:** Let  $a, b \in S$  and a lub of  $\{a, b\}$  exists. Suppose  $c, d \in S$  are two lubs of  $\{a, b\}$ . Then  $c$  and  $d$  are upper bounds of  $\{a, b\}$ . Because  $c$  is a lub of  $\{a, b\}$  and  $d$  is an upper bound of  $\{a, b\}$ ,  $c \leq d$ . Similarly,  $d \leq c$ . Thus, we have  $c \leq d$  and  $d \leq c$ . Therefore, by the antisymmetric property of the relation  $\leq$ , it follows that  $c = d$ . Hence, the lub is unique. ■

**Notation 3.2.30:** The lub of  $\{a, b\}$  in  $(S, \leq)$ , if it exists, is denoted by  $a \vee b$ .

**DEFINITION 3.2.31** ▶ Let  $(S, \leq)$  be a poset and let  $\{a, b\}$  be a subset of  $S$ . An element  $c \in S$  is called a **lower bound** of  $\{a, b\}$  if  $c \leq a$  and  $c \leq b$ .

An element  $d \in S$  is called a **greatest lower bound (glb)** of  $\{a, b\}$  if

- (i)  $d$  is a lower bound of  $\{a, b\}$ ; and
- (ii) if  $c \in S$  is a lower bound of  $\{a, b\}$ , then  $c \leq d$ .

Proceeding as in the proof of the Theorem 3.2.29, we can prove the following theorem.

**Theorem 3.2.32:** In a poset  $(S, \leq)$ , if a subset  $\{a, b\}$  of  $S$  has a glb, then this glb is unique.

**Notation 3.2.33:** The glb of  $\{a, b\}$  in  $(S, \leq)$ , if it exists, is denoted by  $a \wedge b$ .

We have seen several examples of posets in which lub (glb) need not exist. Next, we discuss those posets for which lub and glb exist.

**DEFINITION 3.2.34** ► A poset  $(L, \leq)$  is called a **lattice** if  $a \wedge b$  and  $a \vee b$  exist in  $L$  for all  $a, b \in L$ .

**EXAMPLE 3.2.35**

Let  $L$  be the set of all nonnegative real numbers. Then  $(L, \leq)$  is a poset, where  $\leq$  denotes the usual “less than or equal to” relation. Let  $a, b \in L$ . Now  $\max\{a, b\} \in L$  and  $\min\{a, b\} \in L$ . It is easy to see that  $\max\{a, b\}$  is the lub of  $\{a, b\}$  and  $\min\{a, b\}$  is the glb of  $\{a, b\}$ . For example,  $\max\{2, 6\} = 6 = 2 \vee 6$  and  $\min\{2, 6\} = 2 = 2 \wedge 6$ . Hence,  $(L, \leq)$  is a lattice.

**EXAMPLE 3.2.36**

Let  $S$  be a set. Then  $(\mathcal{P}(S), \subseteq)$  is a poset, where  $\subseteq$  is the set inclusion relation. For  $A, B \in \mathcal{P}(S)$ , we can show that  $A \vee B = A \cup B$  and  $A \wedge B = A \cap B$ . Hence,  $(\mathcal{P}(S), \subseteq)$  is a lattice.

In the following theorem, we collect several useful properties of lattices.

**Theorem 3.2.37:** Let  $(L, \leq)$  be a lattice and  $a, b, c \in L$ . Then

- (L1)  $a \vee b = b \vee a$ ,  $a \wedge b = b \wedge a$  (commutative laws),
- (L2)  $a \vee (b \vee c) = (a \vee b) \vee c$ ,  $a \wedge (b \wedge c) = (a \wedge b) \wedge c$  (associative laws),
- (L3)  $a \vee a = a$ ,  $a \wedge a = a$  (idempotent laws),
- (L4)  $a \vee (a \wedge b) = a$ ,  $a \wedge (a \vee b) = a$  (absorption laws).

**Proof:** We only prove (L1) and (L4) and leave others as exercises.

(L1):  $a \vee b = \text{lub of } \{a, b\} = \text{lub of } \{b, a\} = b \vee a$ . Note that the proof follows from the fact that the set  $\{a, b\}$  is the same as the set  $\{b, a\}$ .

(L4): Now,  $a \leq a$  and  $a \wedge b \leq a$ . Hence,  $a$  is an upper bound of  $\{a, a \wedge b\}$ . Thus, by the definition of least upper bound,  $a \vee (a \wedge b) \leq a$ . Because  $a \vee (a \wedge b)$  is the lub of  $\{a, a \wedge b\}$ , we have  $a \leq a \vee (a \wedge b)$ . Hence,  $a = a \vee (a \wedge b)$  because  $\leq$  is antisymmetric. ■

The proof of the following result is left as an exercise.

**Theorem 3.2.38:** Let  $(S, \leq)$  be a poset and  $a, b \in S$ . Then the following conditions are equivalent.

- (i)  $a \leq b$
- (ii)  $a \vee b = b$
- (iii)  $a \wedge b = a$

**DEFINITION 3.2.39** ► A lattice  $(L, \leq)$  is called **distributive** if it satisfies

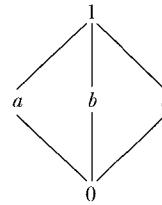
$$(D1) \quad a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$$

for all  $a, b, c \in L$ .

The lattices defined in Examples 3.2.35 and 3.2.36 are distributive lattices.

**EXAMPLE 3.2.40**

Consider the lattice given in Figure 3.25.



**FIGURE 3.25**  
Nondistributive lattice

Because  $a \wedge (b \vee c) = a \wedge 1 = a \neq 0 = 0 \vee 0 = (a \wedge b) \vee (a \wedge c)$ , this is not a distributive lattice.

**Theorem 3.2.41:** A lattice  $(L, \leq)$  is distributive if and only if

$$(D2) \quad a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$$

for all  $a, b, c \in L$ .

**Proof:** Suppose  $(L, \leq)$  is distributive. Let  $a, b, c \in L$ . Then

$$\begin{aligned} (a \vee b) \wedge (a \vee c) &= ((a \vee b) \wedge a) \vee ((a \vee b) \wedge c) && \text{by D1} \\ &= (a \wedge (a \vee b)) \vee ((a \vee b) \wedge c) && \text{by L1} \\ &= a \vee ((a \vee b) \wedge c) && \text{by L4} \\ &= a \vee (c \wedge (a \vee b)) && \text{by L1} \\ &= a \vee ((c \wedge a) \vee (c \wedge b)) && \text{by D1} \\ &= (a \vee (c \wedge a)) \vee (c \wedge b) && \text{by L2} \\ &= (a \vee (c \wedge a)) \vee (b \wedge c) && \text{by L1} \\ &= a \vee (b \wedge c) && \text{by L4.} \end{aligned}$$

Hence,  $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$ . Similarly,  $D2 \Rightarrow D1$ . ■

**Theorem 3.2.42:** In a distributive lattice  $(L, \leq)$ ,

$$a \wedge b = a \wedge c \quad \text{and} \quad a \vee b = a \vee c \quad \text{imply that} \quad b = c$$

for all  $a, b, c \in L$ .

**Proof:** Now,  $b = b \wedge (a \vee b) = b \wedge (a \vee c) = (b \wedge a) \vee (b \wedge c) = (a \wedge c) \vee (b \wedge c) = (c \wedge a) \vee (c \wedge b) = c \wedge (a \vee b) = c \wedge (a \vee c) = c$ . ■

**REMARK 3.2.43** ▶ Note that a poset  $(L, \leq)$  may not contain a greatest element, but from the antisymmetric property of the relation  $\leq$ , it can be shown that if there exists a greatest

element in a poset, then it is unique. Similarly, a poset may contain at most one least element.

We denote the greatest element of a poset, if it exists, by 1 and the least element, if it exists, by 0.

Notice that here 1 and 0 are merely notations; these are not necessarily the integers 1 and 0. For example, for the poset  $(\mathcal{P}(A), \leq)$ , where  $A$  is a set, the greatest element is  $A$  and the least element is  $\emptyset$ . Thus in notation we can write for the poset  $(\mathcal{P}(A), \leq)$ ,  $1 = A$  and  $0 = \emptyset$ .

---

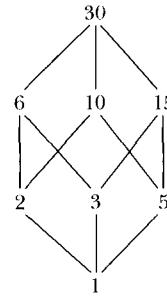
**DEFINITION 3.2.44** ▶ Let  $(L, \leq)$  be a lattice with 1 and 0. If  $a \in L$ , then an element  $b \in L$  is said to be a **complement** of  $a$  if  $a \vee b = 1$  and  $a \wedge b = 0$ .

**EXAMPLE 3.2.45**

Let  $D_{30}$  denote the set of all positive divisors of 30. Then

$$D_{30} = \{1, 2, 3, 5, 6, 10, 15, 30\}.$$

Now,  $(D_{30}, \leq)$  is a poset where  $a \leq b$  if and only if  $a$  divides  $b$  ( $\leq$  is the divisibility relation). Because 1 divides all elements of  $D_{30}$ , it follows that  $1 \leq m$ , for all  $m \in D_{30}$ . Therefore, 1 is the least element of this poset. Again, every member of  $D_{30}$  divides 30. Hence,  $m \leq 30$  for all  $m \in D_{30}$ . This shows that 30 is the greatest element of this poset. The Hasse diagram of  $D_{30}$  is given in Figure 3.26.



**FIGURE 3.26**  
Hasse diagram of  $D_{30}$

Let  $a, b \in D_{30}$ . Let  $d = \gcd\{a, b\}$  and  $m = \text{lcm}\{a, b\}$ . Now  $d \mid a$  and  $d \mid b$ . Hence,  $d \leq a$  and  $d \leq b$ . This shows that  $d$  is a lower bound of  $\{a, b\}$ . Let  $c \in D_{30}$  and  $c \leq a, c \leq b$ . Then,  $c \mid a$  and  $c \mid b$  and because  $d = \gcd\{a, b\}$ , it follows that  $c \mid d$ , so  $c \leq d$ . Thus, we find that  $d = \text{glb}\{a, b\}$ . Because all the positive divisors of  $a, b$  are also divisors of 30,  $d \in D_{30}$ , so  $d = a \wedge b$ . Similarly, we can show that

$$m \in D_{30}$$

and  $m = a \vee b$ .

Hence,  $(D_{30}, \leq)$  is a lattice with the least element integer 1 and the greatest element 30.

Now for any element  $a \in D_{30}$ ,  $\frac{30}{a} \in D_{30}$ . Using the properties of gcd and lcm, we can show that  $a \wedge \frac{30}{a} = 1$  and  $a \vee \frac{30}{a} = 30$ . For example,

$$10 \wedge \frac{30}{10} = \gcd\{10, 3\} = 1$$

and

$$10 \vee \frac{30}{10} = \text{lcm}\{10, 3\} = 30.$$

Hence, 3 is a complement of 10 in this lattice.

**REMARK 3.2.46** ▶ For any positive integer  $n$ , we can construct the lattice  $(D_n, \leq)$ , where  $D_n$  is the set of all positive divisors of  $n$ ,  $a \leq b$  if and only if  $a$  divides  $b$ . In the lattice,  $a \vee b = \text{lub}\{a, b\} = \text{lcm}\{a, b\}$  and  $a \wedge b = \text{glb}\{a, b\} = \text{gcd}\{a, b\}$  for any  $a, b \in D_n$ .

**Theorem 3.2.47:** In a distributive lattice  $(L, \leq)$  with 1 and 0, every element has at most one complement.

**Proof:** Let  $a \in L$ . Suppose  $b, c$  are two complements of  $a$  in  $L$ . Then  $a \vee b = 1$ ,  $a \wedge b = 0$ ,  $a \vee c = 1$ , and  $a \wedge c = 0$ . Hence,  $a \vee b = a \vee c$  and  $a \wedge b = a \wedge c$ . Then, by Theorem 3.2.42, it follows that  $b = c$ . ■

We now introduce the definition of *Boolean algebra*, named after the famous mathematician George Boole (1813–1864). Boole tried to formalize the process of logical reasoning using symbols instead of words. There are several equivalent definitions of Boolean algebra. Here we define Boolean algebra with the help of a lattice.

**DEFINITION 3.2.48** ▶ A distributive lattice  $(L, \leq)$  with the greatest element 1 and the least element 0 is called a **Boolean algebra** if every element has a complement in  $L$ .

From the above theorem, it follows that in a Boolean algebra  $(L, \leq)$  every element  $a \in L$  has a unique complement. The complement of  $a$  in  $L$  is denoted by  $a'$ .

#### EXAMPLE 3.2.49

Let  $P(S)$  be the set of all subsets of a nonempty set  $S$ . Then  $(P(S), \leq)$  is a poset, where  $A \leq B$  if and only if  $A \subseteq B$ , for all  $A, B \in P(S)$ . This poset is a lattice, where  $A \vee B = A \cup B$  and  $A \wedge B = A \cap B$ , for all  $A, B \in P(S)$ . The subset  $S$  is the greatest element 1, and the empty subset  $\emptyset$  is the least element 0. Also, for each  $A \in P(S)$ , the set complement of  $A$  in  $S$  is the complement of  $A$  in this lattice. Hence, the lattice  $(P(S), \leq)$  is a Boolean algebra. We will discuss more about Boolean algebra in Chapter 12.

## WORKED-OUT EXERCISES

**Exercise 1:** For each of the following relations, draw the digraph. Determine which are antisymmetric. Also determine which are partial orders.

- (a)  $(S, R)$ , where  $S = \{2, 6, 8, 10, 20\}$  and  $R$  denotes the divisibility relation
- (b)  $(S, R)$ , where  $S = \{1, 5, 6, 8, 10\}$  and  $R$  denotes the relation

$$R = \{(1, 1), (5, 5), (6, 6), (8, 8), (10, 10), (1, 5), (5, 6), (1, 6)\}$$

- (c)  $(S, R)$ , where  $S = \{1, 5, 6, 8, 10\}$  and  $R$  denotes the relation

$$R = \{(1, 1), (5, 5), (6, 6), (8, 8), (10, 10), (1, 6), (8, 6), (6, 1)\}$$

#### Solution:

- (a) Here  $R = \{(2, 2), (6, 6), (8, 8), (10, 10), (20, 20), (2, 6), (2, 8), (2, 10), (2, 20), (10, 20)\}$ . Because  $(a, a) \in R$ , for

all  $a \in S$ , the relation is reflexive. There are no distinct elements  $a, b \in S$  such that  $(a, b) \in R$  and  $(b, a) \in R$ . Hence, the relation is antisymmetric. The relation is also transitive, because if  $(a, b) \in R$  and  $(b, c) \in R$ , then  $(a, c) \in R$  for all  $a, b, c \in S$ . Hence, the relation  $R$  is a partial order. The digraph of this relation is shown in Figure 3.27(a).

- (b) Because  $(a, a) \in R$ , for all  $a \in S$ , the relation is reflexive. There are no distinct elements  $a, b \in S$  such that  $(a, b) \in R$  and  $(b, a) \in R$ . Hence, the relation is antisymmetric. The relation is also transitive, because if  $(a, b) \in R$  and  $(b, c) \in R$ , then  $(a, c) \in R$  for all  $a, b, c \in S$ . Hence, the relation  $R$  is a partial order. The digraph of this relation is shown in Figure 3.27(b).
- (c) Because  $(a, a) \in R$ , for all  $a \in S$ , the relation is reflexive. This relation is not antisymmetric, because  $(1, 6), (6, 1) \in R$ , but  $1 \neq 6$ . Hence, this relation is also not a partial order. The digraph of this relation is shown in Figure 3.27(c).

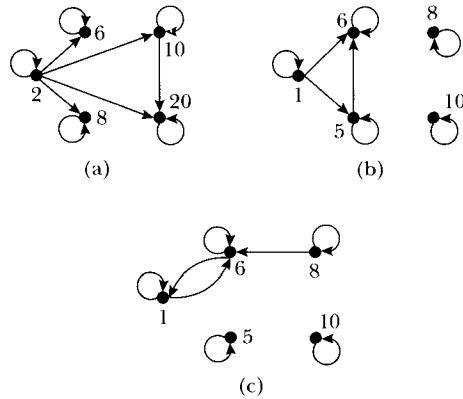
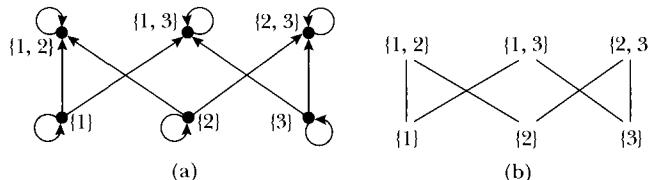


FIGURE 3.27 Digraphs

**Exercise 2:** Let  $S = \{1, 2, 3\}$  and  $T$  be the set of all proper nonempty subsets of  $S$ . In the poset  $(T, \leq)$ , where  $\leq$  is the set inclusion relation. Draw the digraph of the relation  $\leq$  and the Hasse diagram of the poset. Find the maximal and minimal elements.

**Solution:** The digraph is shown in the Figure 3.28(a), and the Hasse diagram is shown in Figure 3.28(b).

FIGURE 3.28 Digraph and Hasse diagram of  $S = \{1, 2, 3\}$ 

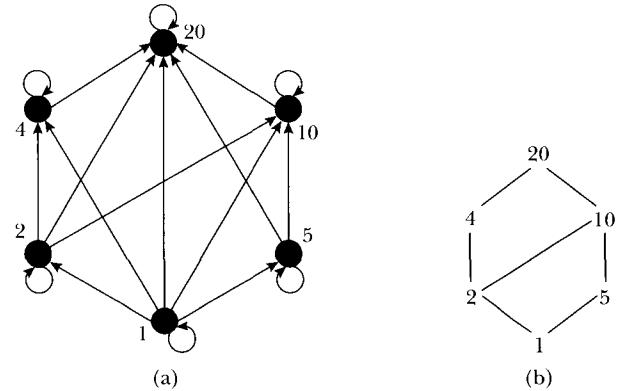
In this poset,  $\{1\}$ ,  $\{2\}$ , and  $\{3\}$  are minimal elements, and  $\{1, 2\}$ ,  $\{1, 3\}$ ,  $\{2, 3\}$  are maximal elements.

**Exercise 3:** Draw the digraph of the divisibility relation and the Hasse Diagram of the poset  $(D_{20}, \leq)$ .

**Solution:** We have

$$D_{20} = \{1, 2, 4, 5, 10, 20\}.$$

The digraph is shown in Figure 3.29(a), and the Hasse diagram is shown in Figure 3.29(b).

FIGURE 3.29 Digraph and Hasse diagram of  $D_{20}$ 

**Exercise 4:** Define a relation  $R$  on the set  $\mathbb{Z}$  of all integers by  $m R n$  if and only if  $m^2 = n^2$ . Is  $R$  a partial order?

**Solution:** Because  $m^2 = m^2$  for all  $m \in \mathbb{Z}$ , it follows that the relation is reflexive. Now  $(2)^2 = (-2)^2$  implies that  $2 R -2$  and  $-2 R 2$ , but  $-2 \neq 2$ . Hence,  $R$  is not antisymmetric, and therefore  $R$  is not a partial order.

**Exercise 5:** Let  $(S_1, \leq_1)$  and  $(S_2, \leq_2)$  be two posets, where  $S_1 = \{1, 2, 4\}$ ,  $S_2 = \{1, 2, 3, 6\}$ , and both the relations  $\leq_1, \leq_2$  are divisibility relations. With respect to lexicographic order on  $S_1 \times S_2$ , find all pairs  $(a, b) \in S_1 \times S_2$  such that  $(a, b) \leq (2, 3)$ .

**Solution:** Note that  $(a, b) \leq (c, d)$  if and only if  $a <_1 c$  or  $a = c$  and  $b \leq_2 d$ . We find those pairs  $(a, b) \in S_1 \times S_2$  such that  $(a, b) \leq (2, 3)$ . The pairs are  $(1, b) \in S_1 \times S_2$ ,  $b \in S_2$  and  $(2, b) \in S_1 \times S_2$  such that  $b$  divides 3. Hence, the pairs are  $(1, 1), (1, 2), (1, 3), (1, 6), (2, 1)$ , and  $(2, 3)$ .

**Exercise 6:** Consider the poset  $(S, \leq)$ , where  $S = \{2, 4, 3, 6, 12\}$  and the partial order is the divisibility relation. Find a linear order on  $S$  compatible with the given partial order.

**Solution:** 2 is a minimal element of  $S$ . Let  $a_1 = 2$  and  $S_1 = S - \{2\} = \{4, 3, 6, 12\}$ . Then  $S_1$  is also a poset under the divisibility relation. Also,  $S_1$  has a minimal element  $a_2 = 3$ . Let

$$S_2 = S - \{2, 3\} = \{4, 6, 12\}.$$

Now  $S_2$  has a minimal element  $a_3 = 4$ . Let

$$S_3 = S - \{2, 3, 4\} = \{6, 12\}.$$

$S_3$  has a minimal element  $a_4 = 6$ . Let

$$S_4 = S - \{2, 3, 4, 6\} = \{12\}.$$

Finally,  $a_5 = 12$  is a minimal element of  $\{12\}$ . We now de-

fine the partial order  $\leq_1$  on  $S$  by  $2 \leq_1 3 \leq_1 4 \leq_1 6 \leq_1 12$ . It follows that this is a linear order.

Notice that  $4 \leq 12$  because 4 divides 12. In the relation  $\leq_1$ , we have

$$4 \leq_1 6 \leq_1 12,$$

which implies that  $4 \leq_1 12$  by the property of transitivity. Similarly, we can verify that the compatibility holds for other elements. So it follows that the linear order  $\leq_1$  on  $S$  is compatible with the relation  $\leq$ .

**Exercise 7:** Show that every chain is a distributive lattice.

**Solution:** Let  $(L, \leq)$  be a chain and  $a, b, c \in L$ . Because  $L$  is a chain, either  $a \leq b$  or  $b \leq a$ . If  $a \leq b$ , then  $a \vee b = b$  and  $a \wedge b = a$ . If  $b \leq a$ , then  $a \vee b = a$  and  $a \wedge b = b$ . Hence, for any two elements  $a, b \in L$ ,  $a \wedge b$  and  $a \vee b$  exist in  $L$ . Suppose  $a \leq b$ .

**Case 1:**  $b \leq c$

Now  $a \wedge (b \vee c) = a \wedge c = a$  and  $(a \wedge b) \vee (a \wedge c) = a \vee a = a$ . Hence, we have

$$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c).$$

**Case 2:**  $c \leq b$

**Subcase 2a:**  $a \leq c$

In this case, we have  $a \leq c \leq b$ . Now  $a \wedge (b \vee c) = a \wedge b = a$  and  $(a \wedge b) \vee (a \wedge c) = a \vee c = a$ . Hence,

$$(a \wedge b) \vee (a \wedge c) = (a \wedge b) \vee (a \wedge c) = a.$$

Hence,

$$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c).$$

**Subcase 2b:**  $c \leq a$

In this case, we have  $c \leq a \leq b$ . Now  $a \wedge (b \vee c) = a \wedge b = a$  and  $(a \wedge b) \vee (a \wedge c) = a \vee c = a$ . Hence,

$$(a \wedge b) \vee (a \wedge c) = (a \wedge b) \vee (a \wedge c) = a.$$

Similarly, if  $b \leq a$ , then  $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ .

**Exercise 8:** In a lattice  $(L, \leq)$ , prove that  $(a \wedge b) \vee (a \wedge c) \leq a \wedge (b \vee (a \wedge c))$  for all  $a, b, c \in L$ .

**Solution:** Now  $a \wedge b \leq a$ ,  $a \wedge c \leq a$ . Therefore,  $(a \wedge b) \vee (a \wedge c) \leq a$ . Again,  $a \wedge b \leq b$  implies

$$(a \wedge b) \vee (a \wedge c) \leq b \vee (a \wedge c).$$

Thus, we find that  $(a \wedge b) \vee (a \wedge c)$  is a lower bound of  $\{a, b \vee (a \wedge c)\}$ . But  $a \wedge (b \vee (a \wedge c))$  is the glb of  $\{a, b \vee (a \wedge c)\}$ . Hence,

$$(a \wedge b) \vee (a \wedge c) \leq a \wedge (b \vee (a \wedge c)).$$

**Exercise 9:** Consider the lattice  $(D_{20}, \leq)$ , where  $\leq$  denotes the divisibility relation. Find  $4 \wedge (5 \vee 10)$  and  $(2 \vee (2 \wedge 5)) \vee 4$ . Is this lattice a Boolean algebra?

**Solution:**  $D_{20} = \{1, 2, 4, 5, 10, 20\}$ . Now

$$\begin{aligned} 4 \wedge (5 \vee 10) &= 4 \wedge 10 && \text{because } 5 \vee 10 = \text{lcm}\{5, 10\} = 10 \\ &= 2 && \text{because } 4 \wedge 10 = \text{gcd}\{4, 10\} = 2. \end{aligned}$$

Also,

$$\begin{aligned} (2 \vee (2 \wedge 5)) \vee 4 &= (2 \vee 1) \vee 4 && \text{because } 2 \wedge 5 = \text{gcd}\{2, 5\} = 1 \\ &= 2 \vee 4 && \text{because } 2 \vee 1 = \text{lcm}\{2, 1\} = 2 \\ &= 4 && \text{because } 2 \vee 4 = \text{lcm}\{2, 4\} = 4. \end{aligned}$$

The Hasse diagram of  $D_{20}$  is shown in Figure 3.29. In this lattice, the least element is 1 and the greatest element is 20. Now,

$$\begin{aligned} 2 \wedge 1 &= 1, & 2 \vee 1 &\neq 20, & 2 \wedge 4 &\neq 1, \\ 2 \wedge 5 &= 1, & 2 \vee 5 &\neq 20, & 2 \wedge 10 &\neq 1. \end{aligned}$$

Hence, 2 has no complement in  $D_{20}$ . Therefore, this is not a Boolean algebra.

**Exercise 10:** Consider the lattice  $(S, \leq)$ , where  $S = \{1, 2, 4, 5, 8, 9\}$  and  $\leq$  denotes the usual “less than or equality” relation. Find  $4 \wedge (5 \vee 9)$  and  $(2 \vee (2 \wedge 8)) \vee 4$ . Is this lattice a Boolean algebra?

**Solution:** In this lattice,  $a \vee b = \max\{a, b\}$  and  $a \wedge b = \min\{a, b\}$ . The Hasse diagram of  $S$  is given in Figure 3.30.



FIGURE 3.30

Hasse diagram of  $S$

Now,

$$\begin{aligned} (2 \vee (2 \wedge 8)) \vee 4 &= (2 \vee \min\{2, 8\}) \vee 4 \\ 4 \wedge (5 \vee 9) &= (2 \vee 2) \vee 4 \\ &= 2 \wedge 9 && \text{and} \\ &= \min\{4, 9\} && = \max\{2, 2\} \vee 4 \\ &= 4 && = \max\{2, 4\} \\ & && = 4. \end{aligned}$$

This is a chain. Hence,  $S$  is a distributive lattice with the greatest element 9 and the least element 1. In this lattice, suppose there exists an element  $b$  such that  $2 \vee b = 9$  and  $2 \wedge b = 1$ . Then  $\max\{2, b\} = 9$  implies that  $b = 9$ . On the other hand,  $\min\{2, b\} = 1$  implies that  $b = 1$ . Thus, we find that 2 has no complement in this lattice. Hence,  $S$  is not a Boolean algebra.

## SECTION REVIEW

---

### Key Terms

antisymmetric	lexicographic order	topological ordering
partial order	dictionary order	upper bound
partially ordered set	closed	least upper bound (lub)
poset	covers	lower bound
dual	Hasse diagram	greatest lower bound (glb)
comparable	minimal element	lattice
linearly ordered set	maximal element	distributive
totally ordered set	greatest element	complement
chain	least element	Boolean algebra
product partial order	compatible	

### Some Key Definitions

1. A relation  $R$  on a set  $S$  is called antisymmetric if for all  $a, b \in S$ ,  $a R b$  and  $b R a$ , then  $a = b$ .
2. A relation  $R$  on a set  $A$  is called a partial order on  $A$  if  $R$  is reflexive, antisymmetric, and transitive.
3. A set  $A$  together with a partial order relation  $R$  is called a partially ordered set, or simply poset, and we denote this poset by  $(A, R)$ .
4. Let  $(S, \leq)$  be a poset and  $a, b \in S$ . If either  $a \leq b$  or  $b \leq a$ , then we say that  $a$  and  $b$  are comparable. The poset  $(S, \leq)$  is called a linearly ordered set, or a totally ordered set, or a chain, if for all  $a, b \in S$  either  $a \leq b$  or  $b \leq a$ .
5. Let  $(S, \leq)$  be poset. An element  $a \in S$  is called
  - (i) a minimal element if there is no element  $b \in S$  such that  $b < a$ ,
  - (ii) a maximal element if there is no element  $b \in S$  such that  $a < b$ ,
  - (iii) a greatest element if  $b \leq a$  for all  $b \in S$ ,
  - (iv) a least element if  $a \leq b$  for all  $b \in S$ .
6. Let  $(S, \leq)$  be a poset and let  $\{a, b\}$  be a subset of  $S$ . An element  $c \in S$  is called an upper bound of  $\{a, b\}$  if  $a \leq c$  and  $b \leq c$ .
7. An element  $d \in S$  is called a least upper bound (lub) of  $\{a, b\}$  if
  - (i)  $d$  is an upper bound of  $\{a, b\}$ ; and
  - (ii) if  $c \in S$  is an upper bound of  $\{a, b\}$ , then  $d \leq c$ .
8. Let  $(S, \leq)$  be a poset and let  $\{a, b\}$  be a subset of  $S$ . An element  $c \in S$  is called a lower bound of  $\{a, b\}$  if  $c \leq a$  and  $c \leq b$ . An element  $d \in S$  is called a greatest lower bound (glb) of  $\{a, b\}$  if
  - (i)  $d$  is a lower bound of  $\{a, b\}$ ; and
  - (ii) if  $c \in S$  is a lower bound of  $\{a, b\}$ , then  $c \leq d$ .
9. A poset  $(L, \leq)$  is called a lattice if  $a \wedge b$  and  $a \vee b$  exist in  $L$  for all  $a, b \in L$ .

10. A lattice  $(L, \leq)$  is called distributive if it satisfies

$$(D1) \quad a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c) \text{ for all } a, b, c \in L.$$

## Some Key Results

1. Let  $(S, \leq)$  be a poset such that  $S$  is a finite nonempty set. Then this poset has a minimal element.
2. In a poset  $(S, \leq)$ , if a subset  $\{a, b\}$  of  $S$  has a lub, then this lub is unique.
3. In a poset  $(S, \leq)$ , if a subset  $\{a, b\}$  of  $S$  has a glb, then this glb is unique.
4. Let  $(L, \leq)$  be a lattice and  $a, b, c \in L$ . Then
  - (L1)  $a \vee b = b \vee a, a \wedge b = b \wedge a,$
  - (L2)  $a \vee (b \vee c) = (a \vee b) \vee c, a \wedge (b \wedge c) = (a \wedge b) \wedge c,$
  - (L3)  $a \vee a = a, a \wedge a = a,$
  - (L4)  $a \vee (a \wedge b) = a, a \wedge (a \vee b) = a.$

## EXERCISES

---

1. For each of the following relations draw the digraph. Determine which relations are antisymmetric.
  - a.  $(S, R)$ , where  $S = \{5, 6, 8, 10, 20\}$  and  $R$  denotes the divisibility relation
  - b.  $(S, R)$ , where  $S = \{1, 5, 6, 8, 10\}$  and  $R$  denotes the relation
$$R = \{(1, 1), (5, 5), (6, 6), (8, 8), (10, 10), (1, 5), (5, 6), (1, 6)\}$$
- c.  $(S, R)$ , where  $S = \{1, 5, 6, 8, 10\}$  and  $R$  denotes the relation
$$R = \{(1, 1), (5, 5), (6, 6), (8, 8), (10, 10), (1, 5), (8, 6), (1, 6)\}$$
2. Determine which of the following relations are antisymmetric.
  - a.  $(S, R)$ , where  $S = \mathbb{Z}$ ,  $a R b$  if and only if  $a = b^n$  for some positive integer  $n$
  - b.  $(S, R)$ , where  $S = \mathbb{Z}$ ,  $a R b$  if and only if  $a = nb$  for some positive integer  $n$
3. Define a relation  $R$  on the set  $\mathbb{Z}$  of all integers by  $m R n$  if and only if  $|m| = |n|$ . Is  $R$  a partial order?
4. Define a relation  $R$  on the set  $\mathbb{Z}$  of all integers by  $m R n$  if and only if  $mn \geq 0$ . Is  $R$  a partial order?
5. Define a relation  $R$  on the set  $\mathbb{Z} \times \mathbb{Z}$  by  $(m, t) R (n, r)$  if and only if  $m = n$  and  $t - r \geq 0$ . Is  $R$  antisymmetric?
6. Draw the Hasse diagram for each of the following posets.
  - a.  $(\{a \mid a \text{ is a positive divisor of } 20\}, \leq)$ , where  $\leq$  denotes the divisibility relation
  - b.  $(\mathbb{N}, \leq)$ , where  $\leq$  denotes the natural order relation
  - c.  $(\mathcal{P}(S), \leq)$ ,  $S = \{1, 2, 3, 4\}$ , where  $\leq$  denotes the set inclusion relation
- d.  $(\mathcal{P}(S) - \{\emptyset\}, \leq)$ ,  $S = \{1, 2, 3\}$ , where  $\leq$  denotes the set inclusion relation
7. Let  $S = \{1, 2, 3\}$  and
$$A = \{\{2\}, \{3\}, \{2, 3\}, \{1, 3\}, S\}.$$

Draw the digraph of the partial order  $\leq$  defined by the set inclusion  $\subseteq$  on  $A$ . Also draw the Hasse diagram of the poset  $(A, \leq)$ . Find all maximal and minimal elements of this poset.

8. Let  $S = \{1, 2, 3, 6, 9, 18\}$ . Consider the partial order  $\leq$  defined by the divisibility relation on  $S$ . Draw the digraph of this partial order and the Hasse diagram of the poset  $(S, \leq)$ . Find all maximal and minimal elements of this poset.
9. Give an example of a relation  $R$  that is antisymmetric, but not reflexive.
10. Give an example of a poset  $(P, \leq)$  such that  $P$  has two elements  $a$  and  $b$  for which  $a \wedge b$  does not exist.
11. Show that  $(\mathbb{R}, \leq)$  is not a poset, where  $a \leq b$  means that  $b = ad$  for some  $d \in \mathbb{R}$ .
12. Let  $A$  be the set of first 12 positive integers. Define a relation  $R$  on  $A$  by  $x R y$  if and only if  $x$  is a divisor of  $y$  for all  $x, y \in A$ . Prove that  $(A, R)$  is a poset.
13. Define the relation  $\geq$  on the set  $\mathbb{C}$  of complex numbers by  $a + ib \geq c + id$  if and only if  $a \geq c$  and  $b \geq d$  for all  $a, b, c, d \in \mathbb{R}$ . Prove that  $\geq$  is a partial order on the set of complex numbers  $\mathbb{C}$ . Is it a linear order on  $\mathbb{C}$ ? Justify your answer.
14. Let  $\leq_1$  and  $\leq_2$  be two partial orders on a set  $S$ . Is  $\leq_1 \cap \leq_2$  a partial order on  $S$ ?
15. Let  $(A, \leq)$  and  $(B, \leq)$  be two posets. Prove that  $(A \times B, \leq)$  is a poset, where  $(a, b) \leq (c, d)$  if and only if  $a \leq c$  and  $b \leq d$ .

16. Let  $(A, \leq)$  and  $(B, \leq)$  be two posets. Prove that  $(A \times B, \preceq)$  is a poset, where  $\preceq$  denotes lexicographic order. If  $(A, \leq)$  and  $(B, \leq)$  are linearly ordered sets, then prove that  $(A \times B, \preceq)$  is a linearly ordered set.
17. Let  $(S_1, \leq_1)$  and  $(S_2, \leq_2)$  be two posets, where  $S_1 = \{1, 2, 4\}$ ,  $S_2 = \{1, 2, 3, 6\}$ , and both the relations  $\leq_1, \leq_2$  are divisibility relations. With respect to lexicographic order on  $S_1 \times S_2$ , find all pairs  $(a, b) \in S_1 \times S_2$  such that  $(a, b) \preceq (4, 2)$ .
18. Let  $(S_1, \leq_1)$  and  $(S_2, \leq_2)$  be two posets, where  $S_1 = \{1, 2, 4\}$ ,  $S_2 = \{1, 2, 3, 6\}$ , and both the relations  $\leq_1, \leq_2$  are the usual “less than or equal to” relations. With respect to lexicographic order on  $S_1 \times S_2$ , find all pairs  $(a, b) \in S_1 \times S_2$  such that  $(a, b) \preceq (2, 3)$ .
19. Consider the poset  $(\{2, 3, 6, 12\}, |)$ , where the partial order  $|$  is the divisibility relation. Find a linear order  $\leq$  on  $\{2, 3, 6, 12\}$  such that  $\leq$  is compatible with  $|$ .
20. Find a compatible linear order for the poset  $(\{3, 6, 7, 12, 14, 18, 21\}, |)$ , where the partial order  $|$  is the divisibility relation.
21. Arrange the following words according to dictionary order.
- real, relation, relative, reliable, reason, invitation, invite
  - poset, pond, posses, party, partial, orange, organize
22. Consider the poset  $(S, \leq)$ , where  $S = \{m \mid m \text{ is a positive divisor of } 48 \text{ and } 1 < m < 48\}$  and the relation  $\leq$  is the divisibility relation.
- Find all minimal and maximal elements.
  - Find all lower bounds of  $\{12, 16\}$ .
  - Find all upper bounds of  $\{12, 16\}$ .
  - Find the glb and lub of  $\{12, 16\}$ .
  - Does this poset contain the least element and the greatest element?
  - Is this poset a lattice?
23. Consider the lattice  $(S, \leq)$ , where  $S = \{3, 4, 5, 8, 9, 10\}$   $\leq$  denotes the usual “less than or equal to” relation. Find  $4 \wedge (5 \vee 9)$  and  $(3 \vee (3 \wedge 8)) \vee 4$ . Is this lattice a Boolean algebra?
24. Let  $S = \mathbb{Z} \times \mathbb{Z} \times \mathbb{Z}$ . Consider the usual order relation on  $\mathbb{Z}$ . In the poset  $(\mathbb{Z} \times \mathbb{Z} \times \mathbb{Z}, \preceq)$ , arrange the following elements in increasing order according to the lexicographic order  $\preceq$ .
- $(1, 0, 5), (0, 9, 0), (3, -3, 5), (-4, 9, 2), (0, 0, 2), (54, 123, -312)$
  - $(1, 0, 0), (1, 0, 1), (0, 1, 0), (0, 0, 1), (1, 1, 0), (0, 1, 1)$
25. Justify by examples.
- In a poset  $(S, \leq)$ , a subset  $\{a, b\}$  of  $S$  may not have an upper bound.
  - In a poset  $(S, \leq)$ , a subset  $\{a, b\}$  of  $S$  may have more than one upper bound.
  - In a poset  $(S, \leq)$ , a subset  $\{a, b\}$  of  $S$  may not have a lub.
26. Justify by examples.
- In a poset  $(S, \leq)$ , a subset  $\{a, b\}$  of  $S$  may not have a lower bound.

- b. In a poset  $(S, \leq)$ , a subset  $\{a, b\}$  of  $S$  may have more than one lower bound.
- c. In a poset  $(S, \leq)$ , a subset  $\{a, b\}$  of  $S$  may not have a glb.

27. Which of the posets in Figure 3.31 are lattices?

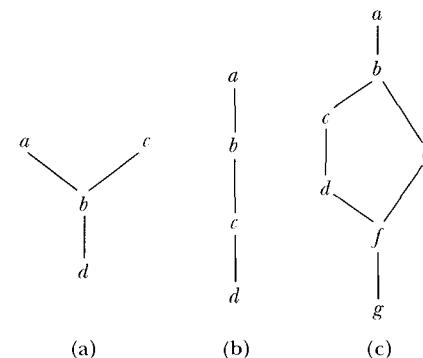


FIGURE 3.31 Posets

28. Let  $D_{40}$  denote the set of all positive divisors of 40. Consider the lattice

$$(D_{40}, \leq),$$

where  $\leq$  denotes the divisibility relation. Find  $4 \wedge (8 \vee 10)$  and  $(2 \vee (2 \wedge 8)) \vee 20$ .

29. Let  $D_{42}$  denote the set of all positive divisors of 42. Consider the lattice

$$(D_{42}, \leq),$$

where  $\leq$  denotes the divisibility relation. Find  $4 \wedge (6 \vee 14)$  and  $(2 \vee (2 \wedge 8)) \vee 21$ .

30. In a lattice  $(L, \leq)$ , prove the following. For all  $a, b, c \in L$ ,

- $a \vee (b \wedge c) \leq (a \vee b) \wedge (a \vee c)$
- $(a \wedge b) \vee (a \wedge c) \leq a \wedge (b \vee c)$
- $(a \wedge b) \vee (b \wedge c) \vee (c \wedge a) \leq (a \vee b) \wedge (b \vee c) \wedge (c \vee a)$

- d. if  $a \leq c$ , then  $a \vee (b \wedge c) \leq (a \vee b) \wedge c$

31. A lattice  $(L, \leq)$  is called a **modular lattice** if for all  $a, b, c \in L$ ,  $a \leq c$  implies

$$a \vee (b \wedge c) = (a \vee b) \wedge c.$$

In a modular lattice  $(L, \leq)$ , prove that for all  $a, b, c \in L$ ,  $a \leq c$ ,  $a \wedge b = c \wedge b$ , and  $a \vee b = c \vee b$  imply that  $a = c$ .

32. Prove that every distributive lattice is a modular lattice.

33. Give an example of a lattice that is modular but not distributive.

34. Prove that a lattice  $(L, \leq)$  is distributive if and only if for all  $a, b, c \in L$ ,

$$(a \wedge b) \vee (b \wedge c) \vee (c \wedge a) = (a \vee b) \wedge (b \vee c) \wedge (c \vee a).$$

35. Determine whether the following assertions are true or false. If true, prove the result; if false, give a counterexample.

  - The relation  $R = \{(a, b) \in \mathbb{Z} \times \mathbb{Z} \mid |a - b| \leq 1\}$  is a partial order on  $\mathbb{Z}$ .
  - The relation  $R = \{(a, b) \in \mathbb{Z} \times \mathbb{Z} \mid |a| \leq |b|\}$  is a partial order on  $\mathbb{Z}$ .
  - The relation  $R = \{(a, b) \in S \times S \mid a \text{ divides } b \text{ in } \mathbb{N}\}$  is a partial order on  $S = \{1, 2, 3, 4, 6, 12\}$ .

### 3.3 APPLICATION: RELATIONAL DATABASE

In the preceding sections, we discussed relations in detail. We now describe an application of relations to database theory.

In the business world, generating relevant information in a timely manner is crucial to the success of a business. For example, stockbrokers rely on stock-market reports to buy, sell, and hold stocks. Students rely on mid-semester grade reports to decide which subjects need more study. To produce relevant information efficiently, we need quick access to data (raw facts) from which to generate appropriate information. Therefore, data collection, storage, and retrieval are some of the most important activities of an organization.

Managing data efficiently requires the use of a computer database, especially if we are dealing with large quantities of data. A **database** is a shared and integrated computer structure that stores

- end-user data; i.e., raw facts that are of interest to the end user;
  - **metadata**, i.e., data about data through which data are integrated.

We can think of a database as a well-organized electronic file cabinet whose contents are managed by software known as a **database management system**; that is, a collection of programs to manage the data and control the accessibility of the data.

Consider the following table which lists various facts associated with students.

**Table:** Student

ID	Name	Rank	Major	EmpID
3456	Peter	Sr	CSC	745
9324	Ashley	Soph	Math	848
8723	Randy	Jr	CSC	745
2367	Sheila	Sr	Arts	467
8236	Anita	Fr	Drama	848
7623	Jackson	Sr	Math	848

The column headings are ID, Name, Rank, Major, and EmpID. Each row in the table describes a student's ID, name, college standing, major, and the ID of the advisor. The rows, among other things, specify which ID is associated with, i.e., related to, which student and the advisor of each student. In other words, we can think of each row as describing the relationships between ID, name, rank, major, and advisor's ID. Let us think of column headings as sets consisting of elements

in those columns. To be specific, consider the following sets.

$$\text{ID} = \{3456, 9324, 8723, 2367, 8236, 7623\}$$

$$\text{Name} = \{\text{Peter}, \text{Ashley}, \text{Randy}, \text{Sheila}, \text{Anita}, \text{Jackson}\}$$

$$\text{Rank} = \{\text{Fr}, \text{Scph}, \text{Jr}, \text{Sr}\}$$

$$\text{Major} = \{\text{CSC}, \text{Math}, \text{Arts}, \text{Drama}\}$$

$$\text{EmpID} = \{745, 848, 467\}$$

The entry in each row is a 5-tuple of the form

$$(\text{ID}, \text{name}, \text{rank}, \text{major}, \text{employeeID}).$$

For example, if  $r_1$  denotes the elements of the first row, then

$$r_1 = (3456, \text{Peter}, \text{Sr}, \text{CSC}, 745).$$

Following this convention, the table *Student* can be described as a set as follows.

$$\begin{aligned} \text{Student} = & \{(3456, \text{Peter}, \text{Sr}, \text{CSC}, 745), (9324, \text{Ashley}, \text{Soph}, \text{Math}, 848), \\ & (8723, \text{Randy}, \text{Jr}, \text{CSC}, 745), (2367, \text{Sheila}, \text{Sr}, \text{Arts}, 467), \\ & (8236, \text{Anita}, \text{Fr}, \text{Drama}, 848), (7623, \text{Jackson}, \text{Sr}, \text{Math}, 848)\} \end{aligned}$$

From this it follows that

$$\text{Student} \subseteq \text{ID} \times \text{Name} \times \text{Rank} \times \text{Major} \times \text{EmpID};$$

that is, we can think of *Student* as a 5-ary relation on the sets *ID*, *Name*, *Rank*, *Major*, *EmpID*.

The database management system based on the theory of relations was developed by E. F. Codd, at IBM, in 1970. His work was considered a major breakthrough for both users and designers. However, because of the lack of the computer power, at that time, the simplicity of the design caused computer overhead. Over the years not only has computer power grown exponentially, but the costs have diminished rapidly. Today's microcomputers, which are more powerful than their mainframe ancestors, can run sophisticated relational database systems, such as XDB, ORACLE, Ingress, and other mainframe relational software.

In a **relational database** system, tables are considered as relations. That is, a table is an ***n*-ary relation**, where *n* is the number of columns in the tables. In this section, when we say table or *n*-ary relation, we mean the same thing.

The headings of the columns of a table are called **attributes**, or **fields**, and each row is called a **record**. For example, the fields of the *Student* table are *ID*, *Name*, *Rank*, *Major*, and *EmpID*. The **domain** of a field is the set of all (possible) elements in that column.



**Edgar Codd**  
(b. 1923)

Codd was born in Portland, UK. He served as an active captain in the Royal Air Force (1942–1945). After the war, he moved to Tennessee and became an instruc-

### Historical Notes

tor of mathematics at the University of Tennessee. Codd received his Ph.D. in computer science from the University of Michigan in 1963.

Codd's groundbreaking work was accomplished while he was working at the IBM research lab (1949–1984). In 1970, he revolutionized the thinking behind computer databases by develop-

ing the idea of a relational database. A relational database allows for the definition of data structures, storage and retrieval operations, and integrity constraints. The database tables are organized by key fields and no two rows are identical. Even today, this database structure remains the one most widely used.

Consider the *Student* table. In this table, we might have more than one student with the same name, however, the ID of each student will be different. In other words, each entry in the ID column uniquely identifies the row containing that ID. Such a field is called a *primary key*.

Sometimes, a primary key may consist of more than one field. For example, suppose that we have a table, say *Courses*, with fields *StudentID*, *CourseID*, and *Grade*, that stores the courses a student is taking or has taken and the grade for the course. Assume that if a student retakes a course, only the most recent information about the course is kept. That is, the new grade replaces the old grade. Now, because a student may take more than one course, and a course may be taken by more than one student, a student's ID will appear more than once in the *StudentID* column. Similarly, in the *CourseID* column, a course's ID will appear more than once. Moreover, in the *Grade* column, a grade will appear more than once. It follows that no entry in a column can uniquely identify its row. However, it can be checked that the tuple (*StudentID*, *CourseID*) can uniquely identify each row. Thus, in this scenario, a primary key consists of two fields.

Because each table is an  $n$ -ary relation and an  $n$ -ary relation is a set, we can take the union, intersection, and difference of two tables with the same fields.

To minimize data redundancy, information is spread over several tables. Therefore, information is usually retrieved from more than one table. For example, to find the names of students and the names of their advisor, we need to choose students' names from the *Student* table and advisors' names from the *Professor* table.

The **join** operation allows us to link more than one table to retrieve relevant information. Tables are joined using common field(s). In fact, the join operation is the real power behind the relational database. Let us now explain how the join operation works.

Suppose that we have the following table.

**Table:** Professor

EmpID	Name	Phone	Office
745	Jacob	X4832	G340
848	Wendy	X9823	S290
467	William	X7823	G348

We can join *Student* and *Professor* tables using the common field *EmpID*. (In reality, the common field need not have the same name. However, the columns must contain similar data. For example, in the *Student* table, we could have named the column *EmpID* as *AdvisorID*.)

A join is a three-step process.

**Step 1.** We create a Cartesian product of the table. For example, for the tables *Student* and *Professor*, the following data are generated.

ID	Name	Rank	Major	EmpID	EmpID	Name	Phone	Office
3456	Peter	Sr	CSC	745	745	Jacob	X4832	G340
9324	Ashley	Soph	Math	848	745	Jacob	X4832	G340
8723	Randy	Jr	CSC	745	745	Jacob	X4832	G340
2367	Sheila	Sr	Arts	467	745	Jacob	X4832	G340

ID	Name	Rank	Major	EmpID	EmpID	Name	Phone	Office
8236	Anita	Fr	Drama	848	745	Jacob	X4832	G340
7623	Jackson	Sr	Math	848	745	Jacob	X4832	G340
3456	Peter	Sr	CSC	745	848	Wendy	X9823	S290
9324	Ashley	Soph	Math	848	848	Wendy	X9823	S290
8723	Randy	Jr	CSC	745	848	Wendy	X9823	S290
2367	Sheila	Sr	Arts	467	848	Wendy	X9823	S290
8236	Anita	Fr	Drama	848	848	Wendy	X9823	S290
7623	Jackson	Sr	Math	848	848	Wendy	X9823	S290
3456	Peter	Sr	CSC	745	467	William	X7823	G348
9324	Ashley	Soph	Math	848	467	William	X7823	G348
8723	Randy	Jr	CSC	745	467	William	X7823	G348
2367	Sheila	Sr	Arts	467	467	William	X7823	G348
8236	Anita	Fr	Drama	848	467	William	X7823	G348
7623	Jackson	Sr	Math	848	467	William	X7823	G348

**Step 2.** We perform a select on the data generated in step (1) to select only those rows that have the same value in both of the EmpID columns. Therefore, we get

ID	Name	Rank	Major	EmpID	EmpID	Name	Phone	Office
3456	Peter	Sr	CSC	745	745	Jacob	X4832	G340
8723	Randy	Jr	CSC	745	745	Jacob	X4832	G340
9324	Ashley	Soph	Math	848	848	Wendy	X9823	S290
8236	Anita	Fr	Drama	848	848	Wendy	X9823	S290
7623	Jackson	Sr	Math	848	848	Wendy	X9823	S290
2367	Sheila	Sr	Arts	467	467	William	X7823	G348

**Step 3.** In this step the duplicated columns are deleted. Thus, we have

ID	Name	Rank	Major	EmpID	Name	Phone	Office
3456	Peter	Sr	CSC	745	Jacob	X4832	G340
8723	Randy	Jr	CSC	745	Jacob	X4832	G340
9324	Ashley	Scph	Math	848	Wendy	X9823	S290
8236	Anita	Fr	Drama	848	Wendy	X9823	S290
7623	Jackson	Sr	Math	848	Wendy	X9823	S290
2367	Sheila	Sr	Arts	467	William	X7823	G348

This information can be further filtered to select more specific data. For example, to select only the students' names, their advisors' names, and the office phone numbers, we can eliminate the other columns.

## Structured Query Language (SQL)

Information from a database is retrieved via a query. A **query** is a request to the database for some information. A relational database management system provides a standard language, called **structured query language (SQL)**. SQL contains about 30 commands to query the database. All major commercially available relational database management systems provide extensions to the basic SQL commands.

An SQL contains commands to **create** tables, **insert** data into tables, **update** tables, **delete** tables, and so on. Once the tables are created, we can use commands to manipulate data into those tables. The most commonly used command for this purpose is the **select** command. The select command allows the user to do the following:

- Specify what information is to be retrieved and from which tables.
- Specify conditions to retrieve the data in a specific form.
- Specify how the retrieved data are to be displayed.

Next, we give a few examples that illustrate how the select command works. (To find a detailed description of this command, refer to the reference section in the appendices.)

The general syntax to use the select command is

```
select column1, column2, ..., columnN
      from table1, table2, ..., tableM
      where where_clause;
```

The where statement in the third line is optional.

To select all the columns of a table, we can use the command

```
select *
      from table
      where where_clause;
```

The following select statement outputs all the names of the students from the *Student* table.

```
select Name
      from Student;
```

The output of this command would be:

Name
Peter
Ashley
Randy
Sheila
Anita
Jackson

**EXAMPLE 3.3.1**

Consider the following select statement.

```
select Student.Name StudentName, Professor.Name AdvisorName
  from Student, Professor
 where Student.EmpID = Professor.EmpID;
```

The `where_clause` statement specifies to join the *Student* and *Professor* tables at the field `EmpID` in both the tables and to output the students' names and their advisors. (Notice that in this query, we have told the system when the data are output to name the column containing students' names `StudentName` and the column containing advisors' names `AdvisorName`.) Because the join is performed on the `EmpID` and both tables have the same column name, in the `where_clause`, `Student.EmpID` means `EmpID` from the *Student* table, and so on.) The output is as follows.

StudentName	AdvisorName
Peter	Jacob
Randy	Jacob
Ashley	Wendy
Anita	Wendy
Jackson	Wendy
Sheila	William

Every major university or college has a course in relational database theory. In this course, students study the operations of tables and SQL in detail. From the applications point of view, we are only interested in describing the application of relations to the relational database.



## WORKED-OUT EXERCISES

**Exercise 1:** Suppose the *Student* and *Professor* tables are as previously given. In parts (b)–(d), write an SQL query to do the following.

- (a) Write the *Professor* table as a set.
- (b) Output the names of the students and their majors.
- (c) Output the names of students whose major is CSC (computer science).
- (d) Output the names of the professors, their phone numbers, and office numbers.
- (e) Output the names of the students and their advisors and the phone numbers of the advisors.

### Solution:

- (a) `Professor = {(745, Jacob, X4832, G340), (848, Wendy, X9823, S290), (467, William, X7823, G348)}`.
- (b) `select Name, Major
 from Student;`
- (c) `select Name
 from Student
 where major = 'CSC';`  
 (Typically, nonnumeric characters are enclosed in single quotation marks.)

- (d) select Name, Phone, Office  
from Professor;
- (e) select Student.Name StudentName, Professor.Name AdvisorName, Phone  
from Student, Professor  
where Student.EmpID = Professor.EmpID;

**Exercise 2:** Suppose we have the following tables.

**Table:** Courses

ID	CourseID	Grade
3456	CSC527	A
8236	PHY345	B
8723	CSC527	A
3456	MTH248	A
8236	ENG150	B
2367	CSC590	C
2367	MTH550	A
9324	ENG150	B
8723	MTH248	C
2367	ENG150	D
9324	CSC590	A

**Table:** CourseDescriptions

CourseID	CourseName	Hours
CSC527	Discrete Math	3
CSC590	Data Structures	3
MTH248	Calculus	4
MTH550	Linear Algebra	3
ENG150	Composition	3
PHY345	Classical Physics	4

(a) Write each table as a set.

(b) Write an SQL query that outputs the student's name, course ID, course name, hours, and grade. (Assume that the *Student* table is as previously given.)

**Solution:**

(a) Courses = {(3456, CSC527, A), (8236, PHY345, B),  
(8723, CSC527, A), (3456, MTH248, A),  
(8236, ENG150, B), (2367, CSC590, C),  
(2367, MTH550, A), (9324, ENG150, B),  
(8723, MTH248, C), (2367, ENG150, D),  
(9324, CSC590, A)}.

CoursesDescriptions = {(CSC527, Discrete Math, 3),  
(CSC590, Data Structures, 3),  
(MTH248, Calculus, 4),  
(MTH550, Linear Algebra, 3),  
(ENG150, Composition, 3),  
(PHY345, Classical Physics, 4)}.

(b)

```
select Name, CourseID, CourseName, Hours, Grade
      from Student, Courses, CoursesDescriptions
      where Student.ID = Courses.ID
      and Course.CourseID = CoursesDescriptions.CourseID;
```

## SECTION REVIEW

### Key Terms

database	fields	create
metadata	record	insert
database management system	domain	update
relational database	join	delete
<i>n</i> -ary relation	query	select
attributes	structured query language (SQL)	

## EXERCISES

---

Use the following tables to answer the exercises given below.

**Table:** Supplier

ID	Name	PhNumber
100	J&M	222-2222
200	A Soft	333-3333
300	Adams &Co	444-4444
400	Ware D	555-5555

**Table:** Customer

ID	Name	PhNumber
11205	John	232-3212
11345	Lisa	812-9999
35263	Bill	111-9020
23457	Laurie	178-2340
99234	Kate	444-2222

**Table:** Product

PCode	PName	PrdPrice	UnitsInStock	SupplierID
501-8B6	Power Drill	78.90	134	200
602-95D	Saw	125.89	90	300
772-6AD	Ladder	112.90	50	200
345-132	Dish Washer	457.89	45	400
728-992	Blender	25.68	200	100
445-6SE	Fan	34.99	350	200

**Table:** Order

OrdNum	PCode	Quantity	CustId
11345	501-8B6	5	11205
11345	345-132	2	11205
11456	445-6SE	7	99234
12567	501-8B6	12	23457
12578	345-132	9	11345
23418	602-95D	10	23457

1. Write *Supplier* table as a set of ordered tuples.
2. Write *Product* table as a set of ordered tuples.
3. Write *Customer* table as a set of ordered tuples.
4. Write *Order* table as a set of ordered tuples.
5. What is the primary key in the *Supplier* table?
6. What is the primary key in the *Customer* table?
7. What is the primary key in the *Product* table?
8. Write an SQL query to print the names of all the customers and their phone numbers.
9. Write an SQL query to print the names of all the products, their prices, and the number of units in stock.
10. Write an SQL query to print the names of all the suppliers and their phone numbers.

11. Write an SQL query to print the names of all the products, such that the number of units in stock is greater than 75.
12. Write an SQL query to print the names of all the products and the names of suppliers who supply those items.
13. Write an SQL query that outputs all the products' names supplied by A Soft.
14. Write an SQL query that outputs all the products bought by Lisa.
15. Write an SQL query that outputs the names of the suppliers of the products bought by Bill.
16. Show the three steps of the join operation to list the names of the products and the names of the suppliers who supply those items.

## PROGRAMMING EXERCISES

---

1. Write a program to test whether a relation is reflexive or symmetric.
2. Write a program to test whether a relation is antisymmetric.
3. Write a program to construct the reflexive and/or symmetric closure of a relation.
4. Write a program that, given the relation on a finite poset, outputs, if any, minimum, maximum, the least, and the greatest elements of the poset.

## Matrices and Closures of Relations

**The objectives of this chapter are to:**

- Learn about matrices and their relationship with relations
- Become familiar with Boolean matrices
- Learn the relationship between Boolean matrices and different closures of a relation
- Explore how to find the transitive closure using Warshall's algorithm

In Chapter 3, we showed that a relation can be described in different ways. For example, we can use the set-builder form or the roster form. If the relation is on a finite set, we can describe it using arrow diagrams and digraphs. In Section 3.1, we discussed relations and their closures and remarked that it is relatively easier to determine the reflexive and/or symmetric closure than the transitive closure of a relation. We are particularly interested in transitive relations, as they tell which element can be reached from other elements. Therefore, if a relation is not transitive, we want to develop a method to determine its transitive closure. In this chapter, we introduce matrices, which will not only provide us another convenient way to represent relations on finite sets, but will also allow us to develop algorithms to determine the transitive closures (as well as reflexive and symmetric closures).

## 4.1 MATRICES

The term *matrix* was first introduced by J. J. Sylvester, in 1850. In 1858, Arthur Cayley began the systematic development of matrices. Matrices have applications in every branch of science and engineering. For example, they allow us to solve equations in several variables in a convenient way as well as to develop and implement algorithms in computers. Let us first explain what we mean by a matrix.

We start with an example: Suppose that the percentages of carbohydrates, fats, and proteins in bread, butter, and cheese produced by a company *A* are as shown in the following table.

	Carbohydrates	Fats	Proteins
Bread	.60	.03	.09
Butter	.00	.90	.02
Cheese	.00	.45	.25

Let us also consider similar information from another company *B*.

	Carbohydrates	Fats	Proteins
Bread	.70	.03	.10
Butter	.00	.76	.03
Cheese	.00	.45	.35

We would like to compare the data of two tables. To do so, we construct the following tables:

$$A = \begin{bmatrix} .60 & .03 & .09 \\ .00 & .90 & .02 \\ .00 & .45 & .25 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} .70 & .03 & .10 \\ .00 & .76 & .03 \\ .00 & .45 & .35 \end{bmatrix}$$

Each table is an array of real numbers arranged in three rows and three columns.

Let us consider another example: The following tables show how many students in two different schools were enrolled in various classes in 2002, 2003, and 2004.



**Arthur Cayley**

(1821–1895)

Born on August 16, 1821, in Cambridge, England, Cayley entered Trinity College at the age of 17 as a pensioner. In 1842, he graduated as senior wrangler. Between 1849 and 1863, he worked as a lawyer, but during this time he also wrote approximately 300 mathematics papers. In 1863, Cayley was elected to the new Sadlerian chair of pure mathematics at

### Historical Notes

Cambridge, where he remained until his death.

For most of his life, Cayley worked on mathematics, theoretical dynamics, and mathematical astronomy. In 1876, he published his only book, *Treatise on Elliptic Functions*. He, along with J. J. Sylvester, his lifelong friend, are considered to be the founders of invariant theory. He is also responsible for matrix theory. The square notation used for determinants is due to Cayley. He proved the Cayley-Hamilton theo-

rem. He was also one of the first mathematicians to consider geometry of more than three dimensions. In 1854, Cayley published a paper that is generally regarded as the earliest work on abstract group theory. He is best known for the theorem that every finite group is isomorphic to a suitable permutation group. In his article of 1854, he introduced a procedure for defining a finite group by listing its elements in the form of a multiplication table, known as a Cayley table.

	Algebra	Calculus	Trigonometry	Discrete Math	Physics
2002	95	80	65	90	85
2003	80	70	86	70	75
2004	94	80	80	70	75

and

	Algebra	Calculus	Trigonometry	Discrete Math	Physics
2002	85	80	70	90	90
2003	90	70	76	87	70
2004	95	90	80	80	75

This information can be compared by considering the following tables:

$$C = \begin{bmatrix} 95 & 80 & 65 & 90 & 85 \\ 80 & 70 & 86 & 70 & 75 \\ 94 & 80 & 80 & 70 & 75 \end{bmatrix} \quad \text{and} \quad D = \begin{bmatrix} 85 & 80 & 70 & 90 & 90 \\ 90 & 70 & 76 & 87 & 70 \\ 95 & 90 & 80 & 80 & 75 \end{bmatrix}$$

Here each table is an array of numbers arranged in three rows and five columns. So we see that these tables are a concise method for presenting information. Each of these tables is an example of a matrix.

A **matrix**  $A$  of size  $m \times n$  is a **rectangular array** of numbers  $a_{ij}$ ,  $i = 1, 2, \dots, m$  and  $j = 1, 2, \dots, n$  arranged in  $m$  rows and  $n$  columns, written as:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1j} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2j} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \dots & a_{ij} & \dots & a_{in} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mj} & \dots & a_{mn} \end{bmatrix}$$

Sometimes we write the matrix  $A$  as  $A = [a_{ij}]_{m \times n}$  or simply as  $A = [a_{ij}]$ . The first subscript,  $i$ , ranging from 1 to  $m$ , identifies the rows and the second subscript,  $j$ , ranging from 1 to  $n$ , identifies the columns. The element  $a_{ij}$  at the intersection of  $i$ th row and  $j$ th column is called the  **$(i, j)$ th element** (or **entry**) of  $A$ .

### EXAMPLE 4.1.1

Let

$$A = \begin{bmatrix} 4 & 5 & 2 \\ -3 & 6 & 0 \\ 11 & 23 & 5 \\ 35 & 5 & -5 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 7 & 8 & 1 & 11 & 13 \\ 2 & -10 & 20 & 3 & -9 \end{bmatrix}.$$



**James Joseph Sylvester**  
(1814–1897)  
Sylvester attended primary school in London, England. He attended St John's College in Cambridge and subsequently taught physics for three years at the University of Lon-

### Historical Notes

don. Sylvester did important work on matrix theory, along with Cayley, and discovered the discriminant of a cube equation in 1851. He used matrix theory to study higher dimensional geometry. In 1854, he became professor of mathematics at the Royal Academy at Woolrich.

In 1877, Sylvester accepted a chair at Johns Hopkins University and founded the *American Journal of Mathematics* in 1878, the first mathematical journal in the United States. He worked for a short time with Chebyshev on mechanical linkages and finished out his career at Oxford from 1883 until 1892.

Then  $A$  is a  $4 \times 3$  matrix and  $B$  is a  $2 \times 5$  matrix. The  $(3, 2)$ th element of  $A$  is 23 and  $(1, 3)$ th element of  $A$  is 2. Similarly, the  $(1, 4)$ th element of  $B$  is 11 and the  $(2, 5)$ th element of  $B$  is  $-9$ .

Let  $A = [a_{ij}]_{m \times n}$  be an  $m \times n$  matrix. Then  $[a_{i1} \ a_{i2} \ \dots \ a_{ij} \ \dots \ a_{in}]$  is the  $i$ th row of  $A$  and

$$\begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{ij} \\ \vdots \\ a_{mj} \end{bmatrix}$$

is the  $j$ th column of  $A$ . Notice that the  $i$ th row of  $A$  is a  $1 \times n$  matrix and the  $j$ th row of  $A$  is an  $m \times 1$  matrix.

### EXAMPLE 4.1.2

Let

$$A = \begin{bmatrix} 4 & 5 & 2 & -1 \\ -23 & 6 & 0 & 3 \\ 11 & 23 & 5 & 4 \\ 35 & 5 & -56 & -8 \\ 18 & 6 & -3 & 12 \end{bmatrix}.$$

Then  $A$  is a  $5 \times 4$  matrix. The second row of  $A$  is:

$$[-23 \ 6 \ 0 \ 3].$$

The fourth column of  $A$  is:

$$\begin{bmatrix} -1 \\ 3 \\ 4 \\ -8 \\ 12 \end{bmatrix}.$$

**DEFINITION 4.1.3** ▶ Two matrices  $A = [a_{ij}]$  and  $B = [b_{ij}]$  are said to be **equal**, written  $A = B$ , if they have the same size (i.e., they have the same number of rows and the same number of columns), and their corresponding elements are the same (i.e.,  $a_{ij} = b_{ij}$  for all  $i$  and  $j$ ).

**DEFINITION 4.1.4** ▶ Let  $A = [a_{ij}]_{m \times n}$  be an  $m \times n$  matrix. If  $m = n$ , i.e., the number of rows of  $A$  is the same as the number of columns of  $A$ , then  $A$  is called a **square matrix**.

### EXAMPLE 4.1.5

Let

$$A = \begin{bmatrix} 4 & 5 & 2 & 3 \\ -3 & 6 & 0 & 2 \\ 11 & 23 & 5 & 11 \\ 35 & 5 & -5 & 9 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 7 & 8 & 1 \\ 2 & -10 & 20 \\ 6 & 23 & 7 \end{bmatrix}.$$

Then  $A$  is a  $4 \times 4$  square matrix and  $B$  is a  $3 \times 3$  square matrix.

**DEFINITION 4.1.6** ▶ Let  $A = [a_{ij}]_{m \times n}$  be an  $m \times n$  matrix. If  $a_{ij} = 0$  for all  $i$  and  $j$ ,  $i = 1, 2, \dots, m$ ;  $j = 1, 2, \dots, n$ ; i.e., each element of  $A$  is 0, then  $A$  is called a **zero matrix**.

**EXAMPLE 4.1.7** Let

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

Then  $A$  is a  $4 \times 3$  zero matrix and  $B$  is a  $3 \times 3$  zero matrix. Notice that  $B$  is also a square matrix.

**DEFINITION 4.1.8** ▶ Let  $A$  be an  $n \times n$  square matrix:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1j} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2j} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \dots & a_{ij} & \dots & a_{in} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nj} & \dots & a_{nn} \end{bmatrix}.$$

The elements  $a_{11}, a_{22}, a_{33}, \dots, a_{nn}$  are called the **diagonal elements** of  $A$ .

**EXAMPLE 4.1.9** Let

$$A = \begin{bmatrix} 4 & 5 & 2 & 3 \\ -3 & 6 & 0 & 2 \\ 11 & 23 & 5 & 11 \\ 35 & 5 & -5 & 9 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 7 & 8 & 1 \\ 2 & -10 & 20 \\ 6 & 23 & 7 \end{bmatrix}.$$

Now  $A$  is a  $4 \times 4$  square matrix and  $B$  is a  $3 \times 3$  square matrix. The diagonal elements of  $A$  are 4, 6, 5, and 9. The diagonal elements of  $B$  are 7, -10, and 7.

**DEFINITION 4.1.10** ▶ Let  $A = [a_{ij}]_{n \times n}$  be an  $n \times n$  square matrix. Then  $A$  is called a **diagonal matrix** if  $a_{ij} = 0$  for all  $i \neq j$ ,  $i = 1, 2, \dots, n$ ;  $j = 1, 2, \dots, n$ ; i.e., all nondiagonal elements of  $A$  are zero.

**EXAMPLE 4.1.11** The following are diagonal matrices:

$$\begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix}, \quad \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \begin{bmatrix} 7 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & -5 & 0 \\ 0 & 0 & 0 & 12 \\ 0 & 0 & 0 & 6 \end{bmatrix}, \quad \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 6 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

**DEFINITION 4.1.12** ▶ Let  $A = [a_{ij}]_{n \times n}$  be an  $n \times n$  diagonal matrix. Then  $A$  is called an **identity matrix**, written  $I_n$ , if  $a_{11} = 1, a_{22} = 1, \dots, a_{nn} = 1$ ; i.e., all diagonal elements are 1.

Notice that

$$I_1 = [1], \quad I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad I_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad I_5 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

---

**DEFINITION 4.1.13** ▶ Let  $A = [a_{ij}]_{m \times n}$  and  $B = [b_{ij}]_{m \times n}$  be  $m \times n$  matrices. The **sum** of  $A$  and  $B$ , written  $A + B$ , is the  $m \times n$  matrix

$$A + B = [c_{ij}]_{m \times n},$$

where  $c_{ij} = a_{ij} + b_{ij}$ , for all  $i$  and  $j$ ,  $i = 1, 2, \dots, m$ ;  $j = 1, 2, \dots, n$ .

Notice that two matrices are added only if they have the same number of rows and the same number of columns. Moreover, to determine the sum of two matrices, their corresponding elements are added.

**EXAMPLE 4.1.14**

Let

$$A = \begin{bmatrix} 4 & 5 & 2 \\ -3 & 6 & 0 \\ -2 & 11 & 5 \\ 0 & 5 & -56 \\ 1 & 6 & -3 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 & 3 & -5 \\ 4 & -5 & 7 \\ -3 & 6 & -3 \\ 0 & -4 & 18 \\ 3 & -13 & 0 \end{bmatrix}.$$

Then  $A$  and  $B$  are  $5 \times 3$  matrices. The sum of  $A$  and  $B$  is:

$$A + B = \begin{bmatrix} 4+0 & 5+3 & 2+(-5) \\ -3+4 & 6+(-5) & 0+7 \\ -2+(-3) & 11+6 & 5+(-3) \\ 0+0 & 5+(-4) & -56+18 \\ 1+3 & 6+(-13) & -3+0 \end{bmatrix} = \begin{bmatrix} 4 & 8 & -3 \\ 1 & 1 & 7 \\ -5 & 17 & 2 \\ 0 & 1 & -38 \\ 4 & -7 & -3 \end{bmatrix}.$$

---

**DEFINITION 4.1.15** ▶ Let  $A = [a_{ij}]_{m \times n}$  and let  $k$  be a number. The matrix  $kA$  is the  $m \times n$  matrix defined by:

$$kA = [ka_{ij}]_{m \times n}.$$

Suppose that

$$A = \begin{bmatrix} 3 & 8 \\ 2 & -6 \\ 4 & 0 \end{bmatrix}.$$

Then

$$3A = \begin{bmatrix} 3 \cdot 3 & 3 \cdot 8 \\ 3 \cdot 2 & 3 \cdot (-6) \\ 3 \cdot 4 & 3 \cdot 0 \end{bmatrix} = \begin{bmatrix} 9 & 24 \\ 6 & -18 \\ 12 & 0 \end{bmatrix}.$$

If  $A = [a_{ij}]_{m \times n}$ , then  $-A$  is defined to be the matrix  $(-1)A$ . Now,

$$-A = (-1)A = (-1) \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} = \begin{bmatrix} -a_{11} & -a_{12} & \dots & -a_{1n} \\ -a_{21} & -a_{22} & \dots & -a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ -a_{m1} & -a_{m2} & \dots & -a_{mn} \end{bmatrix}.$$

For example,

$$\text{if } A = \begin{bmatrix} 4 & 5 & 2 \\ -3 & 6 & 0 \\ -2 & 11 & 5 \\ 0 & 5 & -56 \\ 1 & 6 & -3 \end{bmatrix}, \quad \text{then } -A = \begin{bmatrix} -4 & -5 & -2 \\ 3 & -6 & 0 \\ 2 & -11 & -5 \\ 0 & -5 & 56 \\ -1 & -6 & 3 \end{bmatrix}.$$

Let  $A = [a_{ij}]_{m \times n}$  and  $B = [b_{ij}]_{m \times n}$  be  $m \times n$  matrices. The **difference** of  $A$  and  $B$ , written  $A - B$ , is the  $m \times n$  matrix

$$A - B = A + (-B).$$

It follows that

$$A - B = [a_{ij} - b_{ij}]_{m \times n}.$$

For example, if

$$A = \begin{bmatrix} 4 & 5 & 2 \\ -3 & 6 & 0 \\ -2 & 11 & 5 \\ 0 & 5 & -56 \\ 1 & 6 & -3 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 & 3 & -5 \\ 4 & -5 & 7 \\ -3 & 6 & -3 \\ 0 & -4 & 18 \\ 3 & -13 & 0 \end{bmatrix},$$

then  $A - B$  is:

$$A - B = \begin{bmatrix} 4 - 0 & 5 - 3 & 2 - (-5) \\ -3 - 4 & 6 - (-5) & 0 - 7 \\ -2 - (-3) & 11 - 6 & 5 - (-3) \\ 0 - 0 & 5 - (-4) & -56 - 18 \\ 1 - 3 & 6 - (-13) & -3 - 0 \end{bmatrix} = \begin{bmatrix} 4 & 2 & 7 \\ -7 & 11 & -7 \\ 1 & 5 & 8 \\ 0 & 9 & -74 \\ -2 & 19 & -3 \end{bmatrix}.$$

The following theorem lists various properties of matrices, the proofs of which are left as exercises.

**Theorem 4.1.16:** Let  $A$ ,  $B$ , and  $C$  be  $m \times n$  matrices. Let  $\mathbf{0}$  denote the  $m \times n$  zero matrix. Then

- (i)  $A + B = B + A$ .
- (ii)  $A + (B + C) = (A + B) + C$ .
- (iii)  $A - A = \mathbf{0} = -A + A$ .
- (iv) If  $k$  is a number, then  $k(A + B) = kA + kB$ ,  $k(A - B) = kA - kB$ .

Next, we describe the multiplication of matrices.

**DEFINITION 4.1.17** ▶ Let  $A = [a_{ij}]_{m \times n}$  be an  $m \times n$  matrix and  $B = [b_{jk}]_{n \times p}$  be an  $n \times p$  matrix. The **multiplication** of  $A$  and  $B$ , written  $AB$ , is the  $m \times p$  matrix

$$AB = [c_{ik}]_{m \times p},$$

where

$$c_{ik} = a_{i1}b_{1k} + a_{i2}b_{2k} + \cdots + a_{in}b_{nk} = \sum_{j=1}^n a_{ij}b_{jk},$$

for all  $i$  and  $k$ ,  $i = 1, 2, \dots, m$ ;  $k = 1, 2, \dots, p$ .

Notice that the multiplication  $AB$  of matrices  $A$  and  $B$  is defined only if the number of columns of  $A$  is the same as the number of rows of  $B$ . Moreover, if  $AB$  is defined, then the number of rows of  $AB$  is the same as the number of rows of  $A$  and the number of columns of  $AB$  is the same as the number of columns of  $B$ .

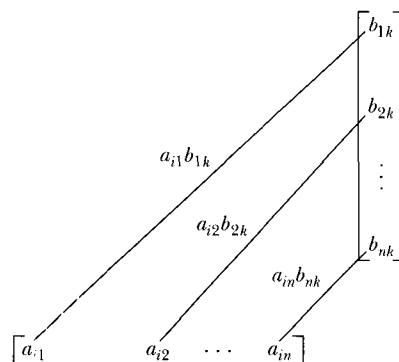
Let  $A = [a_{ij}]_{m \times n}$  be an  $m \times n$  matrix and  $B = [b_{jk}]_{n \times p}$  be an  $n \times p$  matrix. Then  $AB$  is defined. Notice that to determine the  $(i, k)$ th element of  $AB$ , we take the  $i$ th row of  $A$  and the  $k$ th column of  $B$ , multiply the corresponding elements, and add the result. To be specific, suppose that

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1j} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2j} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \dots & a_{ij} & \dots & a_{in} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mj} & \dots & a_{mn} \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1k} & \dots & b_{1p} \\ b_{21} & b_{22} & \dots & b_{2k} & \dots & b_{2p} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ b_{i1} & b_{i2} & \dots & b_{ik} & \dots & b_{ip} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ b_{n1} & b_{n2} & \dots & b_{nk} & \dots & b_{np} \end{bmatrix}.$$

The  $i$ th row of  $A$  is  $[a_{i1} \ a_{i2} \ \dots \ a_{ij} \ \dots \ a_{in}]$  and the  $k$ th column of  $B$  is

$$\begin{bmatrix} b_{1k} \\ b_{2k} \\ \vdots \\ b_{ik} \\ \vdots \\ b_{nk} \end{bmatrix}.$$

Multiply the corresponding elements as in Figure 4.1.



**FIGURE 4.1** Multiplying the  $i$ th row of  $A$  and the  $j$ th column of  $B$

**EXAMPLE 4.1.18**

Let

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 6 & -2 \\ 9 & 1 \\ -1 & 4 \\ -3 & 0 \end{bmatrix}.$$

Because  $A$  is a  $3 \times 4$  matrix and  $B$  is a  $4 \times 2$  matrix,  $AB$  is a  $3 \times 2$  matrix. To determine, say the  $(2, 1)$ th element of  $AB$ , take the second row of  $A$  and the first column of  $B$ , multiply the corresponding elements, and add the result. For example, the second row of  $A$  is  $[5 \ 6 \ 7 \ 8]$  and the first column of  $B$  is

$$\begin{bmatrix} 6 \\ 9 \\ -1 \\ -3 \end{bmatrix}.$$

Multiply the corresponding elements and then add the results, i.e.,  $5 \cdot 6 + 6 \cdot 9 + 7 \cdot (-1) + 8 \cdot (-3) = 53$ . Similarly, we can calculate the other elements of  $AB$ . That is,

$$\begin{aligned} AB &= \begin{bmatrix} 1 \cdot 6 + 2 \cdot 9 + 3 \cdot (-1) + 4 \cdot (-3) & 1 \cdot (-2) + 2 \cdot 1 + 3 \cdot 4 + 4 \cdot 0 \\ 5 \cdot 6 + 6 \cdot 9 + 7 \cdot (-1) + 8 \cdot (-3) & 5 \cdot (-2) + 6 \cdot 1 + 7 \cdot 4 + 8 \cdot 0 \\ 9 \cdot 6 + 10 \cdot 9 + 11 \cdot (-1) + 12 \cdot (-3) & 9 \cdot (-2) + 10 \cdot 1 + 11 \cdot 4 + 12 \cdot 0 \end{bmatrix} \\ &= \begin{bmatrix} 9 & 12 \\ 53 & 24 \\ 97 & 36 \end{bmatrix}. \end{aligned}$$

Let  $A$  be a  $4 \times 3$  matrix and  $B$  be a  $3 \times 5$  matrix. Because the number of columns is the same as the number of rows of  $B$ , the multiplication  $AB$  is defined. What about the multiplication  $BA$ ? To form the multiplication  $BA$ , the number of columns of  $B$  must be the same as the number of rows of  $A$ . However, the number of columns of  $B$  is 5 and the number of rows of  $A$  is 4. Hence, the product  $BA$  is not defined.

Let  $A$  be a  $4 \times 3$  matrix and  $B$  be a  $3 \times 4$  matrix. Then both the products  $AB$  and  $BA$  are defined. Notice that  $AB$  is a  $4 \times 4$  matrix while  $BA$  is a  $3 \times 3$  matrix. Hence, even though both products  $AB$  and  $BA$  are defined, because both matrices are of different sizes,  $AB \neq BA$ .

Next consider the case where  $A$  and  $B$  are square matrices of the same size. To be specific, suppose that  $A$  and  $B$  are  $2 \times 2$  matrices. Then  $AB$  and  $BA$  are  $2 \times 2$  matrices. Even though  $AB$  and  $BA$  are of the same size,  $AB$  is not necessarily the same as  $BA$ . For example, let

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}.$$

Then

$$AB = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 \cdot 1 + 2 \cdot 1 & 1 \cdot 1 + 2 \cdot 0 \\ 3 \cdot 1 + 4 \cdot 1 & 3 \cdot 1 + 4 \cdot 0 \end{bmatrix} = \begin{bmatrix} 3 & 1 \\ 7 & 3 \end{bmatrix}$$

and

$$BA = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 1 \cdot 1 + 1 \cdot 3 & 1 \cdot 2 + 1 \cdot 4 \\ 1 \cdot 1 + 0 \cdot 3 & 1 \cdot 2 + 0 \cdot 4 \end{bmatrix} = \begin{bmatrix} 4 & 6 \\ 1 & 2 \end{bmatrix}.$$

Hence,  $AB \neq BA$ . Thus, matrix multiplication is not commutative.

In general, suppose that  $A$  and  $B$  are matrices such that  $AB$  is defined. For the matrix  $BA$ , any one of the following four cases is possible.

- (i) The multiplication  $BA$  is not defined.
- (ii) The multiplication  $BA$  is defined, but  $AB$  and  $BA$  are of different sizes.
- (iii) The multiplication  $BA$  is defined,  $AB$  and  $BA$  are of the same size, but  $AB \neq BA$ .
- (iv) The multiplication  $BA$  is defined,  $AB$  and  $BA$  are of the same size, and  $AB = BA$ .

Let  $A$  be an  $m \times n$  matrix. We can show that

$$AI_n = A \quad \text{and} \quad I_m A = A,$$

where  $I_n$  is the  $n \times n$  identity matrix and  $I_m$  is the  $m \times m$  identity matrix.

For example, suppose that  $A$  is the  $2 \times 3$  matrix given by:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}.$$

Then

$$\begin{aligned} AI_3 &= \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 \cdot 1 + 2 \cdot 0 + 3 \cdot 0 & 1 \cdot 0 + 2 \cdot 1 + 3 \cdot 0 & 1 \cdot 0 + 2 \cdot 0 + 3 \cdot 1 \\ 4 \cdot 1 + 5 \cdot 0 + 6 \cdot 0 & 4 \cdot 0 + 5 \cdot 1 + 6 \cdot 0 & 4 \cdot 0 + 5 \cdot 0 + 6 \cdot 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}. \end{aligned}$$

Similarly,

$$\begin{aligned} I_2 A &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \\ &= \begin{bmatrix} 1 \cdot 1 + 0 \cdot 4 & 1 \cdot 2 + 0 \cdot 5 & 1 \cdot 3 + 0 \cdot 6 \\ 0 \cdot 1 + 1 \cdot 4 & 0 \cdot 2 + 1 \cdot 5 & 0 \cdot 3 + 1 \cdot 6 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}. \end{aligned}$$

Hence,  $I_2 A = A = AI_3$ .

The following theorem lists various properties of matrix multiplication. We leave the proof as an exercise.

**Theorem 4.1.19:** Let  $A$  be an  $m \times n$  matrix,  $B$  be an  $n \times p$  matrix, and  $C$  be an  $p \times q$  matrix. Then

- (i)  $A(BC) = (AB)C$ ,
- (ii)  $I_m A = A = AI_n$ .

**DEFINITION 4.1.20** ▶ Let  $A$  be an  $n \times n$  matrix. We define  $A^m$ ,  $m \in \mathbb{N}$  as follows:

$$A^1 = A$$

$$A^m = AA^{m-1}, \quad m > 1.$$

Notice that  $A^2 = AA$ ,  $A^3 = AA^2 = A(AA) = (AA)A = A^2A$ . By induction, we can prove that if  $m > 1$ , then  $A^m = A^{m-1}A$ .

Let  $A$  be an  $m \times n$  matrix. Then  $A$  has  $m$  rows and  $n$  columns. A convenient way to store  $A$  in computer memory is to use a two-dimensional array of  $m$  rows and  $n$  columns. Using this convention, we describe an algorithm to multiply two matrices.

### ALGORITHM 4.1: Multiply Matrices.

*Input:*  $A, B$  matrices  
 $m, n, p$ —positive integers such that  $m \times n$  specifies the size of  $A$ ,  $n \times p$  specifies the size of  $B$

*Output:*  $C$ —an  $m \times p$  matrix such that  $C = AB$

```

1. procedure matrixMultiplication(A, B, C, m, n, p)
2. begin
3.   for i := 1 to m do
4.     for k := 1 to p do
5.       begin
6.         C[i, k] := 0
7.         for j := 1 to n do
8.           C[i, k] := C[i, k] + A[i, j] * B[j, k];
9.       end
10.    end

```

## Transpose of a Matrix

**DEFINITION 4.1.21** ▶ Let  $A = [a_{ij}]_{m \times n}$  be an  $m \times n$  matrix. The **transpose** of  $A$ , written  $A^T$ , is the  $n \times m$  matrix defined by

$$A^T = [b_{ji}]_{n \times m},$$

where  $b_{ij} = a_{ji}$  for all  $i$  and  $j$ ,  $i = 1, 2, \dots, n$ ;  $j = 1, 2, \dots, m$ ; i.e., the  $(i, j)$ th element of  $A^T$  is the same as the  $(j, i)$ th element of  $A$ .

### EXAMPLE 4.1.22

Let  $A$  be the  $4 \times 3$  matrix given by:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \end{bmatrix}.$$

Then  $A^T$  is the  $3 \times 4$  matrix given by:

$$A^T = \begin{bmatrix} 1 & 4 & 7 & 10 \\ 2 & 5 & 8 & 11 \\ 3 & 6 & 9 & 12 \end{bmatrix}.$$

Notice that the rows of  $A$  are the columns of  $A^T$  and the columns of  $A$  are the rows of  $A^T$ .

The following theorem lists some properties of the transpose of a matrix.

### Theorem 4.1.23:

- (i) Let  $A$  and  $B$  be  $m \times n$  matrices. Then

$$(A + B)^T = A^T + B^T.$$

- (ii) Let  $A$  be an  $m \times n$  matrix and  $B$  be an  $n \times p$  matrix. Then

$$(AB)^T = B^T A^T.$$

## Symmetric Matrices

**DEFINITION 4.1.24** ▶ Let  $A = [a_{ij}]$  be an  $n \times n$  matrix. Then  $A$  is called **symmetric** if  $a_{ij} = a_{ji}$  for all  $i$  and  $j$ ,  $i = 1, 2, \dots, n; j = 1, 2, \dots, n$ ; i.e., the  $(i, j)$ th element of  $A$  is the same as the  $(j, i)$ th element of  $A$ .

### EXAMPLE 4.1.25

The following matrices are symmetric:

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 5 & 6 \\ 3 & 6 & 9 \end{bmatrix}, \quad \begin{bmatrix} 1 & 4 & -7 & 10 \\ 4 & 0 & 8 & 6 \\ -7 & 8 & -5 & -23 \\ 10 & 6 & -23 & 2 \end{bmatrix}.$$

The following theorem lists some properties of symmetric matrices.

**Theorem 4.1.26:** Let  $A$  and  $B$  be  $n \times n$  symmetric matrices and  $c \in \mathbb{R}$ . Then  $A + B$  and  $cA$  are symmetric matrices.

## Boolean (Zero-One) Matrices

Now that we understand what a matrix is, let us consider matrices whose entries are 0 or 1. These are the matrices that allow us to represent matrices in a convenient way in computer memory and to design and implement algorithms to determine the transitive closure of a relation.

Before defining Boolean matrices, we note that the set  $\{0, 1\}$  is a lattice under the usual “less than or equal to” relation, where for all  $a, b \in \{0, 1\}$ ,  $a \vee b = \max\{a, b\}$  and  $a \wedge b = \min\{a, b\}$ . Then

$$a \vee b = \begin{cases} 1 & \text{if } a = 1, \text{ or } b = 1, \text{ or } (a = 1 \text{ and } b = 1), \\ 0 & \text{otherwise.} \end{cases}$$

and

$$a \wedge b = \begin{cases} 1 & \text{if } a = 1 \text{ and } b = 1 \\ 0 & \text{otherwise.} \end{cases}$$

For example,

$$1 \vee 1 = 1, \quad 1 \vee 0 = 1, \quad 0 \vee 1 = 1, \quad \text{and} \quad 0 \vee 0 = 0,$$

and

$$1 \wedge 1 = 1, \quad 1 \wedge 0 = 0, \quad 0 \wedge 1 = 0, \quad \text{and} \quad 0 \wedge 0 = 0.$$

This lattice is a chain,  $0 < 1$ , of two elements, and hence it is a distributive lattice. Also in this lattice 0 is the complement of 1 and 1 is the complement of 0. This lattice is therefore a Boolean algebra of two elements. Matrices with entries from this Boolean algebra are called Boolean matrices.

---

**DEFINITION 4.1.27** ▶ Let  $A = [a_{ij}]_{m \times n}$  and  $B = [b_{ij}]_{m \times n}$  be  $m \times n$  Boolean matrices.

- (i) The **Boolean join** (or **join**) of  $A$  and  $B$ , written  $A \vee B$ , is the  $m \times n$  matrix defined by:

$$A \vee B = [a_{ij} \vee b_{ij}]_{m \times n}.$$

- (ii) The **Boolean meet** (or **meet**) of  $A$  and  $B$ , written  $A \wedge B$ , is the  $m \times n$  matrix defined by:

$$A \wedge B = [a_{ij} \wedge b_{ij}]_{m \times n}.$$

### EXAMPLE 4.1.28

Let

$$A = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

Then

$$A \vee B = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \vee \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 \vee 1 & 0 \vee 0 & 1 \vee 1 \\ 1 \vee 1 & 0 \vee 1 & 1 \vee 0 \\ 0 \vee 0 & 1 \vee 0 & 1 \vee 1 \\ 0 \vee 0 & 0 \vee 1 & 0 \vee 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

Similarly,

$$A \wedge B = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \wedge \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 \wedge 1 & 0 \wedge 0 & 1 \wedge 1 \\ 1 \wedge 1 & 0 \wedge 1 & 1 \wedge 0 \\ 0 \wedge 0 & 1 \wedge 0 & 1 \wedge 1 \\ 0 \wedge 0 & 0 \wedge 1 & 0 \wedge 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

---

**DEFINITION 4.1.29** ▶ Let  $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n \in \{0, 1\}$ . The expression

$$(a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \cdots \vee (a_n \wedge b_n)$$

is called the **join of meet expression**.

**Theorem 4.1.30:** Let  $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n \in \{0, 1\}$ . Then the following conditions are equivalent.

- (i)  $(a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \dots \vee (a_n \wedge b_n) = 1$
- (ii)  $a_i \wedge b_i = 1$  for some  $i, 1 \leq i \leq n$ .
- (iii)  $a_i = 1$  and  $b_i = 1$  for some  $i, 1 \leq i \leq n$

**Proof:** (i)  $\Rightarrow$  (ii): Suppose

$$(a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \dots \vee (a_n \wedge b_n) = 1.$$

If  $a_i \wedge b_i = 0$  for all  $i, i = 1, 2, \dots, n$ , then

$$(a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \dots \vee (a_n \wedge b_n) = 0 \vee 0 \vee \dots \vee 0 = 0,$$

which is a contradiction. Hence,  $a_i \wedge b_i = 1$  for some  $i, i = 1, 2, \dots, n$ .

(ii)  $\Rightarrow$  (iii): Suppose  $a_i \wedge b_i = 1$ , where  $1 \leq i \leq n$ . Now,  $0 \wedge 1 = 0$  and  $1 \wedge 0 = 0$ . Therefore, if either  $a_i = 0$  or  $b_i = 0$ , then  $a_i \wedge b_i = 0$ , which is a contradiction. Hence,  $a_i = 1$  and  $b_i = 1$ .

(iii)  $\Rightarrow$  (i): Suppose that  $a_i = 1$  and  $b_i = 1$  for some  $i, i = 1, 2, \dots, n$ . Then  $a_i \wedge b_i = 1$ . This implies that

$$\begin{aligned} & (a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \dots \vee (a_n \wedge b_n) \\ &= (a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \dots \vee (a_i \wedge b_i) \vee \dots \vee (a_n \wedge b_n) \\ &= (a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \dots \vee 1 \vee \dots \vee (a_n \wedge b_n) \\ &= 1 \end{aligned}$$

because  $1 \vee 1 = 1$  and  $1 \vee 0 = 1$ . ■

Let  $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n \in \{0, 1\}$ . Suppose that we want to evaluate the join of meet expression

$$(a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \dots \vee (a_n \wedge b_n).$$

We evaluate this expression from left to right. First we evaluate  $a_1 \wedge b_1$ , then  $a_2 \wedge b_2$ , and so on. By Theorem 4.1.30, it follows that as soon as we find the first  $a_i \wedge b_i = 1$ , we can conclude that  $(a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \dots \vee (a_n \wedge b_n) = 1$ .

For example, let us evaluate

$$(1 \wedge 0) \vee (1 \wedge 1) \vee (0 \wedge 1) \vee (1 \wedge 0) \vee (1 \wedge 1) \vee (1 \wedge 0).$$

Now the first term  $1 \wedge 0 = 0$ , but the second term  $1 \wedge 1 = 1$ . Therefore, we no longer need to evaluate the remaining terms. Because the second term  $1 \wedge 1 = 1$ , we can conclude that  $(1 \wedge 0) \vee (1 \wedge 1) \vee (0 \wedge 1) \vee (1 \wedge 0) \vee (1 \wedge 1) \vee (1 \wedge 0) = 1$ . Even though the fifth term  $1 \wedge 1$  is also 1, we can stop evaluating after the second term.

Now consider the expression

$$(1 \wedge 0) \vee (1 \wedge 0) \vee (0 \wedge 0) \vee (1 \wedge 0) \vee (1 \wedge 0) \vee (1 \wedge 1).$$

Here we must evaluate all the terms because only the last term evaluates to 1. Because at least one term evaluates to 1,

$$(1 \wedge 0) \vee (1 \wedge 0) \vee (0 \wedge 0) \vee (1 \wedge 0) \vee (1 \wedge 0) \vee (1 \wedge 1) = 1.$$

Next consider the join of meet expression

$$(1 \wedge 0) \vee (1 \wedge 0) \vee (0 \wedge 0) \vee (1 \wedge 0) \vee (1 \wedge 0) \vee (0 \wedge 0) \vee (0 \wedge 1).$$

Notice that here all terms evaluate to 0. Hence,

$$(1 \wedge 0) \vee (1 \wedge 0) \vee (0 \wedge 0) \vee (1 \wedge 0) \vee (1 \wedge 0) \vee (0 \wedge 0) \vee (0 \wedge 1) = 0.$$

**DEFINITION 4.1.31** ▶ Let  $A = [a_{ij}]_{m \times n}$  and  $B = [b_{jk}]_{n \times p}$  be Boolean matrices. The **Boolean product** (or **product**) of  $A$  and  $B$ , written  $A \odot B$ , is the  $m \times p$  matrix defined by:

$$A \odot B = [c_{ij}]_{m \times p},$$

where

$$c_{ik} = (a_{i1} \wedge b_{1k}) \vee (a_{i2} \wedge b_{2k}) \vee \cdots \vee (a_{in} \wedge b_{nk}) = \bigvee_{j=1}^n (a_{ij} \wedge b_{jk}),$$

for all  $i$  and  $k$ ,  $i = 1, 2, \dots, m$ ;  $k = 1, 2, \dots, p$ .

Notice that the expression in Definition 4.1.31,  $(a_{i1} \wedge b_{1k}) \vee (a_{i2} \wedge b_{2k}) \vee \cdots \vee (a_{in} \wedge b_{nk})$  is a join of meet expression.

Moreover, notice that the Boolean product is similar to the product of matrices defined earlier. The only difference is that  $+$  is replaced with  $\vee$  and multiplication is replaced with  $\wedge$ .

### EXAMPLE 4.1.32

(i)

$$\begin{aligned} & \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} (0 \wedge 1) \vee (0 \wedge 0) \vee (1 \wedge 0) & (0 \wedge 0) \vee (0 \wedge 1) \vee (1 \wedge 0) & (0 \wedge 1) \vee (0 \wedge 1) \vee (1 \wedge 0) \\ (1 \wedge 1) \vee (0 \wedge 0) \vee (1 \wedge 0) & (1 \wedge 0) \vee (0 \wedge 1) \vee (1 \wedge 0) & (1 \wedge 1) \vee (0 \wedge 1) \vee (1 \wedge 0) \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}. \end{aligned}$$

(ii)

$$\begin{aligned} & \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} (1 \wedge 1) \vee (1 \wedge 0) \vee (0 \wedge 0) & (1 \wedge 0) \vee (1 \wedge 1) \vee (0 \wedge 0) \\ (0 \wedge 1) \vee (1 \wedge 0) \vee (1 \wedge 0) & (0 \wedge 0) \vee (1 \wedge 1) \vee (1 \wedge 0) \\ (0 \wedge 1) \vee (0 \wedge 0) \vee (1 \wedge 0) & (0 \wedge 0) \vee (0 \wedge 1) \vee (1 \wedge 0) \\ (1 \wedge 1) \vee (0 \wedge 0) \vee (1 \wedge 0) & (1 \wedge 0) \vee (0 \wedge 1) \vee (1 \wedge 0) \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}. \end{aligned}$$

To compute the  $(i, k)$ th element of  $A \odot B$ , look at the  $i$ th row of  $A$  and  $k$ th column of  $B$ . Compare the corresponding elements. If any two corresponding elements are 1, then the  $(i, k)$ th element is 1; otherwise, the  $(i, k)$ th element is 0.

For example, let

$$A = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}.$$

Now,  $A$  is a  $3 \times 4$  matrix and  $B$  is a  $4 \times 2$  matrix. Therefore,  $A \odot B$  is defined and it is a  $3 \times 2$  matrix. Suppose that we want to determine the  $(2, 1)$ th element of  $A \odot B$ . We consider the second row of  $A$  and first column of  $B$ . Let us write them side by side:

Second Row of $A$	First Column of $B$
$\begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$	$\longleftrightarrow$
$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$	$\longleftrightarrow$
$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$	$\longleftrightarrow$
$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$	$\longleftrightarrow$

Notice that the fourth element of the second row of  $A$  is 1 and the fourth element of the first column of  $B$  is 1. Therefore, the  $(2, 1)$ th element of  $A \odot B$  is 1. In fact, as soon as we can determine that the corresponding elements from the  $i$ th row of  $A$  and the  $j$ th column of  $B$  are both 1, we can conclude that the  $(i, j)$ th element of  $A \odot B$  is 1. For example, if we want to determine the  $(1, 2)$ th element of  $A \odot B$ , we consider the first row of  $A$  and the second column of  $B$ . Let us write them side by side:

First Row of $A$	Second Column of $B$
$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix}$	$\longleftrightarrow$
$\begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$	$\longleftrightarrow$
$\begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$	$\longleftrightarrow$
$\begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$	$\longleftrightarrow$

We notice that the corresponding first two elements are different, but the corresponding second elements are both 1. Thus we can conclude that the  $(1, 2)$ th element of  $A \odot B$  is 1.

#### ALGORITHM 4.2: Multiply Boolean Matrices.

*Input:*  $A, B$  Boolean matrices

$m, n, p$ —positive integers such that  $m \times n$  specifies the size of  $A$ ,  $n \times p$  specifies the size of  $B$

*Output:*  $C$ —an  $m \times p$  matrix such that  $C = A \odot B$

```

1. procedure booleanProduct ( $A, B, C, m, n, p$ )
2. begin
3.   for  $i := 1$  to  $m$  do
4.     for  $k := 1$  to  $p$  do
5.       begin
6.          $C[i, k] := 0$ 
7.         for  $j := 1$  to  $n$  do
8.           if  $A[i, j] = 1$  and  $B[j, k] = 1$  then
9.             begin

```

```

10.      C[i,k] := 1;
11.      exit this for loop
12.    end
13.  end
14. end

```

## WORKED-OUT EXERCISES

**Exercise 1:** Let  $A = (a_{ij})$  be the matrix given by

$$A = \begin{bmatrix} 2 & 0 & 6 & -1 \\ 4 & 9 & 0 & 0 \\ 5 & 0 & 1 & -8 \end{bmatrix}.$$

- (a) What is the size of  $A$ ?
- (b) What are the entries of the second row of  $A$ ?
- (c) Find the entries  $a_{23}, a_{31}, a_{22}, a_{34}$ .
- (d) What is the transpose of  $A$ ?

**Solution:**

- (a) The size of  $A$  is  $3 \times 4$ .
- (b) The entries of the second row are: 4, 9, 0, and 0.
- (c)  $a_{23} = 0, a_{31} = 5, a_{22} = 9, a_{34} = -8$ .

$$(d) A^T = \begin{bmatrix} 2 & 4 & 5 \\ 0 & 9 & 0 \\ 6 & 0 & 1 \\ -1 & 0 & -8 \end{bmatrix}$$

**Exercise 2:** Let

$$A = \begin{bmatrix} 1 & -6 & 2 \\ 0 & -3 & 1 \\ -4 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 2 & -5 \\ -6 & -4 & 2 \\ 0 & 5 & 0 \end{bmatrix},$$

$$C = \begin{bmatrix} 0 \\ 3 \\ -2 \end{bmatrix}, \quad \text{and} \quad D = \begin{bmatrix} 7 \\ 0 \\ 0 \end{bmatrix}$$

be matrices. Compute the following if they exist.

- (a)  $A + B$
- (b)  $5A$
- (c)  $A + D$
- (d)  $C - 6D$
- (e)  $4B + 7A$

**Solution:**

$$\begin{aligned}
(a) A + B &= \begin{bmatrix} 1 & -6 & 2 \\ 0 & -3 & 1 \\ -4 & 1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 2 & -5 \\ -6 & -4 & 2 \\ 0 & 5 & 0 \end{bmatrix} \\
&= \begin{bmatrix} 1+0 & -6+2 & 2-5 \\ 0-6 & -3-4 & 1+2 \\ -4+0 & 1+5 & 0+0 \end{bmatrix} \\
&= \begin{bmatrix} 1 & -4 & -3 \\ -6 & -7 & 3 \\ -4 & 6 & 0 \end{bmatrix}
\end{aligned}$$

$$(b) 5A = \begin{bmatrix} 5 \cdot 1 & 5 \cdot -6 & 5 \cdot 2 \\ 5 \cdot 0 & 5 \cdot -3 & 5 \cdot 1 \\ 5 \cdot -4 & 5 \cdot 1 & 5 \cdot 0 \end{bmatrix} = \begin{bmatrix} 5 & -30 & 10 \\ 0 & -15 & 5 \\ -20 & 5 & 0 \end{bmatrix}$$

(c)  $A$  and  $D$  are not of the same order, so  $A + D$  does not exist.

$$\begin{aligned}
(d) \text{First compute } -6D. \text{ Here } D &= \begin{bmatrix} 7 \\ 0 \\ 0 \end{bmatrix}. \text{ Hence, } -6D = \\
&\begin{bmatrix} -6 \cdot 7 \\ -6 \cdot 0 \\ -6 \cdot 0 \end{bmatrix} = \begin{bmatrix} -42 \\ 0 \\ 0 \end{bmatrix}. \text{ Now } C = \begin{bmatrix} 0 \\ 3 \\ -2 \end{bmatrix}. \text{ Hence,}
\end{aligned}$$

$$C - 6D = C + (-6D) = \begin{bmatrix} 0-42 \\ 3+0 \\ -2+0 \end{bmatrix} = \begin{bmatrix} -42 \\ 3 \\ -2 \end{bmatrix}.$$

$$(e) 4B = \begin{bmatrix} 4 \cdot 0 & 4 \cdot 2 & 4 \cdot -5 \\ 4 \cdot -6 & 4 \cdot -4 & 4 \cdot 2 \\ 4 \cdot 0 & 4 \cdot 5 & 4 \cdot 0 \end{bmatrix} = \begin{bmatrix} 0 & 8 & -20 \\ -24 & -16 & 8 \\ 0 & 20 & 0 \end{bmatrix}$$

and

$$7A = \begin{bmatrix} 7 \cdot 1 & 7 \cdot -6 & 7 \cdot 2 \\ 7 \cdot 0 & 7 \cdot -3 & 7 \cdot 1 \\ 7 \cdot -4 & 7 \cdot 1 & 7 \cdot 0 \end{bmatrix} = \begin{bmatrix} 7 & -42 & 14 \\ 0 & -21 & 7 \\ -28 & 7 & 0 \end{bmatrix}$$

Hence,

$$\begin{aligned}
4B + 7A &= \begin{bmatrix} 0 & 8 & -20 \\ -24 & -16 & 8 \\ 0 & 20 & 0 \end{bmatrix} + \begin{bmatrix} 7 & -42 & 14 \\ 0 & -21 & 7 \\ -28 & 7 & 0 \end{bmatrix} \\
&= \begin{bmatrix} 0+7 & 8-42 & -20+14 \\ -24+0 & -16-21 & 8+7 \\ 0-28 & 20+7 & 0+0 \end{bmatrix} \\
&= \begin{bmatrix} 7 & -34 & -6 \\ -24 & -37 & 15 \\ -28 & 27 & 0 \end{bmatrix}
\end{aligned}$$

**Exercise 3:** Let  $A = [1 \ 6], B = \begin{bmatrix} 1 \\ -5 \end{bmatrix}, C = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, D = \begin{bmatrix} -1 & 0 \\ 0 & -4 \end{bmatrix}$ , and  $E = \begin{bmatrix} -1 & 0 & 3 \\ 1 & 0 & -2 \end{bmatrix}$  be matrices. Compute the following if they exist.

- (a)  $AB$
- (b)  $BA$
- (c)  $AC$
- (d)  $CA$
- (e)  $CE$
- (f)  $EC$
- (g)  $A^2$
- (h)  $D^2$

**Solution:**

- (a)  $A$  is a  $1 \times 2$  matrix and  $B$  is a  $2 \times 1$  matrix. Hence,  $AB$  exists and it is a  $1 \times 1$  matrix.

$$AB = [1 \quad 6] \cdot \begin{bmatrix} 1 \\ -5 \end{bmatrix} = [1 \cdot 1 + 6 \cdot -5] = [-29]$$

- (b)  $B$  is a  $2 \times 1$  matrix and  $A$  is a  $1 \times 2$  matrix. Hence,  $BA$  exists and it is a  $2 \times 2$  matrix.

$$BA = \begin{bmatrix} 1 \\ -5 \end{bmatrix} \cdot [1 \quad 6] = \begin{bmatrix} 1 \cdot 1 & 1 \cdot 6 \\ -5 \cdot 1 & -5 \cdot 6 \end{bmatrix} = \begin{bmatrix} 1 & 6 \\ -5 & -30 \end{bmatrix}$$

- (c)  $A$  is a  $1 \times 2$  matrix and  $C$  is a  $2 \times 2$  matrix. Hence,  $AC$  exists and it is a  $1 \times 2$  matrix.

$$AC = [1 \quad 6] \cdot \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = [1 \cdot 1 + 6 \cdot 3 \quad 1 \cdot 2 + 6 \cdot 4] = [19 \quad 26]$$

- (d)  $C$  is a  $2 \times 2$  matrix and  $A$  is a  $1 \times 2$  matrix. Hence,  $CA$  does not exist.

- (e)  $C$  is a  $2 \times 2$  matrix and  $E$  is a  $2 \times 3$  matrix. Hence,  $CE$  exists and it is a  $2 \times 3$  matrix.

$$\begin{aligned}CE &= \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} -1 & 0 & 3 \\ 1 & 0 & -2 \end{bmatrix} \\&= \begin{bmatrix} 1 \cdot -1 + 2 \cdot 1 & 1 \cdot 0 + 2 \cdot 0 & 1 \cdot 3 + 2 \cdot -2 \\ 3 \cdot -1 + 4 \cdot 1 & 3 \cdot 0 + 4 \cdot 0 & 3 \cdot 3 + 4 \cdot -2 \end{bmatrix} \\&= \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & 1 \end{bmatrix}\end{aligned}$$

- (f)  $E$  is a  $2 \times 3$  matrix and  $C$  is a  $2 \times 2$  matrix. Hence,  $EC$  does not exist.

- (g) A is a  $1 \times 2$  matrix. The product of a  $1 \times 2$  matrix with a  $1 \times 2$  matrix does not exist. Hence,  $A^2$  does not exist.

- (h)  $D$  is a  $2 \times 2$  matrix . Hence,  $D^2 = D \cdot D$  is a  $2 \times 2$  matrix and

$$\begin{aligned}
 D^2 &= D \cdot D = \begin{bmatrix} -1 & 0 \\ 0 & -4 \end{bmatrix} \cdot \begin{bmatrix} -1 & 0 \\ 0 & -4 \end{bmatrix} \\
 &= \begin{bmatrix} -1 \cdot -1 + 0 \cdot 0 & -1 \cdot 0 + 0 \cdot -4 \\ 0 \cdot -1 + (-4) \cdot 0 & 0 \cdot 0 + (-4) \cdot -4 \end{bmatrix} \\
 &= \begin{bmatrix} 1 & 0 \\ 0 & 16 \end{bmatrix}.
 \end{aligned}$$

**Exercise 4:** Let  $A = \begin{bmatrix} 1 & 0 \end{bmatrix}$ ,  $B = \begin{bmatrix} 0 & 2 \\ 3 & 4 \end{bmatrix}$ , and  $C = \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix}$ . Determine the following.

Determine the following.

$$(a) \quad (AB)^T \qquad \qquad \qquad (b) \quad (BC)^T$$

### Solution:

$$\begin{aligned}
 (a) \ AB &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 3 & 4 \end{bmatrix} \\
 &= [1 \cdot 0 + 0 \cdot 3 \quad 1 \cdot 2 + 0 \cdot 4] \\
 &= [0 \quad 6].
 \end{aligned}$$

$$\text{Thus, } (AB)^T = \begin{bmatrix} 0 \\ 6 \end{bmatrix}$$

$$\begin{aligned}
 (b) \quad BC &= \begin{bmatrix} 0 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix} \\
 &= \begin{bmatrix} 0 \cdot 1 + 2 \cdot 0 & 0 \cdot 0 + 2 \cdot 4 \\ 3 \cdot 1 + 4 \cdot 0 & 3 \cdot 0 + 4 \cdot 4 \end{bmatrix} \\
 &= \begin{bmatrix} 0 & 8 \\ 3 & 16 \end{bmatrix}.
 \end{aligned}$$

$$\text{Hence, } (BC)^T = \begin{bmatrix} 0 & 3 \\ 8 & 16 \end{bmatrix}.$$

**Exercise 5:** Let  $A = \begin{bmatrix} 1 & 0 \\ -1 & 4 \end{bmatrix}$ . Express  $A^2 - 2A + I_2$  as a  $2 \times 2$  matrix.

**Solution:**

$$\begin{aligned}
 A^2 &= \begin{bmatrix} 1 & 0 \\ -1 & 4 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -1 & 4 \end{bmatrix} \\
 &= \begin{bmatrix} 1 \cdot 1 + 0 \cdot (-1) & 1 \cdot 0 + 0 \cdot 4 \\ (-1) \cdot 1 + 4 \cdot (-1) & (-1) \cdot 0 + 4 \cdot 4 \end{bmatrix} \\
 &= \begin{bmatrix} 1 & 0 \\ -5 & 16 \end{bmatrix}.
 \end{aligned}$$

Thus,

$$\begin{aligned}
 A^2 - 2A + I_2 &= \begin{bmatrix} 1 & 0 \\ -5 & 16 \end{bmatrix} - 2 \begin{bmatrix} 1 & 0 \\ -1 & 4 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} 1 & 0 \\ -5 & 16 \end{bmatrix} - \begin{bmatrix} 2 & 0 \\ -2 & 8 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} 1 - 2 + 1 & 0 - 0 + 0 \\ -5 - (-2) + 0 & 16 - 8 + 1 \end{bmatrix} \\
 &= \begin{bmatrix} 0 & 0 \\ -3 & 9 \end{bmatrix}.
 \end{aligned}$$

**Exercise 6:** Let  $A$ ,  $B$ , and  $C$  be Boolean matrices such that

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

Determine the following.

- (a)  $A \wedge B$   
 (b)  $A \wedge C$   
 (c)  $(A \wedge B) \vee (A \wedge C)$

**Solution:**

$$\begin{aligned}
 \text{(a)} \quad A \wedge B &= \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} \wedge \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} 1 \wedge 0 & 0 \wedge 0 & 1 \wedge 1 \\ 0 \wedge 1 & 1 \wedge 0 & 0 \wedge 0 \\ 1 \wedge 1 & 1 \wedge 1 & 0 \wedge 1 \end{bmatrix} \\
 &= \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}.
 \end{aligned}$$

$$(b) A \wedge C = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} \wedge \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 \wedge 1 & 0 \wedge 0 & 1 \wedge 0 \\ 0 \wedge 1 & 1 \wedge 0 & 0 \wedge 1 \\ 1 \wedge 0 & 1 \wedge 1 & 0 \wedge 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

$$(c) (A \wedge B) \vee (A \wedge C) = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix} \vee \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0 \vee 1 & 0 \vee 0 & 1 \vee 0 \\ 0 \vee 0 & 0 \vee 0 & 0 \vee 0 \\ 1 \vee 0 & 1 \vee 1 & 0 \vee 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}.$$

**Exercise 7:** Let  $A = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \odot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$ . Find  $x, y$ , and  $z$ .

**Solution:** Now

$$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \odot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} (1 \wedge x) \vee (0 \wedge y) \vee (1 \wedge z) \\ (1 \wedge x) \vee (1 \wedge y) \vee (1 \wedge z) \\ (1 \wedge x) \vee (1 \wedge y) \vee (0 \wedge z) \end{bmatrix}$$

$$= \begin{bmatrix} x \vee 0 \vee z \\ x \vee y \vee z \\ x \vee y \end{bmatrix}$$

$$= \begin{bmatrix} x \vee z \\ x \vee y \vee z \\ x \vee y \end{bmatrix}.$$

Thus,

$$\begin{bmatrix} x \vee z \\ x \vee y \vee z \\ x \vee y \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}.$$

This implies that

$$x \vee z = 0, \quad (4.1)$$

$$x \vee y \vee z = 1, \quad (4.2)$$

and

$$x \vee y = 1. \quad (4.3)$$

Now, (4.1) implies that  $x = 0$  and  $z = 0$ . Either of the two remaining equations implies that  $y = 1$ .

## SECTION REVIEW

### Key Terms

matrix	diagonal matrix	join
rectangular array	identity matrix	Boolean meet
th element	sum	meet
entry	difference	join of meet expression
equal	multiplication	Boolean product
square matrix	transpose	product
zero matrix	symmetric	
diagonal elements	Boolean join	

### Some Key Definitions

- Let  $A = [a_{ij}]_{m \times n}$  and  $B = [b_{ij}]_{m \times n}$  be  $m \times n$  matrices. The sum of  $A$  and  $B$ , written  $A + B$ , is the  $m \times n$  matrix

$$A + B = [c_{ij}]_{m \times n},$$

where  $c_{ij} = a_{ij} + b_{ij}$ , for all  $i$  and  $j$ ,  $i = 1, 2, \dots, m$ ;  $j = 1, 2, \dots, n$ .

2. Let  $A = [a_{ij}]_{m \times n}$  be an  $m \times n$  matrix and  $B = [b_{jk}]_{n \times p}$  be an  $n \times p$  matrix. The multiplication of  $A$  and  $B$ , written  $AB$ , is the  $m \times p$  matrix

$$AB = [c_{ik}]_{m \times p},$$

where

$$c_{ik} = a_{i1}b_{1k} + a_{i2}b_{2k} + \cdots + a_{in}b_{nk} = \sum_{j=1}^n a_{ij}b_{jk},$$

for all  $i$  and  $k$ ,  $i = 1, 2, \dots, m$ ;  $k = 1, 2, \dots, p$ .

3. Let  $A = [a_{ij}]_{m \times n}$  be an  $m \times n$  matrix. The transpose of  $A$ , written  $A^T$ , is the  $n \times m$  matrix defined by

$$A^T = [b_{ij}]_{n \times m},$$

where  $b_{ij} = a_{ji}$  for all  $i$  and  $j$ ,  $i = 1, 2, \dots, n$ ;  $j = 1, 2, \dots, m$ ; i.e., the  $(i, j)$ th element of  $A^T$  is the same as the  $(j, i)$ th element of  $A$ . Notice that the rows of  $A$  are the columns of  $A^T$  and the columns of  $A$  are the rows of  $A^T$ .

4. Let  $A = [a_{ij}]$  be an  $n \times n$  matrix. Then  $A$  is called symmetric if  $a_{ij} = a_{ji}$  for all  $i$  and  $j$ ,  $i = 1, 2, \dots, n$ ;  $j = 1, 2, \dots, n$ ; i.e., the  $(i, j)$ th element of  $A$  is the same as the  $(j, i)$ th element of  $A$ .
5. Let  $A = [a_{ij}]_{m \times n}$  and  $B = [b_{ij}]_{m \times n}$  be  $m \times n$  Boolean matrices.

- (i) The Boolean join (or join) of  $A$  and  $B$ , written  $A \vee B$ , is the  $m \times n$  matrix defined by  $A \vee B = [a_{ij} \vee b_{ij}]_{m \times n}$ .
- (ii) The Boolean meet (or meet) of  $A$  and  $B$ , written  $A \wedge B$ , is the  $m \times n$  matrix defined by  $A \wedge B = [a_{ij} \wedge b_{ij}]_{m \times n}$ .

6. Let  $A = [a_{ij}]_{m \times n}$  and  $B = [b_{jk}]_{n \times p}$  be Boolean matrices. The Boolean product (or product) of  $A$  and  $B$ , written  $A \odot B$ , is the  $m \times p$  matrix defined by  $A \odot B = [c_{ij}]_{m \times p}$ , where

$$c_{ik} = (a_{i1} \wedge b_{1k}) \vee (a_{i2} \wedge b_{2k}) \vee \cdots \vee (a_{in} \wedge b_{nk}) = \bigvee_{j=1}^n (a_{ij} \wedge b_{jk}),$$

for all  $i$  and  $k$ ,  $i = 1, 2, \dots, m$ ;  $k = 1, 2, \dots, p$ .

## Some Key Results

1. Let  $A$ ,  $B$ , and  $C$  be  $m \times n$  matrices. Let  $\mathbf{0}$  denote the  $m \times n$  zero matrix. Then
- (i)  $A + B = B + A$ .
  - (ii)  $A + (B + C) = (A + B) + C$ .
  - (iii)  $A - A = \mathbf{0} = -A + A$ .
  - (iv) If  $k$  is a number, then  $k(A + B) = kA + kB$ ,  $k(A - B) = kA - kB$ .
2. Let  $A$  be an  $m \times n$  matrix,  $B$  be an  $n \times p$  matrix, and  $C$  be an  $p \times q$  matrix. Then
- (i)  $A(BC) = (AB)C$ .
  - (ii)  $I_m A = A = A I_n$ .

3. (i) Let  $A$  and  $B$  be  $m \times n$  matrices. Then  $(A + B)^T = A^T + B^T$ .  
(ii) Let  $A$  be an  $m \times n$  matrix and  $B$  be an  $n \times p$  matrix. Then  $(AB)^T = B^T A^T$ .
4. Let  $A$  and  $B$  be  $n \times n$  symmetric matrices and  $c \in \mathbb{R}$ . Then  $A + B$  and  $cA$  are symmetric matrices.
5. Let  $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n \in \{0, 1\}$ . Then the following conditions are equivalent.
- (i)  $(a_1 \wedge b_1) \vee (a_2 \wedge b_2) \vee \dots \vee (a_n \wedge b_n) = 1$
  - (ii)  $a_i \wedge b_i = 1$  for some  $i, 1 \leq i \leq n$
  - (iii)  $a_i = 1$  and  $b_i = 1$  for some  $i, 1 \leq i \leq n$

## EXERCISES

1. Let  $A$ ,  $B$ , and  $C$  be the matrices defined as follows.

$$A = \begin{bmatrix} 7 & -2 & 5 \\ 1 & 0 & 9 \end{bmatrix}, \quad B = \begin{bmatrix} 12 & 35 & -4 \\ 3 & -49 & 17 \\ 8 & 0 & 35 \end{bmatrix}, \quad C = \begin{bmatrix} 5 \\ 23 \end{bmatrix}$$

Determine the following, if they exist.

- a.  $a_{11}$     b.  $a_{23}$     c.  $b_{13}$     d.  $b_{22}$   
e.  $b_{24}$     f.  $c_{12}$     g.  $c_{21}$

2. Let  $A$ ,  $B$ , and  $C$  be the matrices defined as follows.

$$A = \begin{bmatrix} 6 & 3 & 2 \\ 1 & -2 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 4 & -2 & -1 \\ -2 & 0 & 5 \end{bmatrix}, \quad C = \begin{bmatrix} 2 & -5 & -4 \\ 1 & 3 & 7 \\ -6 & 0 & -1 \end{bmatrix}$$

Compute the following, if they exist.

- a.  $A + B$     b.  $A - B$     c.  $A + C$   
d.  $2A + B$     e.  $A - 2C$

3. Let  $A = \begin{bmatrix} 4 \\ -1 \end{bmatrix}$ ,  $B = \begin{bmatrix} 1 & -6 & 2 \\ 5 & -3 & 1 \\ -4 & 1 & 2 \end{bmatrix}$ ,  $C = \begin{bmatrix} 3 & 2 & -5 \\ -6 & 4 & 8 \\ 0 & 5 & 9 \end{bmatrix}$ ,

and  $D = \begin{bmatrix} 7 \\ -2 \\ 0 \end{bmatrix}$  be matrices. Compute the following if they exist.

- a.  $A + B$     b.  $5A$     c.  $A - 3D$   
d.  $2A - 6D$     e.  $4B + 7C$

4. Let  $A = \begin{bmatrix} 8 & -2 & 4 \\ 0 & -5 & 0 \\ 2 & 6 & -9 \end{bmatrix}$ ,  $B = \begin{bmatrix} 9 & 1 & 10 \\ 11 & 3 & -2 \\ 6 & 7 & -4 \end{bmatrix}$ , and  $C = \begin{bmatrix} 7 & -3 & 0 \\ 12 & 4 & 5 \\ 3 & 0 & 2 \end{bmatrix}$  be matrices. Verify that  $A + (B + C) = (A + B) + C$ .

5. Suppose that  $\begin{bmatrix} a+2b & c+d \\ 2a-b & c-2d \end{bmatrix} = \begin{bmatrix} 5 & -4 \\ 0 & 17 \end{bmatrix}$ . Determine  $a$ ,  $b$ ,  $c$ , and  $d$ .

6. Prove Theorem 4.1.16.

7. Let  $A = \begin{bmatrix} -1 & 0 \\ 0 & -4 \end{bmatrix}$ ,  $B = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ ,  $C = \begin{bmatrix} 4 \\ 9 \end{bmatrix}$ , and  $D = \begin{bmatrix} -2 & 0 & 3 \\ 1 & -7 & 7 \end{bmatrix}$  be matrices. Compute the following if they exist.

- a.  $AB$     b.  $BA$     c.  $AC$   
d.  $CA$     e.  $BD$     f.  $A^2 + B^2$
8. Let  $A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ ,  $B = \begin{bmatrix} -5 & 2 & 4 \\ 4 & -7 & 5 \\ -6 & 0 & 1 \end{bmatrix}$ ,  $C = \begin{bmatrix} 6 & 6 & 3 \end{bmatrix}$ , and  $D = \begin{bmatrix} 4 & 1 \\ -5 & 3 \\ 2 & 0 \end{bmatrix}$  be matrices. Compute the following if they exist.

- a.  $AB$     b.  $BA$     c.  $AC$     d.  $CA$   
e.  $BD$     f.  $DB$     g.  $A^2$     h.  $CD$
9. Let  $A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -2 & 4 \end{bmatrix}$ ,  $B = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ ,  $C = \begin{bmatrix} 2 & 3 \\ 4 & -1 \\ 0 & 2 \end{bmatrix}$ ,  $D = \begin{bmatrix} 7 & 1 \\ 2 & -5 \end{bmatrix}$ ,  $E = \begin{bmatrix} 3 & -2 \\ 4 & 0 \end{bmatrix}$  be matrices. Compute the following if they exist.

- a.  $AB$     b.  $BC$   
c.  $DA$     d.  $DA - 5BC$   
e.  $D^2 + E^2$     f.  $CE + CD$   
g.  $C(E + D)$
10. Let  $A$  be a  $4 \times 3$  matrix,  $B$  be a  $3 \times 4$  matrix,  $C$  be a  $4 \times 2$  matrix,  $D$  be a  $2 \times 4$  matrix,  $E$  be a  $7 \times 4$  matrix, and  $F$  be a  $3 \times 5$  matrix. Determine which of the following matrix expressions are valid. If a matrix expression is valid, then give the size of the matrix given the expression.

- a.  $AB$     b.  $BC$     c.  $BCD$   
d.  $ABCDE$     e.  $AB + CE$     f.  $2(CDE) + EB$

11. Let  $I_3$  be the identity matrix of size  $3 \times 3$  and  $\mathbf{0}_3$  be the  $3 \times 3$  zero matrix. Let  $A$  and  $B$  be matrices given by:

$$A = \begin{bmatrix} 3 & 2 & -1 \\ 0 & 5 & 0 \\ 2 & 7 & 9 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 1 & 2 & -3 \\ -4 & 5 & 6 \\ 0 & 9 & 5 \end{bmatrix}.$$

Verify the following expressions.

- a.  $I_3 A = A = A I_3$     b.  $A + \mathbf{0}_3 = A = \mathbf{0}_3 + A$   
c.  $\mathbf{0}_3 B = \mathbf{0}_3 = B \mathbf{0}_3$
12. Let  $A = \begin{bmatrix} 0 & 2 & -4 \\ 1 & 5 & 6 \\ 2 & 0 & 9 \end{bmatrix}$  and  $B = \begin{bmatrix} 2 & -1 & -3 \\ 0 & 4 & 5 \\ 1 & 2 & 0 \end{bmatrix}$  be matrices. Let  $C = AB$  and  $D = BA$ . Determine the following

entries of  $C$  and  $D$  without completely computing matrices  $C$  and  $D$ .

- a.  $c_{13}$
- b.  $c_{22}$
- c.  $c_{32}$
- d.  $d_{11}$
- e.  $d_{23}$
- f.  $d_{33}$

13. Let  $A = \begin{bmatrix} 1 & -2 \\ -5 & 2 \\ 7 & 9 \end{bmatrix}$  and  $B = \begin{bmatrix} 0 & 1 & 2 \\ 0 & 4 & 0 \end{bmatrix}$  be matrices. Let  $C = AB$  and  $D = BA$ .

- a. What are the sizes of  $C$  and  $D$ ?
- b. Compute the following entries, if they exist, of matrices  $C$  and  $D$  completely computing  $C$  and  $D$ .
- a.  $c_{22}$
- b.  $c_{32}$
- c.  $d_{11}$
- d.  $d_{21}$
- e.  $d_{32}$

14. Let  $A = [a_{ij}]_{m \times n}$  be an  $m \times n$  matrix and  $B = [b_{jk}]_{n \times p}$  be an  $n \times p$  matrix.

- a. If the  $i$ th row,  $1 \leq i \leq m$ , of  $A$  is all zero, then prove that the  $i$ th row of  $AB$  is zero.
- b. If the  $j$ th column,  $1 \leq j \leq p$ , of  $B$  is all zero, then prove that the  $j$ th column of  $AB$  is zero.

15. Let  $A = [a_{ij}]_{m \times n}$  be an  $m \times n$  matrix and  $B = [b_{jk}]_{n \times p}$  be an  $n \times p$  matrix. Let

$$B_j = \begin{bmatrix} b_{1j} \\ b_{2j} \\ \vdots \\ b_{nj} \end{bmatrix}$$

denote the  $j$ th column of  $B$ ,  $j = 1, 2, \dots, p$ . Then we can write  $B = [B_1 \ B_2 \ \dots \ B_j \ \dots \ B_p]$ . Let  $C = AB$ . Using the same convention, we can write  $C = [C_1 \ C_2 \ \dots \ C_j \ \dots \ C_p]$ , where  $C_j$  denotes the  $j$ th column of  $C$ . Prove that  $C_j = AB_j$  for all  $j$ ,  $j = 1, 2, \dots, p$ .

16. Let  $A = \begin{bmatrix} 1 & 2 & 3 \\ 0 & -1 & 2 \\ 4 & 0 & 1 \end{bmatrix}$ ,  $B = \begin{bmatrix} 2 & 4 \\ 1 & -2 \\ 0 & 1 \end{bmatrix}$ , and  $C = \begin{bmatrix} 4 & 1 \\ 1 & -6 \end{bmatrix}$  be matrices. Verify that  $A(BC) = (AB)C$ .

17. Let  $A = \begin{bmatrix} 2 & 3 & 4 \\ 4 & 7 & 0 \\ 1 & 6 & -9 \end{bmatrix}$ ,  $B = \begin{bmatrix} 3 & -4 \\ 0 & 3 \\ -2 & 7 \end{bmatrix}$ , and  $C = \begin{bmatrix} 3 & 1 \\ 0 & 6 \\ -6 & 4 \end{bmatrix}$  be matrices. Verify that  $A(B + C) = AB + AC$ .

18. Let  $A$  be a  $5 \times 3$  matrix,  $B$  be a  $3 \times 4$  matrix,  $C$  be a  $9 \times 7$  matrix, and  $D$  be a  $4 \times 9$  matrix. Determine the size of the following matrix expressions, if they exist.
- a.  $ABC$
  - b.  $ABD$
  - c.  $BDC$
  - d.  $ABDC$

19. Let  $A$  and  $B$  be  $n \times n$  matrices. Simplify the following matrix expression.

- a.  $A(2A - 3B) + 3B(A + B) - 2A^2 - 3B^2 + 4AB$
- b.  $(A + B)^2 - (A + B)(A - B)$

20. Let  $A = \begin{bmatrix} 1 & 0 \\ -1 & 0 \end{bmatrix}$  be a matrix. Determine all matrices  $B$  such that  $AB = BA$ .

21. Let  $A$  and  $B$  be  $n \times n$  diagonal matrices. Prove that  $AB = BA$ .

22. Prove Theorem 4.1.19.

23. An  $n \times n$  matrix  $A$  is called **idempotent** if  $A^2 = A$ . Let  $A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$ . Show that  $A$  is idempotent.

24. Determine  $a$ ,  $b$ , and  $c$  such that  $\begin{bmatrix} 1 & a \\ b & c \end{bmatrix}$  is idempotent.

25. Determine  $a$ ,  $b$ , and  $c$  such that  $\begin{bmatrix} a & b \\ 0 & c \end{bmatrix}$  is idempotent.

26. Prove that if  $A$  and  $B$  are idempotent matrices and  $AB = BA$ , then  $AB$  is idempotent.

27. Prove that if  $A$  is an idempotent matrix and  $m$  is a positive integer, then  $A^m = A$ .

28. A matrix  $A$  is called **nilpotent** if there exists a positive integer  $m$  such that  $A^m = \mathbf{0}$ . The least positive integer  $p$  such that  $A^p = \mathbf{0}$  is called the **degree of nilpotency**. Let  $A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ . Show that  $A$  is nilpotent. What is the degree of nilpotency?

29. Prove that if the matrix  $A$  is idempotent and nilpotent, then  $A = \mathbf{0}$ .

30. Let  $A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ . Prove that  $A^2 = I_2$ . Is  $A$  nilpotent? Justify your answer.

31. Let  $A$  be a diagonal matrix. Prove that  $A^T = A$ .

32. Let  $A$  be an  $n \times n$  matrix. Prove that for all positive integers  $m$ ,  $(A^m)^T = (A^T)^m$ .

33. Let  $A$  be an  $n \times n$  matrix. Prove that  $A$  is idempotent if and only if  $A^T$  is idempotent.

34. Let  $A$  and  $B$  be  $m \times n$  matrices. Prove that  $A = B$  if and only if  $A^T = B^T$ .

35. Prove Theorem 4.1.23.

36. Let  $A = \begin{bmatrix} 1 & 9 & 3 \\ * & 2 & * \\ * & -7 & 13 \end{bmatrix}$ . Determine the entries indicated with a \* so that the matrix  $A$  is symmetric.

37. Prove Theorem 4.1.26.

38. Let  $a, b, c \in \{0, 1\}$ . Verify that

- a.  $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$ .
- b.  $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ .

39. Let  $A$ ,  $B$ , and  $C$  be Boolean matrices such that

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}, \quad C = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

Determine the following.

- a.  $A \wedge B$
- b.  $A \vee B$
- c.  $B \vee C$
- d.  $(A \vee B) \vee C$
- e.  $(A \wedge B) \wedge C$
- f.  $A \vee (B \wedge C)$
- g.  $A \wedge (B \vee C)$

40. Let  $A$ ,  $B$ , and  $C$  be Boolean matrices such that

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

Determine the following.

- a.  $A \odot B$
- b.  $B \odot C$
- c.  $(A \odot B) \odot C$
- d.  $A \odot (B \odot C)$

41. Let  $A$ ,  $B$ ,  $C$  be  $m \times n$  Boolean matrices. Prove that

- a.  $A \wedge B = B \wedge A$
- b.  $A \vee B = B \vee A$
- c.  $A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$ .
- d.  $A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$ .

42. Let  $A$  be an  $m \times n$  Boolean matrix,  $B$  be an  $n \times p$  Boolean matrix, and  $C$  be a  $p \times q$  Boolean matrix. Prove that

$$A \odot (B \odot C) = (A \odot B) \odot C.$$

## 4.2 THE MATRIX OF A RELATION AND CLOSURES

We are now ready to discuss how to represent a relation on a finite set as a matrix. We will then discuss how to use the matrix representation to determine the transitive closure as well as reflexive and symmetric closures.

Let  $A = \{a_1, a_2, \dots, a_n\}$  and  $B = \{b_1, b_2, \dots, b_p\}$  be finite nonempty sets. Let  $R$  be a relation from  $A$  into  $B$ . Then  $R \subseteq A \times B$ . Let

$$M_R = [m_{ij}]_{n \times p}$$

be the Boolean  $n \times p$  matrix, where

$$m_{ij} = \begin{cases} 1 & \text{if } (a_i, b_j) \in R, \text{ i.e., } a_i R b_j, \\ 0 & \text{otherwise.} \end{cases}$$

The matrix  $M_R$  is called the **matrix of the relation  $R$** .

Let us elaborate on  $M_R$  a bit more. Let us write  $M_R$  in the expanded form as:

$$M_R = \begin{bmatrix} m_{11} & m_{12} & \dots & m_{1j} & \dots & m_{1p} \\ m_{21} & m_{22} & \dots & m_{2j} & \dots & m_{2p} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ m_{i1} & m_{i2} & \dots & m_{ij} & \dots & m_{ip} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ m_{n1} & m_{n2} & \dots & m_{nj} & \dots & m_{np} \end{bmatrix}.$$

Let us consider the elements of row 1. If  $m_{11} = 1$ , then  $(a_1, b_1) \in R$  and if  $m_{12} = 1$ , then  $(a_1, b_2) \in R$ . In general, if  $m_{1j} = 1$ , then  $(a_1, b_j) \in R$ , where  $1 \leq j \leq p$ . It follows that the elements of the first row indicate the elements of  $B$  to which  $a_1$  is related. Similarly, the elements of the second row indicate the elements of  $B$  to which  $a_2$  is related. In general, the elements of row  $i$  indicate the elements of  $B$  to which  $a_i$  is related, where  $1 \leq i \leq n$ .

Now consider the elements of column 1 : If  $m_{11} = 1$ , then  $(a_1, b_1) \in R$ ; if  $m_{21} = 1$ , then  $(a_2, b_1) \in R$ ; in general, if  $m_{i1} = 1$ , then  $(a_i, b_1) \in R$ , where  $1 \leq i \leq n$ . It follows that the elements of the first column indicate the elements of  $A$  that are related to  $b_1$ . Similarly, the elements of the second column indicate the elements of  $A$  that are related to  $b_2$ . In general, the elements of column  $j$  indicate the elements of  $A$  that are related to  $b_j$ , where  $1 \leq j \leq p$ .

### EXAMPLE 4.2.1

Let  $A = \{a, b, c, d\}$  and  $B = \{1, 2, 3\}$ . Let  $R = \{(a, 1), (a, 3), (b, 2), (b, 3), (d, 1)\}$ . If we let  $a_1 = a$ ,  $a_2 = b$ ,  $a_3 = c$ ,  $a_4 = d$ ,  $b_1 = 1$ ,  $b_2 = 2$ , and  $b_3 = 3$ , then

$$M_R = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}. \quad (4.4)$$

If we do not number the elements of  $A$  and  $B$ , then sometimes we write  $M_R$  as:

$$M_R = \begin{array}{c} 1 \ 2 \ 3 \\ a \left[ \begin{array}{ccc} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{array} \right] \\ b \\ c \\ d \end{array},$$

indicating that the rows are numbered according to the elements of  $A$  and the columns are numbered according to the elements of  $B$ . If no confusion arises, then we write  $M_R$  as in (4.4).

Given a relation from a finite set into a finite set, we can write the matrix of the relation as described above. Notice that in the above definition, we have assumed an ordering of the elements of the given set. In Example 4.2.1, we assumed that  $a_1 = a$ ,  $a_2 = b$ ,  $a_3 = c$ ,  $a_4 = d$ ,  $b_1 = 1$ ,  $b_2 = 2$ , and  $b_3 = 3$ . This means that the elements of the set  $A$  are arranged as  $a$ -1st,  $b$ -2nd,  $c$ -3rd,  $d$ -4th and the elements of  $B$  are arranged as 1-1st, 2-2nd, 3-3rd. Using this ordering, we defined the elements of rows and columns of the associated matrix. When we discussed the graphical representation of a relation, we pointed out that how we draw the digraph is not important; the main point is to show the relationship between the vertices. Likewise in the case of matrix representation, how we arrange the elements of the set is not important, the main point is to show the relationship between the elements. Let us again consider Example 4.2.1.

Let  $A = \{a, b, c, d\}$  and  $E = \{1, 2, 3\}$ . Let  $R = \{(a, 1), (a, 3), (b, 2), (b, 3), (d, 1)\}$ . Suppose the elements of  $A$  are arranged as:  $b$  the 1st element,  $c$  the 2nd element,  $a$  the 3rd element, and  $d$  the 4th element; and the elements of  $B$  are arranged as 3 the 1st element, 1 the 2nd element, and 2 the 3rd element. Then the matrix of  $R$  is (for convenience we show how the rows and columns correspond to the elements of  $A$  and  $B$ , respectively):

$$M_R = \begin{matrix} & \begin{matrix} 3 & 1 & 2 \end{matrix} \\ \begin{matrix} b \\ c \\ a \\ d \end{matrix} & \left[ \begin{matrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{matrix} \right] \end{matrix}.$$

This matrix is different than the matrix given in Example 4.2.1, yet it also shows the same relationship between the elements. For example, in this matrix the element at the (3, 2)th position, i.e., at the third row and second column position, is 1, which implies that  $(a, 1) \in R$ ; the element in the (4, 1)th position is 0, which implies that  $(d, 3) \notin R$ .

---

**REMARK 4.2.2** ► If in the discussion we do not specify any ordering of the elements of the set, we assume that the elements are ordered as they are listed in the set. Using this convention only, we draw the matrix of the relation.

We now consider the reverse situation.

Let  $A = \{a_1, a_2, \dots, a_n\}$  and  $B = \{b_1, b_2, \dots, b_p\}$  be finite nonempty sets. Notice that we have numbered the elements of the sets. We use this ordering in the discussion. Let  $S = [s_{ij}]_{n \times p}$  be a Boolean matrix. Then we can define a relation  $R$  from  $A$  into  $B$  such that  $M_R := S$  as follows: Let  $R$  be the set

$$R = \{(a_i, b_j) \mid s_{ij} = 1, i = 1, 2, \dots, n, j = 1, 2, \dots, p\}.$$

Then  $R \subseteq A \times B$ . Notice that  $(a_i, b_j) \in R$  if and only if  $s_{ij} = 1$  for all  $i$  and  $j$ .

Let  $M_R = [m_{ij}]_{n \times p}$  be the matrix of the relation  $R$ . Now  $m_{ij} = 1$  if and only if  $(a_i, b_j) \in R$  if and only if  $s_{ij} = 1$  for all  $i$  and  $j$ . Hence,  $m_{ij} = s_{ij}$  for all  $i$  and  $j$ . This implies that  $M_R = S$ .

**EXAMPLE 4.2.3**

Let  $A = \{a, b, c\}$  and let  $R$  be a relation on  $A$  whose matrix is

$$M_R = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}.$$

We did not indicate the ordering of the elements of  $A$ . So the elements of  $A$  are ordered as they are listed in the set. Therefore,  $a$  is the 1st element,  $b$  is the 2nd element, and  $c$  is the 3rd element. Now from the matrix, the (1, 1)th element is 1. Thus,  $(a, a) \in R$ . Also the (1, 2)th element is 0 implies  $(a, b) \notin R$ . Thus, we find that the relation  $R$  is  $\{(a, a), (a, c), (b, b), (c, a), (c, b)\}$ .

The matrix of a relation is very useful in computing the elements of a relation. For example, the following theorem shows how to determine the union, intersection, and inverse of a relation.

**Theorem 4.2.4:** Let  $A = \{a_1, a_2, \dots, a_n\}$  and  $B = \{b_1, b_2, \dots, b_p\}$  be finite nonempty sets. Let  $R$  and  $S$  be relations from  $A$  into  $B$ . Then

- (i)  $M_{R \cup S} = M_R \vee M_S$ ,
- (ii)  $M_{R \cap S} = M_R \wedge M_S$ ,
- (iii)  $M_{R^{-1}} = (M_R)^T$ , the transpose of  $M_R$ .

**Proof:** We only prove part (i) and leave the others as exercises.

- (i) Let  $M_{R \cup S} = [m_{ij}]$ ,  $M_R = [r_{ij}]$ , and  $M_S = [s_{ij}]$ . Then  $M_R \vee M_S = [r_{ij} \vee s_{ij}]$ . Now the  $(i, j)$ th element of  $M_{R \cup S}$  is  $m_{ij}$  and the  $(i, j)$ th element of  $M_R \vee M_S$  is  $r_{ij} \vee s_{ij}$ , for all  $i, j, i = 1, 2, \dots, n; j = 1, 2, \dots, p$ . We show that  $m_{ij} = 1$  if and only if  $r_{ij} \vee s_{ij} = 1$ .

First suppose that  $m_{ij} = 1$ . This implies that  $(a_i, b_j) \in R \cup S$ , which in turn implies that  $(a_i, b_j) \in R$  or  $(a_i, b_j) \in S$ . If  $(a_i, b_j) \in R$ , then  $r_{ij} = 1$ , and if  $(a_i, b_j) \in S$ , then  $s_{ij} = 1$ . Thus, either  $r_{ij} = 1$  or  $s_{ij} = 1$ . This implies that  $r_{ij} \vee s_{ij} = 1$ .

Now suppose that  $r_{ij} \vee s_{ij} = 1$ . Then  $r_{ij} = 1$  or  $s_{ij} = 1$ . If  $r_{ij} = 1$ , then  $(a_i, b_j) \in R$ , and if  $s_{ij} = 1$ , then  $(a_i, b_j) \in S$ . Thus,  $(a_i, b_j) \in R$  or  $(a_i, b_j) \in S$ . This implies that  $(a_i, b_j) \in R \cup S$ . Hence  $m_{ij} = 1$ .

Because  $m_{ij} = 1$  if and only if  $r_{ij} \vee s_{ij} = 1$ , for all  $i, j, i = 1, 2, \dots, n; j = 1, 2, \dots, p$ , we must have  $M_{R \cup S} = M_R \vee M_S$ . ■

**EXAMPLE 4.2.5**

Let  $A$  and  $B$  be finite sets. Let  $R$  and  $S$  be relations from  $A$  into  $B$  such that

$$M_R = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \quad \text{and} \quad M_S = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Because  $M_{R \cup S} = M_R \vee M_S$ , we have

$$\begin{aligned} M_{R \cup S} &= \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \vee \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 \vee 0 & 0 \vee 1 & 1 \vee 1 \\ 1 \vee 0 & 0 \vee 0 & 0 \vee 1 \\ 0 \vee 0 & 1 \vee 0 & 0 \vee 0 \\ 0 \vee 0 & 1 \vee 0 & 1 \vee 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}. \end{aligned}$$

Similarly,

$$\begin{aligned} M_{R \cap S} &= \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \wedge \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 \wedge 0 & 0 \wedge 1 & 1 \wedge 1 \\ 1 \wedge 0 & 0 \wedge 0 & 0 \wedge 1 \\ 0 \wedge 0 & 1 \wedge 0 & 0 \wedge 0 \\ 0 \wedge 0 & 1 \wedge 0 & 1 \wedge 1 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \end{aligned}$$

Also,

$$M_{R^{-1}} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}^T = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

Recall that  $M_{R^{-1}}$  is the matrix of the relation from  $B$  into  $A$ .

The next theorem shows how to compute the matrix of the composition of relations.

**Theorem 4.2.6:** Let  $A$ ,  $B$ , and  $C$  be finite sets. Let  $R$  be a relation from  $A$  into  $B$  and  $S$  be a relation from  $B$  into  $C$ . Then

$$M_{S \circ R} = M_R \odot M_S;$$

i.e., the matrix  $M_{S \circ R}$  of the relation  $S \circ R$  is same as the matrix of the Boolean products of  $M_R$  and  $M_S$ .

**Proof:** Let  $A = \{a_1, a_2, \dots, a_n\}$ ,  $B = \{b_1, b_2, \dots, b_p\}$ , and  $C = \{c_1, c_2, \dots, c_q\}$ . Then  $M_R$  is an  $n \times p$  matrix and  $M_S$  is a  $p \times q$  matrix. Hence, the product  $M_R \odot M_S$  is defined and is an  $n \times q$  matrix.

Now  $S \circ R$  is a relation from  $A$  into  $C$ . Because  $A$  has  $n$  elements and  $C$  has  $q$  elements,  $M_{S \circ R}$  is an  $n \times q$  matrix.

Let us write  $M_R = [r_{ij}]$ ,  $M_S = [s_{ij}]$ ,

$$M_{S \circ R} = [m_{ij}], \text{ and } M_R \odot M_S = [t_{ij}].$$

To show that  $M_{S \circ R} = M_R \odot M_S$ , we show that their corresponding entries are the same. Because these are Boolean matrices, we only need to show that  $m_{ij} = 1$  if and only if  $t_{ij} = 1$ , for all  $i, j$ ,  $i = 1, 2, \dots, n$ ;  $j = 1, 2, \dots, q$ .

Suppose that  $m_{ij} = 1$ . This implies that  $(a_i, c_j) \in S \circ R$ . From this and by the definition of the composition of relations, it follows that there exists  $b_k \in B$ ,  $1 \leq k \leq p$ , such that  $(a_i, b_k) \in R$  and  $(b_k, c_j) \in S$ .

Because  $(a_i, b_k) \in R$ , we have  $r_{ik} = 1$ . Similarly, because  $(b_k, c_j) \in S$ , we have  $s_{kj} = 1$ . Now,  $t_{ij}$  is the  $(i, j)$ th element of the Boolean product  $M_R \odot M_S$ . Hence, by the definition of the product of the Boolean matrices

$$\begin{aligned} t_{ij} &= (r_{i1} \wedge s_{1j}) \vee \dots \vee (r_{ik} \wedge s_{kj}) \vee \dots \vee (r_{ip} \wedge s_{pj}) \\ &= (r_{i1} \wedge s_{1j}) \vee \dots \vee (1 \wedge 1) \vee \dots \vee (r_{ip} \wedge s_{pj}) \\ &= (r_{i1} \wedge s_{1j}) \vee \dots \vee 1 \vee \dots \vee (r_{ip} \wedge s_{pj}) \\ &= 1, \end{aligned} \quad \text{by Theorem 4.1.30.}$$

Now suppose that  $t_{ij} = 1$ . This implies that

$$1 = t_{ij} = (r_{i1} \wedge s_{1j}) \vee \dots \vee (r_{ik} \wedge s_{kj}) \vee \dots \vee (r_{ip} \wedge s_{pj}).$$

By Theorem 4.1.30, we must have  $r_{ik} \wedge s_{kj} = 1$  for some  $k$ ,  $1 \leq k \leq p$ . This implies that  $r_{ik} = 1$  and  $s_{kj} = 1$  for some  $k$ ,  $1 \leq k \leq p$ . Because  $r_{ik} = 1$ , we must have  $(a_i, b_k) \in R$ . Similarly, because  $s_{kj} = 1$ , we must have  $(b_k, c_j) \in S$ .

Now  $(a_i, b_k) \in R$  and  $(b_k, c_j) \in S$ , and therefore by the definition of the composition of relations,  $(a_i, c_j) \in S \circ R$ . This implies that  $m_{ij} = 1$ .

We have thus proved that  $m_{ij} = 1$  if and only if  $t_{ij} = 1$ , for all  $i, j$ ,  $i = 1, 2, \dots, n$ ;  $j = 1, 2, \dots, q$ . It now follows that  $M_{S \circ R} = M_R \odot M_S$ . ■

The following corollary is an immediate consequence of Theorem 4.2.6.

**Corollary 4.2.7:** Let  $A$  be a finite set and  $R$  be a relation on  $A$ , i.e.,  $R \subseteq A \times A$ . Let  $M_{R^2}$  be the matrix of the relation  $R^2 = R \circ R$ . Then

$$M_{R^2} = M_R \odot M_R.$$

As in the proof Theorem 3.1.33, we can show that a relation  $R$  on a set  $A$  is transitive if and only if  $R \circ R \subseteq R$ . If  $R$  is a relation on a finite set  $A$ , then Corollary 4.2.7 implies that if  $M_R \odot M_R = M_R$ , then  $R^2 = R$ . Therefore,  $R$  is a transitive relation.

#### EXAMPLE 4.2.8

Let  $A = \{a, b, c\}$  and  $R$  be a relation on  $A$  whose matrix is

$$M_R = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}.$$

Now

$$\begin{aligned} M_R \odot M_R &= \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \\ &= M_R. \end{aligned}$$

Thus,  $R$  is transitive. Let us verify this directly by considering the elements of  $R$ . From the matrix of  $R$ , we have  $R = \{(a, a), (b, a), (c, a), (c, b), (c, c)\}$ . In this relation, we find that whenever  $a R b$  and  $b R c$  hold,  $a R c$  also holds. Hence,  $R$  is transitive.

The following theorem allows us to determine the relation  $R^n$  on a finite  $A$  for any positive integer  $n$ . It also generalizes Corollary 4.2.7.

**Theorem 4.2.9:** Let  $A$  be a finite set and  $R$  be a relation on  $A$ , i.e.,  $R \subseteq A \times A$ . Let  $M_{R^n}$  be the matrix of the relation  $R^n$ . Then for all  $n \geq 1$ ,

$$M_{R^n} = \underbrace{M_R \odot M_R \odot \cdots \odot M_R}_{n \text{ times}}$$

**Proof:** We prove the result by induction on  $n$ .

*Basis step:* Suppose  $n = 1$ . Then  $R^n = R^1 = R$ . Therefore,  $M_{R^n} = M_{R^1} = M_R$ .

*Inductive hypothesis:* Suppose that the result is true for  $n = k$ , i.e.,

$$M_{R^k} = \underbrace{M_R \odot M_R \odot \cdots \odot M_R}_{k \text{ times}},$$

where  $k \geq 1$ .

*Inductive step:* We wish to show that

$$M_{R^{k+1}} = \underbrace{M_R \odot M_R \odot \cdots \odot M_R}_{k+1 \text{ times}}$$

By definition,  $R^{k+1} = R \circ R^k$ . Let us write  $S = R^k$ . Then  $R^{k+1} = R \circ S$ . Hence

$$M_{R^{k+1}} = M_{R \circ S} = M_S \odot M_R \quad (4.5)$$

by Theorem 4.2.6. Now by the inductive hypothesis,

$$M_S = M_{R^k} = \underbrace{M_R \odot M_R \odot \cdots \odot M_R}_{k \text{ times}}$$

Substitute  $M_S$  into (4.5), to get

$$\begin{aligned} M_{R^{k+1}} &= M_S \odot M_R \\ &= \underbrace{M_R \odot M_R \odot \cdots \odot M_R}_{k \text{ times}} \odot M_R \\ &= \underbrace{M_R \odot M_R \odot \cdots \odot M_R}_{k+1 \text{ times}} \end{aligned}$$

This proves the inductive step. Hence, by induction,  $n \geq 1$ ,

$$M_{R^n} = \underbrace{M_R \odot M_R \odot \cdots \odot M_R}_{n \text{ times}}. \blacksquare$$

**Theorem 4.2.10:** Let  $A$  be a finite set with  $n$  elements and  $R$  be a relation on  $A$ . Let  $M_R = [m_{ij}]$  be the matrix of  $R$ . Then

- (i)  $R$  is reflexive if and only if  $m_{ii} = 1$ , for all  $i = 1, 2, \dots, n$ .
- (ii)  $R$  is symmetric if and only if  $M_R = M_{R^{-1}} = (M_R)^T$ , where  $(M_R)^T$  is the transpose of the matrix  $M_R$ .

Theorem 4.2.10 tells us that from the matrix representation of a relation we can immediately determine whether the relation is reflexive and/or symmetric. For reflexivity, we only need to examine the diagonal elements of the matrix. The relation is reflexive if and only if all the diagonal elements are 1. Also the relation is symmetric if and only if the corresponding matrix  $M_R$  is symmetric, i.e.,  $a_{ij} = a_{ji}$ , for all  $i$  and  $j$ . For example, suppose  $R$  is the relation on the set  $A = \{a_1, a_2, a_3, a_4\}$  defined by the matrix

$$M_R = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

Because all the diagonal elements are 1,  $R$  is reflexive. However,  $a_{12} = 1 \neq 0 = a_{21}$ . Therefore,  $M_R$  is not a symmetric matrix. Hence,  $R$  is not symmetric.

Now consider the relation  $R$  on  $A = \{a, b, c\}$  defined by the matrix

$$M_R = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}.$$

Because  $M_R$  is symmetric,  $R$  is symmetric. Now the diagonal element at position  $(2, 2)$  is not 1. Therefore,  $R$  is not reflexive. Let us examine this by direct computation. For this matrix,  $R = \{(a, a), (a, b), (b, a), (b, c), (c, b), (c, c)\}$ . In this relation, we find that for all  $a, b \in A$  whenever  $a R b$  holds  $b R a$  also holds. Also  $(b, b) \notin R$ . Hence,  $R$  is symmetric but not reflexive.

In a similar manner, we can construct a relation such that the matrix of the relation would indicate that the relation is both reflexive and symmetric.

The following theorem follows from Theorem 3.1.57.

**Theorem 4.2.11:** Let  $A = \{a_1, a_2, \dots, a_n\}$  be a finite set  $n \geq 1$ . Let  $R$  be a relation on  $A$ . Then

$$M_{R^\infty} = M_R \vee M_{R^2} \vee \cdots \vee M_{R^n}.$$

Theorem 4.2.11 tells us that to determine the transitive closure of a relation  $R$  on a finite set  $A$  with  $n$  elements, we can compute the matrices  $M_R, M_{R^2}, \dots$ , and  $M_{R^n}$  and then take their Boolean join. The following example shows how to determine the transitive closure by determining these matrices and applying Theorem 4.2.11.

**EXAMPLE 4.2.12**

Let  $A = \{a_1, a_2, a_3, a_4, a_5\}$ . Let  $R$  be a relation on  $A$  given by:

$$R = \{(a_1, a_1), (a_1, a_2), (a_1, a_4), (a_2, a_3), (a_3, a_3), (a_3, a_5), (a_4, a_4), (a_5, a_2)\}.$$

The matrix  $M_R$  is:

$$M_R = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

Now,

$$M_{R^2} = M_R \odot M_R = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix},$$

$$M_{R^3} = M_R \odot M_{R^2} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Similarly,

$$M_{R^4} = M_R \odot M_{R^3} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

$$\text{and } M_{R^5} = M_R \odot M_{R^4} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

Hence,

$$M_{R^\infty} = M_R \vee M_{R^2} \vee M_{R^3} \vee M_{R^4} \vee M_{R^5} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

From the matrix  $M_{R^\infty}$ , we can conclude that the transitive closure of  $R$  is:

$$\begin{aligned} R^\infty = \{(a_1, a_1), (a_1, a_2), (a_1, a_3), (a_1, a_4), (a_1, a_5), \\ (a_2, a_2), (a_2, a_3), (a_2, a_5), \\ (a_3, a_2), (a_3, a_3), (a_3, a_5), \\ (a_4, a_4), (a_5, a_2), (a_5, a_3), (a_5, a_5)\}. \end{aligned}$$

**REMARK 4.2.13** ▶ Let  $R$  be a relation of a finite set  $A$ . Let us show how expensive it would be to determine the transitive closure using Theorem 4.2.11. The matrix  $M_R$  of  $R$  is an  $n \times n$  matrix. Thus, there are  $n^2$  elements in  $M_R$ . To determine the transitive closure of  $R$  using Theorem 4.2.11, we need to compute the matrices  $M_{R^2}, M_{R^3}, \dots, M_{R^n}$ , and then take their Boolean join. Now  $M_{R^2} = M_R \odot M_R$ . Therefore, to determine the, say  $(i, j)$ th, element of  $M_{R^2}$ , we take the  $i$ th row of  $M_R$  and  $j$ th column of  $M_R$ . Next we take the Boolean meet of the corresponding elements and the Boolean join of the result. This may require  $n$  Boolean meet operations and  $n - 1$  Boolean join expressions. A total of  $n + (n - 1)$  operations to determine one element of  $M_{R^2}$ . Because there are  $n^2$  elements, we may need a total of  $(n + (n - 1))n^2 \approx 2n^3$  for large values of  $n$ . (Here the symbol  $\approx$  stands for approximately equal to.) It follows that to determine these matrices, in the worst case, we need approximately  $n \cdot 2n^3 = 2n^4$  operations. Finally, to determine the transitive closure, we need to take the Boolean join of these  $n$  matrices, which would require another  $n^2 \cdot n$  Boolean join operations. As we can see, in terms of computer time, for large values of  $n$ , determining the transitive closure using Theorem 4.2.11 could be very time-consuming. Hence, this may not be an efficient way of determining the transitive closure. In the next section, we describe an efficient algorithm to determine the transitive closure.

The following algorithm uses Theorem 4.2.11 to determine the transitive closure of a relation.

#### ALGORITHM 4.3: Compute the transitive closure.

*Input:*  $M$ —Boolean matrices of the relation  $R$   
 $n$ —positive integers such that  $n \times n$  specifies the size of  $M$

*Output:*  $T$ —an  $n \times n$  Boolean matrix such that  $T = M_{R^\infty}$

1. **procedure** **transitiveClosure**( $M, T, n$ )
2. **begin**
3.    $A = M;$
4.    $T = M;$
5.   **for**  $i := 2$  **to**  $n$  **do**
6.     **begin**
7.        $A = A \odot M; // A = M^i$
8.        $T = T \vee A; // T = M \vee M^2 \vee \dots \vee M^i$
9.     **end**
10. **end**

## Warshall's Algorithm for Determining the Transitive Closure

The preceding section describes how to find the transitive closure of a relation  $R$  by computing the matrices  $M_{R^n}$  and then taking the Boolean join. We remarked that this way of determining the transitive closure is expensive in terms of computer time. In this section, we describe an efficient algorithm, called **Warshall's algorithm**, to determine the transitive closure.

Let  $A = \{a_1, a_2, \dots, a_n\}$  be a finite set,  $n \geq 1$ , and let  $R$  be a relation on  $A$ . Warshall's algorithm determines the transitive closure by constructing a sequence of  $n$  Boolean matrices as follows:

1. Let  $W_0 = M_R = [m_{ij}]$ .

2. Construct the  $n \times n$  Boolean matrix  $W_1 = [w_{ij}]$  as follows:

For all  $i, j; i = 1, 2, \dots, n; j = 1, 2, \dots, n$ , the  $(i, j)$ th element of  $W_1$  is 1 if and only if either the  $(i, j)$ th element of  $W_0$  is 1 or  $(a_i, a_j) \in R$ . That is,

$$w_{ij} = 1 \text{ if and only if either } m_{ij} = 1 \text{ or } (m_{i1} = 1 \text{ and } m_{1j} = 1).$$

for all  $i, j, i = 1, 2, \dots, n; j = 1, 2, \dots, n$ .

3. Suppose that the  $n \times n$  Boolean matrix  $W_{k-1} = [s_{ij}]$  has been constructed, where  $1 < k < n$ . Construct the  $n \times n$  Boolean matrix (see pp. 269–270)  $W_k = [t_{ij}]$  from  $W_{k-1}$  as follows: For all  $i, j, i = 1, 2, \dots, n; j = 1, 2, \dots, n$ ,

$$t_{ij} = 1 \text{ if and only if either } s_{ij} = 1 \text{ or } (s_{ik} = 1 \text{ and } s_{kj} = 1).$$

Let us elaborate on step (3) a bit more.

To construct  $W_k$ :

- (i) First transfer all 1's of  $W_{k-1}$  to  $W_k$ ; i.e., if an element of  $W_{k-1}$  is 1, then make the corresponding entry of  $W_k$  also 1.
- (ii) Consider the  $k$ th row and  $k$ th column of  $W_{k-1}$ . Let  $p_1, p_2, \dots, p_r, 1 \leq r \leq n$ , be the position in the  $k$ th column of  $W_{k-1}$ , where the entry is 1, and  $q_1, q_2, \dots, q_t, 1 \leq t \leq n$ , be the position in the  $k$ th row of  $W_{k-1}$ , where the entry is 1. Then

$$s_{p_1k} = s_{p_2k} = s_{p_3k} = \dots = s_{p_rk} = 1$$

and

$$s_{kp_1} = s_{kp_2} = s_{kp_3} = \dots = s_{kp_t} = 1.$$



**Stephen Warshall**  
(b. 1935–)

Warshall was born and educated in Brooklyn, NY. He earned his undergraduate degree in mathematics at Harvard in 1956. Although he continued his education by taking a variety of postgraduate classes in a range of disciplines, he never completed an advanced degree.

### Historical Notes

After Harvard, Warshall worked for the U.S. Army in their Operations Research Office. He held a post there until 1958 when he left to pursue other opportunities at Technical Operations. Here he had the chance to develop a computer lab for the purpose of creating military software. Warshall, a devoted computer science researcher, continued to work his way up the corporate ladder to eventually sit on the

board of a company called Applied Data Researchers. Warshall's work has had a great impact on how contemporary design and computer languages operate. He is best known for proving the transitive closure of a relation. The algorithm that he created to solve this problem bears his name.

So make

$$t_{p_u q_v} = 1,$$

for all  $u = 1, 2, \dots, r$ ,  $v = 1, 2, \dots, t$ . That is, set the  $(p_u, q_v)$ th element of  $W_k$  to 1 for all  $u = 1, 2, \dots, r$ ,  $v = 1, 2, \dots, t$ .

We will prove below that the Boolean matrix  $W_n$  is, in fact, the transitive closure of  $R$ . However, first we illustrate, in the next example, Warshall's algorithm.

#### EXAMPLE 4.2.14

Let the set  $A$  and  $R$  be a relation on  $A$  as defined in Example 4.2.12. That is, Let  $A = \{a_1, a_2, a_3, a_4, a_5\}$ . Let  $R$  be a relation on  $A$  given by:

$$R = \{(a_1, a_1), (a_1, a_2), (a_1, a_4), (a_2, a_3), (a_3, a_3), (a_3, a_5), (a_4, a_4), (a_5, a_2)\}.$$

In Example 4.2.12, we determined the transitive closure of  $R$  using Theorem 4.2.11. In this example, we determine the transitive closure of  $R$  using Warshall's algorithm.

Set  $W_0 = M_R$ , i.e.,

$$W_0 = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

##### 1. Construct $W_1$ :

First transfer all 1's of  $W_0$  to  $W_1$ ; i.e., set  $W_1 = W_0$ .

$$W_1 = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

In column 1 of  $W_0$ : Nonzero entry is at position 1.

In row 1 of  $W_0$ : Nonzero entries are at positions 1, 2, and 4.

At the positions (1, 1), (1, 2), and (1, 4) of  $W_1$  make the entries 1. Therefore,

$$W_1 = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

##### 2. Construct $W_2$ :

First transfer all 1's of  $W_1$  to  $W_2$ ; i.e., set  $W_2 = W_1$ .

$$W_2 = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

In column 2 of  $W_1$ : Nonzero entries are at positions 1 and 5.

In row 2 of  $W_1$ : Nonzero entry is at position 3.

At the positions (1, 3) and (5, 3) of  $W_2$  make the entries 1. Therefore,

$$W_2 = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}.$$

3. Construct  $W_3$ :

First transfer all 1's of  $W_2$  to  $W_3$ ; i.e., set  $W_3 = W_2$ .

$$W_3 = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}.$$

In column 3 of  $W_2$ : Nonzero entries are at positions 1, 2, 3, and 5.

In row 3 of  $W_2$ : Nonzero entries are at positions 3 and 5.

At the positions (1, 3), (1, 5), (2, 3), (2, 5), (3, 3), (3, 5), (5, 3), and (5, 5) of  $W_3$  make the entries 1. Therefore,

$$W_3 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

4. Construct  $W_4$ :

First transfer all 1's of  $W_3$  to  $W_4$ ; i.e., set  $W_4 = W_3$ .

$$W_4 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

In column 4 of  $W_3$ : Nonzero entries are at positions 1 and 4.

In row 4 of  $W_3$ : Nonzero entry is at position 4.

At the positions (1, 4) and (4, 4) of  $W_4$  make the entries 1. Therefore,

$$W_4 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

5. Construct  $W_5$ :

First transfer all 1's of  $W_4$  to  $W_5$ ; i.e., set  $W_5 = W_4$ .

$$W_5 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

In column 5 of  $W_4$ : Nonzero entries are at positions 1, 2, 3, and 5.

In row 5 of  $W_4$ : Nonzero entries are at positions 2, 3, and 5.

At the positions (1, 2), (1, 3), (1, 5), (2, 2), (2, 3), (2, 5), (3, 2), (3, 3), (3, 5), (5, 2), (5, 3), and (5, 5) of  $W_5$  make the entries 1. Therefore,

$$W_5 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

From  $W_5$ , we can conclude that the transitive closure of  $R$  is:

$$\begin{aligned} R^\infty = & \{(a_1, a_1), (a_1, a_2), (a_1, a_3), (a_1, a_4), (a_1, a_5), \\ & (a_2, a_2), (a_2, a_3), (a_2, a_5), \\ & (a_3, a_2), (a_3, a_3), (a_3, a_5), \\ & (a_4, a_4), (a_5, a_2), (a_5, a_3), (a_5, a_5)\}. \end{aligned}$$

Notice that  $W_5 = M_{R^\infty}$ , which is computed in Example 4.2.12.

Now that we know how to use Warshall's algorithm, next we show that Warshall's algorithm correctly computes the transitive closure. The following two theorems do just that.

**Theorem 4.2.15:** Let  $A = \{a_1, a_2, \dots, a_n\}$  be a finite set,  $n \geq 1$ , and let  $R$  be a relation on  $A$ . Let  $W_m = [t_{ij}]_{n \times n}$  be the matrix as defined at the beginning of this section (see p. 266), where  $1 \leq m \leq n$ . Then for all  $i, j$ ,  $1 \leq i \leq n$ ;  $1 \leq j \leq n$ ;  $t_{ij} = 1$  if and only if either  $(a_i, a_j) \in R$  or there exists  $a_{i_1}, a_{i_2}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_m\}$  such that  $(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_j) \in R$ .

**Proof:** We prove the result by induction on  $m$ .

*Basis step:* Suppose that  $m = 1$ . Let  $W_1 = [t_{ij}]$  and  $W_0 = [s_{ij}]$ . Let  $t_{ij} = 1$ . Then either  $s_{ij} = 1$  or  $s_{i1} = 1$  and  $s_{1j} = 1$ . If  $s_{ij} = 1$ , then  $(a_i, a_j) \in R$ .

Suppose that  $s_{i1} = 1$  and  $s_{1j} = 1$ . Then  $(a_i, a_1), (a_1, a_j) \in R$ . We can take  $q = 1$  and  $a_{i_1} = a_1 \in \{a_1\}$ .

Conversely, suppose that either  $(a_i, a_j) \in R$  or  $(a_i, a_1), (a_1, a_j) \in R$ . If  $(a_i, a_j) \in R$ , then  $s_{ij} = 1$ , which implies that  $t_{ij} = 1$ . If  $(a_i, a_1), (a_1, a_j) \in R$ , then  $s_{i1} = 1$  and  $s_{1j} = 1$ , so  $t_{ij} = 1$ .

*Inductive hypothesis:* Suppose that the result is true for  $m = k - 1$ , where  $k > 1$ .

*Inductive step:* Consider  $W_k = [t_{ij}]$ . Let us write  $W_{k-1} = [s_{ij}]$ .

First suppose that  $t_{ij} = 1$ . Then either  $s_{ij} = 1$  or  $(s_{ik} = 1 \text{ and } s_{kj} = 1)$ . Suppose that  $s_{ij} = 1$ . Then by the inductive hypothesis, either  $(a_i, a_j) \in R$  or there exists  $a_{i_1}, a_{i_2}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_{k-1}\}$  such that  $(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_j) \in R$ . Now  $\{a_1, a_2, \dots, a_{k-1}\} \subseteq \{a_1, a_2, \dots, a_{k-1}, a_k\}$ . Hence either  $(a_i, a_j) \in R$  or there exists  $a_{i_1}, a_{i_2}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_k\}$  such that  $(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_j) \in R$ .

Next suppose that  $s_{ik} = 1$  and  $s_{kj} = 1$ . Because  $s_{ik} = 1$ , by the inductive hypothesis, either  $(a_i, a_k) \in R$  or there exists  $a_{i_1}, a_{i_2}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_{k-1}\}$  such that  $(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_k) \in R$ . Similarly, because  $s_{kj} = 1$ , either  $(a_k, a_j) \in R$  or there exists  $a'_{k_1}, a'_{k_2}, \dots, a'_{k_r} \in \{a_1, a_2, \dots, a_{k-1}\}$  such that  $(a_k, a'_{k_1}), (a'_{k_1}, a'_{k_2}), \dots, (a'_{k_r}, a_j) \in R$ . We have four cases:

**Case 1:**  $(a_i, a_k) \in R$  and  $(a_k, a_j) \in R$ . Then  $(a_i, a_k), (a_k, a_j) \in R$  and  $a_k \in \{a_1, a_2, \dots, a_k\}$ .

**Case 2:**  $(a_i, a_k) \in R$  and there exists  $a'_{k_1}, a'_{k_2}, \dots, a'_{k_r} \in \{a_1, a_2, \dots, a_{k-1}\}$  such that  $(a_k, a'_{k_1}), (a'_{k_1}, a'_{k_2}), \dots, (a'_{k_r}, a_j) \in R$ . Then  $(a_i, a_k), (a_k, a'_{k_1}), (a'_{k_1}, a'_{k_2}), \dots, (a'_{k_r}, a_j) \in R$  and  $a_k, a'_{k_1}, a'_{k_2}, \dots, a'_{k_r} \in \{a_1, a_2, \dots, a_k\}$ .

**Case 3:**  $(a_k, a_j) \in R$  and there exists  $a_{i_1}, a_{i_2}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_{k-1}\}$  such that  $(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_k) \in R$ . Then  $a_{i_1}, a_{i_2}, \dots, a_{i_q}, a_k \in \{a_1, a_2, \dots, a_{k-1}, a_k\}$  such that  $(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_k), (a_k, a_j) \in R$ .

**Case 4:** There exists  $a_{i_1}, a_{i_2}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_{k-1}\}$  such that

$$(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_k) \in R$$

and there exists  $a'_{k_1}, a'_{k_2}, \dots, a'_{k_r} \in \{a_1, a_2, \dots, a_{k-1}\}$  such that

$$(a_k, a'_{k_1}), (a'_{k_1}, a'_{k_2}), \dots, (a'_{k_r}, a_j) \in R.$$

Then  $a_{i_1}, a_{i_2}, \dots, a_{i_q}, a_k, a'_{k_1}, a'_{k_2}, \dots, a'_{k_r} \in \{a_1, a_2, \dots, a_{k-1}, a_k\}$  such that

$$(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_k), (a_k, a'_{k_1}), (a'_{k_1}, a'_{k_2}), \dots, (a'_{k_r}, a_j) \in R.$$

Thus, either  $(a_i, a_j) \in R$  or there exists  $a_{i_1}, a_{i_2}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_k\}$  such that  $(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_j) \in R$ . Hence, the result is true for  $k$ . The result now follows by induction.

Conversely, assume that either  $(a_i, a_j) \in R$  or there exists  $a_{i_1}, a_{i_2}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_k\}$  such that

$$(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_j) \in R.$$

If  $(a_i, a_j) \in R$ , then  $(i, j)$ th entry of  $W_0$  is 1, so  $t_{ij} = 1$ . Suppose that there exists  $a_{i_1}, a_{i_2}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_k\}$  such that

$$(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_j) \in R.$$

By Lemma 3.1.54, we may assume that all the elements  $a_{i_1}, a_{i_2}, \dots, a_{i_q}$  are distinct.

Suppose that  $a_{i_l} \neq a_k$  for all  $l = 1, 2, \dots, q$ . Then  $a_{i_1}, a_{i_2}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_{k-1}\}$  such that  $(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_j) \in R$ . Therefore, by the inductive hypothesis,  $s_{ij} = 1$ . Thus,  $t_{ij} = 1$ .

Suppose that there exists  $t$ ,  $1 \leq t \leq q$  such that  $a_{i_t} = a_k$ . It follows that the elements  $a_{i_1}, a_{i_2}, \dots, a_{i_{t-1}}, a_{i_{t+1}}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_{k-1}\}$ . Notice that

$$(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_{t-1}}, a_k) \in R$$

and  $a_{i_1}, a_{i_2}, \dots, a_{i_{t-1}} \in \{a_1, a_2, \dots, a_{k-1}\}$ . Hence, by the inductive hypothesis,  $s_{ik} = 1$ . Similarly,  $s_{kj} = 1$ . This implies that  $t_{ij} = 1$ . ■

**Theorem 4.2.16:** Let  $A = \{a_1, a_2, \dots, a_n\}$  be a finite set,  $n \geq 1$ , and let  $R$  be a relation on  $A$ . Let  $W_k = [t_{ij}]_{n \times n}$  be the matrix as defined at the beginning of this section (see p. 266),  $1 \leq k \leq n$ . Then  $W_n = M_{R^\infty}$ ; that is,  $W_n$  is the transitive closure of  $R$ .

**Proof:** Let  $W_n = [t_{ij}]$  and  $M_{R^\infty} = [m_{ij}]$ . Let  $1 \leq i \leq n$  and  $1 \leq j \leq n$ . We show that  $t_{ij} = 1$  if and only if  $m_{ij} = 1$ .

Suppose that  $t_{ij} = 1$ . Then by Theorem 4.2.15, either  $(a_i, a_j) \in R$  or there exists  $a_{i_1}, a_{i_2}, \dots, a_{i_q} \in \{a_1, a_2, \dots, a_n\}$  such that  $(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_j) \in R$ . Because  $(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_q}, a_j) \in R$ , we have  $(a_i, a_j) \in R^q \subseteq R^\infty$ . Hence,  $m_{ij} = 1$ .

Conversely, suppose that  $m_{ij} = 1$ . Then  $(a_i, a_j) \in R^\infty$ . By Theorem 3.1.57,

$$R^\infty = R \cup R^2 \cup \dots \cup R^n.$$

It follows that  $(a_i, a_j) \in R^k$ , for some  $k$ ,  $1 \leq k \leq n$ . Now because  $(a_i, a_j) \in R^k$ , there exists  $a_{i_1}, a_{i_2}, \dots, a_{i_k} \in A = \{a_1, a_2, \dots, a_n\}$  such that  $(a_i, a_{i_1}), (a_{i_1}, a_{i_2}), \dots, (a_{i_k}, a_j) \in R$ . This implies, by Theorem 4.2.15, that the  $(i, j)$ th entry of  $W_n$  is 1, i.e.,  $t_{ij} = 1$ .

Consequently,  $t_{ij} = 1$  if and only if  $m_{ij} = 1$ . Hence,  $W_n = M_{R^\infty}$ . ■

---

**REMARK 4.2.17** ▶ We might think that to determine the transitive closure using Warshall's requires us to separately compute the matrices  $W_1, W_2, \dots, W_n$ , so we will need memory space for each of these matrices. However, to determine the next matrix, first we transfer all the 1's of the current matrix to the next matrix. Next by considering the appropriate row and column of the current matrix, we make additional entries of the next matrix 1 as described above. It follows that we can do all these computations using just one matrix. This is in fact how we implement Warshall's algorithm in computer memory.

#### ALGORITHM 4.4: Warshall's Algorithm.

*Input:*  $M$ —Boolean matrices of the relation  $R$   
*n*—positive integers such that  $n \times n$  specifies the size of  $M$

*Output:*  $W$ —an  $n \times n$  Boolean matrix such that  $W = M_{R^\infty}$

```

1. procedure WarshallAlgorithm( $M, W, n$ )
2. begin
3.    $W = M;$ 
4.   for  $k := 1$  to  $n$  do
5.     for  $i := 1$  to  $n$  do
6.       for  $j := 1$  to  $n$  do
7.         if  $W[i, j] \neq 1$  then
8.           if  $W[i, k] = 1$  and  $W[k, j] = 1$  then
9.              $W[i, j] = 1;$ 
10. end
```

---

**REMARK 4.2.18** ▶ It can be shown that the number of operations required by Warshall's algorithm is  $\leq an^3$ , for some positive constant  $a$ .

## WORKED-OUT EXERCISES

**Exercise 1:** Let  $R = \{(1, 1), (1, 3), (2, 2), (2, 3), (3, 3), (3, 4)\}$  and  $S = \{(1, 4), (2, 1), (3, 1), (3, 2), (3, 3)\}$  be relations on the set  $X = \{1, 2, 3, 4\}$ . Find  $M_R$ ,  $M_S$ ,  $M_{R \cap S}$ ,  $M_{R \cup S}$ , and  $M_{R^{-1}}$ . Also, verify that

- (a)  $M_{R \cup S} = M_R \vee M_S$ ,
- (b)  $M_{R \cap S} = M_R \wedge M_S$ ,
- (c)  $M_{R^{-1}} = (M_R)^T$ , the transpose of  $M_R$ .

**Solution:** Now

$$M_R = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad M_S = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Also,  $R \cap S = \{(3, 3)\}$ ,  $R \cup S = \{(1, 1), (1, 3), (1, 4), (2, 1), (2, 2), (2, 3), (3, 1), (3, 2), (3, 3), (3, 4)\}$ . Thus,

$$M_{R \cap S} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad M_{R \cup S} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Because  $R^{-1} = \{(1, 1), (3, 1), (2, 2), (3, 2), (3, 3), (4, 3)\}$ , we have

$$M_{R^{-1}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

Next,

$$\begin{aligned} M_R \vee M_S &= \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \vee \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 1 \vee 0 & 0 \vee 0 & 1 \vee 0 & 0 \vee 1 \\ 0 \vee 1 & 1 \vee 0 & 1 \vee 0 & 0 \vee 0 \\ 0 \vee 1 & 0 \vee 1 & 1 \vee 1 & 1 \vee 0 \\ 0 \vee 0 & 0 \vee 0 & 0 \vee 0 & 0 \vee 0 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\ &= M_{R \cup S}. \end{aligned}$$

Also,

$$\begin{aligned} M_R \wedge M_S &= \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \wedge \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 1 \wedge 0 & 0 \wedge 0 & 1 \wedge 0 & 0 \wedge 1 \\ 0 \wedge 1 & 1 \wedge 0 & 1 \wedge 0 & 0 \wedge 0 \\ 0 \wedge 1 & 0 \wedge 1 & 1 \wedge 1 & 1 \wedge 0 \\ 0 \wedge 0 & 0 \wedge 0 & 0 \wedge 0 & 0 \wedge 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\ &= M_{R \cap S}. \end{aligned}$$

$$(M_R)^T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = M_{R^{-1}}.$$

**Exercise 2:** Let  $A = \{a, b, c\}$  and  $B = \{1, 2, 3, 4\}$  be sets. Let  $R$  be a relation from  $A$  into  $B$  and  $S$  be a relation from  $B$  into  $A$  such that the matrices of  $R$  and  $S$  are

$$M_R = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix} \quad \text{and} \quad M_S = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Find

- (a)  $R$  and  $S$ ;
- (b)  $M_R \odot M_S$ ;
- (c) From part (b), find  $S \circ R$ .

**Solution:**

- (a) Because the (1, 2) entry of  $M_R$  is 1, we have  $(a, 2) \in R$ . Similarly, we can look at the other entries of  $M_R$  that are 1 and find the element of  $R$ . Therefore,  $R = \{(a, 2), (a, 3), (b, 1), (b, 2), (b, 4), (c, 2), (c, 4)\}$ . Similarly,  $S = \{(1, a), (1, b), (2, a), (2, c), (4, c)\}$ .

- (b) We have

$$M_R \odot M_S = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

- (c) First note that  $S \circ R$  is a relation from  $A$  into  $A$ . From part (b), we have,

$$S \circ R = \{(a, a), (a, c), (b, a), (b, b), (b, c), (c, a), (c, b), (c, c)\}.$$

**Exercise 3:** Let  $A = \{a_1, a_2, a_3, a_4\}$ . Let  $R$  be the relation on  $A$  given by:

$$R = \{(a_1, a_1), (a_1, a_2), (a_1, a_3), (a_2, a_3), (a_3, a_3), (a_4, a_1), (a_4, a_4)\}.$$

- (a) Find the matrices  $M_R$  and  $M_{R^2}$ .
- (b) Find  $R^2$ .

**Solution:**

- (a) We have

$$M_R = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

Now

$$\begin{aligned} M_{R^2} &= M_R \odot M_R = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}. \end{aligned}$$

$$(b) R^2 = \{(a_1, a_1), (a_1, a_2), (a_1, a_3), (a_2, a_3), (a_3, a_3), (a_4, a_1), (a_4, a_2), (a_4, a_3), (a_4, a_4)\}$$

**Exercise 4:** Let  $R$  be the relation on the set  $A = \{a_1, a_2, a_3, a_4\}$  defined by the matrix

$$M_R = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

Find the transitive closure of  $R$ .

**Solution:** Because  $A$  has four elements, we have

$$R^\infty = R \cup R^2 \cup R^3 \cup R^4.$$

That is, we need to determine  $M_{R^2}$ ,  $M_{R^3}$ , and  $M_{R^4}$ . Notice that  $M_R$  is the same as the matrix in the preceding worked-out exercise. Therefore, we only need to determine  $M_{R^3}$  and  $M_{R^4}$ . Now

$$\begin{aligned} M_{R^3} = M_R \odot M_{R^2} &= \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}. \end{aligned}$$

We see that  $M_{R^3} = M_{R^2}$  and from this it follows that  $M_{R^4} = M_{R^3}$ . Now,  $M_{R^\infty} = M_R \vee M_{R^2} \vee M_{R^3} \vee M_{R^4} = M_R \vee M_{R^2}$  because  $M_{R^2} = M_{R^3} = M_{R^4}$ . Hence

$$\begin{aligned} M_{R^\infty} &= \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \vee \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}. \end{aligned}$$

Hence, the transitive closure is

$$R^\infty = \{(a_1, a_1), (a_1, a_2), (a_1, a_3), (a_2, a_3), (a_3, a_3), (a_4, a_1), (a_4, a_2), (a_4, a_3), (a_4, a_4)\}.$$

**Exercise 5:** Let  $R$  be the relation on a set  $A = \{a_1, a_2, a_3, a_4\}$  defined by the following matrix:

$$M_R = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

Find the transitive closure of  $R$  by Warshall's algorithm.

**Solution:** Set  $W_0 = M_R$ , i.e.,

$$W_0 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

(1) Construct  $W_1$ :

First transfer all 1's of  $W_0$  to  $W_1$ ; i.e., set  $W_1 = W_0$ .

$$W_1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

In column 1 of  $W_0$ : Nonzero entries are at positions 1 and 4.

In row 1 of  $W_0$ : Nonzero entries are at positions 1 and 2.

At the positions (1, 1), (1, 2), (4, 1), and (4, 2) of  $W_1$  make the entries 1. Therefore,

$$W_1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

(2) Construct  $W_2$ :

First transfer all 1's of  $W_1$  to  $W_2$ ; i.e., set  $W_2 = W_1$ .

$$W_2 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

In column 2 of  $W_1$ : Nonzero entries are at positions 1, 2, and 4.

In row 2 of  $W_1$ : Nonzero entry is at position 2.

At the positions (1, 2), (2, 2), and (4, 2) of  $W_2$  make the entries 1. Therefore,

$$W_2 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

(3) Construct  $W_3$ :

First transfer all 1's of  $W_2$  to  $W_3$ ; i.e., set  $W_3 = W_2$ .

$$W_3 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

In column 3 of  $W_2$ : Nonzero entry is at position 3.

In row 3 of  $W_2$ : Nonzero entry is at position 3.

At the position (3, 3) of  $W_3$  make the entry 1. Therefore,

$$W_3 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

(4) Construct  $W_4$ :First transfer all 1's of  $W_3$  to  $W_4$ ; i.e., set  $W_4 = W_3$ .

$$W_4 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

In column 4 of  $W_3$ : Nonzero entry is at position 4.In row 4 of  $W_3$ : Nonzero entries are at positions 1, 2, and 4.At the positions (4, 1), (4, 2), and (4, 4) of  $W_4$  make the entries 1. Therefore,

$$W_4 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

From  $W_4$ , we can conclude that the transitive closure of  $R$  is:

$$R^\infty = \{(a_1, a_1), (a_1, a_2), (a_2, a_2), (a_3, a_3), (a_4, a_1), (a_4, a_2), (a_4, a_4)\}.$$

## SECTION REVIEW

---

### Key Terms

matrix of a relation

Warshall's algorithm

### Key Definition

- Let  $A = \{a_1, a_2, \dots, a_n\}$  and  $B = \{b_1, b_2, \dots, b_p\}$  be finite nonempty sets. Let  $R$  be a relation from  $A$  into  $B$ . Then  $R \subseteq A \times B$ . Let  $M_R = [m_{ij}]_{n \times p}$  be the Boolean  $n \times p$  matrix, where

$$m_{ij} = \begin{cases} 1 & \text{if } (a_i, b_j) \in R, \text{ i.e., } a_i R b_j, \\ 0 & \text{otherwise.} \end{cases}$$

The matrix  $M_R$  is called the matrix of the relation  $R$ .

### Some Key Results

- Let  $A$ ,  $B$ , and  $C$  be finite sets. Let  $R$  be a relation from  $A$  into  $B$  and  $S$  be a relation from  $B$  into  $C$ . Then

$$M_{S \circ R} = M_R \odot M_S;$$

i.e., the matrix  $M_{S \circ R}$  of the relation  $S \circ R$  is same as the matrix of the Boolean products of  $M_R$  and  $M_S$ .

- Let  $A$  be a finite set and  $R$  be a relation on  $A$ , i.e.,  $R \subseteq A \times A$ . Let  $M_{R^n}$  be the matrix of the relation  $R^n$ . Then for all  $n \geq 1$ ,

$$M_{R^n} = \underbrace{M_R \odot M_R \odot \cdots \odot M_R}_{n \text{ times}}.$$

- Let  $A = \{a_1, a_2, \dots, a_n\}$  be a finite set,  $n \geq 1$ . Let  $R$  be a relation on  $A$ . Then

$$M_{R^\infty} = M_R \vee M_{R^2} \vee \cdots \vee M_{R^n}.$$

- Let  $A = \{a_1, a_2, \dots, a_n\}$  be a finite set,  $n \geq 1$ , and  $R$  be a relation on  $A$ .

- Let  $W_0 = M_R = [m_{ij}]$ .

- (ii) Construct the  $n \times n$  Boolean matrix  $W_1 = [w_{ij}]$  as follows:  
For all  $i, j$ ,  $i = 1, 2, \dots, n$ ;  $j = 1, 2, \dots, n$ , the  $(i, j)$ th element of  $W_1$  is 1 if and only if either the  $(i, j)$ th element of  $W_0$  is 1 or  $(a_i, a_l), (a_l, a_j) \in R$ . That is,  $w_{ij} = 1$  if and only if either  $m_{ij} = 1$  or  $(m_{il} = 1 \text{ and } m_{lj} = 1)$ , for all  $i, j$ ,  $i = 1, 2, \dots, n$ ;  $j = 1, 2, \dots, n$ .

(iii) Suppose that the  $n \times n$  Boolean matrix  $W_{k-1} = [s_{ij}]$  has been constructed, where  $1 < k < n$ . Construct the  $n \times n$  Boolean matrix  $W_k = [t_{ij}]$  from  $W_{k-1}$  as follows:  
For all  $i, j$ ,  $i = 1, 2, \dots, n$ ;  $j = 1, 2, \dots, n$ ,  $t_{ij} = 1$  if and only if either  $s_{ij} = 1$  or  $(s_{ik} = 1 \text{ and } s_{kj} = 1)$ .

Then  $W_n = M_{R^\infty}$ , that is,  $W_n$  is the transitive closure of  $R$ .

# **EXERCISES**

*In these exercises, to find the matrix of a relation assume that the elements of the set are ordered as they are listed in the set.*

1. Let

$$R = \{(a, 3), (a, 2), (b, 4), (c, 3), (c, 1), (b, 3)\}$$

and

$$S = \{(a, 1), (b, 1), (b, 3), (c, 4), (c, 2)\}$$

be relations from the set  $A = \{a, b, c\}$  into the set  $B = \{1, 2, 3, 4\}$ . Determine  $M_R$ ,  $M_S$ ,  $M_{R \cup S}$ ,  $M_{R \cap S}$ , and  $M_{R^{-1}}$ .

- Let  $R = \{(1, 1), (1, 2), (3, 4), (3, 3), (3, 1), (1, 3)\}$  and  $S = \{(2, 2), (2, 1), (3, 3), (1, 4), (3, 1)\}$  be two relations on the set  $X = \{1, 2, 3, 4\}$ . Find  $M_R$ ,  $M_S$ ,  $M_{R \cup S}$ ,  $M_{R \cap S}$ , and  $M_{R^{-1}}$ . Also verify that
    - $M_{R \cup S} = M_R \vee M_S$ .
    - $M_{R \cap S} = M_R \wedge M_S$ .
    - $M_{R^{-1}} = (M_R)^T$ , the transpose of  $M_R$ .
  - Prove Theorem 4.2.4(ii) and (iii).
  - Let  $R = \{(a, b) \in A \mid a \text{ divides } b\}$ , where  $A = \{1, 2, 3, 4\}$ . Find the matrix  $M_R$  of  $R$ . Then determine whether  $R$  is reflexive, symmetric, or transitive.
  - Let  $A = \{a, b, c, d\}$ ,  $B = \{1, 2, 3\}$ , and  $C = \{x, y, z\}$ . Let  $R$  be a relation from  $A$  into  $B$  and  $S$  be a relation from  $B$  into  $C$  such that the matrices of  $R$  and  $S$  are

$$M_R = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{and} \quad M_S = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

- a. Write  $R$  and  $S$  as sets.
  - b. Determine  $M_R \odot M_S$ .
  - c. Using part (b) determine  $S \circ R$ .

6. Let  $A = \{a_1, a_2, a_3, a_4\}$ . Let  $R$  be the relation on  $A$  given by:

$$R = \{(a_1, a_2), (a_1, a_3), (a_1, a_4), (a_2, a_3), (a_3, a_1), (a_4, a_4)\}.$$

- a. Determine the matrices  $M_R$  and  $M_{R^2}$ .
  - b. Find  $R^3$ .

7. Let  $R = \{(1, 1), (1, 2), (1, 3), (1, 4), (2, 1), (2, 2), (2, 3), (2, 4), (3, 4)\}$  be a relation on the set  $X = \{1, 2, 3, 4\}$ . Find  $M_R$  and  $M_R \odot M_R$ . From this determine whether  $R$  is transitive.

8. Let  $R$  be the relation on the set  $A = \{a_1, a_2, a_3, a_4\}$  such that the matrix of  $R$  is

$$M_R = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

- a. Is  $R$  reflexive?
  - b. Is  $R$  symmetric?
  - c. Determine the transitive closure of  $R$ .
  - . Let  $R = \{(a_1, a_1), (a_2, a_2), (a_3, a_1), (a_3, a_2)\}$  be a relation on a set  $A = \{a_1, a_2, a_3\}$ . Compute the following.
    - a.  $M_R$
    - b.  $M_{R^{-1}}$
    - c.  $M_{R \cup R^{-1}}$
    - d.  $M_{R^2}$

$$M_R = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

Write the relation  $R$  as a set of ordered pairs.

- Let  $R = \{(a, b) \in A \mid a + b = 4\}$  be a relation on the set  $A = \{1, 2, 3, 4\}$ . Find the matrix representation  $M_R$  of  $R$ . From  $M_R$  determine whether  $R$  is reflexive, symmetric, or transitive. If  $R$  is not transitive, then using Warshall's algorithm find the transitive closure of  $R$ .
  - Let  $R$  be the relation on the set  $A = \{a_1, a_2, a_3, a_4\}$  such that the matrix of  $R$  is

$$M_R = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

Find  $M_R \odot M_R$  and write the relation  $R^2$  as ordered pairs.

13. Let  $R$  be the relation on the set  $A = \{1, 2, 3, 4\}$  such that the matrix of  $R$  is

$$M_R = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Find the transitive closure of  $R$ .

14. Let  $R$  be the relation on the set  $A = \{a_1, a_2, a_3, a_4\}$  such that the matrix of  $R$  is

$$M_R = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

Use Warshall's algorithm to find the transitive closure of  $R$ .

15. Let  $R$  be the relation on the set  $A = \{a_1, a_2, a_3, a_4\}$  such that the matrix of  $R$  is

$$M_R = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

Find the transitive closure of  $R$  by Warshall's algorithm.

16. Let  $R$  be the relation on the set  $A = \{a_1, a_2, a_3, a_4, a_5\}$  such that the matrix of  $R$  is

$$M_R = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

Find the transitive closure of  $R$  by Warshall's algorithm.

## PROGRAMMING EXERCISES

1. Write a program to add, subtract, and multiply two matrices, and determine the transpose of a matrix.
2. Write a program to find the Boolean join, meet, and product of two Boolean matrices.
3. Write a program that uses the formula  $M_{R^\infty} = M_R \vee M_{R^2} \vee \dots \vee M_{R^n}$  to determine the transitive closure of a relation.
4. Write a program to implement Warshall's algorithm to implement the transitive closure.
5. Write a program that does the following: Given a relation  $R$  determine the smallest equivalent relation  $S$  such that  $R \subseteq S$ .

## Functions

**The objectives of this chapter are to:**

- Learn about functions
- Explore various properties of functions
- Learn about sequences and strings
- Become familiar with the representation of strings in computer memory
- Learn about binary operations

In Chapter 3, we studied relations extensively. In this chapter, we study a special type of relation known as a function. Before formally beginning, let us present a brief history of the functions concept. The term *function* was first coined around 1694 by the famous German mathematician Leibnitz in the context of the slope of a curve. Later, in 1749, the Swiss mathematician Leonhard Euler (1707–1783) defined a function as a law governing the interdependence of variable quantities. Our present-day understanding of functions is attributed to Dirichlet, who in 1837 proposed the definition of a function as a “*rule of correspondence that assigns a unique value of the dependent variable to every permitted value of an independent variable.*” In the next section, we shall see that this idea, in essence, lies at the core of the formal definition of a function.

Functions are one of the most important concepts in mathematics and computer science. They allow us to study the relationship between different (algebraic) structures and they allow us to put an algebraic structure on a set. We use the notion of functions extensively in algorithm analysis as a means of putting a measure

on the algorithm. (We formally introduce algorithm analysis in Chapter 9).

In the first two sections, we present a formal treatment of functions. In the third section, we use the concept of functions to study sequences and strings. In the last section, we illustrate how functions are used to put an algebraic structure on a set in the form of binary operations

## 5.1 FUNCTIONS

Let  $S$  be the set of all students in the Discrete Mathematics class at a local university. Assume that  $S$  is not empty. We can associate a positive integer with each student in the following way: If *Jessica* is a student of this class and she is 20 years old, then we associate 20 with *Jessica*. In general, if  $n$  is the age of the student  $s$ , then we associate  $n$  with  $s$  and express it as  $f : s \mapsto n$ .

We find that this correspondence establishes a relation between students and positive integers. That is, we can consider  $f \subseteq S \times \mathbb{N}$ . This relation has the following important properties.

1. We can associate a fixed positive integer with each student; i.e., for each  $s \in S$ , there is positive integer  $n$  such that  $(s, n) \in f$ . For example,  $(\text{Jessica}, 20) \in f$ .
2. We cannot associate two distinct positive integers with the same student because a student cannot have two different ages. That is, if  $s \in S$  and  $m$  and  $n$  are positive integers such that  $(s, n) \in f$  and  $(s, m) \in f$ , then we must have  $n = m$ .



**Johann Peter Gustave Lejeune Dirichlet**  
(1805–1859)

Dirichlet was born in the Belgian town of Richelet. He was an eager pupil and would spend any extra money he had on mathematics books. Dirichlet attended the Gymnasium in Bonn and the Jesuit College in Cologne. For university, he went to Paris, as French universities were more exacting than German ones at that time. Eventually, Dirichlet's influence would help to

### HISTORICAL NOTES

make German universities the most rigorous in the world. Dirichlet's drive to achieve is best demonstrated by his perseverance in returning as quickly as possible to his classes after having contracted smallpox.

Upon completing his studies at the College de France, Dirichlet was employed by General Maximilien Foy, a leading figure in the Napoleonic wars. During this time he attracted notice for a paper he published on Fermat's Last Theorem. The fame and genius of this paper paved the way for his return to a

teaching career in Germany upon the death of General Foy. While in Germany, Dirichlet taught concurrently at the Military College and the University of Berlin, a post he held for 27 years. In 1855, he accepted a position at Göttingen that opened upon the death of Gauss. It is interesting that Dirichlet would replace Gauss at Göttingen because as a young man, he was always carrying a copy of Gauss's *Disquisitiones arithmeticæ*. Dirichlet remained at Göttingen until his death in 1859.

This type of relation is called a function from a set  $A$  to a set  $B$ . In the preceding example,  $A$  is the set of all students in the Discrete Mathematics class and  $B = \mathbb{N}$ , the set of all positive integers.

We noticed in Chapter 3 that the mathematical concept of relation is very close to the idea of relation as used outside mathematics. But the mathematical notion of function is different from its meaning in the ordinary sense. We often hear statements such as: "The function of eyes is to see things," "The function of the F1 key on your keyboard is to make help available," and so on. However, mathematically speaking, a function is nothing but a special type of correspondence or relation.

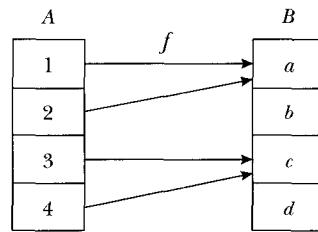
Given that functions are special types of relations, let us consider the following relation: Let  $A = \{1, 2, 3, 4\}$  and  $B = \{a, b, c, d\}$  be sets and  $f$  be the set

$$f = \{(1, a), (2, a), (3, c), (4, c)\}.$$

Because  $f \subseteq A \times B$ ,  $f$  is a relation from  $A$  into  $B$ . We have  $(1, a) \in f$ , which here we write as  $f(1) = a$ , and so on. Thus,

$$\begin{aligned} f(1) &= a, \\ f(2) &= a, \\ f(3) &= c, \\ f(4) &= c. \end{aligned}$$

The arrow diagram in Figure 5.1 represents this relation  $f$  from  $A$  to  $B$ .



**FIGURE 5.1** Arrow diagram of  $f$



### Gottfried Wilhelm Leibnitz

(1646–1716)

Leibnitz was born in Leipzig, Germany, to a professor and the daughter of an attorney. In addition to attending a respectable school, he had taught himself Latin and some Greek by the age of 12. At age 14, because he was an advanced student, he was enrolled at the University of Leipzig where he studied philosophy and mathematics. In-

### Historical Notes

tent on a career in law and denied a degree from his university, he moved to Nuremberg and began a career as a diplomat. He worked for both the court of France and the House of Hanover, whose heir Georg Ludwig ascended the English throne in 1714 as George I.

It was during his years with the Hanovers that Leibnitz had the most time to pursue his other interests, particularly mathematics. He published most of his mathematical theories between 1682 and 1692 in *Acta Eruditio-*

*rum*, a journal he founded with Otto Mencke. His greatest and perhaps most controversial contribution to mathematics was his discovery of calculus. It is still disputed today whether this achievement was based solely on his own work or was derived from early work by Newton of which he may have had knowledge. Regardless of the origin, it is Leibnitz's system of notation for differential calculus that is more commonly applied today.

Let us note the following:

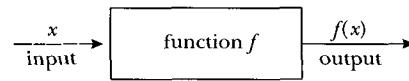
- (i) The domain of  $f$  is the set  $A$ , i.e.,  $\mathcal{D}(f) = \{1, 2, 3, 4\} = A$ . In other words, there is an arrow originating from each element of  $A$  to an element of  $B$ .
- (ii) An element of  $A$  is related to only one element of  $B$ . In other words, for an element  $x \in A$ , there exists a unique  $y \in B$  such that

$$f(x) = y.$$

For example, 1 is related to  $a$  only, 2 is related to  $a$  only, 3 is related to  $c$  only, and 4 is related to  $c$  only. (Notice that both 1 and 2 are related to  $a$ . However, 1 as well as 2 is related to only one element of  $B$ .) In the diagram, we can say that from each element of  $A$ , *only* one arrow is originating to an element of  $B$ .

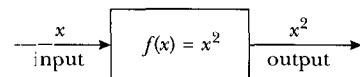
As remarked previously, such a relation  $f$  is called a function, and we write  $f : A \rightarrow B$  and say that  $f$  is a function from  $A$  into  $B$ . We can think of  $f$  as a rule that determines  $f(a)$  for each  $a \in A$ .

A student of computer science may think of a function  $f$  as a machine with input and output that processes elements of a set  $A$  to produce elements of a set  $B$ . If an element  $x \in A$  is placed in the input of the machine, the element  $f(x) \in B$  appears in the output (see Figure 5.2).



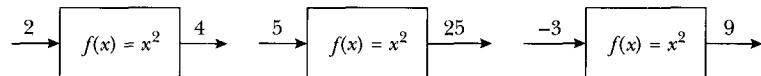
**FIGURE 5.2** Function machine

For example, consider the function  $f : \mathbb{R} \rightarrow \mathbb{R}$  defined by  $f(x) = x^2$  for all  $x \in \mathbb{R}$ . We can describe this as a function machine as shown in Figure 5.3.



**FIGURE 5.3** Function machine of the function  $f(x) = x^2$

Figure 5.4 shows some of the inputs and outputs of the function machine of Figure 5.3.



**FIGURE 5.4** Function machine showing inputs and outputs

As we can see in Figure 5.4, corresponding to each input, there is exactly one output. Moreover, for each real number, this function machine gives an output. These are the properties that make a relation a function.

The concept of functions is of paramount importance in mathematics. As remarked earlier, this concept, among others, enables us to study the relationship between various algebraic structures. Intuitively, a function from a set  $A$  into a set  $B$  is a rule of correspondence between the elements of these sets in the sense that it associates with each element  $a \in A$  a unique element  $b \in B$ . More formally, we have the following definition.

---

**DEFINITION 5.1.1** ▶ Let  $A$  and  $B$  be nonempty sets and  $f$  be a relation from  $A$  into  $B$ . Then  $f$  is called a **function** from  $A$  into  $B$  if

- (i) the domain of  $f$  is  $A$ , i.e.,  $\mathcal{D}(f) = A$ , and
- (ii) for all  $(a, b), (a', b') \in f$ ,  $a = a'$  implies  $b = b'$ . In this case, we say that  $f$  is **well defined** or **single valued**.

To indicate that  $f$  is a function from  $A$  into  $B$ , we usually write  $f : A \rightarrow B$  and say that  $f$  is a function from  $A$  into  $B$ .

In Definition 5.1.1, the first condition says that every element of  $A$  associates with some element of  $B$ , and the second condition means that for every  $a \in A$ , there is exactly one  $b \in B$  such that  $(a, b) \in f$ , i.e.,  $f(a) = b$ . The element  $b$  is called the **image** of  $a$  under  $f$ , and  $a$  is called a **preimage** of  $b$  under  $f$ . Sometimes we also say that  $a$  is **mapped** to  $b$  under  $f$  or simply  $a$  is mapped to  $b$ .

So in order to show that a relation  $f$  from  $A$  to  $B$  is a function, we must verify the following two conditions:

1. The domain of  $f$  is  $A$ , which means that every element of  $A$  has some image in  $B$ , and
2. an element of  $A$  cannot have more than one image in  $B$ .

Let us now give examples of relations, some of which are functions and some are not.

### EXAMPLE 5.1.2

Let  $A$  be the set of all workers in a factory. Assume that each worker has a fixed monthly salary. With each worker we associate a rational number by the following rule  $f$ . If  $n$  is the amount of monthly salary of a worker  $a$ , then  $f : a \mapsto n$ . Because for each worker there corresponds a unique positive rational number,  $f$  is a function from  $A$  into  $\mathbb{Q}$ .

### EXAMPLE 5.1.3

- (i) Let  $f$  be a relation from  $\mathbb{R}$  into  $\mathbb{R}$  given by

$$f = \{(x, 7x + 3) \mid x \in \mathbb{R}\}.$$

Now  $\mathcal{D}(f) = \{x \mid x \in \mathbb{R}\} = \mathbb{R}$ . Thus,  $f$  satisfies part (i) of Definition 5.1.1. Next let  $(x, 7x + 3), (y, 7y + 3) \in f$ . Suppose  $x = y$ . Then  $7x = 7y$ , therefore  $7x + 3 = 7y + 3$ . This shows that  $f$  is well defined and therefore  $f$  satisfies part (ii) of Definition 5.1.1. Hence,  $f$  is a function from  $\mathbb{R}$  into  $\mathbb{R}$ .

- (ii) Let  $g$  be a relation from  $\mathbb{R}$  into  $\mathbb{R}$  given by

$$g = \{(x, 3x^2 + 5x + 2) \mid x \in \mathbb{R}\}.$$

Now  $\mathcal{D}(g) = \{x \mid x \in \mathbb{R}\} = \mathbb{R}$ . Thus,  $f$  satisfies part (i) of Definition 5.1.1. Next let

$$(x, 3x^2 + 5x + 2), (y, 3y^2 + 5y + 2) \in g.$$

Suppose  $x = y$ . Then  $3x^2 + 5x + 2 = 3y^2 + 5y + 2$ . This shows that  $g$  is well defined. Therefore,  $g$  satisfies part (ii) of Definition 5.1.1. Consequently,  $g$  is a function from  $\mathbb{R}$  into  $\mathbb{R}$ .

### EXAMPLE 5.1.4

Let  $f$  be the relation from  $\mathbb{Q}$  into  $\mathbb{Q}$  given by

$$f = \left\{ \left( \frac{p}{q}, q + p \right) \mid p, q \in \mathbb{Z}, q \neq 0 \right\}.$$

Observe that

$$\mathcal{D}(f) = \left\{ \frac{p}{q} \in \mathbb{Q} \mid p, q \in \mathbb{Z}, q \neq 0 \right\} = \mathbb{Q}.$$

However, we have  $\frac{3}{5} = \frac{9}{15} \in \mathbb{Q}$  and  $(\frac{3}{5}, 8), (\frac{9}{15}, 24) \in f$ . But

$$f\left(\frac{3}{5}\right) = 8 \neq 24 = f\left(\frac{9}{15}\right).$$

Thus  $f$  is not well defined. Hence,  $f$  is not a function from  $\mathbb{Q}$  into  $\mathbb{Q}$ .

**DEFINITION 5.1.5** ▶ Let  $A$  and  $B$  be sets and  $f : A \rightarrow B$  be a function. The set  $A$  is referred to as the **domain** of the function and the set  $B$  is called the **codomain**, or **target**, of  $f$ . The set

$$f(A) = \{f(x) \mid x \in A\}$$

is a subset of the codomain  $B$ . The set  $f(A)$  is called the **range** of the function  $f$ , or the *image* of the set  $A$  under the function  $f$ , denoted by  $\text{Im}(f)$  or  $I(f)$ .

### EXAMPLE 5.1.6

- (i) Let  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  be a function defined by  $f(n) = 2n + 1$  for all  $n \in \mathbb{Z}$ . Observe for all  $n \in \mathbb{Z}$ ,  $2n + 1$  is an odd integer. Thus,

$$\text{Im}(f) = \{f(m) \mid m \in \mathbb{Z}\} = \{2m + 1 \mid m \in \mathbb{Z}\} = \text{set of all odd integers}.$$

- (ii) Let  $g : \mathbb{R} \rightarrow \mathbb{R}$  be a function defined by  $g(x) = x^2$  for all  $x \in \mathbb{R}$ . Then

$$\begin{aligned} \text{Im}(g) &= \{y \in \mathbb{R} \mid y = g(x) \text{ for some } x \in \mathbb{R}\} \\ &= \{y \in \mathbb{R} \mid y = x^2 \text{ for some } x \in \mathbb{R}\}. \\ &= \{y \in \mathbb{R} \mid y \geq 0\}. \end{aligned}$$

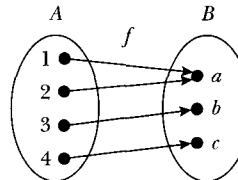
- (iii) Let  $A = \{1, 2, 3, 4, 5\}$  and  $B = \{a, b, c, d\}$ . Let  $h : A \rightarrow B$  be defined by  $h(1) = a$ ,  $h(2) = a$ ,  $h(3) = c$ ,  $h(4) = d$ , and  $h(5) = c$ . Then  $\text{Im}(f) = \{a, c, d\}$ .

There are different ways to describe a function. A function with a finite domain can be described simply by listing the images of the elements of the domain. For example,  $f : \{1, 2, 3, 4\} \rightarrow \{a, b, c\}$  by stipulating  $f(1) = a$ ,  $f(2) = a$ ,  $f(3) = b$ ,  $f(4) = c$  describes a function from  $\{1, 2, 3, 4\}$  to  $\{a, b, c\}$ .

We can also describe this  $f$  in the following way:

$$\begin{aligned} f : 1 &\mapsto a \\ 2 &\mapsto a \\ 3 &\mapsto b \\ 4 &\mapsto c. \end{aligned}$$

Every function is a relation. Therefore, functions on finite sets can be described by arrow diagrams. In the case of functions, we may draw the arrow diagram slightly differently. If  $f : A \rightarrow B$  is a function from a finite set  $A$  into a finite set  $B$ , then in the arrow diagram, the elements of  $A$  are enclosed in ellipses rather than individual boxes. For example, we may draw the arrow diagram for the preceding function as shown in Figure 5.5.



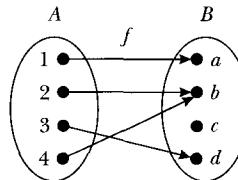
**FIGURE 5.5** Arrow diagram of  $f$

**REMARK 5.1.7** ▶ To determine from its arrow diagram whether a relation  $f$  from a set  $A$  into a set  $B$  is a function, we do two things: (1) Check to see if there is an arrow from each element of  $A$  to an element of  $B$ . This would ensure that the domain of  $f$  is the set  $A$ , i.e.,  $D(f) = A$ ; (2) Check to see that there is only one arrow from each element of  $A$  to an element of  $B$ . This would ensure that  $f$  is well defined.

**EXAMPLE 5.1.8**

Let  $A = \{1, 2, 3, 4\}$  and  $B = \{a, b, c, d\}$  be sets.

- (i) The arrow diagram in Figure 5.6 represents the relation  $f$  from  $A$  into  $B$ .

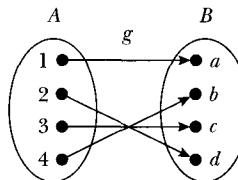


**FIGURE 5.6** Arrow diagram of  $f$

Notice that every element of  $A$  has some image in  $B$ . Also note that an element of  $A$  is related to only one element of  $B$ ; i.e., for each  $a \in A$  there exists a unique element  $b \in B$  such that  $f(a) = b$ . Hence,  $f$  is a function from  $A$  into  $B$ . The image of  $f$  is the set  $\text{Im}(f) = \{a, b, d\}$ .

Let us look at the arrow diagram of  $f$ . There is an arrow originating from each element of  $A$  to an element of  $B$ . Therefore,  $D(f) = A$ . Also, there is only one arrow from each element of  $A$  to an element of  $B$ . Thus,  $f$  is well defined.

- (ii) The arrow diagram in Figure 5.7 represents the relation  $g$  from  $A$  into  $B$ .



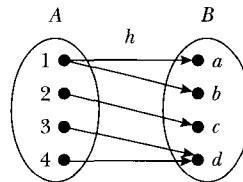
**FIGURE 5.7** Arrow diagram of  $g$

We find that every element of  $A$  has some image in  $B$ . Therefore,  $D(g) = A$ . Also, for each  $a \in A$ , there exists a unique element  $b \in B$  such that  $g(a) = b$ . Hence,  $g$  is a function from  $A$  into  $B$ . For this function the image of  $g$  is  $\text{Im}(g) = \{a, b, c, d\} = B$ .

Let us look at the arrow diagram of  $g$ . There is an arrow originating from each element of  $A$  to an element of  $B$ . Therefore,  $D(g) = A$ . Also,

there is only one arrow from each element of  $A$  to an element of  $B$ . Thus,  $g$  is well defined.

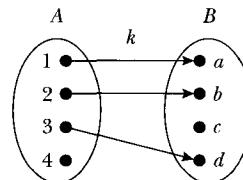
- (iii) Let  $h$  be the relation described by the arrow diagram in Figure 5.8.



**FIGURE 5.8** Arrow diagram of  $h$

Notice that every element of  $A$  has some image in  $B$ ; i.e., there is an arrow originating from each element of  $A$  to an element of  $B$ . Therefore,  $D(h) = A$ . However, element 1 has two images in  $B$ ; i.e., there are two arrows originating from 1, one going to  $a$  and another going to  $b$ , so  $h$  is not well defined. Thus, we see that the first condition of Definition 5.1.1 is satisfied, but the second one is not. Hence,  $h$  is not a function.

- (iv) The arrow diagram in Figure 5.9 represents a relation from  $A$  into  $B$ .



**FIGURE 5.9** Arrow diagram of  $k$

Here, we see that not every element of  $A$  has an image in  $B$ . For example, the element 4 has no image in  $B$ . In other words, there is no arrow originating from 4. Therefore,  $4 \notin D(k)$ , so  $D(k) \neq A$ . This implies that  $k$  is not a function from  $A$  into  $B$ .

If the domain and the range of a function are numbers, then the function is typically defined by means of an algebraic formula. For example (as shown in Example 5.1.3),

$$f(x) = 7x + 3 \quad \text{and} \quad g(x) = 3x^2 + 5x + 2$$

are functions from  $\mathbb{R}$  into  $\mathbb{R}$ . For simplicity and easy reference, we call such functions **numeric functions**.

We can also define numeric functions in such a way so that different expressions are used to find the image of an element. For example, consider the function  $f : \mathbb{N} \rightarrow \mathbb{N}$  defined by: For all  $n \in \mathbb{N}$

$$f(n) = \begin{cases} 1 & \text{if } n = 1, \\ n - 1 & \text{if } n > 1. \end{cases}$$

To determine the image of a natural number, under this  $f$ , we do the following: If the number is greater than 1, we use the expression  $n - 1$ ; otherwise the image

is 1. For example,  $f(1) = 1$  and  $f(2) = 2 - 1 = 1$ . For another example, consider the function  $g : \mathbb{Z} \rightarrow \{1, -1\}$  defined by: For all  $n \in \mathbb{Z}$

$$g(n) = \begin{cases} 1 & \text{if } n \text{ is even,} \\ -1 & \text{if } n \text{ is odd.} \end{cases}$$

Here the image of any even integer is 1 and the image of any odd integer is  $-1$ .

Next consider the function  $h : \mathbb{R} \rightarrow \mathbb{R}$  defined by: For all  $x \in \mathbb{R}$

$$h(x) = \begin{cases} 2x - 5 & \text{if } x \geq 0, \\ x^2 + 3 & \text{if } x < 0. \end{cases}$$

Here to determine the image of a nonnegative real number, we use the expression  $2x - 5$ ; to determine the image of a negative real number, we use the expression  $x^2 + 3$ .

**DEFINITION 5.1.9** ► A function  $f : A \rightarrow A$  is said to be the **identity function** if  $f(x) = x$  for all  $x \in A$ . This function is usually denoted by  $i_A$ .

**DEFINITION 5.1.10** ► A function  $f : A \rightarrow B$  is said to be a **constant function** if there exists  $b \in B$  such that  $f(x) = b$  for all  $x \in A$ . That is, all elements of  $A$  are mapped to only one element of  $B$ .

Notice that under a constant function every element of the domain set goes to some fixed element in the codomain.

**EXAMPLE 5.1.11** Let  $f : \mathbb{Z} \rightarrow \{1, 2, 5, 7, 10\}$  be defined by  $f(x) = 5$  for all  $x \in \mathbb{Z}$ . Then  $f$  is a constant function and  $\text{Im}(f) = \{5\}$ .

Let  $f : X \rightarrow Y$  and  $g : X \rightarrow Y$  be functions. Because

$$f \subseteq X \times Y \quad \text{and} \quad g \subseteq X \times Y,$$

we can show that  $f = g$  if and only if

$$f(x) = g(x)$$

for all  $x \in X$ . (See Exercise 14, p. 298.)

**EXAMPLE 5.1.12** Let  $f : \mathbb{R}^+ \rightarrow \mathbb{R}$  be given by

$$f(x) = x + \frac{x}{|x|}$$

and  $g : \mathbb{R}^+ \rightarrow \mathbb{R}$  be given by

$$g(x) = x + 1,$$

for all  $x \in \mathbb{R}^+$ , where  $\mathbb{R}^+$  is the set of all positive real numbers. Notice that if  $x \in \mathbb{R}^+$ , then  $x \neq 0$  and  $|x| = x$ , therefore  $f(x) = x + \frac{x}{|x|} = x + \frac{x}{x} = x + 1 = g(x)$ . Hence,  $f = g$ .

## One-One, Onto, and One-to-One Correspondence

The functions that we considered in the preceding section are not subject to satisfying any specific conditions. To make this topic interesting, in this section we discuss functions that satisfy certain conditions.

**DEFINITION 5.1.13** ▶ Let  $A$  and  $B$  be sets and  $f : A \rightarrow B$ . Then

- (i)  $f$  is called **one-one** (or **injective** or **injection**) if for all  $a_1, a_2 \in A$ ,

$$a_1 \neq a_2 \Rightarrow f(a_1) \neq f(a_2)$$

(i.e., images of distinct elements of the domain are distinct).

- (ii)  $f$  is called **onto**  $B$  (or **surjective** or **surjection**) if for every  $b \in B$  there exists at least one  $a \in A$  such that  $f(a) = b$ , i.e.,

$$\text{Im}(f) = B.$$

- (iii)  $f$  is called **one-to-one correspondence** (or **bijection** or **bijection**) if  $f$  is both one-one and onto.

**REMARK 5.1.14** ▶ Note that Definition 5.1.13(i) is equivalent to the following:  $f$  is called a *one-one* (*injective* or *injection*) if for all  $a_1, a_2 \in A$ ,

$$f(a_1) = f(a_2) \Rightarrow a_1 = a_2.$$

**REMARK 5.1.15** ▶ Let  $f : A \rightarrow B$  be a function from the set  $A$  into the set  $B$ . If  $f$  is onto  $B$ , then sometimes we simply say that  $f$  is onto or  $f$  is a function from  $A$  onto  $B$ .

### EXAMPLE 5.1.16

Let  $A = \{1, 2, 3, 4\}$  and  $B = \{a, b, c, d\}$ . Let  $f : A \rightarrow B$  be a function such that the arrow diagram of  $f$  is as shown in Figure 5.10.

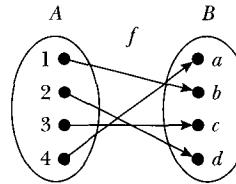


FIGURE 5.10 Arrow diagram of  $f$

Notice that the arrows from a distinct element of  $A$  go to a distinct element of  $B$ . That is, every element of  $B$  has at most one arrow coming to it. It follows that if  $a_1, a_2 \in A$  and  $a_1 \neq a_2$ , then  $f(a_1) \neq f(a_2)$ . Hence,  $f$  is one-one.

Next, we note that each element of  $B$  has an arrow coming to it. That is, each element of  $B$  has a preimage. Therefore,  $\text{Im}(f) = B$ . Hence,  $f$  is onto  $B$ . It also follows that  $f$  is a one-to-one correspondence.

### EXAMPLE 5.1.17

Let  $A$  be a set and consider the identity function  $i_A : A \rightarrow A$ . Then  $i_A(a) = a$  for all  $a \in A$ . Suppose  $a_1, a_2 \in A$  and  $a_1 \neq a_2$ . By the definition of  $i_A$ , we have  $i_A(a_1) = a_1$  and  $i_A(a_2) = a_2$ . It follows that  $i_A(a_1) \neq i_A(a_2)$ . Thus,  $i_A$  is one-one. We also note that every element is its own preimage as  $a = i_A(a)$  for all  $a \in A$ . Hence,  $i_A$  is onto  $A$ . It also follows that  $i_A$  is a one-to-one correspondence.

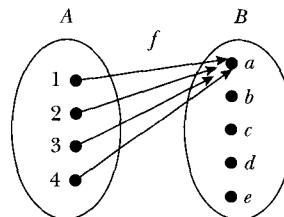
**EXAMPLE 5.1.18**

Let  $A = \{1, 2, 3, 4\}$  and  $B = \{a, b, c, d, e\}$ . We consider the following functions from  $A$  into  $B$ .

$$(i) \quad f : 1 \mapsto a, 2 \mapsto a, 3 \mapsto a, 4 \mapsto a$$

For this function we find that the images of distinct elements of the domain are not distinct. For example  $1 \neq 2$ , but  $f(1) = a = f(2)$ . Moreover, we find that  $\text{Im}(f) = \{a\} \neq B$ . Hence,  $f$  is neither one-one nor onto  $B$ .

Let us draw the arrow diagram of  $f$  (see Figure 5.11).



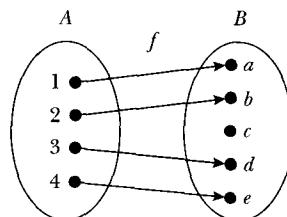
**FIGURE 5.11** Arrow diagram of  $f$

In the arrow diagram, we see that there are several arrows coming from distinct elements of  $A$  to the element  $a$  of  $B$ . Therefore,  $f$  is not one-one. We also notice that there is no arrow from any element of  $A$  to the element  $b \in B$ . Thus,  $b$  has no preimage. Hence,  $f$  is not onto  $B$ .

$$(ii) \quad f : 1 \mapsto a, 2 \mapsto b, 3 \mapsto d, 4 \mapsto e$$

For this function, we find that the images of distinct elements of the domain are distinct. Thus,  $f$  is one-one. In this function, for the element  $c$  of  $B$ , the codomain, there is no element  $x$  in the domain such that  $f(x) = c$ ; i.e.,  $c$  has no preimage. Hence,  $f$  is not onto  $B$ .

Let us draw the arrow diagram of  $f$  (see Figure 5.12).



**FIGURE 5.12** Arrow diagram of  $f$

In the arrow diagram, we see that arrows from distinct elements of  $A$  go to distinct elements of  $B$ ; i.e., an element of  $B$  has at most one arrow coming to it. Therefore,  $f$  is one-one. However, element  $c$  of  $B$  has no arrow coming to it; i.e., it has no preimage. Therefore,  $f$  is not onto  $B$ .

**EXAMPLE 5.1.19**

Let  $B = \{1, -1\}$  and let  $f : \mathbb{Z} \rightarrow B$  be defined by

$$f(n) = \begin{cases} 1 & \text{if } n \text{ is even,} \\ -1 & \text{if } n \text{ is odd.} \end{cases}$$

For this function  $f$ , we have  $f(2) = 1 = f(4)$ , but  $2 \neq 4$ . Therefore,  $f$  is not one-one. If we consider the arrow diagram of  $f$ , we will find that arrows from all even numbers end at 1, and arrows from all odd numbers end at  $-1$ . In other

words, the element  $1 \in B$  has more than one arrow coming to it. This is enough to conclude that  $f$  is not one-one.

Now the image of  $f$ ,  $\text{Im}(f) = \{1, -1\} = B$ . Hence,  $f$  is onto  $B$ .

**EXAMPLE 5.1.20**

Let the function  $f : \mathbb{N} \rightarrow \mathbb{Z}$  be defined by

$$f(n) = 2n + 5 \quad \text{for all } n \in \mathbb{N}.$$

In the preceding examples, to show the function is one-one, we showed that if elements of the domain are distinct, then their images are distinct. Here we use the equivalent definition of a one-one function given in Remark 5.1.14. That is, we show that if the images of the two elements of the domain are the same, then those elements of the domain are also the same.

Let  $n, m \in \mathbb{N}$ . Suppose that  $f(n) = f(m)$ . Then

$$\begin{aligned} f(n) &= f(m) \\ \Rightarrow 2n + 5 &= 2m + 5 \\ \Rightarrow n &= m. \end{aligned}$$

This shows that  $f$  is one-one.

Consider  $8 \in \mathbb{Z}$ . Suppose that there exists  $n \in \mathbb{N}$  such that  $f(n) = 8$ . This implies that  $2n + 5 = 8$ , so  $n = \frac{3}{2}$ , i.e.,  $\frac{3}{2} \in \mathbb{N}$ , which is a contradiction. It follows that 8 has no preimage. Hence,  $f$  is not onto  $\mathbb{Z}$ .

Notice that to show  $f$  is not onto  $\mathbb{Z}$ , we chose the element  $8 \in \mathbb{Z}$  and proved that it has no preimage. There is nothing special in choosing 8. We can, in fact, choose any even number or any nonpositive integer and show that there are no preimages for such elements. This is due to the fact that if  $n$  is positive, then its image  $2n + 5$  is positive as well as an odd integer. This implies that the image of  $f$  does not contain any nonpositive as well as even integers. However, to prove that  $f$  is not onto  $\mathbb{Z}$ , we only need to show that there is an element in  $\mathbb{Z}$  that has no preimage.

**EXAMPLE 5.1.21**

- (i) Let  $B = \mathbb{N} \cup \{0\}$ . Consider the relation  $f$  from  $\mathbb{Z}$  into  $B$  defined by:  $f(n) = |n|$  for all  $n \in \mathbb{Z}$ . It can be checked that  $f$  is a function. Now

$$f(4) = |4| = 4 = |-4| = f(-4), \quad \text{but } 4 \neq -4.$$

This implies that  $f$  is not one-one.

Let  $b \in B$ . Then  $b$  is a nonnegative integer. Moreover,  $f(b) = b$ . This implies that a preimage of  $b$  is  $b$  itself. It now follows that  $f$  is onto  $B$ .

- (ii) Consider the relation  $f$  on  $\mathbb{Z}$  defined by  $f(n) = |n|$  for all  $n \in \mathbb{Z}$ . Then  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  is a function. As in part (i),  $f$  is not one-one. Moreover,  $f(n) = |n| \geq 0$ , so negative integers have no preimages. It follows that  $f$  is not onto  $\mathbb{Z}$ .

**EXAMPLE 5.1.22**

Let  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  be defined by  $f(n) = 9n + 5$  for all  $n \in \mathbb{Z}$ . Let  $n_1, n_2 \in \mathbb{Z}$ . Now

$$\begin{aligned} f(n_1) &= f(n_2) \\ \Rightarrow 9n_1 + 5 &= 9n_2 + 5 \\ \Rightarrow n_1 &= n_2. \end{aligned}$$

Hence,  $f$  is a one-one function.

However,  $f$  is not onto  $\mathbb{Z}$ . Indeed, here,  $4 \in \mathbb{Z}$  has no preimage under  $f$ . For, if  $f(n) = 4$  for some  $n \in \mathbb{Z}$ , then  $9n + 5 = 4$ , i.e.,  $n = \frac{-1}{9} \notin \mathbb{Z}$ , a contradiction.

Observe that if we consider  $f_1 : \mathbb{R} \rightarrow \mathbb{R}$  by defining

$$f_1(x) = 9x + 5$$

for all  $x \in \mathbb{R}$ , then  $f_1$  becomes an example of a function which is both one-one and onto  $\mathbb{R}$  and hence a one-to-one correspondence.

## Composition

Given two functions  $f : A \rightarrow B$  and  $g : B \rightarrow C$ , we now construct a function  $h : A \rightarrow C$  with the help of  $f$  and  $g$  as follows.

Let  $a \in A$ . Then  $f(a) \in B$ . Let  $f(a) = b$ . Now  $g(b) \in C$ . Suppose  $g(b) = c$ . So with each element  $a \in A$ , we can associate an element  $c \in C$ . The association

$$a \mapsto b \mapsto c$$

i.e.,

$$a \mapsto f(a) \mapsto g(f(a))$$

defines a function  $h : A \rightarrow C$  by

$$h(a) = g(f(a))$$

for all  $a \in A$ .

Let us, in fact, verify that  $h$  satisfies the required conditions of a function (Definition 5.1.1).

We note that

$$h = \{(a, g(f(a))) \in A \times C \mid a \in A\}$$

is a relation from  $A$  into  $C$ . From this it follows that the domain of  $h$  is  $A$ .

Now let  $(a, c), (a, d) \in A \times C$ . Then  $c = g(f(a))$  and  $d = g(f(a))$ . Because  $f$  and  $g$  are functions,  $f(a)$  and  $g(f(a))$  are unique. Hence,  $c = d$ . This implies that for every  $a \in A$ , there is exactly one  $c \in C$  such that  $(a, c) \in h$ , i.e.,  $h(a) = c$ .

The function  $h : A \rightarrow C$  is called the composite of  $f$  and  $g$ . The following is the formal definition.

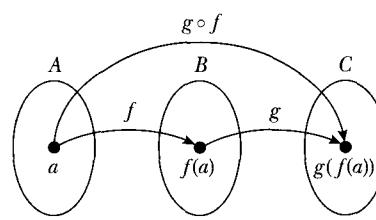
---

**DEFINITION 5.1.23** ▶ Let  $f : A \rightarrow B$  and  $g : B \rightarrow C$  be functions. The **composition** of  $f$  and  $g$ , written  $g \circ f$ , is the function from  $A$  to  $C$  defined as

$$(g \circ f)(a) = g(f(a)), \quad \text{for all } a \in A.$$

We sometimes write the composition,  $g \circ f$ , of the function  $f$  and  $g$  as  $gf$ .

Let  $f : A \rightarrow B$  and  $g : B \rightarrow C$  be functions. Pictorially,  $g \circ f$  is described in Figure 5.13.

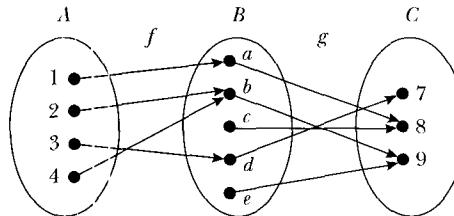


**FIGURE 5.13** Composition of the functions  $f$  and  $g$

We now consider some examples of the composition of two functions.

**EXAMPLE 5.1.24**

Let  $A = \{1, 2, 3, 4\}$ ,  $B = \{a, b, c, d, e\}$ , and  $C = \{7, 8, 9\}$ . Consider the functions  $f : A \rightarrow B$ ,  $g : B \rightarrow C$  as defined by the arrow diagrams in Figure 5.14.

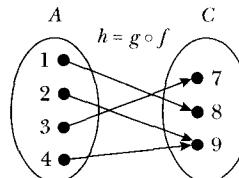


**FIGURE 5.14** Arrow diagram of the functions  $f$  and  $g$

Let us note that

$$\begin{aligned} 1 &\xrightarrow{f} a \xrightarrow{g} 8 \Rightarrow 1 \xrightarrow{g \circ f} 8 \\ 2 &\xrightarrow{f} b \xrightarrow{g} 9 \Rightarrow 2 \xrightarrow{g \circ f} 9 \\ 3 &\xrightarrow{f} d \xrightarrow{g} 7 \Rightarrow 3 \xrightarrow{g \circ f} 7 \\ 4 &\xrightarrow{f} b \xrightarrow{g} 9 \Rightarrow 4 \xrightarrow{g \circ f} 9 \end{aligned}$$

The arrow diagram in Figure 5.15 describes the function  $h = g \circ f : A \rightarrow C$ .



**FIGURE 5.15** Arrow diagram of  $h = g \circ f$

**EXAMPLE 5.1.25**

Consider the functions  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  and  $g : \mathbb{Z} \rightarrow \mathbb{Z}$  defined by

$$f(n) = (-1)^n,$$

for all  $n \in \mathbb{Z}$  and

$$g(n) = 2n,$$

for all  $n \in \mathbb{Z}$ . Then  $g \circ f : \mathbb{Z} \rightarrow \mathbb{Z}$  is given by

$$(g \circ f)(n) = g(f(n)) = g((-1)^n) = 2(-1)^n,$$

for all  $n \in \mathbb{Z}$ . Notice that

$$(g \circ f)(n) = 2(-1)^n = \begin{cases} 2 & \text{if } n \text{ is even,} \\ -2 & \text{if } n \text{ is odd.} \end{cases}$$

Notice that here we can also define  $f \circ g : \mathbb{Z} \rightarrow \mathbb{Z}$  by

$$(f \circ g)(n) = f(g(n)) = f(2n) = (-1)^{2n},$$

i.e.,

$$(f \circ g)(n) = 1$$

for all  $n \in \mathbb{Z}$ .

We note that  $(g \circ f)(2) = 2(-1)^2 = 2(1) = 2 \neq 1 = (f \circ g)(2)$ . Hence,  $g \circ f \neq f \circ g$ .

**REMARK 5.1.26** ▶ Example 5.1.25 shows that in general,  $g \circ f \neq f \circ g$  even when both are defined. That is, in general, the composition of functions is noncommutative.

### EXAMPLE 5.1.27

Consider the functions  $f : \mathbb{Z} \rightarrow \mathbb{Q}$  and  $g : \mathbb{Q} \rightarrow \mathbb{Q}$  given by

$$f(x) = \frac{1}{5}x$$

for all  $x \in \mathbb{Z}$  and

$$g(x) = x^2 + 1$$

for all  $x \in \mathbb{Q}$ . Observe that  $g \circ f : \mathbb{Z} \rightarrow \mathbb{Q}$  is given by

$$(g \circ f)(x) = g(f(x)) = g\left(\frac{x}{5}\right) = \left(\frac{x}{5}\right)^2 + 1 = \frac{x^2}{25} + 1$$

for all  $x \in \mathbb{Z}$ .

We leave the proof of the following theorem as an exercise.

**Theorem 5.1.28:** Let  $f : A \rightarrow B$  be a function from a set  $A$  into a set  $B$ .

Consider the identity functions  $i_A : A \rightarrow A$  and  $i_B : B \rightarrow B$ . Then

$$f \circ i_A = f = i_B \circ f.$$

Let us now describe an important property of the composition of functions.

**Theorem 5.1.29:** Let  $f : A \rightarrow B$ ,  $g : B \rightarrow C$ , and  $h : C \rightarrow D$ . Then

$$h \circ (g \circ f) = (h \circ g) \circ f;$$

i.e., composition of functions is associative, provided the composition is defined.

**Proof:** Observe that  $h \circ (g \circ f) : A \rightarrow D$  and  $(h \circ g) \circ f : A \rightarrow D$ . Let  $x \in A$ . Then

$$\begin{aligned}(h \circ (g \circ f))(x) &= h((g \circ f)(x)) \\ &= h(g(f(x)))\end{aligned}$$

and

$$\begin{aligned}((h \circ g) \circ f)(x) &= (h \circ g)(f(x)) \\ &= h(g(f(x))).\end{aligned}$$

Hence,  $(h \circ (g \circ f))(x) = ((h \circ g) \circ f)(x)$ . Because  $x$  is an arbitrary element of  $A$ , we conclude that  $h \circ (g \circ f) = (h \circ g) \circ f$ . ■

**Theorem 5.1.30:** Suppose that  $f : A \rightarrow B$  and  $g : B \rightarrow C$ . The following assertions hold.

- (i) If both  $f$  and  $g$  are one-one, then  $g \circ f$  is also one-one.
- (ii) If  $f$  is onto  $B$  and  $g$  is onto  $C$ , then  $g \circ f$  is also onto  $C$ .
- (iii) If both  $f$  and  $g$  are one-to-one correspondences, then  $g \circ f$  is also a one-to-one correspondence.

**Proof:**

- (i) Let  $a, a' \in A$ . Suppose that

$$(g \circ f)(a) = (g \circ f)(a').$$

We have

$$\begin{aligned} (g \circ f)(a) &= (g \circ f)(a') \\ \Rightarrow g(f(a)) &= g(f(a')) \\ \Rightarrow f(a) &= f(a'), \quad \text{because } g \text{ is one-one} \\ \Rightarrow a &= a', \quad \text{because } f \text{ is one-one.} \end{aligned}$$

Hence,  $g \circ f$  is one-one.

- (ii) Let  $c \in C$ . Because  $c \in C$  and  $g$  is onto  $C$ , there exists  $b \in B$  such that  $g(b) = c$ . Now  $b \in B$  and  $f$  is onto  $B$ , so there exists  $a \in A$  such that  $f(a) = b$ . Thus,

$$(g \circ f)(a) = g(f(a)) = g(b) = c.$$

Hence,  $g \circ f$  is onto  $C$ .

- (iii) This follows from parts (i) and (ii). ■

Using the notion of composition of functions, we can define the power of a function as described next.

Let  $f : A \rightarrow A$  be a function. Now  $f$  is a relation on  $A$ , so we can define  $f^2$ ,  $f^3$ ,  $f^4$ , and so on as follows: For all  $a \in A$ ,

$$\begin{aligned} f^1(a) &= f(a) \\ f^2(a) &= (f \circ f)(a) = f(f(a)) \\ f^3(a) &= (f \circ f^2)(a) = f(f^2(a)) \\ &\vdots \end{aligned}$$

Formally, we define  $f^n$  for all  $n \in \mathbb{N}$  as follows: For all  $a \in A$ ,

$$\begin{aligned} f^1(a) &= f(a) \\ f^n(a) &= (f \circ f^{n-1})(a) \quad \text{if } n > 1. \end{aligned}$$

It is interesting to observe that if  $A$  is a finite set and  $f : A \rightarrow A$  is a one-one function on  $A$ , then  $f$  is automatically onto  $A$ . In order to prove this assertion, we note that if  $f : A \rightarrow A$  is a one-one function, then from Theorem 5.1.30(i), we find  $f \circ f : A \rightarrow A$  is a one-one function and by induction we can show that

$f^n : A \rightarrow A$  is a one-one function for all integers  $n \geq 1$  (see Worked-Out Exercise 7, p. 295).

**Theorem 5.1.31:** Let  $A$  be a finite set  $A$  and  $f : A \rightarrow A$  be a function. If  $f$  is one-one, then  $f$  is onto  $A$  and hence a one-to-one correspondence.

**Proof:** Let  $a \in A$ . Now  $f^n : A \rightarrow A$ , so  $f^n(a) \in A$  for all  $n \geq 1$ . This implies that

$$\{a, f(a), f^2(a), \dots\} \subseteq A.$$

Because  $A$  is finite, it follows that  $\{a, f(a), f^2(a), \dots\}$  is finite. Therefore, there must exist positive integers  $r$  and  $s$  such that  $r > s$  and

$$f^r(a) = f^s(a).$$

Now

$$\begin{aligned} f^r(a) &= f^s(a) \\ \Rightarrow (f^s \circ f^{r-s})(a) &= f^s(a) \\ \Rightarrow f^s(f^{r-s}(a)) &= f^s(a) \\ \Rightarrow f^{r-s}(a) &= a, \quad \text{because } f^s \text{ is one-one.} \end{aligned}$$

Let

$$a' = f^{r-s-1}(a) \in A.$$

Then

$$f(a') = f(f^{r-s-1}(a)) = f^{r-s}(a) = a.$$

We can now conclude that  $f$  is onto  $A$ . Consequently,  $f$  is a one-to-one correspondence. ■



## WORKED-OUT EXERCISES

**Exercise 1:** Determine which of the relations  $f$  are functions from the set  $X$  to the set  $Y$ .

- (a)  $X = \{-2, -1, 0, 1, 2\}$ ,  $Y = \{-3, 4, 5\}$ , and  $f = \{(-2, -3), (-1, -3), (0, 4), (1, 5), (2, -3)\}$ .
- (b)  $X = \{-2, -1, 0, 1, 2\}$ ,  $Y = \{-3, 4, 5\}$ , and  $f = \{(-2, -3), (1, 4), (2, 5)\}$ .
- (c)  $X = Y = \{-3, -1, 0, 2\}$ , and  $f = \{(-3, -1), (-3, 0), (-1, 2), (0, 2), (2, -1)\}$ .
- (d)  $X = Y = \text{the set of all integers}$ , and  $f = \{(a, b) \in \mathbb{Z} \times \mathbb{Z} \mid b = a + 1\}$ .
- (e)  $X = Y = \text{the set of all integers}$ , and  $f = \{(a, b) \in \mathbb{Z} \times \mathbb{Z} \mid b = \sqrt{a}\}$ .

In case any of these relations are functions, determine if they are one-one, onto  $Y$ , and/or one-to-one correspondences.

**Solution:**

- (a) The domain of  $f$ ,  $D(f) = \{-2, -1, 0, 1, 2\} = X$ . Moreover, for any  $a \in X$  there are no two distinct elements

$b$  and  $c$  in  $Y$  such that  $(a, b) \in f$  and  $(a, c) \in f$ . Hence,  $f$  is a function. Now  $\text{Im}(f) = \{-3, 4, 5\} = Y$ . Thus,  $f$  is onto  $Y$ . However,  $f(-2) = -3 = f(-1)$  and  $-2 \neq -3$ . Hence,  $f$  is not one-one.

- (b) The domain of  $f$ ,  $D(f) = \{-2, 1, 2\} \neq X$ . Hence,  $f$  is not a function on  $X$ .
- (c) The domain of  $f$ ,  $D(f) = \{-3, -1, 0, 2\} = X$ . Now  $(-3, -1) \in f$  and  $(-3, 0) \in f$ , so  $f(-3) = -1$  and simultaneously  $f(-3) = 0$ , so  $f$  is not well defined. Hence,  $f$  is not a function.
- (d) The domain of  $f$ ,  $D(f) = \mathbb{Z}$ . Let  $b, c \in \mathbb{Z}$  be such that  $(a, b) \in f$  and  $(a, c) \in f$ . Then  $b = a + 1$  and  $c = a + 1$ . Thus,  $b = a + 1 = c$ . Hence,  $f$  is well defined. Consequently,  $f$  is a function.

To show  $f$  is one-one, let  $a, b \in \mathbb{Z}$  and  $a \neq b$ . Then  $a + 1 \neq b + 1$ , so  $f(a) \neq f(b)$ . Thus,  $f$  is one-one. (To show  $f$  is one-one, we can also argue as follows: Let  $a, b \in \mathbb{Z}$  and  $f(a) = f(b)$ . This implies that  $a + 1 = b + 1$ , so  $a = b$ . Hence,  $f$  is one-one.)

To show  $f$  is onto  $\mathbb{Z}$ , let  $a \in \mathbb{Z}$ . Then  $a - 1 \in \mathbb{Z}$  and  $f(a - 1) = (a - 1) + 1 = a$ ; i.e.,  $a - 1$  is the preimage of  $a$ . Hence, every element of  $\mathbb{Z}$  has a preimage and so  $\text{Im}(f) = \mathbb{Z}$ . This shows that  $f$  is onto  $\mathbb{Z}$ . Consequently,  $f$  is a one-to-one correspondence.

- (e)  $2 \in \mathbb{Z}$ , but  $\sqrt{2} \notin \mathbb{Z}$ . Thus  $(2, \sqrt{2}) \notin f$ . Hence, the domain of  $f$ ,  $D(f) \neq \mathbb{Z}$ . This implies that  $f$  is not a function.

**Exercise 2:** Let  $f$  be the function from the set  $X = \{2, 3, 4, 5, 6, 7\}$  into the set  $Y = \{0, 1, 2, 3, 4\}$  defined by  $f(x) = 2x \pmod{5}$ . Write  $f$  as a set of ordered pairs. Is  $f$  one-one or onto  $Y$ ?

**Solution:** (Recall that  $m \pmod{n}$  is the remainder when  $m$  is divided by  $n$ ).  $2 \cdot 2 \pmod{5} = 4$ ,  $2 \cdot 3 \pmod{5} = 1$ ,  $2 \cdot 4 \pmod{5} = 3$ ,  $2 \cdot 5 \pmod{5} = 0$ ,  $2 \cdot 6 \pmod{5} = 2$ ,  $2 \cdot 7 \pmod{5} = 4$ . Hence,  $f = \{(2, 4), (3, 1), (4, 3), (5, 0), (6, 2), (7, 4)\}$ . Now  $2 \neq 7$  and  $f(2) = 4 = f(7)$ . Thus,  $f$  is not one-one. Again, the range of  $f$ ,  $\text{Im}(f) = \{4, 1, 3, 0, 2\} = Y$ , so  $f$  is onto  $Y$ .

**Exercise 3:** Determine which of the following functions are one-one, onto, or both one-one and onto.

- (a)  $f : \mathbb{N} \rightarrow \mathbb{Z} - \{0\}$  defined by  $f(n) = -n$  for all  $n \in \mathbb{N}$ .
- (b)  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  defined by  $f(x) = x - 4$  for all  $x \in \mathbb{Z}$ .
- (c)  $f : \mathbb{R} \rightarrow \mathbb{R}$  defined by  $f(x) = |x| + x$  for all  $x \in \mathbb{R}$ .
- (d)  $f : \mathbb{R} \rightarrow \mathbb{R}$  defined by  $f(x) = x^3$  for all  $x \in \mathbb{R}$ .
- (e)  $f : \mathbb{C} \rightarrow \mathbb{R}$  defined by  $f(z) = |z|$  for all  $z \in \mathbb{C}$ .

**Solution:**

- (a) Let  $n, m \in \mathbb{N}$ . Suppose that  $f(n) = f(m)$ . Then  $-n = -m$ , so  $n = m$ . Therefore,  $f$  is one-one. Notice that for all  $n \in \mathbb{N}$ ,  $f(n) = -n < 0$ . This indicates that positive integers are not in the range of  $f$  so  $f$  is not onto  $\mathbb{Z} - \{0\}$ . To be specific, consider  $3 \in \mathbb{Z} - \{0\}$ . Suppose that  $3$  has a preimage. Then there exists  $n \in \mathbb{N}$  such that  $f(n) = 3$ . This implies that  $3 = f(n) = -n < 0$ , a contradiction. Hence,  $f$  is not onto  $\mathbb{Z} - \{0\}$ .

- (b) Let  $x, y \in \mathbb{Z}$ . Suppose that  $f(x) = f(y)$ . Then  $x - 4 = y - 4$ , so  $x = y$ . This shows that  $f$  is one-one.

Now  $f$  is onto  $\mathbb{Z}$  if and only if for all  $y \in \mathbb{Z}$  there exists  $x \in \mathbb{Z}$  such that  $f(x) = y$ .

Let  $y \in \mathbb{Z}$ . If  $f(x) = y$ , then  $x - 4 = y$  or  $x = y + 4$ . Also,  $y + 4 \in \mathbb{Z}$ . Thus, we can take  $x$  to be  $y + 4$ . Now  $f(y + 4) = y + 4 - 4 = y$ ; i.e.,  $y + 4$  is the preimage of  $y$ . Therefore,  $f$  is onto  $\mathbb{Z}$ . Hence,  $f$  is one-one and onto  $\mathbb{Z}$ .

- (c) Consider  $-1, -2 \in \mathbb{R}$ . Now  $f(-1) = |-1| + (-1) = 1 + (-1) = 0$ , and  $f(-2) = |-2| + (-2) = 2 + (-2) = 0$ . This shows that  $-1 \neq -2$ , but  $f(-1) = 0 = f(-2)$ . Therefore,  $f$  is not one-one.

Now for any  $x \in \mathbb{R}$ ,

$$f(x) = \begin{cases} |x| + x = x + x = 2x, & \text{if } x \geq 0, \\ |x| + x = -x + x = 0, & \text{if } x < 0. \end{cases}$$

Thus, range of  $f$ ,  $\text{Im}(f) \neq \mathbb{R}$ . Hence,  $f$  is not onto  $\mathbb{R}$ .

- (d) Let  $x$  and  $y$  be two elements of  $\mathbb{R}$ . Suppose that  $f(x) = f(y)$ . Now

$$\begin{aligned} f(x) &= f(y) \\ \Rightarrow x^3 &= y^3 \\ \Rightarrow x^3 - y^3 &= 0 \\ \Rightarrow (x - y)(x^2 + xy + y^2) &= 0 \\ \Rightarrow (x - y) \left( (x + \frac{1}{2}y)^2 + \frac{3}{4}y^2 \right) &= 0 \\ \Rightarrow x - y = 0 \text{ or } (x + \frac{1}{2}y)^2 + \frac{3}{4}y^2 &= 0. \end{aligned}$$

If  $x - y \neq 0$ , then  $x \neq y$  and this implies that  $(x + \frac{1}{2}y)^2 + \frac{3}{4}y^2 > 0$ . Thus, it follows that  $x - y = 0$ , which in turn implies that  $x = y$ . Hence,  $f$  is one-one.

Now let  $a \in \mathbb{R}$ . Because the equation  $x^3 = a$  has a solution  $b$  in  $\mathbb{R}$ , there exists an element  $b$  in  $\mathbb{R}$  such that  $f(b) = b^3 = a$ . Hence,  $f$  is onto  $\mathbb{R}$ . Consequently,  $f$  is a one-to-one correspondence.

- (e) Let  $z = x + iy$  be any complex number, where  $x, y \in \mathbb{R}$ . Then  $|z| = \sqrt{x^2 + y^2} \geq 0$ . Therefore, the range of  $f$ ,  $\text{Im}(f) \neq \mathbb{R}$ . Thus,  $f$  is not onto  $\mathbb{R}$ . Also for  $z_1 = 2 + 3i$  and  $z_2 = 2 - 3i$ ,  $|z_1| = \sqrt{2^2 + 3^2} = \sqrt{13}$  and  $|z_2| = \sqrt{2^2 + 3^2} = \sqrt{13}$ . Because  $2 + 3i \neq 2 - 3i$ ,  $z_1 \neq z_2$ . However,  $f(z_1) = \sqrt{13} = f(z_2)$ . Hence,  $f$  is not one-one.

**Exercise 4:** Let  $f$  be the function from the set  $\mathbb{N}$  into the set  $X = \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$  defined by  $f(x) = x \pmod{7}$  for all  $x \in \mathbb{N}$ . Find  $\text{Im}(f)$ . Is  $f$  onto  $X$ ? Is  $f$  one-one?

**Solution:** We know that for any positive integer  $n$ ,  $n \pmod{7}$  is the remainder when  $n$  is divided by 7. Now by the division algorithm,  $n = 7t + r$ , where  $0 \leq r < 7$ . Then  $n \pmod{7} = r$ . Hence,  $\text{Im}(f) = \{0, 1, 2, 3, 4, 5, 6\}$ . Because  $\{0, 1, 2, 3, 4, 5, 6\} \neq \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$ , it follows that  $f$  is not onto  $X$ .

Again,  $10 \pmod{7} = 3 = 17 \pmod{7}$ . Hence,  $f(10) = f(17)$ . However,  $10 \neq 17$ . Therefore,  $f$  is not one-one.

**Exercise 5:** Let  $f : \mathbb{R} \rightarrow \mathbb{R}$  defined by  $f(x) = x^2 - 4x$ . Find  $\text{Im}(f)$ . Is  $f$  onto  $\mathbb{R}$ ? Is  $f$  one-one?

**Solution:** Let  $y \in \text{Im}(f)$ . Then  $f(x) = y$  for some  $x \in \mathbb{R}$ , i.e.,  $y = f(x) = x^2 - 4x$ . Now

$$\begin{aligned} y &= x^2 - 4x \\ \Rightarrow y + 4 &= x^2 - 4x + 4 \\ \Rightarrow y + 4 &= (x - 2)^2 \\ \Rightarrow y + 4 &\geq 0 \\ \Rightarrow y &\geq -4 \end{aligned}$$

This implies that  $\text{Im}(f) = \{y \in \mathbb{R} \mid y \geq -4\}$ . From this it also follows that  $\text{Im}(f) \neq \mathbb{R}$ , so  $f$  is not onto  $\mathbb{R}$ .

We can also show that  $f$  is not onto  $\mathbb{R}$  by finding an element that has no preimage. For example, consider  $-5 \in \mathbb{R}$ . Suppose that  $f(x) = -5$  for some  $x \in \mathbb{R}$ . Then

$$\begin{aligned} -5 &= x^2 - 4x \\ \Rightarrow -5 + 4 &= x^2 - 4x + 4 \\ \Rightarrow -1 &= (x - 2)^2, \end{aligned}$$

which is impossible because  $(x - 2)^2 \geq 0$ .

Moreover,  $f$  is not one-one as  $0 \neq 4$  and  $f(0) = 0 = f(4)$ .

**Exercise 6:** Suppose that  $f : A \rightarrow B$  and  $g : B \rightarrow C$ . Then prove that

- if  $g \circ f$  is one-one, then  $f$  is one-one;
- if  $g \circ f$  is onto  $C$ , then  $g$  is onto  $C$ ;
- if  $g \circ f$  is a one-one and onto  $C$ , then  $f$  is one-one and  $g$  is onto  $C$ .

**Solution:**

- Suppose that  $g \circ f$  is one-one. To show  $f$  is one-one, let  $a_1, a_2 \in A$  and  $f(a_1) = f(a_2)$ . Now  $f(a_1) = f(a_2)$  and  $g$  is a function from  $B$  to  $C$ . Therefore,

$$g(f(a_1)) = g(f(a_2)),$$

i.e.,

$$(g \circ f)(a_1) = (g \circ f)(a_2).$$

This implies that  $a_1 = a_2$ , because  $g \circ f$  is one-one. Hence,  $f$  is one-one.

- Suppose that  $g \circ f$  is onto  $C$ . To show  $g$  is onto  $C$ , let  $c \in C$ . Because  $g \circ f$  is onto  $C$  and  $c \in C$ , there exists  $a \in A$  such that

$$(g \circ f)(a) = c.$$

This implies that

$$g(f(a)) = c.$$

Let  $b = f(a) \in B$ . Then we have

$$c = (g \circ f)(a) = g(f(a)) = g(b).$$

That is,  $b = f(a)$  is the preimage of  $c$ . Because  $c$  is an arbitrary element of  $C$ , we can conclude that  $g$  is onto  $C$ .

- This follows from parts (i) and (ii).

**Exercise 7:** Let  $A$  be any set and  $f : A \rightarrow A$  be a one-one function. Then  $f^n : A \rightarrow A$  is a one-one function for all integers  $n \geq 1$ .

**Solution:** If possible, suppose there exists an integer  $n > 1$  such that  $f^n$  is not one-one. Let  $k$  be the smallest such integer. That is,  $f, f^2, \dots, f^{k-1}$  are one-one, but  $f^k$  is not one-one,  $k > 1$ . Because  $f^k$  is not one-one, there exist  $a, b \in A$  such that  $a \neq b$  and  $f^k(a) = f^k(b)$ . Now,

$$\begin{aligned} f^k(a) &= f^k(b) \\ \Rightarrow (f \circ f^{k-1})(a) &= (f \circ f^{k-1})(b) \\ \Rightarrow f(f^{k-1}(a)) &= f(f^{k-1}(b)) \\ \Rightarrow f^{k-1}(a) &= f^{k-1}(b), \quad \text{because } f \text{ is one-one} \\ \Rightarrow a &= b, \quad \text{because } f^{k-1} \text{ is one-one.} \end{aligned}$$

This is a contradiction as  $a \neq b$ . Consequently,  $f^n$  is one-one for all integers  $n \geq 1$ .

**Exercise 8:** Let  $S = \{x \in \mathbb{R} \mid -1 < x < 1\}$ . Show that the function  $f : \mathbb{R} \rightarrow S$  defined by

$$f(x) = \frac{x}{1+|x|}$$

is a one-one and onto function.

**Solution:** Let  $x \in \mathbb{R}$ . Then

$$\begin{aligned} -|x| &\leq x \leq |x|, \\ -1 - |x| &< -|x|, \end{aligned}$$

and

$$|x| \leq 1 + |x|.$$

Hence,  $-1 - |x| < x < 1 + |x|$ . Thus,  $-1 < \frac{x}{1+|x|} < 1$  and so  $-1 < f(x) < 1$ . This shows that  $f(x) \in S$ .

Let  $x, y \in \mathbb{R}$  and  $f(x) = f(y)$ . Then  $\frac{x}{1+|x|} = \frac{y}{1+|y|}$ . Thus,  $\frac{|x|}{1+|x|} = \frac{|y|}{1+|y|}$ . This implies that  $|x| + |x||y| = |y| + |x||y|$  and so  $|x| = |y|$ . Now  $\frac{x}{1+|x|} = \frac{y}{1+|y|}$  implies that  $x \geq 0$  if and only if  $y \geq 0$ . Therefore, because  $|x| = |y|$ ,  $x = y$ . Thus,  $f$  is one-one.

Now let  $z \in \mathbb{R}$  and  $-1 < z < 1$ . We show that there exists  $y \in S$  such that  $f(y) = z$ . For this, first suppose that  $0 \leq z < 1$ . Let  $y \in \mathbb{R}$  be such that  $z = f(y)$ . Then,

$$z = f(y) = \frac{y}{1+|y|}.$$

From this, notice that  $y \geq 0$ . Thus,  $z = \frac{y}{1+y}$ . This implies that  $z(1+y) = y$ . Solve this for  $y$  to get  $y = \frac{z}{1-z}$ . This suggests that to find a preimage  $y$  of  $z$ , we can take  $y$  to be  $\frac{z}{1-z}$ . Let us verify this. Now

$$f\left(\frac{z}{1-z}\right) = \frac{\frac{z}{1-z}}{1 + \left|\frac{z}{1-z}\right|} = \frac{\frac{z}{1-z}}{1 + \frac{z}{1-z}} = z.$$

Now suppose  $-1 < z < 0$ . Here we can show that the preimage of  $z$  is  $\frac{z}{1+z}$ . Indeed,

$$f\left(\frac{z}{1+z}\right) = \frac{\frac{z}{1+z}}{1 + \left|\frac{z}{1+z}\right|} = \frac{\frac{z}{1+z}}{1 + \frac{-z}{1+z}} = z.$$

Hence,  $f$  is onto  $\mathbb{R}$ . Consequently,  $f$  is a one-one and onto function.

## SECTION REVIEW

---

### Key Terms

function	target	onto
well defined	range	surjective
single valued	numeric functions	surjection
image	identity function	one-to-one correspondence
preimage	constant function	bijective
mapped	one-one	bijection
domain	injective	composition
codomain	injection	

### Some Key Definitions

- Let  $A$  and  $B$  be nonempty sets and  $f$  be a relation from  $A$  into  $B$ . Then  $f$  is called a function from  $A$  into  $B$ , if
  - the domain of  $f$  is  $A$ , i.e.,  $\mathcal{D}(f) = A$ , and
  - for all  $(a, b), (a', b') \in f$ ,  $a = a'$  implies  $b = b'$ . In this case, we say that  $f$  is well defined or single valued.
- Let  $A$  and  $B$  be sets and  $f : A \rightarrow B$  be a function. The set  $A$  is referred to as the domain of the function and the set  $B$  is called the codomain, or target, of  $f$ . The set

$$f(A) = \{f(x) \mid x \in A\}$$

is a subset of the codomain  $B$ . The set  $f(A)$  is called the range of the function  $f$ , or the image of the set  $A$  under the function  $f$ , denoted by  $\text{Im}(f)$  or  $\mathcal{I}(f)$ .

- A function  $f : A \rightarrow A$  is said to be the identity function if  $f(x) = x$  for all  $x \in A$ . This function is usually denoted by  $i_A$ .
- A function  $f : A \rightarrow B$  is said to be a constant function if there exists  $b \in B$  such that  $f(x) = b$  for all  $x \in A$ . That is, all elements of  $A$  are mapped to only one element of  $E$ .
- Let  $A$  and  $B$  be sets and  $f : A \rightarrow B$ . Then,

- $f$  is called one-one (or injective or injection) if for all  $a_1, a_2 \in A$ ,

$$a_1 \neq a_2 \Rightarrow f(a_1) \neq f(a_2)$$

(i.e., images of distinct elements of the domain are distinct).

- $f$  is called onto  $B$  (or surjective or surjection) if for every  $b \in B$  there exists at least one  $a \in A$  such that  $f(a) = b$ , i.e.,

$$\text{Im}(f) = B.$$

- $f$  is called one-to-one correspondence (or bijective or bijection) if  $f$  is both one-one and onto.

6. Let  $f : A \rightarrow B$  and  $g : B \rightarrow C$  be functions. The composition of  $f$  and  $g$ , written  $g \circ f$ , is the function from  $A$  to  $C$  defined as

$$(g \circ f)(a) = g(f(a)), \quad \text{for all } a \in A.$$

## Some Key Results

1. Let  $f : A \rightarrow B$ ,  $g : B \rightarrow C$ , and  $h : C \rightarrow D$ . Then  $h \circ (g \circ f) = (h \circ g) \circ f$ ; i.e., composition of functions is associative, provided the composition is defined.
2. Suppose that  $f : A \rightarrow B$  and  $g : B \rightarrow C$ . The following assertions hold.
  - (i) If both  $f$  and  $g$  are one-one, then  $g \circ f$  is also one-one.
  - (ii) If  $f$  is onto  $B$  and  $g$  are onto  $C$ , then  $g \circ f$  is also onto  $C$ .
  - (iii) If both  $f$  and  $g$  are one-to-one correspondences, then  $g \circ f$  is also a one-to-one correspondence.

## EXERCISES

---

1. Determine which of the relations  $f$  are functions from the set  $X$  to the set  $Y$ .
  - a.  $X = \{-3, -2, -1, 0, 1, 2\}$ ,  $Y = \{3, 4, 5, 6, 7\}$ , and  $f = \{(-2, 3), (-1, 6), (0, 4), (1, 5), (2, 7)\}$ .
  - b.  $X = \{-3, -2, -1, 0, 1, 2\}$ ,  $Y = \{3, 4, 5, 6, 7\}$ , and  $f = \{(-3, 3), (-2, 3), (0, 4), (-2, 6), (1, 5), (2, 7)\}$ .
  - c.  $X = \{-3, -2, -1, 0, 1, 2\}$ ,  $Y = \{3, 4, 5, 6, 7\}$ , and  $f = \{(-2, 3), (0, 4), (-3, 6), (-1, 7), (1, 5), (2, 7)\}$ .
  - d.  $X = Y = \{-3, -1, 0, 2\}$ , and  $f = \{(-3, -1), (-1, 2), (0, 2), (2, -1)\}$ .
  - e.  $X = Y = \text{the set of all integers}$ , and  $f = \{(a, b) \in \mathbb{Z} \times \mathbb{Z} \mid b = 2a - 1\}$ .
  - f.  $X = Y = \text{the set of all integers}$ , and  $f = \{(a, b) \in \mathbb{Z} \times \mathbb{Z} \mid a^4 = b\}$ .
  - g.  $X = \mathbb{Q} = Y$ , defined by  $f(\frac{n}{m}) = n + m$  for all  $\frac{n}{m} \in \mathbb{Q}$ .
2. Let  $A = \{-3, -2, -1, 0, 1, 2\}$ . Find the range of the function  $f : A \rightarrow \mathbb{R}$ , defined by  $f(x) = x^2 + 1$  for all  $x \in A$ .
3. Find the range of the function  $f : \mathbb{R} \rightarrow \mathbb{R}$ , defined by  $f(x) = x^2 + x + 1$  for all  $x \in \mathbb{R}$ .
4. Consider the function  $f = \{(x, x^2) \mid x \in S\}$  from the set  $S = \{-3, -2, -1, 0, 1, 2, 3\}$  into  $\mathbb{Z}$ . Is  $f$  one-one? Is  $f$  onto  $\mathbb{Z}$ ?
5. Let  $f$  be the function from the set  $X = \{2, 3, 4, 5, 6, 7, 8\}$  into the set  $Y = \{0, 1, 2, 3, 4, 5, 6\}$ , defined by  $f(x) = 3x \pmod{7}$  for all  $x \in X$ . Write  $f$  as a set of ordered pairs. Is  $f$  one-one or onto  $Y$ ?
6. Let  $f$  be the function from the set  $X = \{1, 2, 3, 4, 5, 6, 7, 8\}$  into the set  $Y = \{0, 1, 2, 3, 4, 5, 6, 7\}$ , defined by  $f(x) = 2x \pmod{8}$  for all  $x \in X$ . Write  $f$  as a set of ordered pairs. Is  $f$  one-one or onto  $Y$ ?
7. Let  $f$  be the function from the set  $X = \{3, 4, 5, 6, 7, 8\}$  into the set  $Y = \{0, 1, 2, 3, 4\}$ , defined by  $f(x) = (2x + 3)(\pmod{5})$  for all  $x \in X$ . Write  $f$  as a set of ordered pairs. Is  $f$  one-one or onto  $Y$ ?
8. Show that the following functions are neither one-one nor onto ( $\mathbb{Z}$  in (a), (b); and  $\mathbb{R}$  in (c), (d), and (e)).
  - a.  $f : \mathbb{Z} \rightarrow \mathbb{Z}$ , defined by  $f(x) = 4x^2 + 3$  for all  $x \in \mathbb{Z}$ .
  - b.  $f : \mathbb{Z} \rightarrow \mathbb{Z}$ , defined by for all  $n \in \mathbb{Z}$
$$f(n) = \begin{cases} 1, & \text{if } n \text{ is even}, \\ -1, & \text{if } n \text{ is odd}. \end{cases}$$
  - c.  $f : \mathbb{R} \rightarrow \mathbb{R}$ , defined by  $f(x) = |x| + 1$  for all  $x \in \mathbb{R}$ .
  - d.  $f : \mathbb{R} \rightarrow \mathbb{R}$ , defined by  $f(x) = \frac{x}{x^2+1}$  for all  $x \in \mathbb{R}$ .
  - e.  $f : \mathbb{R} \rightarrow \mathbb{R}$ , defined by  $f(x) = \cos x$  for all  $x \in \mathbb{R}$ .
9. Show that the following functions are onto  $\mathbb{Z}$ , but not one-one.
  - a.  $f : \mathbb{Z} \rightarrow \mathbb{Z}$ , defined by for all  $n \in \mathbb{Z}$
$$f(n) = \begin{cases} n & \text{if } n \geq 0 \\ n+1 & \text{if } n < 0. \end{cases}$$
  - b.  $f : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$ , defined by  $f(n, m) = n + m$  for all  $n, m \in \mathbb{Z}$ .
10. Show that the following functions are one-one, but not onto  $\mathbb{Z}$ .
  - a.  $f : \mathbb{Z} \rightarrow \mathbb{Z}$ , defined by  $f(n) = 9n + 1$  for all  $n \in \mathbb{Z}$ .
  - b.  $f : \mathbb{Z} \rightarrow \mathbb{Z}$ , defined by  $f(n) = 3^n$  for all  $n \in \mathbb{Z}$ .
11. Show that the following functions are one-to-one correspondences.
  - a.  $f : \mathbb{R} \rightarrow \mathbb{R}$ , defined by  $f(a) = a\sqrt{2}$  for all  $a \in \mathbb{R}$ .
  - b.  $f : \mathbb{R} \rightarrow \mathbb{R}$ , defined by  $f(x) = \frac{2x-1}{3}$  for all  $x \in \mathbb{R}$ .
  - c.  $f : \mathbb{R} \rightarrow \mathbb{R}$ , defined by  $f(x) = x^3 - 1$  for all  $x \in \mathbb{R}$ .
12. Determine which of the following functions are one-one, onto, or both one-one and onto.

- a.  $f : \mathbb{Z} \rightarrow \mathbb{Z}$ , defined by  $f(n) = 4n - 3$  for all  $n \in \mathbb{Z}$ .  
b.  $f : \mathbb{Z} \times \mathbb{N} \rightarrow \mathbb{Z}$ , defined by  $f(n, m) = \frac{n}{m}$  for all  $n \in \mathbb{Z}$  and for all  $m \in \mathbb{N}$ .  
c.  $f : \mathbb{R} \rightarrow \mathbb{R}$ , defined by  $f(x) = x^3 - x$  for all  $x \in \mathbb{R}$ .  
d.  $f : \mathbb{Z} \rightarrow \mathbb{Q}$ , defined by  $f(n) = 2^n$  for all  $n \in \mathbb{Z}$ .  
e.  $f : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ , defined by  $f(x) = \frac{1}{x}$  for all  $x \in \mathbb{R}^+$ .  
f.  $f : \mathbb{R} \rightarrow \mathbb{R}^+$ , defined by  $f(x) = 3^x$  for all  $x \in \mathbb{R}$ .
13. Let  $f : \mathbb{R} \rightarrow \mathbb{R}$ , defined by  $f(x) = x^2 - 6x$  for all  $x \in \mathbb{R}$ . Find  $\text{Im}(f)$ . Is  $f$  onto  $\mathbb{R}$ ? Is  $f$  one-one?
14. Let  $f : X \rightarrow Y$  and  $g : X \rightarrow Y$  be functions. Show that  $f = g$  if and only if  $f(x) = g(x)$  for all  $x \in X$ .
15. Let  $M_2(\mathbb{R})$  denote the set of all matrices over real numbers. Define  $f : M_2(\mathbb{R}) \rightarrow M_2(\mathbb{R})$  by  $f(A) = A^T$  (the transpose of a matrix  $A$ ) for all  $A \in M_2(\mathbb{R})$ . Find  $f(A)$ , when  $A = \begin{bmatrix} 2 & 0 \\ 1 & 0 \end{bmatrix}$ , and  $\begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}$ . Is  $f$  one-one? Is it onto  $M_2(\mathbb{R})$ ?
16. Define  $f : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R} \times \mathbb{R}$  by  $f((x, y)) = (u, v)$ , where
- $$\begin{bmatrix} u & v \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 1 & -1 \end{bmatrix}.$$
- Find  $f((1, 0)), f((0, 2))$ . Is  $f$  one-one? Is it onto  $\mathbb{R} \times \mathbb{R}$ ?
17. Let  $M_2(\mathbb{R})$  denote the set of all matrices over real numbers. Define  $f : M_2(\mathbb{R}) \rightarrow \mathbb{R}$  by  $f(A) = a - d$  for all  $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in M_2(\mathbb{R})$ . Find  $f(A)$ , when  $A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$  and  $\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ . Is  $f$  one-one? Is it onto  $\mathbb{R}$ ?
18. Let  $A = \{1, 2, 3\}$ . List all one-one functions from  $A$  onto  $A$ .
19. Let  $A = \{1, 2, \dots, n\}$ . Show that the number of one-one and onto functions from  $A$  into  $A$  is  $n!$ .
20. Let  $f : A \rightarrow B$  be a function. Define a relation  $R$  on  $A$  by for all  $a, b \in A$ ,  $a R b$  if and only if  $f(a) = f(b)$ . Show that  $R$  is an equivalence relation.
21. Let  $A = \{x \in \mathbb{Z} \mid -3 < x \leq 5\}$ ,  $B = \{x \in \mathbb{Z} \mid 0 < x \leq 8\}$ , and  $C = \{x \in \mathbb{Z} \mid -8 < x \leq 2\}$ . Consider the functions  $f : A \rightarrow B$  defined by  $f(x) = x + 3$  for all  $x \in A$  and  $g : B \rightarrow C$  defined by  $g(x) = -x + 1$  for all  $x \in B$ . Draw the arrow diagrams of the functions  $f : A \rightarrow B$  and  $g : B \rightarrow C$ . Then draw the arrow diagram of  $g \circ f$ .
22. Let  $f : \mathbb{R} \rightarrow \mathbb{R}$  and  $g : \mathbb{R} \rightarrow \mathbb{R}$  be functions defined by  $f(x) = x^2 - 2x + 4$  and  $g(x) = 7x - 2$  for all  $x \in \mathbb{R}$ . Find  $f \circ g$ ,  $g \circ f$ , and  $f \circ g(-2), g \circ f(-2)$ .
23. Let  $f : \mathbb{R}^+ \rightarrow \mathbb{R}^+$  and  $g : \mathbb{R}^+ \rightarrow \mathbb{R}^+$  be functions defined by  $f(x) = \sqrt{x}$  and  $g(x) = 3x + 1$  for all  $x \in \mathbb{R}^+$ , where  $\mathbb{R}^+$  is the set of all positive real numbers. Find  $f \circ g$  and  $g \circ f$ . Is  $f \circ g = g \circ f$ ?
24. Let  $f : \mathbb{Q}^+ \rightarrow \mathbb{R}$  and  $g : \mathbb{R} \rightarrow \mathbb{R}$  be defined by  $f(x) = 1 + \frac{1}{x}$  for all  $x \in \mathbb{Q}^+$  and  $g(x) = x + 1$  for all  $x \in \mathbb{R}$ , where  $\mathbb{Q}^+$  is the set of all positive rational numbers. Find  $g \circ f$ .
25. Let  $f : \mathbb{R} \rightarrow \mathbb{R}$  and  $g : \mathbb{R} \rightarrow \mathbb{R}$  be defined by  $f(x) = 4x - 3$  and  $g(x) = x^2 + 2$  for all  $x \in \mathbb{R}$ . Find  $g \circ f, f \circ g$ , and  $g \circ g$ .
26. Prove Theorem 5.1.28.
27. For the following statement, write the proof if the statement is true, otherwise give a counter example.  
A function  $f : A \rightarrow B$  is one-one if and only if  $g \circ f = h \circ f$  for all functions  $g, h : B \rightarrow A$ .

## 5.2 SPECIAL FUNCTIONS AND CARDINALITY OF A SET

This section continues the discussion of functions. Here we discuss inverse, restriction, and extensions of a function. We also discuss the floor and ceiling functions, which are often encountered in computer science, especially in algorithm analysis. We conclude with a discussion of the cardinality of a set.

### Inverse of a Function

Let  $f : A \rightarrow B$  be a function from a set  $A$  into a set  $B$ . Then  $f \subseteq A \times B$  is a relation from  $A$  into  $B$ . In Chapter 3, we defined the inverse relation  $f^{-1} \subseteq B \times A$ . Now the natural question is: Is  $f^{-1}$  a function from  $B$  into  $A$ ? Before giving the answer, let us consider the following examples of functions.

#### EXAMPLE 5.2.1

- (i) Let  $A = \{1, 2, 3, 4, 5\}$ ,  $B = \{a, b, c, d\}$ , and  $f : A \rightarrow B$  be defined by

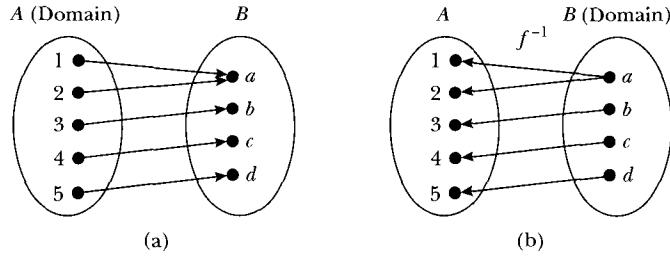
$$f(1) = a, \quad f(2) = a, \quad f(3) = b, \quad f(4) = c, \quad f(5) = d.$$

The arrow diagrams of  $f$  and  $f^{-1}$  are shown in Figure 5.16.

We see that the distinct elements 1 and 2 of  $A$  are both mapped to  $a$ . Therefore, it follows that function  $f$  is not one-one. Because every element of  $B$  has a preimage,  $f$  is onto  $B$ . Hence,  $f$  is onto  $B$  but not one-one.

The inverse relation  $f^{-1} \subseteq B \times A$  is given by

$$f^{-1} := \{(a, 1), (a, 2), (b, 3), (c, 4), (d, 5)\}.$$

FIGURE 5.16 Arrow diagrams of  $f$  and  $f^{-1}$ 

From this, as well from the arrow diagram, we see that each element of  $B$  has an image under  $f^{-1}$ . Therefore, the domain of  $f^{-1}$ ,  $D(f^{-1}) = \{a, b, c, d\} = B$ . Hence,  $f^{-1}$  satisfies the first condition of Definition 5.1.1.

Next, we notice that the element  $a$  of  $B$  has two distinct images, 1 and 2, under  $f^{-1}$ . This implies that  $f^{-1}$  does not satisfy the second condition of Definition 5.1.1. Hence,  $f^{-1}$  is not a function.

- (ii) Let  $A = \{1, 2, 3\}$ ,  $B = \{a, b, c, d\}$ , and  $f : A \rightarrow B$  be defined by

$$f(1) = a, \quad f(2) = b, \quad f(3) = d.$$

For this function  $f$  we see that distinct elements of  $A$  are mapped to distinct elements of  $B$ ; i.e., distinct elements have distinct images. Thus,  $f$  is one-one. Next, we see that no element of  $A$  is mapped to the element  $c$  of  $B$ . Therefore,  $c$  has no preimage, so  $f$  is not onto  $B$ . Hence,  $f$  is one-one but not onto  $B$ .

For this function  $f$ , the relation  $f^{-1} \subseteq B \times A$  is given by

$$f^{-1} = \{(a, 1), (b, 2), (d, 3)\}.$$

From this, we see that  $c \in B$  is not mapped to any element of  $A$ , so  $c$  has no image under  $f^{-1}$ . Therefore, the domain of  $f^{-1}$ ,  $D(f^{-1}) = \{a, b, d\} \neq B$ . Hence,  $f^{-1}$  does not satisfy the first condition of Definition 5.1.1, so we find that  $f^{-1}$  is not a function.

From Example 5.2.1, we may guess that the inverse relation,  $f^{-1}$ , of a function  $f : A \rightarrow B$  is a function if  $f$  is both one-one and onto  $B$ . Let us prove the following theorem.

**Theorem 5.2.2:** Let  $f : A \rightarrow B$  be a function. The inverse relation  $f^{-1} \subseteq B \times A$  is a function from  $B$  into  $A$  if and only if  $f$  is both one-one and onto  $B$ .

**Proof:** Suppose that  $f$  is both one-one and onto  $B$ . We show that  $f^{-1}$  satisfies both conditions of Definition 5.1.1.

Let  $b \in B$ . Because  $f : A \rightarrow B$  is onto  $B$ , there exists  $a \in A$  such that  $f(a) = b$ . Therefore,  $(a, b) \in f$ , which implies that  $(b, a) \in f^{-1}$ . That is,  $f^{-1}(b) = a$ . This shows that for each  $b \in B$ , there exists  $a \in A$  such that  $f^{-1}(b) = a$ . Hence,  $D(f^{-1}) = B$ , so  $f^{-1}$  satisfies the first condition of Definition 5.1.1.

Next suppose that  $(b, a), (b, d) \in f^{-1}$ ; i.e.,  $f^{-1}(b) = a$  and  $f^{-1}(b) = d$  for some  $a, d \in A$ . Then  $(a, b), (d, b) \in f$ . This implies that  $f(a) = b$  and  $f(d) = b$ , so  $f(a) = f(d)$ . Now  $f$  is one-one and therefore  $f(a) = f(d)$  implies  $a = d$ . This shows that

$b$  has only one image under  $f^{-1}$ . Thus,  $f^{-1}$  satisfies the second condition of Definition 5.1.1.

Consequently,  $f^{-1} : B \rightarrow A$  is a function.

Conversely, assume that  $f^{-1} : B \rightarrow A$  is a function. First we show that  $f$  is one-one.

Let  $a_1, a_2 \in A$  and  $f(a_1) = f(a_2)$ . Let us write  $f(a_1) = f(a_2) = b \in B$ . This implies that  $(a_1, b), (a_2, b) \in f$ , so by the definition of the inverse relation,  $(b, a_1), (b, a_2) \in f^{-1}$ . Because  $f^{-1} : B \rightarrow A$  is a function, it follows that  $a_1 = a_2$ . This shows that  $f : A \rightarrow B$  is one-one.

Next we show that  $f$  is onto  $B$ . Let  $b \in B$ . There exists  $a \in A$  such that  $f^{-1}(b) = a$  because  $f^{-1}$  is a function. This implies that  $f(a) = b$ , so  $b$  has a preimage in  $A$ . Because  $b$  is an arbitrary element of  $B$ , we can conclude that every element  $b \in B$  has a preimage in  $A$ . Hence,  $f : A \rightarrow B$  is onto  $B$ . ■

From Chapter 3, we know that the inverse of a relation always exists. However, from Theorem 5.2.2, we find that the inverse,  $f^{-1}$ , of a function  $f$  is a function if and only if  $f$  is a one-to-one correspondence.

**Corollary 5.2.3:** Let  $f : A \rightarrow B$  be a one-one and onto function. Then  $f^{-1} : B \rightarrow A$  is a one-one and onto function. Moreover,  $f^{-1} \circ f = i_A$  and  $f \circ f^{-1} = i_B$ .

**Proof:** In Chapter 3, we proved that for any relation,  $(f^{-1})^{-1} = f$ . Because  $f$  and  $f^{-1}$  are functions, it follows from Theorem 5.2.2 that  $f^{-1}$  is one-one and onto.

To show  $f^{-1} \circ f = i_A$ , let  $a \in A$ . Suppose  $f(a) = b$  for some  $b \in B$ . Then  $f^{-1}(b) = a$ . Thus,

$$(f^{-1} \circ f)(a) = f^{-1}(f(a)) = f^{-1}(b) = a = i_A(a).$$

This is true for all  $a \in A$ . Hence,  $f^{-1} \circ f = i_A$ . Similarly,  $f \circ f^{-1} = i_B$ . ■

**Theorem 5.2.4:** Let  $f : A \rightarrow B$  be a function such that  $f$  is one-one and onto  $B$ .

- (i) If there exists a function  $g : B \rightarrow A$  such that  $g \circ f = i_A$ , then  $g = f^{-1}$ .
- (ii) If there exists a function  $h : B \rightarrow A$  such that  $f \circ h = i_B$ , then  $h = f^{-1}$ .

**Proof:** Because  $f : A \rightarrow B$  is one-one and onto,  $f^{-1} : B \rightarrow A$  is one-one and onto by Corollary 5.2.3. Also by Corollary 5.2.3,  $f^{-1} \circ f = i_A$  and  $f \circ f^{-1} = i_B$ .

- (i) Suppose there exists a function  $g : B \rightarrow A$  such that  $g \circ f = i_A$ . We show that  $g = f^{-1}$ .

Let  $b \in B$ . Because  $f : A \rightarrow B$  is onto, there exists  $a \in A$  such that  $f(a) = b$ . Then

$$\begin{aligned}
 (f^{-1} \circ f)(a) &= i_A(a) = (g \circ f)(a) \\
 \Rightarrow f^{-1}(f(a)) &= g(f(a)) \\
 \Rightarrow f^{-1}(b) &= g(b), \quad \text{because } f(a) = b.
 \end{aligned}$$

- This is true for all  $b \in B$ . Hence,  $g = f^{-1}$ .
- (ii) Suppose there exists a function  $h : B \rightarrow A$  such that  $f \circ h = i_B$ . We show that  $h = f^{-1}$ .

Let  $b \in B$ . Then

$$\begin{aligned}
 (f \circ f^{-1})(b) &= i_B(b) = (f \circ h)(b) \\
 \Rightarrow f(f^{-1}(b)) &= f(h(b)) \\
 \Rightarrow f^{-1}(b) &= h(b), \quad \text{because } f \text{ is one-one.}
 \end{aligned}$$

This is true for all  $b \in B$ . Hence,  $h = f^{-1}$ . ■

### Theorem 5.2.5: The inverse of a function, if it exists, is unique.

**Proof:** The result follows from Corollary 5.2.3 and Theorem 5.2.4. ■  
Theorem 5.2.4 motivates the following definition.

**DEFINITION 5.2.6** ▶ Let  $f : A \rightarrow B$  be a function from the set  $A$  into the set  $B$ .

- (i)  $f$  is called **left invertible** if there exists  $g : B \rightarrow A$  such that  $g \circ f = i_A$ . Moreover, if such a function  $g$  exists, then  $g$  is called a **left inverse** of  $f$ .
- (ii)  $f$  is called **right invertible** if there exists  $h : A \rightarrow B$  such that  $f \circ h = i_B$ . Moreover, if such a function  $h$  exists, then  $h$  is called a **right inverse** of  $f$ .

It can be shown that  $f$  is a one-one function, then  $f$  is left invertible and if  $f$  is onto, then  $f$  is right invertible (see Worked-Out Exercise 3 of this section and Exercise 13 on page 314).

From Theorem 5.2.4, we find that if  $f : A \rightarrow B$  is a one-one and onto function, then to compute  $f^{-1}$ , we find a function  $g : B \rightarrow A$  such that  $g \circ f = i_A$  or we find a function  $h : B \rightarrow A$  such that  $f \circ h = i_B$ .

#### EXAMPLE 5.2.7

Consider  $f : \mathbb{R} \rightarrow \mathbb{R}$  defined by

$$f(x) = 9x + 5$$

for all  $x \in \mathbb{R}$ . This function  $f$  is both one-one and onto. Hence,  $f^{-1}$  exists.

Let  $y = f^{-1}(x)$ . Then  $f(y) = x$ , which implies that  $x = 9y + 5$ . Solve  $x = 9y + 5$  to get  $y = \frac{x-5}{9}$ . That is,  $f^{-1}(x) = \frac{x-5}{9}$ .

Let us determine  $f^{-1}$  using Theorem 5.2.4. For this, we define  $g : \mathbb{R} \rightarrow \mathbb{R}$  by for all  $x \in \mathbb{R}$ ,

$$g(x) = \frac{x-5}{9}.$$

From the definition of  $g$ , it follows that  $g$  is a function. Now for any  $x \in \mathbb{R}$ ,

$$(g \circ f)(x) = g(f(x)) = g(9x + 5) = \frac{(9x + 5) - 5}{9} = \frac{9x}{9} = x = i_{\mathbb{R}}(x).$$

This implies that  $g \circ f = i_{\mathbb{R}}$ . Thus, by Theorem 5.2.4,  $g = f^{-1}$ .

Hence,  $f^{-1} : \mathbb{R} \rightarrow \mathbb{R}$  is given by  $f^{-1}(x) = \frac{x-5}{9}$  for all  $x \in \mathbb{R}$ .

## Restriction, Extensions, Image, and Preimage

Let  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  be a function such that  $f(x) = 2x + 3$  for all  $x \in \mathbb{Z}$ . Define the function  $g : \mathbb{N} \rightarrow \mathbb{Z}$  by

$$g(n) = 2n + 3$$

for all  $n \in \mathbb{N}$ . Now  $\mathbb{N} \subseteq \mathbb{Z}$ , and for all  $n \in \mathbb{N}$  we find that

$$g(n) = 2n + 3 = f(n).$$

That is,  $\mathbb{N} \subseteq \mathbb{Z}$  and  $g(n)$  and  $f(n)$  are the same on each element of  $n \in \mathbb{N}$ . Such a function  $g$  is called the restriction of  $f$  to  $\mathbb{N}$ . More formally, we have the following definition.

---

**DEFINITION 5.2.8** ▶ Let  $f : A \rightarrow B$  and  $\emptyset \neq A' \subseteq A$ . The **restriction** of  $f$  to  $A'$ , written  $f|_{A'}$ , is defined to be  $f|_{A'} = \{(a', f(a')) \mid a' \in A'\}$ .

Observe that  $f|_{A'}$  is actually the same function  $f$  except that its domain is a subset of the domain of  $f$ .

### EXAMPLE 5.2.9

Let  $f : \mathbb{Z} \rightarrow \{1, -1\}$  be defined by

$$f(n) = \begin{cases} 1 & \text{if } n \text{ is even,} \\ -1 & \text{if } n \text{ is odd,} \end{cases}$$

and let  $g : \mathbb{E} \rightarrow \{1, -1\}$  be defined by  $g(n) = 1$ , where  $\mathbb{E}$  is the set of all even integers. Now  $\mathbb{E} \subseteq \mathbb{Z}$ , and for all  $n \in \mathbb{E}$ ,

$$g(n) = 1 = f(n).$$

Thus,  $g$  is the restriction of  $f$  to  $\mathbb{E}$ .

Here  $g$  is actually the same as the function  $f$  except that its domain  $\mathbb{E}$  is a subset of  $\mathbb{Z}$ .

---

**REMARK 5.2.10** ▶ If  $f : A \rightarrow B$  is a function and  $C$  is a nonempty subset of  $A$ , then finding the restriction of  $f$  to  $C$  is straightforward. In fact,  $f|_C(x) = f(x)$  for all  $x \in C$ . That is, the restriction of  $f$  to  $C$ ,  $f|_C$ , is the same function as  $f$  except that the domain of  $f|_C$  is  $C$ , which is a subset of the domain of  $f$  as  $D(f) = A$ .

We leave the proof of the following theorem as an exercise.

**Theorem 5.2.11:** Let  $f : A \rightarrow B$  be a function and  $C$  be a nonempty subset of  $A$ . Then the restriction of  $f$  to  $C$  is unique.

Just as we can consider the restriction of a function from a set to its subset, we can also consider the extension of a function from a subset to a set.

For example, let  $f : \mathbb{Z} \rightarrow \mathbb{R}$  be a function such that  $f(x) = 2x + 3$  for all  $x \in \mathbb{Z}$ . Now  $\mathbb{Z} \subseteq \mathbb{Q}$ . Define  $h : \mathbb{Q} \rightarrow \mathbb{R}$  by

$$h(x) = \begin{cases} 2x + 3 & \text{if } x \in \mathbb{Z}, \\ 0 & \text{if } x \in \mathbb{Q} - \mathbb{Z}. \end{cases}$$

It follows from the definition of  $h$  that  $h$  is a function and  $h|_{\mathbb{Z}} = f$ . That is, the restriction of  $h$  to  $\mathbb{Z}$  is  $f$ .

Such a function  $h$  is called an extension of  $f$  to  $\mathbb{Q}$ . More formally, we have the following definition.

**DEFINITION 5.2.12** ▶ Let  $f : A \rightarrow B$  and  $A \subseteq A'$ . A function  $g : A' \rightarrow B$  is called an **extension** of  $f$  to  $A'$  if  $g|_A = f$ .

Note that a function may have more than one extension. For example, for the preceding function  $f$ , we can define the function  $k : \mathbb{Q} \rightarrow \mathbb{R}$  by  $k(x) = 2x + 3$  if  $x \in \mathbb{Z}$ , and  $k(x) = 1$  if  $x \in \mathbb{Q} - \mathbb{Z}$ . Then  $k|_{\mathbb{Z}} = f$ , so  $k$  is an extension of  $f$  to  $\mathbb{Q}$ .

### EXAMPLE 5.2.13

Suppose  $A = \{a \in \mathbb{R} \mid a > 0\}$ . Let  $f : A \rightarrow \mathbb{R}$  be given by

$$f(a) = \frac{|a|}{a}$$

for all  $a \in A$ . Observe that  $g : \mathbb{R} \rightarrow \mathbb{R}$  defined by  $g(a) = 1$  for all  $a \in \mathbb{R}$  is an extension of  $f$  to  $\mathbb{R}$ , because  $g|_A = f$ .

### EXAMPLE 5.2.14

Let  $f : \mathbb{R} \rightarrow \mathbb{R}_1$ , where  $\mathbb{R}_1 = \{x \in \mathbb{R} \mid -1 \leq x \leq 1\}$ , be given by  $f(x) = \sin x$  for all  $x \in \mathbb{R}$ . Suppose

$$A = \left\{ x \in \mathbb{R} \mid -\frac{\pi}{2} \leq x \leq \frac{\pi}{2} \right\}.$$

Then the function  $g : A \rightarrow \mathbb{R}_1$  given by  $g(x) = \sin x$  for all  $x \in A$  is a restriction of  $f$  to  $A$ .

Suppose  $f : A \rightarrow B$  is a function. In the preceding sections, we only considered the images and preimages of one element at a time. Now if  $P$  is a subset of  $A$ , then  $P \subseteq \mathcal{D}(f)$ , so we can form the set consisting of all the images of the elements of  $P$ . Similarly, if  $Q$  is a nonempty subset of  $B$ , then we can form the set consisting of all the preimages of the elements of  $Q$ . We begin with the following definitions.

Let  $f : A \rightarrow B$  be a function from a set  $A$  into a set  $B$ . Let  $P \subseteq A$ . The set

$$f(P) = \{f(a) \mid a \in P\} \subseteq B$$

is called the **image** (or **direct image**) of  $P$  under  $f$ .

Let  $X$  and  $Y$  be two nonempty subsets of  $A$ . It is interesting to see that, in general,

$$f(X \cap Y) \neq f(X) \cap f(Y).$$

Indeed, let  $f : \mathbb{R} \rightarrow \mathbb{R}$  be given by  $f(x) = |x|$  for all  $x \in \mathbb{R}$ . Let  $X$  be the set of all nonnegative integers and  $Y$  be the set of all nonpositive integers. Then  $X, Y \subseteq \mathbb{R}$  and  $X \cap Y = \{0\}$ . Further,

$$f(X \cap Y) = f(\{0\}) = \{0\},$$

whereas

$$f(X) = \{x \in \mathbb{R} \mid x \geq 0\} \quad \text{and} \quad f(Y) = \{y \in \mathbb{R} \mid y \geq 0\},$$

and hence

$$f(X) \cap f(Y) = f(X) \neq \{0\} = f(X \cap Y).$$

However, such an equality holds good for all subsets  $X, Y$  of the domain of the function  $f$  if and only if  $f$  is one-one.

**Theorem 5.2.15:** Let  $X$  and  $Y$  be nonempty sets and  $f : X \rightarrow Y$  be a function. Then  $f$  is one-one if and only if

$$f(A \cap B) = f(A) \cap f(B)$$

for all nonempty subsets  $A$  and  $B$  of  $X$ .

**Proof:** Suppose that  $f$  is one-one. Let  $A$  and  $B$  be nonempty subsets of  $X$ . To show that  $f(A \cap B) = f(A) \cap f(B)$ , as usual, we show that  $f(A \cap B) \subseteq f(A) \cap f(B)$  and  $f(A) \cap f(B) \subseteq f(A \cap B)$ . The result then will follow from the equality of sets.

Let  $y \in f(A \cap B)$ . Then

$$\begin{aligned} y &\in f(A \cap B) \\ \Rightarrow y &= f(x) \quad \text{for some } x \in A \cap B \\ \Rightarrow y &= f(x) \quad \text{for some } x \in A \text{ and } x \in B \\ \Rightarrow y &= f(x) \quad \text{for some } f(x) \in f(A) \text{ and } f(x) \in f(B) \\ \Rightarrow y &= f(x) \quad \text{for some } f(x) \in f(A) \cap f(B). \end{aligned}$$

From this it follows that  $f(A \cap B) \subseteq f(A) \cap f(B)$ .

On the other hand, let  $y \in f(A) \cap f(B)$ . Then  $y \in f(A)$  and  $y \in f(B)$ . Thus,  $y = f(a)$  for some  $a \in A$  and  $y = f(b)$  for some  $b \in B$ . Because  $f$  is one-one and  $f(a) = f(b)$ , we find that  $a = b$ . Thus,  $y \in f(A \cap B)$ . Hence,  $f(A) \cap f(B) \subseteq f(A \cap B)$ . Consequently,  $f(A \cap B) = f(A) \cap f(B)$ .

Conversely, suppose that  $f(A \cap B) = f(A) \cap f(B)$  for all subsets  $A$  and  $B$  of  $X$ . We show that  $f$  is one-one by the method of proof by contradiction.

Suppose  $f$  is not one-one. Then there exist  $x, y \in X$  such that  $x \neq y$  and  $f(x) = f(y)$ . Let  $A = \{x\}$  and  $B = \{y\}$ . Then  $f(A) = \{f(x)\}$  and  $f(B) = \{f(y)\} = \{f(x)\}$ . Now  $A \cap B = \emptyset$ , so  $f(A \cap B) = \emptyset$ . However,  $f(A) \cap f(B) = \{f(x)\} \neq \emptyset$ . Thus,  $f(A \cap B) \neq f(A) \cap f(B)$ , a contradiction. Hence,  $f$  is one-one. ■

Let  $f : A \rightarrow B$  be a function. Let  $Q$  be a nonempty subset of  $B$ . Then the set

$$f^{-1}(Q) = \{a \in A \mid f(a) \in Q\}$$

is called the **inverse image** of  $Q$  under  $f$  and  $f^{-1}(Q) \subseteq A$ .

---

**REMARK 5.2.16** ▶ Let  $f : A \rightarrow B$  be a function. Note that  $f^{-1} : B \rightarrow A$  is a relation; it need not be a function. Moreover, if  $Q \subseteq B$ , then  $f^{-1}(Q)$  is to be understood as the set of preimages of the elements of  $Q$  under  $f$ .

## The Floor and Ceiling Functions

The functions we describe in this section, floor and ceiling, are very useful in computer science, especially in algorithm analysis. Most of the programming languages provide them as functions, typically as part of the math library, so programmers can use them without having to write their own code. Our interest here is to describe their basic properties.

---

**DEFINITION 5.2.17** ▶ For any real number  $x$ , the **floor** of  $x$ , written  $\lfloor x \rfloor$ , is the greatest integer less than or equal to  $x$ .

From the definition of  $\lfloor x \rfloor$  it follows that  $\lfloor x \rfloor$  denotes the greatest integer that does not exceed  $x$ .

**EXAMPLE 5.2.18**

Let  $x = 4.15$ . Here 4 is the greatest integer that does not exceed 4.15. Hence,  $\lfloor 4.15 \rfloor = 4$ .

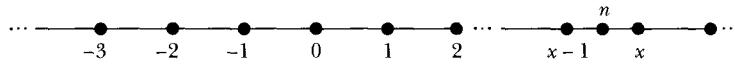
Let  $x = \frac{3}{4} = 0.75$ . Here 0 is the greatest integer that does not exceed  $\frac{3}{4}$ . Hence,  $\lfloor \frac{3}{4} \rfloor = 0$ .

Consider  $x = 48.0$ . The largest integer that does not exceed 48.0 is 48. Thus,  $\lfloor 48.0 \rfloor = 48$ .

Now consider  $x = -3.8$ . Observe that the largest integer that does not exceed  $x$  is  $-4$ . Thus,  $\lfloor -3.8 \rfloor = -4$ .

Similarly,  $\lfloor 56 \rfloor = 56$ ,  $\lfloor -6.78 \rfloor = -7$ .

Let  $x$  be a real number. If  $x$  is an integer, then  $\lfloor x \rfloor = x$ . Suppose  $x$  is not an integer. Then  $x - 1$  is also not an integer. See Figure 5.17.



**FIGURE 5.17** Number line

Thus, there exists an integer  $n$  such that  $x - 1 < n < x$ .

Suppose there exist two integers  $n, m$  such that

$$x - 1 < n < x \quad (5.1)$$

and

$$x - 1 < m < x.$$

Multiply the second inequality by  $-1$ , to get

$$-(x - 1) > -m > -x$$

or

$$-x < -m < -x + 1. \quad (5.2)$$

Add the corresponding sides of (5.1) and (5.2) to get

$$x - 1 - x < n - m < x - x + 1.$$

This implies  $-1 < n - m < 1$ . The only integer between  $-1$  and  $1$  is  $0$ , so  $n - m = 0$ , i.e.,  $n = m$ .

Thus, we find that there exists only one integer  $n$  such that  $x - 1 < n < x$ . From this it follows that  $\lfloor x \rfloor$  is the unique integer satisfying

$$x - 1 < \lfloor x \rfloor \leq x. \quad (5.3)$$

To compute  $\lfloor x \rfloor$ , we can use (5.3) as follows. Let  $x = 4.15$ . Then

$$x - 1 = 4.15 - 1 = 3.15.$$

Hence, the unique integer  $\lfloor 4.15 \rfloor$  satisfying

$$3.15 < \lfloor 4.15 \rfloor \leq 4.15$$

is 4, (the only integer between 3.15 and 4.15 is 4).

For another example, consider  $\frac{7}{9}$ . Now

$$\begin{aligned}\frac{7}{9} - 1 &< \lfloor \frac{7}{9} \rfloor \leq \frac{7}{9} \\ \Rightarrow -\frac{2}{9} &< \lfloor \frac{7}{9} \rfloor \leq \frac{7}{9} \\ \Rightarrow \lfloor \frac{7}{9} \rfloor &= 0.\end{aligned}$$

(The only integer between  $-\frac{2}{9}$  and  $\frac{7}{9}$  is 0.

**DEFINITION 5.2.19** ► For any real number  $x$ , the **ceiling** of  $x$ , written  $\lceil x \rceil$ , is the least integer greater than or equal to  $x$ .

As in the case of  $\lfloor x \rfloor$ , we can show that  $\lceil x \rceil$  is the unique integer satisfying

$$x \leq \lceil x \rceil < x + 1. \quad (5.4)$$

Let  $x = -4.15$ . Then the unique integer  $\lceil -4.15 \rceil$  satisfying

$$-4.15 \leq \lceil -4.15 \rceil < -4.15 + 1$$

i.e.,

$$-4.15 \leq \lceil -4.15 \rceil < -3.15$$

is  $-4$ . Hence,  $\lceil -4.15 \rceil = -4$ .

Similarly,  $\lceil 4.15 \rceil = 5$ ,  $\lceil -3.001 \rceil = -3$ ,  $\lceil 4 \rceil = 4$ ,  $\lceil -15 \rceil = -15$ .

**DEFINITION 5.2.20** ► (i) The function  $f : \mathbb{R} \rightarrow \mathbb{R}$  defined by  $f(x) = \lfloor x \rfloor$ , for all real numbers  $x$ , is called a **floor function**.  
(ii) The function  $g : \mathbb{R} \rightarrow \mathbb{R}$  defined by  $g(x) = \lceil x \rceil$ , for all real numbers  $x$ , is called a **ceiling function**.

Let us consider the floor function,  $f(x) = \lfloor x \rfloor$ . For each real number  $x$ ,  $\lfloor x \rfloor$  is an integer, so  $\text{Im}(f) \subseteq \mathbb{Z}$ . On the other hand, for any integer  $n$ ,  $n = \lfloor n \rfloor = f(n)$ . That is, under the floor function, every integer is its own preimage. It now follows that  $\text{Im}(f) = \mathbb{Z} \neq \mathbb{R}$ . In other words, the floor function is not onto  $\mathbb{R}$ .

Notice that  $f(4.8) = \lfloor 4.8 \rfloor = 4 = \lfloor 4.9 \rfloor = f(4.9)$ . However,  $4.8 \neq 4.9$ . This shows that the floor function is not one-one.

Similarly, if  $g$  denotes the ceiling function, then  $\text{Im}(g) = \mathbb{Z}$ , so  $g$  is not onto  $\mathbb{R}$ . Also  $g$  is not one-one. We record these results in the form of the following theorem.

### Theorem 5.2.21:

- (i) The floor function is neither one-one nor onto  $\mathbb{R}$ .
- (ii) The ceiling function is neither one-one nor onto  $\mathbb{R}$ .

The following theorem gives an important and interesting property of a floor function.

**Theorem 5.2.22:** Let  $x \in \mathbb{R}$  and  $n \in \mathbb{N}$ . Then  $\lfloor x + n \rfloor = \lfloor x \rfloor + n$ .

**Proof:** From (5.3), we have

$$x - 1 < \lfloor x \rfloor \leq x$$

and this implies that

$$x - 1 + n < \lfloor x \rfloor + n \leq x + n, \quad n \in \mathbb{N},$$

i.e.,

$$(x + n) - 1 < \lfloor x \rfloor + n \leq x + n. \quad (5.5)$$

Because  $\lfloor x \rfloor$  is the unique integer satisfying the condition  $x - 1 < \lfloor x \rfloor \leq x$  and  $\lfloor x \rfloor + n$  is an integer, it follows from (5.5) that  $\lfloor x + n \rfloor = \lfloor x \rfloor + n$ . ■

Using Theorem 5.2.22, we can compute that

$$\left\lfloor \frac{70}{9} \right\rfloor = \left\lfloor 7 + \frac{7}{9} \right\rfloor = 7 + \left\lfloor \frac{7}{9} \right\rfloor = 7 + 0 = 7.$$

## Cardinality of a Set

Let  $A = \{1, 2, 3, 4, 5, 6\}$ ,  $B = \{a, b, c, d, e, u\}$ , and  $C = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 23\}$  be sets. Here the sets  $A$  and  $B$  are not equal, however, we can say that they have the same number of elements and the number of elements is 6. The sets  $A$  and  $C$  are not equal,  $A$  is a proper subset of  $C$ , and we also find that the number of elements of  $A$  is less than the number of elements of  $C$ .

Now if we consider the set  $\mathbb{Z}$  of all integers and the set  $\mathbb{E}$  of all even integers, then we find that  $\mathbb{E}$  is a proper subset of  $\mathbb{Z}$ . Can we say that  $\mathbb{Z}$  contains more elements than  $\mathbb{E}$ ?

To answer these questions, we use the concepts of one-one and onto functions.

For the sets  $A = \{1, 2, 3, 4, 5, 6\}$  and  $B = \{a, b, c, d, e, f\}$  we find that there exists a function  $f$  from  $A$  into  $B$  such that  $f$  is one-one and onto  $B$ . For example, if

$$\begin{aligned} f : 1 &\mapsto a \\ 2 &\mapsto b \\ 3 &\mapsto c \\ 4 &\mapsto d \\ 5 &\mapsto e \\ 6 &\mapsto u, \end{aligned}$$

then  $f$  is one-one and onto  $B$ .

This gives us a way to count the number of elements of the set  $B$ , which is 6. We say that the cardinality of the set  $B$  is 6. Also in view of this one-to-one correspondence, we say that the sets  $A$  and  $B$  have the same cardinality.

Now for the sets  $A$  and  $C$ , we cannot define a one-to-one correspondence from  $A$  into  $C$ . Hence, we say that  $A$  and  $C$  are not of the same cardinality.

We can think of cardinality as a measure that compares the size of sets. Roughly speaking, the cardinality of a set denotes the number of elements of a set.

**DEFINITION 5.2.23** ▶ Two sets  $A$  and  $B$  have the same **cardinality** if there exists a one-to-one correspondence from  $A$  into  $B$ .

If two sets  $A$  and  $B$  have the same cardinality, we write  $|A| = |B|$ .

**DEFINITION 5.2.24** ▶ Two sets  $A$  and  $B$  are said to be **equivalent** or **(equipotent)**, written  $A \sim B$ , if there exists a one-one and onto function from  $A$  into  $B$ .

From the definition we find that for any two sets  $A$  and  $B$ ,  $A \sim B$  if and only if  $|A| = |B|$ .

Let  $I_n$  be the set

$$I_n = \{1, 2, 3, 4, 5, \dots, n\},$$

$n \in \mathbb{N}$ . If  $B$  is a set such that there exists a one-to-one correspondence  $f : I_n \rightarrow B$ , then  $I_n$  and  $B$  have the same cardinality, and we say that  $B$  is a finite set with  $n$  elements.

Consider the sets  $A$  and  $B$  with  $n$  elements. Then there exists a one-to-one correspondence  $f : I_n \rightarrow A$ , and there exists a one-to-one correspondence  $g : I_n \rightarrow B$ . Because  $f : I_n \rightarrow A$  is one-one and onto  $A$ , the function  $f^{-1} : A \rightarrow I_n$  is also a one-one and onto function. Hence,  $g \circ f^{-1} : A \rightarrow B$  is a one-to-one correspondence. So it follows that  $|A| = |B|$ . Thus, we find that two finite sets with the same number of elements are equivalent.

The following example answers the question raised at the beginning of this section: Can we say that  $\mathbb{Z}$  contains more elements than  $\mathbb{E}$ ?

### EXAMPLE 5.2.25

We show that the sets  $\mathbb{Z}$  and  $\mathbb{E}$  have the same cardinality. For this purpose, define the function  $f : \mathbb{Z} \rightarrow \mathbb{E}$  by  $f(n) = 2n$  for all  $n \in \mathbb{Z}$ . Next, we show that  $f$  is one-one and onto  $\mathbb{E}$ .

Let  $n, m \in \mathbb{Z}$ . Then

$$\begin{aligned} f(n) &= f(m) \\ \Rightarrow 2n &= 2m \\ \Rightarrow n &= m \\ \Rightarrow f &\text{ is one-one.} \end{aligned}$$

To show  $f$  is onto, let  $m \in \mathbb{E}$ . Because  $m$  is an even integer, there exists an integer  $n \in \mathbb{Z}$  such that  $m = 2n$ . Thus, we have  $m = 2n = f(n)$ ; i.e.,  $n$  is the preimage of  $m$ . This shows that  $f$  is onto  $\mathbb{E}$ . Consequently,  $f$  is a one-to-one correspondence.

Therefore,  $|\mathbb{Z}| = |\mathbb{E}|$ . Even though  $\mathbb{E}$  is a proper subset of  $\mathbb{Z}$ , we find that they have the same cardinality or, roughly speaking, the same number of elements.

Let us consider another example regarding the cardinality of infinite sets.

### EXAMPLE 5.2.26

In this example, we show that the sets  $\mathbb{N}$  and  $\mathbb{Z}$  have the same cardinality.

Define the function  $f : \mathbb{N} \rightarrow \mathbb{Z}$  by

$$f(n) = \begin{cases} \frac{n}{2} & \text{if } n \text{ is even,} \\ -\frac{n-1}{2} & \text{if } n \text{ is odd.} \end{cases}$$

Under this function we find that

$$1 \mapsto 0, 2 \mapsto 1, 3 \mapsto -1, 4 \mapsto 2, 5 \mapsto -2, 6 \mapsto 3, 7 \mapsto -3, \text{ and so on.}$$

To show that  $f$  is both one-one and onto  $\mathbb{Z}$ , first we show that if  $n, m \in \mathbb{N}$  such that  $f(n) = f(m)$ , then  $n$  and  $m$  are either both even or both odd.

Suppose this is not the case. Then there exist  $n, m \in \mathbb{N}$  such that  $f(n) = f(m)$  and one of  $n, m$  is odd and the other is even. To be specific, suppose  $n$  is even and

$m$  is odd. Now

$$\begin{aligned} f(n) &= f(m) \\ \Rightarrow \frac{n}{2} &= -\frac{m-1}{2} \\ \Rightarrow n+m &= 1 \\ \Rightarrow &\text{a contradiction, because the sum of two positive integers } > 1. \end{aligned}$$

We leave it as an exercise to show that  $f$  is both one-one and onto  $\mathbb{Z}$  and hence  $\mathbb{N}$  and  $\mathbb{Z}$  have the same cardinality.

**DEFINITION 5.2.27** ► A set  $A$  is said to be **countable** if either  $|A| = |I_n|$  for some positive integer  $n$  or  $|A| = |\mathbb{N}|$ .

The empty set is also considered a countable set, and we write  $|\emptyset| = 0$ .

**EXAMPLE 5.2.28**

Because  $f : \mathbb{N} \rightarrow \mathbb{Z}$  given by

$$f(n) = \begin{cases} \frac{n}{2} & \text{if } n \text{ is even,} \\ -\frac{n-1}{2} & \text{if } n \text{ is odd} \end{cases}$$

is a one-to-one correspondence, it follows that  $\mathbb{Z}$  is a countable set.

The diagram in Figure 5.18 gives a graphic representation of  $f$  and shows the countability of  $\mathbb{Z}$ .

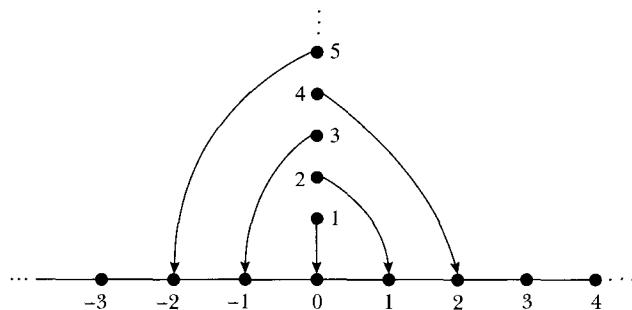


FIGURE 5.18 One-to-one correspondence from  $\mathbb{N}$  into  $\mathbb{Z}$

Again we have seen that the function  $g : \mathbb{Z} \rightarrow \mathbb{E}$  defined by  $g(n) = 2n$ , for all  $n \in \mathbb{Z}$ , is a one-to-one correspondence and thus  $g \circ f : \mathbb{N} \rightarrow \mathbb{E}$  is a one-to-one correspondence. Hence,  $\mathbb{E}$  is a countable set.

**EXAMPLE 5.2.29**

The set  $S = \{n \in \mathbb{N} \mid n \geq 3\}$  is countable. Define the function  $f : \mathbb{N} \rightarrow S$  by  $f(n) = n+2$  for all  $n \in \mathbb{N}$ . Then  $f$  is one-one and onto  $S$  (see Figure 5.19). Hence,  $S$  is countable.

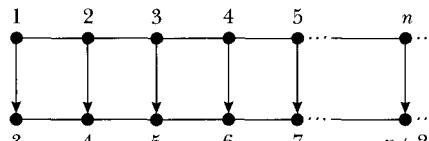


FIGURE 5.19 One-to-one correspondence from  $\mathbb{N}$  into  $S$

Next we state some properties of countable sets without proof.

**Theorem 5.2.30:**

- (i) The union of two countable sets is countable.
- (ii) The Cartesian product of two countable sets is countable.

**DEFINITION 5.2.31** ► A set  $S$  is **uncountable** if there is no one-to-one correspondence between  $\mathbb{N}$  and  $S$ .

Do any uncountable sets exist? The following example shows one.

**EXAMPLE 5.2.32**

Let  $\mathcal{P}(\mathbb{N})$  be the set of all subsets of  $\mathbb{N}$ . Because  $\mathbb{N}$  is an infinite set,  $\mathcal{P}(\mathbb{N})$  is definitely an infinite set. We show that there is no one-to-one correspondence between  $\mathbb{N}$  and  $\mathcal{P}(\mathbb{N})$ .

Suppose there exists a one-to-one correspondence  $f : \mathbb{N} \rightarrow \mathcal{P}(\mathbb{N})$ . We denote  $f(n) = A_n$  for all  $n \in \mathbb{N}$ .

Because  $f : \mathbb{N} \rightarrow \mathcal{P}(\mathbb{N})$  is a one-to-one correspondence, we can list the elements of  $\mathcal{P}(\mathbb{N})$  as

$$A_1, A_2, A_3, A_4, A_5, \dots, A_n, \dots,$$

i.e., we can write

$$\mathcal{P}(\mathbb{N}) = \{A_1, A_2, A_3, A_4, A_5, \dots, A_n, \dots\}$$

Now we define the set  $A$ :

$$A = \{i \in \mathbb{N} \mid i \notin A_i\}.$$

Then  $A$  is a subset of  $\mathbb{N}$ , so  $A \in \mathcal{P}(\mathbb{N})$ . This implies that  $A = A_t$  for some  $t \in \mathbb{N}$ . Thus,  $A_t = A = \{i \in \mathbb{N} \mid i \notin A_i\}$ . Because  $t \notin A_t$ ,  $t \in A$ . But  $A = A_t$ . Thus  $t \in A_t$ , a contradiction. Hence,  $\mathcal{P}(\mathbb{N})$  is uncountable.

In Worked-Out Exercise 9 (p. 144), we proved that if  $A$  is a set with  $n$  elements, then the number of elements in  $\mathcal{P}(A)$ , the set of all subsets of  $A$ , is  $2^n$  and hence a finite set. But Example 5.2.32 shows that if  $A$  is a countable set, then  $\mathcal{P}(A)$  may not be a countable set.

**REMARK 5.2.33** ► At this point, we like to mention that in discrete structures we are mainly interested in finite and countable sets. We gave an example of an uncountable set only to demonstrate that these sets do exist.

We conclude the section by commenting that not all infinite sets have the same cardinal number (follows from the above example that  $|\mathbb{N}| \neq |\mathcal{P}(\mathbb{N})|$ ). In addition, we take the opportunity to state (without proof) a classical theorem of this area, due to Schröder-Bernstein:

Let  $A$  and  $B$  be two sets. If  $A \sim Y$  for some subset  $Y$  of  $B$  and  $B \sim X$  for some subset  $X$  of  $A$ , then  $A \sim B$ .

 WORKED-OUT EXERCISES

**Exercise 1:** Let  $f : \mathbb{Q} \rightarrow \mathbb{Q}$  be the function defined by  $f(x) = 3x + 4$  for all  $x \in \mathbb{Q}$ . Find the inverse of  $f$  if it exists.

**Solution:** Let  $x, y \in \mathbb{Q}$ . If  $f(x) = f(y)$ , then  $3x + 4 = 3y + 4$ , which implies that  $x = y$  so if  $x \neq y$ , then  $f(x) \neq f(y)$ . Therefore,  $f$  is one-one. Let  $u \in \mathbb{Q}$  such that  $f(u) = x$ . Then  $3u + 4 = x$ , which implies that  $u = \frac{x-4}{3}$ . So there exists  $u = \frac{x-4}{3} \in \mathbb{Q}$  such that  $f(u) = f\left(\frac{x-4}{3}\right) = 3\left(\frac{x-4}{3}\right) + 4 = x$ . Hence,  $f$  is onto  $\mathbb{Q}$ . Because  $f$  is both one-one and onto,  $f^{-1}$  exists. If  $f(x) = y$ , then  $f^{-1}(y) = x$ . Also  $y = f(x) = 3x + 4$  implies that  $x = \frac{y-4}{3}$ . Hence,  $f^{-1}(y) = \frac{y-4}{3}$  for all  $y \in \mathbb{R}$ .

**Exercise 2:** Let  $f$  be the function from the set  $N$  into the set  $X = \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$  defined by  $f(x) = x \pmod{7}$  for all  $x \in N$ . Find  $\text{Im}(f)$ . Is  $f$  onto? Is  $f$  one-one? Does  $f^{-1}$  exist?

**Solution:** We know that for any positive integer  $n$ ,  $n \pmod{7}$  is the remainder when  $n$  is divided by 7. Now by the division algorithm,  $n = 7t + r$ , where  $0 \leq r < 7$ . Then  $n \pmod{7} = r$ . Hence,  $\text{Im}(f) = \{0, 1, 2, 3, 4, 5, 6\}$  so  $f$  is not onto  $X$ . Also  $f(0) = 0 \pmod{7} = 7 \pmod{7} = f(7)$  and  $0 \neq 7$ . Therefore,  $f$  is not one-one. Because  $f$  is not one-one and onto,  $f^{-1}$  does not exist.

**Exercise 3:** Let  $f : A \rightarrow B$ . Prove that  $f$  is left invertible if and only if  $f$  is one-one.

**Solution:** Suppose  $f$  is left invertible. Then there exists  $g : B \rightarrow A$  such that

$$g \circ f = i_A.$$

Let  $a, b \in A$  and  $f(a) = f(b)$ . Now

$$\begin{aligned} f(a) &= f(b) \\ \Rightarrow g(f(a)) &= g(f(b)) \\ \Rightarrow (g \circ f)(a) &= (g \circ f)(b) \\ \Rightarrow i_A(a) &= i_A(b) \\ \Rightarrow a &= b. \end{aligned}$$

Thus,  $f$  is one-one.

Conversely, suppose  $f$  is one-one. We construct a function  $g : B \rightarrow A$ , which is a left inverse of  $f$ .

Because  $f : A \rightarrow B$  is one-one, observe that, for any element  $b \in B$ , either there is no preimage in  $A$  or there exists a unique element, say  $a_b \in A$  such that  $f(a_b) = b$ . Let us arbitrarily choose and fix an element  $a_0 \in A$ . We now define  $g : B \rightarrow A$  by

$$g(b) = \begin{cases} a_0 & \text{if } b \text{ has no preimage under } f, \\ a_b & \text{if } a_b \text{ is the unique preimage of } b \text{ under } f, \text{ i.e., } f(a_b) = b \end{cases}$$

for all  $b \in B$ . From the definition of  $g$  it follows that  $g$  is a function from  $B$  into  $A$ . We now show that  $g \circ f = i_A$ . Let  $a \in A$  and  $f(a) = b$  for some  $b \in B$ . Then by the definition of  $g$ ,  $g(b) = a$ . So,

$$(g \circ f)(a) = g(f(a)) = g(b) = a = i_A(a).$$

Because  $a$  is any element of  $A$ , we conclude that  $g \circ f = i_A$ . Hence,  $g$  is a left inverse of  $f$ .

**Exercise 4:** Let  $f : \mathbb{N} \rightarrow \mathbb{N}$  be defined by  $f(n) = n + 3$  for all  $n \in \mathbb{N}$ .

- (a) Show that  $f$  is one-one, but not onto  $\mathbb{N}$ .
- (b) Verify that  $g : \mathbb{N} \rightarrow \mathbb{N}$  defined by: For all  $n \in \mathbb{N}$

$$g(n) = \begin{cases} 3 & \text{if } n \leq 3, \\ n - 3 & \text{if } n > 3 \end{cases}$$

is a left inverse of  $f$ .

- (c) Show that  $f \circ g \neq i_{\mathbb{N}}$ .

**Solution:**

- (a) To show  $f$  is one-one, let  $m, n \in \mathbb{N}$  and  $f(m) = f(n)$ . This implies that  $m + 3 = n + 3$  and so  $m = n$ . Therefore,  $f$  is one-one.

To show  $f$  is not onto  $\mathbb{N}$ , let  $1 \in \mathbb{N}$ . Suppose there exists  $n \in \mathbb{N}$  such that  $f(n) = 1$ . Then

$$\begin{aligned} f(n) &= 1 \\ \Rightarrow n + 3 &= 1 \\ \Rightarrow n &= -2 \notin \mathbb{N}, \end{aligned}$$

which is a contradiction. Hence,  $f$  is not onto  $\mathbb{N}$ .

- (b) We now see that

$$\begin{aligned} (g \circ f)(n) &= g(f(n)) \\ &= g(n + 3) = n + 3 - 3 = n = i_{\mathbb{N}}(n) \end{aligned}$$

for all  $n \in \mathbb{N}$ . Hence,  $g \circ f = i_{\mathbb{N}}$ , so  $g$  is a left inverse of  $f$ .

- (c) Now

$$(f \circ g)(3) = f(g(3)) = f(3) = 6 \neq 3 = i_{\mathbb{N}}(3).$$

This implies that  $f \circ g \neq i_{\mathbb{N}}$ .

**Exercise 5:**

- (a) Let  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  be a function defined by

$$f(x) = \begin{cases} x & \text{if } x \text{ is even,} \\ 4x + 1 & \text{if } x \text{ is odd} \end{cases}$$

for all  $x \in \mathbb{Z}$ . Find a left inverse of  $f$  if one exists.

- (b) Let  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  be the function defined by  $f(x) = |x| + x$  for all  $x \in \mathbb{Z}$ . Find a right inverse of  $f$  if one exists.

**Solution:**

- (a) By Worked-Out Exercise 2,  $f$  has a left inverse if and only if  $f$  is one-one. Before we attempt to find a left inverse of  $f$ , let us first check whether  $f$  is one-one.

Let  $x, y \in \mathbb{R}$  and  $f(x) = f(y)$ . By the definition of  $f$ ,  $f(x)$  is  $x$  if  $x$  is even and  $f(x)$  is  $4x + 1$  if  $x$  is odd. Thus, because  $f(x) = f(y)$ , both  $x$  and  $y$  are either even or odd. If  $x$  and  $y$  are both even, then  $f(x) = x$  and  $f(y) = y$  and so  $x = y$ . Suppose  $x$  and  $y$  are odd. Then  $f(x) = 4x + 1$  and  $f(y) = 4y + 1$ . Then  $4x + 1 = 4y + 1$ , so  $x = y$ . Hence,  $f$  is one-one and  $f$  has a left inverse. Thus, there exists a function  $g : \mathbb{Z} \rightarrow \mathbb{Z}$  such that  $g \circ f = i_{\mathbb{Z}}$ .

Let  $x \in \mathbb{Z}$ . Suppose  $x$  is even. Now  $x = i_{\mathbb{Z}}(x) = (g \circ f)(x) = g(f(x)) = g(x)$ . This means  $g(x) = x$  when  $x$  is even. Now suppose  $x$  is odd. Then  $x = i_{\mathbb{Z}}(x) = (g \circ f)(x) = g(f(x)) = g(4x + 1)$ . Let  $4x + 1 = y$ . Then  $x = \frac{y-1}{4}$ . Now if we take  $g(y) = \frac{y-1}{4}$  when  $y$  is odd, then we find that  $g(4x + 1) = \frac{4x+1-1}{4} = x$ . Thus, our choice of  $g$  is

$$g(x) = \begin{cases} x & \text{if } x \text{ is even,} \\ \frac{x-1}{4} & \text{if } x \text{ is odd.} \end{cases}$$

- (b) As in Worked-Out Exercise 3(c) (p. 294), we can show that  $f$  is not onto  $\mathbb{Z}$ . Because  $f$  is not onto  $\mathbb{Z}$ ,  $f$  does not have a right inverse (see Exercise 13, p. 314).

**Exercise 6:** Find  $\lfloor 5.15 \rfloor$ ,  $\lfloor \sqrt{20} \rfloor$ , and  $\lfloor -3.23 \rfloor$ . (For  $\sqrt{20}$ , consider only the positive square root.)

**Solution:** Because 5 is the greatest integer that does not exceed 5.15, it follows that  $\lfloor 5.15 \rfloor = 5$ .

Now  $\sqrt{20}$  is approximately 4.47, so we find that 4 is the greatest integer that does not exceed 4.4. It follows that  $\lfloor \sqrt{20} \rfloor = 4$ . (To show  $\lfloor \sqrt{20} \rfloor = 4$ , we can also argue as follows:  $4^2 = 16 < 20 < 25 = 5^2$ . Thus,  $4 < \sqrt{20} < 5$ . Hence,  $\lfloor \sqrt{20} \rfloor = 4$ .)

Because -4 is the greatest integer that does not exceed -3.23, it follows that  $\lfloor -3.23 \rfloor = -4$ .

**Exercise 7:** Find  $\lceil 1.34 \rceil$ ,  $\lceil \sqrt{39} \rceil$ , and  $\lceil -7.45 \rceil$ . (For  $\sqrt{39}$ , consider only the positive square root.)

**Solution:** Because 2 is the least integer greater than or equal to 1.34, it follows that  $\lceil 1.34 \rceil = 2$ .

Now  $\sqrt{39}$  is approximately 6.2, so we find that 7 is the least integer that is greater than or equal to 6.2. Hence, it follows that  $\lceil \sqrt{39} \rceil = 7$ . (To show  $\lceil \sqrt{39} \rceil = 7$ , we can also argue as follows:  $6^2 = 36 < 39 < 49 = 7^2$ . Thus,  $6 < \sqrt{39} < 7$ . Hence,  $\lceil \sqrt{39} \rceil = 7$ .)

Because -7 is the least integer that is greater than or equal to -7.45, it follows that  $\lceil -7.45 \rceil = -7$ .

**Exercise 8:** Let  $S = \{x \in \mathbb{R} \mid -1 < x < 1\}$ . Show that the sets  $\mathbb{R}$  and  $S$  have the same cardinality.

**Solution:** From Worked-Out Exercise 8 (p. 295), we find that there exists a one-to-one correspondence  $f : \mathbb{R} \rightarrow S$  defined by

$$f(x) = \frac{x}{1 + |x|}$$

from  $\mathbb{R}$  onto  $S$ . Hence,  $\mathbb{R}$  and  $S$  have the same cardinality.

**Exercise 9:** Show that the set  $S$  of all odd natural numbers is countable.

**Solution:** Define functions  $f : \mathbb{N} \rightarrow \mathbb{N} \cup \{0\}$  by  $f(n) = n - 1$  for all  $n \in \mathbb{N}$  and  $g : \mathbb{N} \cup \{0\} \rightarrow S$  by  $g(n) = 2n + 1$  for all  $n \in \mathbb{N}$ . Then  $f$  and  $g$  are one-one and onto, so  $g \circ f : \mathbb{N} \rightarrow S$  is a one-to-one correspondence. Hence,  $S$  is countable.

**Exercise 10:** Prove that every function  $f : A \rightarrow B$  can be expressed as a composition of a one-one, an onto, and a one-to-one correspondence.

**Solution:** Let  $f : A \rightarrow B$  be a function. Define the relation  $R$  on  $A$  by for all  $a, b \in A$ ,  $a R b$  if and only if  $f(a) = f(b)$ . Then  $R$  is an equivalence relation (see Exercise 20, p. 298). By Theorem 3.1.41, the equivalence relation  $R$  partitions  $A$  into disjoint equivalence classes. Let  $A/R = \{[a] \mid a \in A\}$  denote the set of all  $R$ -equivalence classes.

Now define  $h : A \rightarrow A/R$  by  $h(a) = [a]$ . Then  $h$  is an onto function. Next we define  $g : A/R \rightarrow \text{Im}(f)$  by

$$g([a]) = f(a).$$

Suppose  $[a], [b] \in A/R$ . Then

$$\begin{aligned} [a] &= [b] \\ \Leftrightarrow f(a) &= f(b) \\ \Leftrightarrow g([a]) &= g([b]) \\ \Rightarrow \text{an element of } A/R &\text{ cannot have more than} \\ &\text{one image in } \text{Im}(f). \end{aligned}$$

Thus, we verified that  $g : A/R \rightarrow \text{Im}(f)$  is a function, and we also found that this function is one-one and onto, i.e., a one-to-one correspondence.

Finally, define  $i : \text{Im}(f) \rightarrow B$  by  $i(b) = b$  for all  $b \in \text{Im}(f)$ . This mapping,  $i$ , is one-one.

Notice that the mapping  $i : \text{Im}(f) \rightarrow B$  is the restriction of the identity mapping  $i_B : B \rightarrow B$ .

Now consider the composition  $(i \circ g) \circ h : A \rightarrow B$  of the mappings  $h : A \rightarrow A/R$ ,  $g : A/R \rightarrow \text{Im}(f)$ ,  $i : \text{Im}(f) \rightarrow B$ . We find that

$$\begin{aligned} ((i \circ g) \circ h)(a) &= (i \circ g)(h(a)) \\ &= (i \circ g)([a]) \\ &= i(g([a])) \\ &= i(f(a)) \\ &= f(a), \end{aligned}$$

for all  $a \in A$ . This implies that  $f = (i \circ g) \circ h$  (see Figure 5.20).

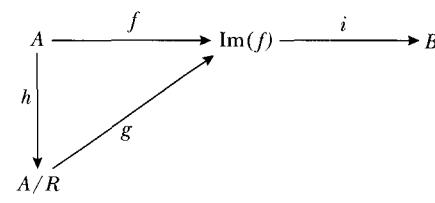


FIGURE 5.20  $f = (i \circ g) \circ h$

## SECTION REVIEW

---

### Key Terms

inverse function	image(s)	cardinality
left invertible	direct image	equivalent
left inverse	inverse image	equipotent
right invertible	floor	countable
right inverse	ceiling	uncountable
restriction	floor function	
extension	ceiling function	

### Some Key Definitions

- Let a function  $f : A \rightarrow B$  be a function from the set  $A$  into the set  $B$ .
  - $f$  is called left invertible if there exists  $g : B \rightarrow A$  such that  $g \circ f = i_A$ . Moreover, if such a function  $g$  exists, then  $g$  is called a left inverse of  $f$ .
  - $f$  is called right invertible if there exists  $h : A \rightarrow B$  such that  $f \circ h = i_B$ . Moreover, if such a function  $h$  exists, then  $h$  is called a right inverse of  $f$ .
- Let  $f : A \rightarrow B$  and  $\emptyset \neq A' \subseteq A$ . The restriction of  $f$  to  $A'$ , written  $f|_{A'}$ , is defined to be  $f|_{A'} = \{(a', f(a')) \mid a' \in A'\}$ .
- Let  $f : A \rightarrow B$  and  $A \subseteq A'$ . A function  $g : A' \rightarrow B$  is called an extension of  $f$  to  $A'$  if  $g|_A = f$ .
- For any real number  $x$ , the floor of  $x$ , written  $\lfloor x \rfloor$ , is the greatest integer less than or equal to  $x$ .
- For any real number  $x$ , the ceiling of  $x$ , written  $\lceil x \rceil$ , is the least integer greater than or equal to  $x$ .
- The function  $f : \mathbb{R} \rightarrow \mathbb{R}$  defined by  $f(x) = \lfloor x \rfloor$ , for all real numbers  $x$ , is called a floor function.
- The function  $g : \mathbb{R} \rightarrow \mathbb{R}$  defined by  $g(x) = \lceil x \rceil$ , for all real numbers  $x$ , is called a ceiling function.
- Two sets  $A$  and  $B$  have the same cardinality if there exists a one-to-one correspondence from  $A$  into  $B$ . If two sets  $A$  and  $B$  have the same cardinality, then we write  $|A| = |B|$ .
- A set  $A$  is said to be a countable set if either  $|A| = |I_n|$  for some positive integer  $n$  or  $|A| = |\mathbb{N}|$ .
- The empty set is also considered a countable set, and we write  $|\emptyset| = 0$ .
- A set  $S$  is uncountable if there is no one-to-one correspondence between  $\mathbb{N}$  and  $S$ .

### Some Key Results

- Let  $f : A \rightarrow B$  be a function. The inverse relation  $f^{-1} \subseteq B \times A$  is a function from  $B$  into  $A$  if and only if  $f$  is both one-one and onto  $B$ .

2. Let  $f : A \rightarrow B$  be a function such that  $f$  is one-one and onto  $B$ .
- If there exists a function  $g : B \rightarrow A$  such that  $g \circ f = i_A$ , then  $g = f^{-1}$ .
  - If there exists a function  $h : B \rightarrow A$  such that  $f \circ h = i_B$ , then  $h = f^{-1}$ .
3. The inverse of a function, if it exists, is unique.
4. Let  $x \in \mathbb{R}$  and  $n \in \mathbb{N}$ . Then  $\lfloor x + n \rfloor = \lfloor x \rfloor + n$ .

## EXERCISES

---

- Let  $X = \{x \in \mathbb{Z} \mid -2 < x \leq 5\}$  and  $Y = \{x \in \mathbb{Z} \mid 1 < x \leq 8\}$ . Consider the function  $f : X \rightarrow Y$  defined by  $f(x) = x + 3$  for all  $x \in X$ . Draw the arrow diagram for the function  $f : X \rightarrow Y$ . Is  $f$  a one-to-one correspondence? Find the inverse of  $f$  if it exists and then show the arrow diagram of  $f^{-1}$ .
- Let  $f : \mathbb{R} \rightarrow \mathbb{R}$  be the function defined by  $f(x) = 2x - 5$  for all  $x \in \mathbb{R}$ . Find the inverse of  $f$  if it exists.
- Let  $f : \mathbb{Q} \rightarrow \mathbb{Q}$  be the function defined by  $f(x) = 8x$  for all  $x \in \mathbb{Q}$ . Find the inverse of  $f$  if it exists.
- Let  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  be the function defined by  $f(x) = -x$  for all  $x \in \mathbb{Z}$ . Find the inverse of  $f$  if it exists.
- Let  $f : \mathbb{R} \rightarrow \mathbb{R}$  be the function defined by  $f(x) = \frac{3}{2}x - 5$  for all  $x \in \mathbb{R}$ . Find the inverse of  $f$  if it exists.
- Let  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  be the function defined by  $f(x) = 4x + 7$  for all  $x \in \mathbb{Z}$ . Find the inverse of  $f$  if it exists.
- Let  $f : \mathbb{R} \rightarrow \mathbb{R}$  be the function defined by  $f(x) = x^2$  for all  $x \in \mathbb{R}$ . Find the inverse of  $f$  if it exists.
- Let  $X = \{x \in \mathbb{R} \mid -2 < x \leq 5\}$  and  $Y = \{x \in \mathbb{R} \mid 1 < x \leq 22\}$ . Show that the function  $f : X \rightarrow Y$  defined by  $f(x) = 3x + 7$  for all  $x \in X$  is a one-to-one correspondence. Find the inverse of  $f$ .
- Let  $X = \{x \in \mathbb{Z} \mid -2 < x \leq 5\}$  and  $Y = \{x \in \mathbb{Z} \mid 1 < x \leq 22\}$ . Find the inverse of the function  $f : X \rightarrow Y$  defined by  $f(x) = 3x + 7$  for all  $x \in X$  if it exists.
- For each of the functions  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  given below, find a left inverse of  $f$  whenever one exists.
  - $f(x) = 4x - 3$  for all  $x \in \mathbb{Z}$ .
  - $f(x) = 7x$  for all  $x \in \mathbb{Z}$ .
- Let  $f : A \rightarrow B$ . Prove that  $f$  is left invertible if and only if  $f$  is one-one.
- Let  $f : \mathbb{N} \rightarrow \mathbb{N}$  be defined by  $f(n) = n + 5$  for all  $n \in \mathbb{N}$ .
  - Show that  $f$  is one-one but not onto.
  - Let  $g : \mathbb{N} \rightarrow \mathbb{N}$  be defined by: For all  $n \in \mathbb{N}$ ,
$$g(n) = \begin{cases} 5 & \text{if } n \leq 5 \\ n - 5 & \text{if } n > 5. \end{cases}$$

Show that  $g \circ f = i_{\mathbb{N}}$ .

  - Show that  $f \circ g \neq i_{\mathbb{N}}$ .
- Prove that  $f : A \rightarrow B$  is right invertible if and only if  $f$  is onto.
- Let  $f : \mathbb{Z} \rightarrow \mathbb{E}^*$  be defined by  $f(x) = |x| + x$  for all  $x \in \mathbb{Z}$ , where  $\mathbb{E}^*$  is the set of nonnegative even integers.

- Show that  $f$  is onto but not one-one.
  - Let  $g : \mathbb{E}^* \rightarrow \mathbb{Z}$  be defined by  $g(x) = \frac{x}{2}$  for all  $x \in \mathbb{E}^*$ . Verify that  $g$  is a right inverse of  $f$ .
  - Show that  $g \circ f \neq i_{\mathbb{Z}}$ .
15. For each of the functions  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  given below, find a right inverse of  $f$  whenever one exists.
- $f(x) = x - 2$  for all  $x \in \mathbb{Z}$ .
  - $f(x) = 9x$  for all  $x \in \mathbb{Z}$ .
  -
- $$f(x) = \begin{cases} x & \text{if } x \text{ is even,} \\ 2x + 1 & \text{if } x \text{ is odd,} \end{cases}$$
- for all  $x \in \mathbb{Z}$ .
16. Show that  $f : \mathbb{Z}^+ \rightarrow \{-1, 0, 1, 2\}$  defined by
- $$f(x) = \begin{cases} -1 & \text{if } x \text{ is of the form } 3n - 1, \\ 0 & \text{if } x \text{ is of the form } 3n, \\ 1 & \text{if } x \text{ is of the form } 3n + 1, \end{cases}$$
- can be expressed as a composition of one-one, onto, and one-to-one correspondences.
17. Given  $f : X \rightarrow Y$  and  $A, B \subseteq X$ , prove that
- $f(A \cup B) = f(A) \cup f(B)$ ,
  - $f(A \cap B) \subseteq f(A) \cap f(B)$ ,
  - $f(A - B) \subseteq f(A) - f(B)$  if  $f$  is one-one.
18. Given  $f : X \rightarrow Y$ , let  $S \subseteq Y$ . Define  $f^{-1}(S) = \{x \in X \mid f(x) \in S\}$ . Let  $A, B \subseteq Y$ . Prove that
- $f^{-1}(A \cup B) = f^{-1}(A) \cup f^{-1}(B)$ ,
  - $f^{-1}(A \cap B) = f^{-1}(A) \cap f^{-1}(B)$ ,
  - $f^{-1}(A - B) = f^{-1}(A) - f^{-1}(B)$ .
19. Let  $f : A \rightarrow B$ . Let  $f^*$  be the inverse relation, i.e.,
- $$f^* = \{(y, x) \in B \times A \mid f(x) = y\}.$$
- Show by an example that  $f^*$  need not be a function.
  - Show that  $f^*$  is a function from  $\text{Im}(f)$  into  $A$  if and only if  $f$  is one-one.
  - Show that  $f^*$  is a function from  $B$  into  $A$  if and only if  $f$  is one-one and onto  $B$ .
  - Show that if  $f^*$  is a function from  $B$  into  $A$ , then  $f^{-1} = f^*$ .

20. Let  $f$  be the function from the set  $\mathbb{N}$  into the set  $B = \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$  defined by  $f(x) = x \pmod{8}$  for all  $x \in \mathbb{N}$ . Draw the arrow diagram of the function  $f|_A : A \rightarrow B$ , where  $A = \{4, 7, 8, 10, 20, 24, 28\}$  and  $f|_A$  is the restriction of  $f$  to  $A$ .
21. Find  $\lfloor 8.13 \rfloor$ ,  $\lfloor \sqrt{221} \rfloor$ ,  $\lfloor -11.23 \rfloor$ , and  $\lfloor -1.23 \rfloor + 1$ .
22. Find  $\lceil 13.44 \rceil$ ,  $\lceil \sqrt{78} \rceil$ ,  $\lceil -1.25 \rceil$ , and  $\lceil 23.31 \rceil - 1$ .
23. Define  $f : \mathbb{R} \rightarrow \mathbb{R}$  by  $f(x) = \lfloor x \rfloor + 2$  and  $g : \mathbb{R} \rightarrow \mathbb{R}$  by  $g(x) = \lceil x \rceil + 1$ . Find  $f(9.11)$  and  $g(-13.02)$ .
24. Let  $f : \mathbb{R} \rightarrow \mathbb{R}$  and  $g : \mathbb{R} \rightarrow \mathbb{R}$  be defined by  $f(x) = \lceil x \rceil$  and  $g(x) = \lfloor x \rfloor + 2$ . Find  $f \circ g$  and  $g \circ f$ .
25. Define  $f : \mathbb{R} \rightarrow \mathbb{R}$  by  $f(x) = \lfloor x \rfloor + 2$  and  $g : \mathbb{R} \rightarrow \mathbb{R}$  by  $g(x) = \lceil x \rceil + 1$ . Find  $f \circ g$  and  $g \circ f$ .
26. Let  $A = \{x \in \mathbb{R} \mid 0 \leq x \leq 1\}$  and  $B = \{x \in \mathbb{R} \mid 5 \leq x \leq 8\}$ . Show that the sets  $A$  and  $B$  have the same cardinality.
27. Show that the set  $S = \{n \in \mathbb{N} \mid n \geq 100\}$  is countable.
28. Show that the sets  $5\mathbb{Z}$  and  $3\mathbb{Z}$  have the same cardinality.
29. Find a bijective function  $f : \mathbb{R}^+ \rightarrow S$ , where  $S = \{x \in \mathbb{R} \mid 0 < x < 1\}$ .
30. Let  $S = \{x \in \mathbb{R} \mid 0 < x < 2\}$ . Show the sets  $S$  and  $\mathbb{R}^+$  have the same cardinality.
31. For any two sets  $A$  and  $B$ , prove that  $A \times B$  and  $B \times A$  have the same cardinality.
32. Show that the set  $\mathbb{N} \cup \{0\}$  is a countable set.
33. For the following statement, write the proof if the statement is true, otherwise give a counterexample.  
A function  $f : A \rightarrow B$  is one-one if and only if for all subsets  $C$  of  $A$ ,  $f(A - C) \supseteq B - f(C)$ .

## 5.3 SEQUENCES AND STRINGS

Maxine is running a successful business and is planning a short vacation trip. She is planning to take a cell phone with her so that in the case of an emergency the manager at work can reach her. To budget her calls, she looks at various plans and chooses the plan that charges \$1.00 for the connection charge and \$.10 for each minute. For example, the charges for a one-minute call are \$1.10, the charges for a two-minute call are \$1.20, and so on. So Maxine makes the following table for the first ten minutes of telephone charges:

Minutes	1	2	3	4	5	6	7	8	9	10
Charges	1.10	1.20	1.30	1.40	1.50	1.60	1.70	1.80	1.90	2.00

Extrapolating from the table, we see that for a 30-minute call, the charges are  $1.00 + 30(0.10) = 4.00$ . In general, for an  $n$ -minute call, the charges are

$$1.00 + n(0.10) = 1.00 + 0.10n.$$

Let us list the telephone charges as follows:

$$1.10, 1.20, 1.30, \dots, (1.00 + 0.1n), \dots \quad (5.6)$$

This is an ordered list of real numbers in which the first element is 1.10, the second element is 1.20, and so on. Such an ordered list of elements is called a **sequence**. If we let  $c_1 = 1.10$ ,  $c_2 = 1.20, \dots, c_n = 1.00 + 0.1n$ , then in symbols, we can write this sequence as follows:

$$c_1, c_2, \dots, c_n, \dots \quad (5.7)$$

Typically, we call  $c_n$  the  **$n$ th term of the sequence**.

Consider the word *computer*. The 1st letter of this word is *c*, the 2nd is *o*, the 3rd is *m*, and so on. So we find an ordered list of letters

$$\begin{aligned} l_1 &= c, & l_2 &= o, & l_3 &= m, & l_4 &= p, \\ l_5 &= u, & l_6 &= t, & l_7 &= e, & l_8 &= r, \end{aligned}$$

where  $l_n$  denotes the  $n$ th letter of the word *computer*. Hence, the list

$$l_1, l_2, l_3, l_4, l_5, l_6, l_7, l_8$$

is a finite sequence.

Informally, a sequence on a set  $X$  is an ordered list of elements of  $X$ . For example, the ordered list

$$C : 1.10, 1.20, 1.30, \dots, (1.00 + 0.1n), \dots$$

is a sequence on the set of real numbers. Similarly, the ordered list

$$S : 2, 4, 6, 8, \dots,$$

is a sequence on the set of positive integers. The first element of  $S$  is 2, the second element is 4, the third element is 6, and so on.

The ordered list may stop after  $n$  elements for some positive integer  $n$  or it may go on forever. If the list stops, we say that the sequence is **finite**, otherwise it is called an *infinite sequence*. Generally, when we speak of a sequence we mean an infinite sequence.

Let us define the function  $f : \mathbb{N} \rightarrow \mathbb{R}$  as follows:

$$f(n) = 1.00 + (0.1)n. \quad (5.8)$$

Then

$$\begin{aligned} f(1) &= 1.00 + (0.1)1 = 1.10 = c_1, \\ f(2) &= 1.00 + (0.1)2 = 1.20 = c_2, \\ &\vdots \\ f(n) &= 1.00 + (0.1)n = c_n, \\ &\vdots \end{aligned}$$

Notice that  $c_1$  is the image of 1,  $c_2$  is the image of 2, and so on.

It follows that the sequence in (5.6), i.e., (5.7), can be realized as a function from  $\mathbb{N}$ , the set of natural numbers into the set  $\mathbb{R}$ , the set of real numbers. Indeed, this is the mathematical definition of a sequence. More formally, we have the following definition.

**DEFINITION 5.3.1** ▶ An **infinite sequence**, or simply a **sequence**, on a nonempty set  $X$  is a function from the set of positive integers  $\mathbb{N}$ , i.e., from the set  $\{1, 2, \dots\}$  into  $X$ .

Let  $f$  be a sequence on a set  $X$ . Then  $f$  is a function from  $\mathbb{N}$  into  $X$ . Typically, we use the subscript notation  $a_n$  (or  $x_n$ , or  $c_n$ , or  $s_n$ ) to denote the image  $f(n)$  of  $n$  in  $X$ .

Suppose  $f$  is a sequence on a set  $X$  given by  $f(n) = a_n$ . Even though the sequence  $f$  is a function, we generally represent  $f$  by listing its images at the integers  $1, 2, 3, \dots$  in the form

$$f : a_1, a_2, \dots, a_n, \dots$$

and refer to this as the sequence  $f$ , or

$$\{a_1, a_2, \dots\}$$

or briefly by

$$\{a_n\},$$

where  $a_n$  is the  $n$ th term of the sequence. We call  $a_1$  the first term of the sequence,  $a_2$  the second term of the sequence, and, in general,  $a_n$  the  $n$ th term of the sequence.

Occasionally, we add an extra term,  $a_0$ , the zeroth term of the sequence. In such a case, the domain of the sequence is extended to include 0, and the sequence is considered to be a function from the set  $\{0, 1, 2, \dots\}$  to  $X$ .

If we explicitly want to point out the first index of the sequence as 1 or 0, we write the sequence as  $\{a_n\}_{n=1}^{\infty}$  or  $\{a_n\}_{n=0}^{\infty}$ . The terms of a sequence may all be different or some may be repeated.

A sequence whose terms are integers is called an **integer sequence**.

### EXAMPLE 5.3.2

Let  $f : \mathbb{N} \rightarrow \mathbb{Z}$  be a function defined by  $f(n) = n^2$ . Then  $f(1) = 1, f(2) = 2^2 = 4, f(3) = 3^2 = 9$ . Thus,

$$1, 4, 9, 16, \dots, n^2, \dots$$

is a sequence on  $\mathbb{Z}$ . Let  $a_n$  denote the  $n$ th term of this sequence. Then  $a_1 = 1, a_2 = 4, a_3 = 9$ , and so on. We denote this sequence by  $\{n^2\}_{n=1}^{\infty}$ , or simply by  $\{n^2\}$ .

From now on we will describe sequences by listing the elements of the sequence, or using the sequence notation such as  $\{a_n\}_{n=1}^{\infty}$  or simply by  $\{a_n\}$ , or just by specifying the  $n$ th term of the sequence.

### EXAMPLE 5.3.3

Consider the list of numbers

$$1, 1, 2, 2, 3, 3, 4, 4, 5, 5, \dots$$

Suppose  $a_1 = 1, a_2 = 1, a_3 = 2, a_4 = 2, a_5 = 3, a_6 = 3$ , and so on. Then  $\{a_n\}_{n=1}^{\infty}$  is a sequence on the set of integers. Notice that not all the terms of this sequence are distinct.

### EXAMPLE 5.3.4

Consider the sequence  $\{a_n\}_{n=1}^{\infty}$ , where  $a_n = \frac{n}{n+1}$ , on the set of rational numbers. The first four terms of this sequence are

$$a_1 = \frac{1}{1+1} = \frac{1}{2}; \quad a_2 = \frac{2}{1+2} = \frac{2}{3}; \quad a_3 = \frac{3}{1+3} = \frac{3}{4}; \quad a_4 = \frac{4}{1+4} = \frac{4}{5}.$$

### EXAMPLE 5.3.5

Let  $S = \{s_n\}_{n=1}^{\infty}$  be a sequence on the set of integers such that the  $n$ th term,  $s_n$ , of  $S$  is given by  $s_n = 2n + 3$ . The first five terms of  $S$  are:  $a_1 = 2 \cdot 1 + 3 = 5$ ;  $a_2 = 2 \cdot 2 + 3 = 7$ ;  $a_3 = 2 \cdot 3 + 3 = 9$ ;  $a_4 = 2 \cdot 4 + 3 = 11$ ; and  $a_5 = 2 \cdot 5 + 3 = 13$ .

From the preceding examples, we see that if we know the  $n$ th term of a sequence, then we can determine all the terms of the sequence.

In the next few examples we illustrate how to determine the  $n$ th term of a sequence, given the listing of the sequence.

### EXAMPLE 5.3.6

Consider the following sequence.

$$2, 4, 6, 8, 10, 12, \dots$$

Let  $\{a_n\}_{n=1}^{\infty}$  denote this sequence. We want to determine  $a_n$ . Let us examine the first few terms of this sequence:

$$\begin{aligned}a_1 &= 2 = 2 \cdot 1, \\a_2 &= 4 = 2 \cdot 2, \\a_3 &= 6 = 2 \cdot 3, \\a_4 &= 8 = 2 \cdot 4, \\a_5 &= 10 = 2 \cdot 5, \\a_6 &= 12 = 2 \cdot 6,\end{aligned}$$

We see that the following pattern emerges: Each term of the sequence is 2 times its subscript. For example, for the fourth term,  $a_4$ , the subscript is 4, so  $a_4 = 2 \cdot 4 = 8$ . From this we can say that

$$a_n = 2n.$$

By using induction, we can in fact show that  $a_n = 2n$  for all  $n \geq 1$ .

### EXAMPLE 5.3.7

Consider the following sequence.

$$1, 2, 4, 7, 11, 16, 22, 29, \dots$$

Let  $\{a_n\}_{n=1}^{\infty}$  denote this sequence. We want to determine  $a_n$ . Let us examine the first few terms of this sequence. In this sequence,  $a_1 = 1, a_2 = 2, a_3 = 4, a_4 = 7, a_5 = 11, a_6 = 16$ . We see that

$$\begin{aligned}a_2 - a_1 &= 2 - 1 = 1 \Rightarrow a_2 = a_1 + 1, \\a_3 - a_2 &= 4 - 2 = 2 \Rightarrow a_3 = a_2 + 2, \\a_4 - a_3 &= 7 - 4 = 3 \Rightarrow a_4 = a_3 + 3, \\a_5 - a_4 &= 11 - 7 = 4 \Rightarrow a_5 = a_4 + 4, \\a_6 - a_5 &= 16 - 11 = 5 \Rightarrow a_6 = a_5 + 5.\end{aligned}$$

The following pattern is emerging: Given the first term, which is 1, each successive term can be obtained from the previous term by adding the subscript (index) of the term in the term itself. For example, to obtain  $a_2$ , the second term, we take the first term,  $a_1$ , which is 1, and add the subscript of  $a_1$ , which is 1, to it. That is,  $a_2 = 1 + 1 = 2$ . Similarly, for the third term,  $a_3$ , we add the subscript of the second term,  $a_2$ , to the value of  $a_2$ , to get  $a_3 = 2 + 2 = 4$ . Let us determine one more term, say  $a_6$ . To do so, we take  $a_5 = 11$  and add the subscript of  $a_5$ , which is 5, to it to get  $a_6 = 11 + 5 = 16$ .

We can now say that  $a_1 = 1$  and

$$a_n = a_{n-1} + (n - 1) \quad \text{for all } n > 1.$$

In Chapter 8, we will discuss how to determine an explicit formula for  $a_n$ , i.e., a formula for  $a_n$  that does not depend on the previous terms.

## Special Sequences

Consider the sequence

$$5, 7, 9, 11, \dots$$

Let  $\{a_n\}_{n=1}^{\infty}$  denote this sequence. Then  $a_1 = 5, a_2 = 7, a_3 = 9, a_4 = 11, \dots$ . Here we see that the difference between the consecutive terms is 2. For example,  $a_2 - a_1 = 7 - 5 = 2$ . Moreover, we can write

$$\begin{aligned}a_1 &= 5 = 5 + 0 = 5 + 0 \cdot 1, \\a_2 &= 7 = 5 + 2 = 5 + 2 \cdot 1, \\a_3 &= 9 = 5 + 4 = 5 + 2 \cdot 2, \\a_4 &= 11 = 5 + 6 = 5 + 3 \cdot 2.\end{aligned}$$

In general,  $a_n = 5 + (n - 1) \cdot 2$ . Let us write  $a = 5$  and  $d = 2$ . Then  $a_n = a + (n - 1)d$ . We can therefore write the preceding sequence as

$$a, a + d, a + 2d, \dots, a + (n - 1)d, a + nd, \dots$$

We often come across such sequences, and there is a special name for them: arithmetic progressions. More formally, we have the following definition.

---

**DEFINITION 5.3.8** ▶ Let  $a$  and  $d$  be real numbers. The sequence

$$a, a + d, a + 2d, \dots, a + (n - 1)d, a + nd, \dots,$$

is called an **arithmetic progression (AP)**. We call  $a$  the **first term** and  $d$  the **common difference** of the sequence. Furthermore,  $a + (n - 1)d$  is the  $n$ th term of the sequence.

### EXAMPLE 5.3.9

Let  $\{a_n\}_{n=1}^{\infty}$  be an AP such that the first term is  $a = 4$  and the common difference is  $d = 5$ . Let us find the first four terms and the  $n$ th term of this sequence.

Now  $a_1 = a = 4, a_2 = a + d = 4 + 5 = 9, a_3 = a + 2d = 4 + 2 \cdot 5 = 14, a_4 = a + 3d = 4 + 3 \cdot 5 = 19$ . Also,

$$a_n = a + (n - 1)d = 4 + (n - 1)5 = 4 + 5(n - 1).$$

Consider the sequence

$$2, 6, 18, 54, \dots$$

Let  $\{a_n\}_{n=1}^{\infty}$  denote this sequence. Then  $a_1 = 2, a_2 = 6, a_3 = 18, a_4 = 54, \dots$ . Here we see that the ratio of the consecutive terms is 3. For example,  $\frac{a_2}{a_1} = \frac{6}{2} = 3$  and  $\frac{a_4}{a_3} = \frac{54}{18} = 3$ . Moreover we can write

$$\begin{aligned}a_1 &= 2, \\a_2 &= 6 = 2 \cdot 3, \\a_3 &= 18 = 2 \cdot 9 = 2 \cdot 3^2, \\a_4 &= 54 = 2 \cdot 27 = 2 \cdot 3^3.\end{aligned}$$

In general,  $a_n = 2 \cdot 3^{n-1}$ . Let us write  $a = 2$  and  $r = 3$ . Then  $a_n = ar^{n-1}$ . We can therefore write the preceding sequence as

$$a, ar, ar^2, \dots, ar^{n-1}, ar^n, \dots$$

We come across such sequences in various contexts, and there is a special name for them: geometric progressions. More formally, we have the following definition.

**DEFINITION 5.3.10** ▶ Let  $a$  and  $r$  be real numbers. The sequence

$$a, ar, ar^2, \dots, ar^{n-1}, ar^n, \dots,$$

is called a **geometric progression (GP)**. We call  $a$  the **first term** and  $r$  the **common ratio** of the sequence. Furthermore  $ar^{n-1}$  is the  $n$ th term of the sequence.

**EXAMPLE 5.3.11**

Let  $\{a_n\}_{n=1}^{\infty}$  be a GP such that the first term is  $a = 3$  and the common ratio is  $r = \frac{1}{2}$ . Let us find the first four terms and the  $n$ th term of this sequence.

Now  $a_1 = a = 3$ ,  $a_2 = ar = 3 \cdot \frac{1}{2} = \frac{3}{2}$ ,  $a_3 = ar^2 = 3 \cdot (\frac{1}{2})^2 = \frac{3}{2^2}$ ,  $a_4 = ar^3 = 3 \cdot (\frac{1}{2})^3 = \frac{3}{2^3}$ . Also,

$$a_n = ar^{n-1} = \frac{3}{2^{n-1}}.$$

## Summation

Let  $\{a_n\}_{n=1}^{\infty}$  be a sequence. Consider the following terms of this sequence:

$$a_m, a_{m+1}, \dots, a_n.$$

Some of the common things we do with these terms are adding them and multiplying them. Let us first consider the addition.

The sum of the terms  $a_m, a_{m+1}, \dots, a_n$ ,

$$a_m + a_{m+1} + \dots + a_n$$

is typically written

$$\sum_{i=m}^n a_i \quad (5.9)$$

or

$$\sum_{i=m}^n a_i. \quad (5.10)$$

We call (5.9) or (5.10) the **sum of the terms**  $a_m, a_{m+1}, \dots, a_n$ . The symbol  $\sum$  is called the **summation symbol**. The variable  $i$  is called the **index**, or **subscript**, of the summation.

There is nothing special about the choice of the variable  $i$ . We could choose  $j$  or  $k$  as the index of the summation and write the sum as

$$\sum_{j=m}^n a_j \quad \text{or} \quad \sum_{k=m}^n a_k.$$

Note that

$$\sum_{i=m}^n a_i = \sum_{j=m}^n a_j = \sum_{k=m}^n a_k.$$

For this reason,  $i$  is also called a **dummy variable**.

Let us again consider the sum  $\sum_{i=m}^n a_i$  in (5.10). The sum starts at  $m$  and ends at  $n$ . We call  $m$  the **lower limit** of the sum,  $n$  the **upper limit** of the sum, and  $a_i$  the **general term** of the sum.

**EXAMPLE 5.3.12**

(i) In this example, we calculate the sum  $\sum_{i=1}^4 i$ . We see that

$$\sum_{i=1}^4 i = 1 + 2 + 3 + 4 = 10.$$

(ii) Let us calculate the sum  $\sum_{i=3}^6 i^2$ . We see that

$$\sum_{i=3}^6 i^2 = 3^2 + 4^2 + 5^2 + 6^2 = 9 + 16 + 25 + 36 = 86.$$

(iii) Let us calculate the sum  $\sum_{i=1}^3 \frac{i}{i+1}$ . We see that

$$\sum_{i=1}^3 \frac{i}{i+1} = \frac{1}{2} + \frac{2}{3} + \frac{3}{4} = \frac{23}{12}.$$

## Change of Index Variable

Consider the sums

$$\sum_{i=1}^3 (i+1) \quad \text{and} \quad \sum_{j=2}^4 j.$$

Let us expand both sums to get

$$\sum_{i=1}^3 (i+1) = (1+1) + (2+1) + (3+1) = 2 + 3 + 4 = 9$$

and

$$\sum_{j=2}^4 j = 2 + 3 + 4 = 9.$$

From this it follows that

$$\sum_{i=1}^3 (i+1) = \sum_{j=2}^4 j.$$

The general term in the first sum is  $i+1$ , and the general term in the second sum is  $j$ . It is easier to expand and simplify the second sum because it only requires us to replace the value of  $j$  starting at the lower limit until the upper limit, while in the first sum we need to substitute the value of  $i$  and also calculate  $i+1$ .

We see there are situations where one sum can be transformed into another sum by changing the appropriate index variable and determining the new lower and upper limits.

To change the index variable in a sum we do the following:

1. Calculate the lower limit of the new index variable.
2. Calculate the upper limit of the new index variable.
3. Find the general term of the summation in terms of the new index variable.

The following examples illustrate how to change the index variable.

**EXAMPLE 5.3.13**

Let us consider the sum  $\sum_{i=1}^3 (i+1)$  and change the index variable to  $j = i + 1$ .

- (i) Lower limit for  $j$ : The lower limit for  $i$  is 1, so the lower limit for  $j$  is  $j = i + 1 = 1 + 1 = 2$ .
- (ii) Upper limit for  $j$ : The upper limit for  $i$  is 3, so the upper limit for  $j$  is  $j = i + 1 = 3 + 1 = 4$ .
- (iii) The general term is  $i + 1 = j$ .

Hence, the equivalent sum is:

$$\sum_{j=2}^4 j.$$

**EXAMPLE 5.3.14**

Consider the sum  $\sum_{i=0}^{10} i(i+1)$ . Let us change the index variable to  $j = i + 1$ .

- (i) Lower limit for  $j = i + 1 = 0 + 1 = 1$ .
- (ii) Upper limit for  $j = i + 1 = 10 + 1 = 11$ .
- (iii) The general term for the new summation is given by

$$i(i+1) = (j-1)j$$

because if  $j = i + 1$ , then  $i = j - 1$ .

Hence, the equivalent sum is

$$\sum_{j=1}^{11} j(j-1).$$

**EXAMPLE 5.3.15**

Consider the sum  $\sum_{i=0}^{n-1} (n^2 + 1 + i)$ . Let us change the index variable to  $j = i + 1$ .

- (i) Lower limit for  $j = i + 1 = 0 + 1 = 1$ .
- (ii) Upper limit for  $j = i - 1 = n - 1 + 1 = n$ .
- (iii) The general term for the new summation is given by

$$n^2 + 1 + i = n^2 + j.$$

Hence, the new sum is

$$\sum_{j=1}^n (n^2 + j).$$

The following theorem lists some of the properties of summation. We leave the proof as an exercise.

**Theorem 5.3.16:** Let  $\{a_n\}_{n=1}^{\infty}$  and  $\{b_n\}_{n=1}^{\infty}$  be sequences of real numbers and let  $c$  be a real number. Suppose  $m$  and  $n$  are integers such that  $1 \leq m \leq n$ . Then

- (i)  $\sum_{i=m}^n a_i + \sum_{i=m}^n b_i = \sum_{i=m}^n (a_i + b_i)$ .
- (ii)  $c \cdot \sum_{i=m}^n a_i = \sum_{i=m}^n c a_i$ .

The following theorem lists some of the commonly known summations, some of which we proved, using induction, in Chapter 2.

**Theorem 5.3.17:**

- (i)  $\sum_{i=1}^n i = \frac{n(n+1)}{2}$  for all integers  $n \geq 1$ .
- (ii)  $\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}$  for all integers  $n \geq 1$ .
- (iii)  $\sum_{i=1}^n i^3 = \frac{n^2(n+1)^2}{4}$  for all integers  $n \geq 1$ .
- (iv) Let  $a$  and  $r$  be real numbers such that  $r \neq 1$ . Then

$$\sum_{i=0}^n ar^i = \begin{cases} \frac{ar^{n+1} - a}{r - 1} & \text{if } r \neq 1, \\ (n+1)a & \text{if } r = 1. \end{cases}$$

**Proof:**

- (i) This is (2.12) (Chapter 2, p. 133), and it was proved there.
- (ii) This was proved in Worked-Out Exercise 1 (p. 142).
- (iii) This is Exercise 4 (p. 147).
- (iv) To prove part (iv), let us first assume that  $r = 1$ . Then

$$\sum_{i=0}^n ar^i = \sum_{i=0}^n a = \underbrace{a + a + \cdots + a}_{n+1 \text{ times}} = (n+1)a.$$

Now suppose  $r \neq 1$ . Let us write

$$A = \sum_{i=0}^n ar^i. \quad (5.11)$$

Multiply both sides of (5.11) with  $r$  to get

$$\begin{aligned} rA &= r \left( \sum_{i=0}^n ar^i \right) \\ &= r(a + ar + ar^2 + \cdots + ar^{n-1} + ar^n) \\ &= ar + ar^2 + \cdots + ar^n + ar^{n+1} \\ &= a + (ar + ar^2 + \cdots + ar^n + ar^{n+1}) - a \quad \text{add and subtract } a \\ &= (a + ar + ar^2 + \cdots + ar^n) + (ar^{n+1} - a) \\ &= \sum_{i=0}^n ar^i + (ar^{n+1} - a) \\ &= A + (ar^{n+1} - a). \end{aligned}$$

Therefore,

$$rA = A + (ar^{n+1} - a).$$

This implies that

$$rA - A = (ar^{n+1} - a),$$

i.e.,

$$A(r - 1) = (ar^{n+1} - a).$$

Because  $r \neq 1$ ,  $r - 1 \neq 0$ , so we divide both sides of the preceding equation to get

$$A = \frac{ar^{n+1} - a}{r - 1}.$$

This proves the result. ■

## Product

Just as we can add the terms of a sequence, we can also multiply the terms of a sequence. Let  $\{a_n\}_{n=1}^{\infty}$  be a sequence. Consider the following terms of this sequence:

$$a_m, a_{m+1}, \dots, a_n.$$

The product of the terms  $a_m, a_{m+1}, \dots, a_n$ ,

$$a_m a_{m+1} \cdots a_n$$

is typically written as

$$\prod_{i=m}^n a_i \quad (5.12)$$

or

$$\prod_{i=m}^n a_i. \quad (5.13)$$

We call (5.12) and (5.13) the **product of the terms**  $a_m, a_{m+1}, \dots, a_n$ . The symbol  $\prod$  is called the **product symbol**. The variable  $i$  is called the **index**, or **subscript**, of the product.

As in the case of summation, there is nothing special about the choice of the variable  $i$ . We could choose  $j$  or  $k$  as the index of the product and write the product as

$$\prod_{j=m}^n a_j \quad \text{or} \quad \prod_{k=m}^n a_k.$$

In the product

$$\prod_{i=m}^n a_i,$$

as in the case of summation,  $m$  is called the lower limit,  $n$  is called the upper limit, and  $a_i$  is called the general term of the product.

### EXAMPLE 5.3.18

Let us evaluate the product  $\prod_{i=1}^4 i$ .

$$\prod_{i=1}^4 i = 1 \cdot 2 \cdot 3 \cdot 4 = 24.$$

### EXAMPLE 5.3.19

Let us evaluate the product  $\prod_{i=3}^6 \frac{i+1}{i}$ .

$$\prod_{i=3}^6 \frac{i+1}{i} = \frac{4}{3} \cdot \frac{5}{4} \cdot \frac{6}{5} \cdot \frac{7}{6} = \frac{7}{3}.$$

Just as we can change the index variable in a sum, we can also change the index variable in a product. The rules for doing this are the same: Calculate the lower and upper limit for the new index variable and the general term for the product.

**Theorem 5.3.20:** Let  $\{a_n\}_{n=1}^{\infty}$  and  $\{b_n\}_{n=1}^{\infty}$  be sequences of real numbers. Suppose  $m$  and  $n$  are integers such that  $1 \leq m \leq n$ . Then

$$\prod_{i=m}^n a_i \cdot \prod_{i=m}^n b_i = \prod_{i=m}^n (a_i b_i)$$

## Strings (Words)

Let  $A$  be the set of the lowercase English alphabet. Consider the following sequence on  $A$ :

$$d, i, s, c, r, e, t, e \quad (5.14)$$

This is a finite sequence and it has eight terms. If  $\{a_k\}_{k=1}^8$  denotes this sequence, then  $a_1 = d$ ,  $a_2 = i$ ,  $a_3 = s$ ,  $a_4 = c$ ,  $a_5 = r$ ,  $a_6 = e$ ,  $a_7 = t$ ,  $a_8 = e$ . If we omit the commas in (5.14), then we can write this sequence as *discrete*.

Finite sequences over a set are of special interest in computer science. We use a special name for finite sequences and call them strings or words. More formally, we have the following definition.

---

**DEFINITION 5.3.21** ▶ Let  $A$  be a nonempty finite set. A **string**, or **word**, over  $A$  is a finite sequence of elements from  $A$ . The set  $A$  is called an **alphabet**.

Even though a string is a sequence, while describing the string we omit the commas. For example, the sequence  $m, a, t, h, e, m, a, t, i, c, s$  over the English alphabet is written as *mathematics*.

The **length** of a string (word)  $s$ , written  $|s|$ , is the number of elements in the string (word). For example, the length of the string *mathematics* is 11, and the length of the string *discrete* is 8.

A string with no element in it is called the **empty string** or **empty word**. We denote the empty string (word) by  $\lambda$ . It follows that the length of the empty string (word)  $\lambda$  is 0, i.e.,  $|\lambda| = 0$ .

If  $s_1$  and  $s_2$  are two strings over a set  $A$ , then the **concatenation** of  $s_1$  and  $s_2$  is the string  $s_1 s_2$ . That is, to obtain the concatenation of  $s_1$  and  $s_2$ , we list the elements of  $s_1$  followed by the elements of  $s_2$ .

For example, suppose  $s_1 = ababedb$  and  $s_2 = caabcbdd$  are two strings over the set  $A = \{a, b, c, d\}$ . Then the concatenation of  $s_1$  and  $s_2$  is  $s_1 s_2 = ababedbcacabcbdd$ .

From the definition of length and the concatenation of strings, it follows that if  $s_1$  and  $s_2$  are strings over a set  $A$ , then

$$|s_1 s_2| = |s_1| + |s_2|.$$

### EXAMPLE 5.3.22

Let  $A = \{0, 1\}$ .

- (i) Let  $s = 01101010$ . Then  $s$  is a string over  $A$  and  $|s| = 8$ .
- (ii) 00, 01, 10, 11 are the only strings of length 2 over  $A$ .
- (iii) If  $s_1 = 1001010$  and  $s_2 = 00111$ , then  $s_1 s_2 = 100101000111$ . Also  $|s_1| = 7$ ,  $|s_2| = 5$ , and  $|s_1 s_2| = 12 = |s_1| + |s_2|$ .

**Theorem 5.3.23:** Let  $A$  be a finite set and let  $s, s_1, s_2, s_3$  be strings over  $A$ . Then

- (i)  $\lambda s = s = s\lambda$ .
- (ii)  $s_1(s_2s_3) = (s_1s_2)s_3$ .
- (iii)  $|s_1s_2| = |s_1| + |s_2|$ .

## Representing Strings into Computer Memory

Writing programs to perform operations on strings requires the ability to store strings in computer memory. A string is a finite sequence on a set, say  $A$ ; that is, the number of elements in a string is finite. Therefore, a convenient way of storing a string into computer memory is to use an array.

For example, let  $A = \{a, b\}$  and  $s = ababbabab$  be a string over  $A$ . The length of  $s$  is  $|s| = 11$ . Therefore, we can use an array of the size 11 to store  $s$  as follows:

[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]
a	b	a	b	b	a	b	a	b	a	b

The top row shows the position of an element in the string.

---

**REMARK 5.3.24** ▶ Programming languages such as C++ and Java provide a data type to manipulate strings. This data type includes algorithms to implement operations such as concatenation, finding the length of a string, determining whether a string is a substring in another string, and finding a substring into another string.

As we remarked in Chapter 2, in computer memory everything is represented as a sequence of 0's and 1's. In other words, the language of a computer is a sequence of 0's and 1's. If we let  $A$  be the set  $A = \{0, 1\}$ , then the language of a computer is strings over  $A$ .

In Chapter 2, we described algorithms to convert a number from its binary (base-2) representation to its decimal (base-10) representation. However, the algorithms given there are recursive. In this section, we describe nonrecursive algorithms to convert a number from binary to decimal and decimal to binary.

When we use the built-in data type of a programming language to manipulate numbers, there is a limit to the largest number that can be stored in computer memory. For example, in Chapter 2, we remarked that if 8 bits are used to store both positive and negative integers, then the largest integer we can store is 127 and the smallest integer we can store is  $-128$ . Typically, in a programming language 32 bits are used to store integers. Some languages may provide 64 bits to store integers.

For the sake of discussion, let us suppose that 32 bits are used to store integers. Then the largest and smallest integers that can be represented are 2,147,483,647 and  $-2,147,483,648$ , respectively. If only nonnegative integers are stored, then the largest integer we can represent is 4,294,967,295.

Let us assume that we are dealing with nonnegative integers only. To write nonrecursive algorithms to convert numbers from base 2 to base 10, we use an array of size 32. The following algorithm converts a nonnegative integer from base 2 to base 10.

**ALGORITHM 5.1:** Nonrecursive algorithm to convert a number from base 2 to base 10.

*Input:*  $L$ —an array of the size 32

*Output:*  $x$ —the decimal representation of the binary number

```

1. function nonRecursiveBinaryToDecimal( $L$ )
2. begin
3.    $x := 0$ ;
4.   for  $i := 1$  to 32 do
5.      $x := x + L[32 + 1 - i] * 2^{i-1}$ ;
6.   return  $x$ ;
7. end
```

Next we describe the nonrecursive algorithm to convert a nonnegative integer from base 10 to base 2.

**ALGORITHM 5.2:** Nonrecursive algorithm to convert a number from base 10 to base 2.

*Input:*  $x$ —the decimal representation of the binary number

*Output:*  $L$ —an array containing the binary representation of  $x$

```

1. procedure nonRecursiveDecimalToBinary( $L, x$ )
2. begin
3.   for  $i := 1$  to 32 do
4.      $L[i] = 0$ ;
5.    $n := 32$ ;
6.   while ( $x \neq 0$ ) do
7.     begin
8.        $L[n] := x \bmod 2$ ;
9.        $n := n - 1$ ;
10.       $x := x \div 2$ ;
11.    end
12.  end
```


**WORKED-OUT EXERCISES**


---

**Exercise 1:** Let  $\{a_n\}_{n=1}^{\infty}$  be a sequence such that  $a_n = 2n^2 - 3$ . Find the first three terms of this sequence.

**Solution:** We have  $a_1 = 2 \cdot 1^2 - 3 = 2 - 3 = -1$ ,  $a_2 = 2 \cdot 2^2 - 3 = 8 - 3 = 5$ , and  $a_3 = 2 \cdot 3^2 - 3 = 18 - 3 = 15$ .

**Exercise 2:** Consider the following sequence.

$$\frac{2}{2}, \frac{3}{2}, \frac{4}{3}, \frac{5}{4}, \frac{6}{5}, \frac{7}{6}, \dots$$

Find the  $n$ th term of this sequence.

**Solution:** Let  $\{a_n\}_{n=1}^{\infty}$  denote this sequence. Then  $a_1 = 2$ ,  $a_2 = \frac{3}{2}$ ,  $a_3 = \frac{4}{3}$ ,  $a_4 = \frac{5}{4}$ ,  $a_5 = \frac{6}{5}$ ,  $a_6 = \frac{7}{6}$ . We note that

$$\begin{aligned} a_1 &= 2 = 1 + 1 = 1 + \frac{1}{1}, \\ a_2 &= \frac{3}{2} = 1 + \frac{1}{2}, \\ a_3 &= \frac{4}{3} = 1 + \frac{1}{3}, \\ a_4 &= \frac{5}{4} = 1 + \frac{1}{4}, \\ a_5 &= \frac{6}{5} = 1 + \frac{1}{5}, \\ a_6 &= \frac{7}{6} = 1 + \frac{1}{6}. \end{aligned}$$

From this we find that  $a_n = 1 + \frac{1}{n}$ .

**Exercise 3:** Evaluate the following.

$$(a) \sum_{k=1}^4 (k^2 - 2k) \quad (b) \sum_{k=1}^{10} 100 \quad (c) \prod_{k=1}^3 4^k$$

**Solution:**

$$\begin{aligned} (a) \sum_{k=1}^4 (k^2 - 2k) &= (1^2 - 2 \cdot 1) + (2^2 - 2 \cdot 2) + (3^2 - 2 \cdot 3) + (4^2 - 2 \cdot 4) \\ &= (1 - 2) + (4 - 4) + (9 - 6) + (16 - 8) \\ &= -1 + 0 + 3 + 8 \\ &= 10. \end{aligned}$$

$$(b) \sum_{k=1}^{10} 100 = \underbrace{100 + 100 + \dots + 100}_{10 \text{ times}} = 10 \cdot 100 = 1000.$$

$$(c) \prod_{k=1}^3 4^k = 4^1 \cdot 4^2 \cdot 4^3 = 4^6 = 4096.$$

**Exercise 4:** Simplify:

$$\sum_{i=1}^n \left( \frac{i}{i+1} - \frac{i+1}{i+2} \right).$$

**Solution:** Let us expand the summation:

$$\sum_{i=1}^n \left( \frac{i}{i+1} - \frac{i+1}{i+2} \right)$$

$$\begin{aligned} &= \left( \frac{1}{2} - \frac{2}{3} \right) + \left( \frac{2}{3} - \frac{3}{4} \right) + \left( \frac{3}{4} - \frac{4}{5} \right) + \dots + \\ &\quad \left( \frac{n-1}{n} - \frac{n}{n+1} \right) + \left( \frac{n}{n+1} - \frac{n+1}{n+2} \right) \\ &= \frac{1}{2} - \frac{2}{3} + \frac{2}{3} - \frac{3}{4} + \frac{3}{4} - \frac{4}{5} + \dots + \frac{n-1}{n} \\ &\quad - \frac{n}{n+1} + \frac{n}{n+1} - \frac{n+1}{n+2} \\ &= \frac{1}{2} - \frac{n+1}{n+2}. \end{aligned}$$

Let us simplify by changing the index variable in one of the summations. First notice that

$$\sum_{i=1}^n \left( \frac{i}{i+1} - \frac{i+1}{i+2} \right) = \sum_{i=1}^n \frac{i}{i+1} - \sum_{i=1}^n \frac{i+1}{i+2}.$$

Let us change the index variable to  $j = i + 1$  in the sum  $\sum_{i=1}^n \frac{i+1}{i+2}$ .

(1) Lower limit: If  $i = 1$ , then  $j = i + 1 = 1 + 1 = 2$ .

(2) Upper limit: If  $i = n$ , then  $j = i + 1 = n + 1$ .

(3) The general term of the new sum:  $\frac{i+1}{i+2} = \frac{i+1}{i+1+1} = \frac{j}{j+1}$ . Thus,

$$\sum_{i=1}^n \frac{i+1}{i+2} = \sum_{j=2}^{n+1} \frac{j}{j+1}.$$

It now follows that

$$\sum_{i=1}^n \frac{i}{i+1} - \sum_{i=1}^n \frac{i+1}{i+2} = \sum_{i=1}^n \frac{i}{i+1} - \sum_{j=2}^{n+1} \frac{j}{j+1}.$$

Next, in both of the summations on the right side change the index variables to  $k$  to get

$$\sum_{i=1}^n \frac{i}{i+1} - \sum_{j=2}^{n+1} \frac{j}{j+1} = \sum_{k=1}^n \frac{k}{k+1} - \sum_{k=2}^{n+1} \frac{k}{k+1}.$$

We can write  $\sum_{k=1}^n \frac{k}{k+1} = \frac{1}{2} + \sum_{k=2}^n \frac{k}{k+1}$  and  $\sum_{k=2}^{n+1} \frac{k}{k+1} = \sum_{k=2}^n \frac{k}{k+1} + \frac{n+1}{n+2}$ . Hence,

$$\begin{aligned} &\sum_{k=1}^n \frac{k}{k+1} - \sum_{k=2}^{n+1} \frac{k}{k+1} \\ &= \left( \frac{1}{2} + \sum_{k=2}^n \frac{k}{k+1} \right) - \left( \sum_{k=2}^n \frac{k}{k+1} + \frac{n+1}{n+2} \right) \\ &= \frac{1}{2} + \sum_{k=2}^n \frac{k}{k+1} - \sum_{k=2}^n \frac{k}{k+1} - \frac{n+1}{n+2} \\ &= \frac{1}{2} + \left( \sum_{k=2}^n \frac{k}{k+1} - \sum_{k=2}^n \frac{k}{k+1} \right) - \frac{n+1}{n+2} \\ &= \frac{1}{2} - \frac{n+1}{n+2}. \end{aligned}$$

**Exercise 5:** Evaluate  $\sum_{i=1}^4 \sum_{j=2}^5 (ij)$ .

**Solution:** To evaluate  $\sum_{i=1}^4 \sum_{j=2}^5 (ij)$ , we first evaluate the inner summation. Therefore,

$$\begin{aligned}\sum_{i=1}^4 \sum_{j=2}^5 (ij) &= \sum_{i=1}^4 (2i + 3i + 4i + 5i) \\&= \sum_{i=1}^4 14i \\&= 14 \sum_{i=1}^4 i \quad \text{by Theorem 5.3.16(ii)} \\&= 14 \frac{4(4+1)}{2} \quad \text{by Theorem 5.3.17(i)} \\&= 140.\end{aligned}$$

**Exercise 6:** In the following arithmetic progressions, the first two terms are given. Write the next four terms.

(a) 7, 11, ...      (b)  $-\frac{1}{2}, 1, \dots$

**Solution:**

- (a) The common difference of the given AP is 4. Hence, the next four terms are 15, 19, 23, and 27.

- (b) The common difference of the given AP is  $1 - (-\frac{1}{2}) = \frac{3}{2}$ . Hence, the next four terms are  $1 + \frac{3}{2} = \frac{5}{2}, \frac{5}{2} + \frac{3}{2} = 4, 4 + \frac{3}{2} = \frac{11}{2}$ , and  $\frac{11}{2} + \frac{3}{2} = 7$ .

**Exercise 7:** If the 8th term of a GP is 20 and the 13th term is 640, then find the 20th term of this GP.

**Solution:** Let  $a$  be the first term and  $r$  be the common ratio of the given GP. Then the  $n$ th term is  $a_n = ar^{n-1}$ . Thus,  $a_8 = ar^{8-1}$  and  $a_{13} = ar^{13-1}$ .

From the given conditions, we have  $20 = ar^7$  and  $640 = ar^{12}$ . So we find that  $\frac{640}{20} = \frac{ar^{12}}{ar^7}$ . This implies  $32 = r^5$ . Hence,  $r = 2$ .

Now from  $20 = ar^7$ , we find that

$$a = \frac{20}{r^7} = \frac{20}{2^7} = \frac{20}{128} = \frac{5}{32}.$$

Hence, the 20th term is

$$a_{20} = \frac{5}{32} 2^{20-1} = 5 \cdot 2^{14} = 81920.$$

## SECTION REVIEW

### Key Terms

sequence	sum of the terms	index
$n$ th term of the sequence	summation symbol	subscript
finite sequence	index	string
infinite sequence	subscript	word
integer sequence	dummy variable	alphabet
arithmetic progression (AP)	lower limit	length
first term	upper limit	empty string
common difference	general term	empty word
geometric progression (GP)	product of the terms	concatenation
common ratio	product symbol	

### Some Key Definitions

- An infinite sequence, or simply a sequence, on a nonempty set  $X$  is a function from the set of positive integers  $N$ , i.e., from the set  $\{1, 2, \dots\}$  into  $X$ .
- A sequence whose terms are integers is called an integer sequence.
- Let  $a$  and  $d$  be real numbers. The sequence  $a, a+d, a+2d, \dots, a+(n-1)d, a+nd, \dots$ , is called an arithmetic progression (AP).

4. Let  $a$  and  $r$  be real numbers. The sequence  $a, ar, ar^2, \dots, ar^{n-1}, ar^n, \dots$ , is called a geometric progression (GP).
5. To change the index variable in a sum we do the following: (1) Calculate the lower limit of the new index variable. (2) Calculate the upper limit of the new index variable. (3) Find the general term of the summation in terms of the new index variable.
6. Let  $A$  be a nonempty finite set. A string, or word, over  $A$  is a finite sequence of elements from  $A$ . The set  $A$  is called an alphabet.
7. The length of a string  $s$ , written  $|s|$ , is the number of elements in the string.
8. A string with no element in it is called the empty string, or empty word.
9. If  $s_1$  and  $s_2$  are two strings over a set  $A$ , then the concatenation of  $s_1$  and  $s_2$  is the string  $s_1 s_2$ .

## Some Key Results

1. Let  $\{a_n\}_{n=1}^{\infty}$  and  $\{b_n\}_{n=1}^{\infty}$  be sequences of real numbers and let  $c$  be a real number. Suppose  $m$  and  $n$  are integers such that  $1 \leq m \leq n$ . Then
  - (i)  $\sum_{i=m}^n a_i + \sum_{i=m}^n b_i = \sum_{i=m}^n (a_i + b_i)$ .
  - (ii)  $c \cdot \sum_{i=m}^n a_i = \sum_{i=m}^n c a_i$ .
2. Let  $A$  be a finite set and let  $s, s_1, s_2, s_3$  be strings over  $A$ . Then
  - (i)  $\lambda s = s = s\lambda$ .
  - (ii)  $s_1(s_2s_3) = (s_1s_2)s_3$ .
  - (iii)  $|s_1s_2| = |s_1| + |s_2|$ .

## EXERCISES

---

In Exercises 1–8, write the first five terms of the sequence whose  $n$ th term is given.

1.  $a_n = 2n + 4, n \geq 1$
2.  $a_n = n^2 - 1, n \geq 0$
3.  $b_n = \frac{3n}{2n-5}, n \geq 1$
4.  $c_n = (-1)^n n, n \geq 1$
5.  $s_n = (-1)^n n^2, n \geq 0$
6.  $a_n = n!, n \geq 0$
7.  $b_n = \frac{n^2}{n!}, n \geq 1$
8.  $d_n = 20 - \left(\frac{7}{n}\right)^2, n \geq 1$

In Exercises 9–14, find  $n$ th term of the sequence.

9.  $1, -1, 1, -1, 1, -1, \dots$
10.  $-1, 2, -3, 4, -5, 6, -7, \dots$
11.  $3, 8, 15, 24, 35, 48, \dots$
12.  $\frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \frac{4}{5}, \frac{5}{6}, \dots$
13.  $\frac{1}{8}, \frac{4}{27}, \frac{9}{64}, \frac{16}{125}, \frac{25}{216}, \dots$
14.  $-4, -1, 4, 11, 20, 32, \dots$

15. Find the first five terms and the  $n$ th term of the arithmetic progression (AP) with the first term  $a$  and the common difference  $d$  specified as follows:

- (i)  $a = 5, d = 3$ ,
- (ii)  $a = -2, d = 6$ ,
- (iii)  $a = 3, d = \frac{1}{2}$ .

16. Find the first five terms and the  $n$ th term of the geometric progression (GP) with the first term  $a$  and the common ratio  $r$  specified as follows:

- (i)  $a = 2, r = 3$ ,
- (ii)  $a = 1, r = \frac{1}{2}$ ,
- (iii)  $a = -2, r = -\frac{1}{3}$ .

17. Write the next four terms of the following arithmetic progressions:

- a.  $8, -11, \dots$
- b.  $-2, -\frac{1}{3}, \dots$

18. Write the next four terms of the following geometric progressions:

- a.  $4, -16, \dots$
- b.  $-2, -\frac{1}{2}, \dots$

19. If the sum of the first three terms of an AP is 225, then find the second term.

20. If the 4th term of an AP is 13 and the 9th term is 33, then find the 100th term of this AP.

21. If the 5th term of an AP is 41 and the 11th term is 20, then find the 15th term of this AP.

22. If the 12th term of an AP is  $-13$  and the sum of the first four terms is 24, then find the first term and common difference of this AP.

23. If the 3rd term of a GP is 4 and the 7th term is 64, then find the 9th term of this GP.
24. If the product of the first three terms of a GP is 125, then find the second term.
25. Suppose three numbers are in GP. If their sum is 19 and their product is 216, then find the numbers.
26. Let  $\{a_n\}_{n=1}^{\infty}$  be a sequence such that  $a_n = 2n + 3$ . Find the following sums and products:

$$\begin{array}{ll} \text{(i)} & \sum_{i=1}^5 a_i \\ & \prod_{i=1}^4 a_i \\ \text{(ii)} & \sum_{i=3}^7 a_i \\ & \prod_{i=2}^5 a_i \end{array}$$

In Exercises 27–35, find the indicated sum or product.

27.  $\sum_{i=1}^7 (2n - 4)$
28.  $\sum_{i=2}^6 n^2$
29.  $\sum_{i=1}^5 (2n^2 - n)$
30.  $\sum_{i=4}^7 \frac{n^2}{n+1}$
31.  $\sum_{j=1}^7 \left( \frac{1}{j} - \frac{1}{j+1} \right)$
32.  $\sum_{k=1}^{10} \left( \frac{k^2}{(k+1)^2} - \frac{(k+1)^2}{(k+2)^2} \right)$
33.  $\prod_{i=1}^4 (-1)^i$
34.  $\prod_{k=2}^4 k^2$
35.  $\prod_{k=21}^5 \frac{(k+1)(k+2)}{k^2 - 3}$

In Exercises 36–43, write the expression using summation or product notation.

36.  $1 - 2 + 3 - 4 + 5 - 6$
37.  $1 - 4 + 9 - 16 + 25 - 36 + 49$
38.  $(1 - \frac{1}{1}) + (1 - \frac{1}{2}) + (1 - \frac{1}{3}) + (1 - \frac{1}{4})$
39.  $\frac{2}{1!} + \frac{3}{2!} + \frac{4}{3!} + \frac{5}{4!} + \cdots + \frac{n+1}{n!}$

40.  $1 + 2^3 + 3^3 + 4^3 + 5^3 + \cdots + n^3$
41.  $1 + a + a^2 + \cdots + a^n$
42.  $1 \cdot 3 \cdot 5 \cdot 7 \cdot 9 \cdot 11$
43.  $\frac{1}{2} \cdot \frac{2}{3} \cdot \frac{3}{4} \cdot \frac{4}{5} \cdot \frac{5}{6}$

In Exercises 44–47, change the index variable as indicated.

44.  $\sum_{i=3}^n (i - 3)$ : Change the index variable to  $j = i - 3$ .
45.  $\sum_{i=1}^n (i - 1)^2$ : Change the index variable to  $j = i - 1$ .
46.  $\sum_{k=1}^n \frac{k}{(k+1)^2}$ : Change the index variable to  $j = k + 1$ .
47.  $\sum_{k=1}^n \frac{2n-k-1}{(k+1)^2}$ : Change the index variable to  $j = k + 1$ .
48. Evaluate the following.

$$\begin{array}{ll} \text{(i)} & \sum_{i=1}^3 \sum_{j=1}^5 (i+j) \\ & \sum_{i=0}^4 \sum_{j=2}^3 (i^j) \\ \text{(ii)} & \sum_{i=1}^5 \sum_{j=1}^4 i \\ & \sum_{i=2}^6 \sum_{j=1}^3 (i^2 j^2) \end{array}$$

49. Write  $\sum_{k=1}^n (k^2 - 2k) + \sum_{k=1}^n (k - 3)$  as a single summation. If possible, simplify the general term.
50. Write  $3 \sum_{k=1}^n (k^2 + 5) + \sum_{k=1}^n (8k - 2k^3)$  as a single summation. If possible, simplify the general term.
51. Write  $(\prod_{k=1}^n \frac{2k}{k^2+1}) \cdot (\prod_{k=1}^n \frac{k^2+1}{k+2})$  as a single product. If possible, simplify the general term.
52. Prove Theorem 5.3.16.
53. Prove Theorem 5.3.20.
54. Let  $\{a_n\}_{n=1}^{\infty}$  be a sequence. The summation  $\sum_{i=1}^n (a_i - a_{i+1})$  is called the *telescoping sum*. Prove that  $\sum_{i=1}^n (a_i - a_{i+1}) = (a_1 - a_{n+1})$ .
55. Let  $A = \{a, b, c\}$  and let  $s_1$ ,  $s_2$ , and  $s_3$  be strings over  $A$  such that  $s_1 = aabbcc$ ,  $s_2 = abcabc$ , and  $s_3 = abccbacab$ . Find the following.
- (i)  $|s_1|$    (ii)  $|s_2|$    (iii)  $|s_3|$   
 (iv)  $s_1 s_2$    (v)  $s_3 s_2$    (vi)  $s_1 s_2 s_3$
56. Let  $A = \{a, b, c\}$ .
- a. Find all strings of length 2 over  $A$ .  
 b. Find all strings of length 3 over  $A$ .
57. Prove Theorem 5.3.23.

## 5.4 BINARY OPERATIONS

The concept of a binary operation<sup>2</sup> is very important in algebra. In this section, we define binary operations and examine their basic properties. Since elementary school, we have been accustomed to the basic operations of addition, multiplication, subtraction, and division of numbers. Each of these operations allows us to associate a unique number for each pair of numbers. For example, for the integers 20 and 4, the operation  $+$  associates the unique integer 24 by adding these numbers, i.e.,  $20 + 4 = 24$ . Similarly,  $20 - 4 = 16$ , so the operation subtraction,  $-$ , associates the unique integer 16 with the pair of integers 20 and 4. We can therefore consider  $+$  as an operation between two integers  $a$  and  $b$  that associates the unique element  $a + b$  with these integers. We can, in fact, regard  $+$  as a function from  $\mathbb{Z} \times \mathbb{Z}$  into  $\mathbb{Z}$  such that for all  $a, b \in \mathbb{Z}$ ,  $+(a, b) = a + b$ . Using this

<sup>2</sup>This section may be skipped without any discontinuation

convention, we say that  $+$  is a binary operation on  $\mathbb{Z}$ . More formally, we have the following definition.

**DEFINITION 5.4.1** ▶ Let  $S$  be a nonempty set. A **binary operation** on  $S$  is a function from  $S \times S$  into  $S$ .

For any ordered pair  $(x, y)$  of elements  $x, y \in S$ , a binary operation on  $S$  assigns a unique member of  $S$ . For example,  $+$  is a binary operation on  $\mathbb{Z}$  which assigns 24 to the pair  $(20, 4)$ .

If  $*$  is a binary operation on  $S$ , we write  $x * y$  for  $*(x, y)$ , where  $x, y \in S$ . Because the image of  $S$  under  $*$  is a subset of  $S$ , we say that  $S$  is **closed under  $*$** .

$\mathbb{Z}$  is closed under  $+$  because if we add two integers we obtain an integer.

Now  $3, 10 \in \mathbb{N}$  and  $3 - 10 = -7 \notin \mathbb{N}$ . Thus,  $-$  (subtraction) is not a binary operation on  $\mathbb{N}$  and we say that  $\mathbb{N}$  is not closed under the operation  $-$ .

**DEFINITION 5.4.2** ▶ Let  $S$  be a nonempty set and  $*$  a binary operation on  $S$ . Then

(i)  $*$  is called **associative** if for all  $x, y, z \in S$ ,

$$x * (y * z) = (x * y) * z.$$

(ii)  $*$  is called **commutative** if for all  $x, y \in S$ ,

$$x * y = y * x.$$

### EXAMPLE 5.4.3

Addition of integers is both associative and commutative. But  $5 - 2 \neq 2 - 5$  and  $(1 - 2) - 3 \neq 1 - (2 - 3)$ . Hence, the binary operation subtraction on  $\mathbb{Z}$  is neither associative nor commutative.

### EXAMPLE 5.4.4

Let  $M_2(\mathbb{R})$  be the set of all  $2 \times 2$  matrices over  $\mathbb{R}$ , i.e.,

$$M_2(\mathbb{R}) = \left\{ \begin{bmatrix} a & b \\ c & d \end{bmatrix} \mid a, b, c, d \in \mathbb{R} \right\}.$$

Let  $+$  denote the usual addition of matrices and  $\cdot$  denote the usual multiplication of matrices. Because addition (multiplication) of  $2 \times 2$  matrices over  $\mathbb{R}$  is a  $2 \times 2$  matrix over  $\mathbb{R}$ , it follows that  $+$  ( $\cdot$ ) is a binary operation on  $M_2(\mathbb{R})$ . Note that  $+$  is both associative and commutative and  $\cdot$  is associative but not commutative. For example,

$$\begin{bmatrix} 1 & 2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \neq \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 0 & 0 \end{bmatrix},$$

so multiplication is not commutative.

### EXAMPLE 5.4.5

Let  $M_2(B)$  be the set of all  $2 \times 2$  matrices over the two elements Boolean algebra  $B = \{0, 1\}$ , i.e.,

$$M_2(B) = \left\{ \begin{bmatrix} a & b \\ c & d \end{bmatrix} \mid a, b, c, d \in B \right\}.$$

Let  $\vee$  denote the Boolean addition of matrices and  $\odot$  denote the Boolean multiplication of matrices. Because Boolean addition and Boolean multiplication of

$2 \times 2$  matrices over  $B$  is a  $2 \times 2$  matrix over  $B$ , it follows that  $\vee$  and  $\odot$  are binary operations on  $M_2(B)$ . Note that  $\vee$  is both associative and commutative and  $\odot$  is associative but not commutative. For example,

$$\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \odot \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \neq \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix},$$

so  $\odot$  is not commutative.

#### EXAMPLE 5.4.6

Let  $A$  be a nonempty set and let  $S$  be the set of all functions from  $A$  into  $A$ . Then,  $\circ$ , the composition of functions, is a binary operation on  $S$ . As shown in Example 5.1.25,  $\circ$  need not be commutative. However, by Theorem 5.1.29,  $\circ$  is associative.

A convenient way to define a binary operation on a finite set  $S$  is by means of an operation table called a **Cayley multiplication table**, or **Cayley table**. These tables were introduced in 1854 by Arthur Cayley.

The following example shows how to define a binary operation using the Cayley table.

Let  $S = \{a, b, c, d\}$ . Define the binary operation  $*$  on  $S$  by the following table.

*	a	b	c	d
a	b	d	c	b
b	c	a	b	d
c	c	c	b	d
d	a	b	d	c

Let  $a, b \in S$ . To determine the element of  $S$  assigned to  $a * b$ , we look at the intersection of the row labeled by  $a$  and the column headed by  $b$ . We see that the element at this position is  $d$ , so we have  $a * b = d$ . Similarly,  $b * a = c$ .

We assume in a Cayley table that the elements of the set are listed across the top of the table in the same order they are listed at the left.

It can be verified that a binary operation defined by a Cayley table is commutative if and only if the entries in the table are symmetric with respect to the diagonal that starts at upper left corner of the table and terminates at the lower right corner. The operation defined by the preceding Cayley table is not commutative because the table is not symmetric with respect to the diagonal.

---

**DEFINITION 5.4.7** ▶ A nonempty set  $S$  together with a finite number of binary operations is called a **mathematical system**. We denote a mathematical system by  $(S, *_1, *_2, *_3, \dots, *_n)$ , where  $S$  is a nonempty set and  $*_1, *_2, *_3, \dots, *_n$  are binary operations on  $S$ .

The set  $\mathbb{Z}$  of all integers with the usual operations of addition, multiplication, and subtraction is a very common mathematical system.

---

**DEFINITION 5.4.8** ▶ A mathematical system  $(S, *)$  with only one binary operation is called a **groupoid**.

---

**DEFINITION 5.4.9** ▶ Let  $(S, *)$  be a mathematical system. An element  $e \in S$  is called an **identity** of  $(S, *)$  if for all  $x \in S$ ,

$$e * x = x = x * e.$$

**EXAMPLE 5.4.10**

Let  $S = \{e, a, b\}$ . Define  $*$  on  $S$  by the following multiplication table.

$*$	$e$	$a$	$b$
$e$	$e$	$a$	$b$
$a$	$a$	$a$	$a$
$b$	$b$	$a$	$a$

We note that  $e * a = a = a * e$ ,  $e * b = b = b * e$ , and  $e * e = e = e * e$ . Thus,  $e$  is an identity of  $(S, *)$ .

**EXAMPLE 5.4.11**

- (i) In Example 5.4.6,  $i_A$  is an identity element of  $(S, \circ)$ .
- (ii) In Example 5.4.4,  $\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$  is an identity element for the mathematical system  $(M_2(\mathbb{R}), +)$  and  $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$  is an identity element for the mathematical system  $(M_2(\mathbb{R}), \cdot)$ .

**Theorem 5.4.12:** An identity element (if it exists) of a mathematical system  $(S, *)$  is unique.

**Proof:** Let  $e, f$  be identities of  $(S, *)$ . Because  $e$  is an identity,  $e * a = a$  for all  $a \in S$ . Substituting  $f$  for  $a$ , we get

$$e * f = f. \quad (5.15)$$

Now  $f$  is an identity, so  $a * f = a$  for all  $a \in S$ . Substituting  $e$  for  $a$ , we get

$$e * f = e. \quad (5.16)$$

From (5.15) and (5.16), we get  $e = f$ . Hence, an identity element (if it exists) is unique. ■

---

**DEFINITION 5.4.13** ▶ A mathematical system  $(S, *)$  with associative binary operation  $*$  is called a **semigroup**.

From the definition, we find that a semigroup is a groupoid  $(S, *)$  such that the binary operation  $*$  is associative. We often use the expression “ $S$  is a semigroup with respect to  $*$ ” to mean that  $(S, *)$  is a semigroup; sometimes this may be further abbreviated as “ $S$  is a semigroup.”

---

**DEFINITION 5.4.14** ▶ A semigroup  $(S, *)$  is called *commutative* if  $*$  is a commutative binary operation; i.e., for all  $x, y \in S$ ,  $x * y = y * x$ .

---

**DEFINITION 5.4.15** ▶ A semigroup  $(S, *)$  is called a **monoid** if  $S$  contains an element  $e$ , called an *identity* of  $(S, *)$ , such that for all  $x \in S$ ,

$$e * x = x = x * e.$$

From Theorem 5.4.12, it follows that a monoid contains one and only one identity element. We denote the identity element of a monoid by 1.

Some well-known examples of semigroups are  $(\mathbb{Z}, +)$ ,  $(\mathbb{Z}, \cdot)$ ,  $(\mathbb{Q}, +)$ ,  $(\mathbb{Q}, \cdot)$ ,  $(\mathbb{R}, +)$ , and  $(\mathbb{R}, \cdot)$ , where  $+$  denotes the usual addition and  $\cdot$  denotes the usual multiplication of numbers.

$(\mathbb{Z}, +)$  is a semigroup with identity element 0 because  $0 + a = a + 0 = a$ , for all  $a \in \mathbb{Z}$ . The identity element of the semigroup  $(\mathbb{Z}, \cdot)$  is 1 because  $1 \cdot a = a \cdot 1 = a$ , for all  $a \in \mathbb{Z}$ .

Notice that each of these semigroups contains infinite number of elements.

We now give an example of semigroups with a finite number of elements.

### EXAMPLE 5.4.16

Let  $X$  be a set with three elements. Then the number of elements in  $X \times X$  is 9.

Let  $\text{Rel}(X)$  be the set of all relations on  $X$ . Recall that a relation  $R$  on  $X$  is a subset of  $X \times X$  and a subset of  $X \times X$  is a relation on  $X$ .

Now the number of subsets of  $X \times X$  is  $2^9$ . Hence,  $\text{Rel}(X)$  is a set with  $2^9$  elements.

Because the composition of relations is a relation, it follows that  $\circ$  is a binary operation on  $\text{Rel}(X)$ . By Theorem 3.1.21, the composition of relations is associative. It now follows that  $(\text{Rel}(X), \circ)$  is a semigroup.

**DEFINITION 5.4.17** ▶ Let  $A$  be a nonempty set. Let  $T_A$  be the set of all functions on  $A$ , i.e.,

$$T_A = \{f \mid f : A \rightarrow A\}.$$

Because composition of functions is a function, the operation  $\circ$ , composition of functions, is a binary operation on  $T_A$ . By Theorem 5.1.29,  $\circ$  is associative. Hence,  $T_A$  is a semigroup with respect to the composition of functions. This semigroup is called the **transformation semigroup** on  $A$ .

**Theorem 5.4.18:** Let  $A$  be a nonempty set with more than two elements.

The transformation semigroup  $T_A$  on  $A$  is a noncommutative monoid.

**Proof:** Let  $i_A$  denote an identity function on  $A$ . Then  $i_A(x) = x$  for all  $x \in A$ . Let  $f : A \rightarrow A$  be any function on  $A$ . Now for all  $x \in A$ ,  $(i_A \circ f)(x) = i_A(f(x)) = f(x)$ . This implies that  $i_A \circ f = f$ . Similarly,  $f \circ i_A = f$ . Hence,  $i_A$  is the identity element of the transformation semigroup  $T_A$ .

Let  $a$ ,  $b$ , and  $c$  be distinct elements in  $A$ . Define  $f : A \rightarrow A$ ,  $g : A \rightarrow A$  by

$$\begin{aligned} f(a) &= b, \\ f(b) &= c, \\ f(x) &= x, \quad \text{for all } x \in A - \{a, b\}, \end{aligned}$$

and

$$\begin{aligned} g(a) &= c, \\ g(b) &= a, \\ g(x) &= x, \quad \text{for all } x \in A - \{a, b\}. \end{aligned}$$

Then

$$(f \circ g)(a) = f(g(a)) = f(c) = c$$

and

$$(g \circ f)(a) = g(f(a)) = g(b) = a.$$

Hence,  $f \circ g \neq g \circ f$ . Therefore,  $T_A$  is a noncommutative monoid. ■

**EXAMPLE 5.4.19**

Let  $A$  be a set with three elements. Then  $T_A$  contains  $3^3 = 27$  elements and the transformation semigroup  $T_A$  on  $A$  is a noncommutative finite monoid.

**DEFINITION 5.4.20**

► Let  $(S, *)$  be a semigroup. An element  $a$  of  $S$  is called an **idempotent** element if  $a * a = a$ . If every element of a semigroup is idempotent, then it is called an **idempotent semigroup**, or a **band**.

Consider the transformation semigroup  $T_A$  on  $A = \{1, 2, 3\}$ . Let  $f : A \rightarrow A$  be defined by  $f(x) = 2$  for all  $x \in A$ . Then  $f$  is a constant function and  $(f \circ f)(x) = f(f(x)) = f(2) = 2 = f(x)$  for all  $x \in A$ . Hence,  $f \circ f = f$ , which implies that  $f$  is an idempotent in the transformation semigroup  $T_A$ . Now consider the function  $g : A \rightarrow A$  defined by

$$g : 1 \mapsto 2, 2 \mapsto 3, 3 \mapsto 1$$

Now  $(g \circ g)(1) = g(g(1)) = g(2) = 3 \neq g(1)$ . This implies that  $g \circ g \neq g$ , which shows that  $g$  is not an idempotent in the transformation semigroup  $T_A$ . Therefore, the transformation semigroup  $T_A$  is not a band.

Let  $S$  be a nonempty set and  $\mathcal{P}(S)$  denote the set of all subsets of  $S$ . Define a binary operation  $*$  on  $S$  by  $A * B = A \cap B$  for all  $A, B \in \mathcal{P}(S)$ . Then  $(\mathcal{P}(S), *)$  is a semigroup. In this semigroup,  $A * A = A \cap A = A$  for all  $A \in \mathcal{P}(S)$ . Hence, this semigroup is an idempotent semigroup. Also  $A * B = A \cap B = B \cap A = B * A$  for all  $A, B \in \mathcal{P}(S)$  implies that this is a commutative semigroup.

**EXAMPLE 5.4.21**

Let  $(L, \leq)$  be a lattice. Define the binary operation  $*$  on  $L$  by  $a * b = a \vee b$  for all  $a, b \in L$ . Because  $(L, \leq)$  is a lattice, the lub $\{a, b\} = a \vee b$  exists in  $L$  and  $(a \vee b) \vee c = a \vee (b \vee c)$  for all  $a, b, c \in L$ . Thus, the binary operation  $*$  is associative. Also  $a * b = a \vee b = b \vee a = b * a$  and  $a * a = a \vee a = a$ . Hence, we find that  $(L, *)$  is a commutative idempotent semigroup.

Let  $A$  be an alphabet. Let  $A^+$  be the set of all nonempty strings on the alphabet  $A$ . If  $u, v \in A^+$ , i.e.,  $u$  and  $v$  are two nonempty strings on  $A$ , then the concatenation of  $u$  and  $v$ ,  $uv \in A^+$ . Thus, the concatenation of strings is a binary operation on  $A^+$ . Also, by Theorem 5.3.23(ii), the concatenation operation is associative. Hence,  $A^+$  becomes a semigroup with respect to the concatenation operation. This semigroup is called a **free semigroup generated by  $A$** .

Let  $A^*$  denote the set of all words including empty word  $\lambda$ . By Theorem 5.3.23(i), for all  $s \in A^*$ ,  $\lambda s = s = s\lambda$ . Thus,  $A^*$  becomes a monoid with identity  $1 = \lambda$ . This monoid is called a **free monoid generated by  $A$** . If  $A$  contains more than one element, then we find that  $A^*$  is a noncommutative monoid.

## WORKED-OUT EXERCISES

**Exercise 1:** Which of the following are associative binary operations?

- $(\mathbb{Z}, *)$ , where  $x * y = (x + y) - (x \cdot y)$  for all  $x, y \in \mathbb{Z}$ .
- $(\mathbb{Z}, *)$ , where  $x * y = \max(x, y)$  for all  $x, y \in \mathbb{Z}$ .
- $(\mathbb{R}, *)$ , where  $x * y = |x + y|$  for all  $x, y \in \mathbb{R}$ .

**Solution:**

- (a) Let  $x, y, z \in \mathbb{Z}$

$$\begin{aligned} (x * y) * z &= ((x + y) - (x \cdot y)) * z \\ &= (x + y) - (x \cdot y) + z - ((x + y) - (x \cdot y)) \cdot z \\ &= x + y + z - x \cdot y - x \cdot z - y \cdot z + x \cdot y \cdot z. \end{aligned}$$

Similarly,

$$x * (y * z) = x + y + z - x \cdot y - x \cdot z - y \cdot z + x \cdot y \cdot z.$$

Thus,  $(x * y) * z = x * (y * z)$  for all  $x, y \in \mathbb{Z}$ . Hence,  $*$  is associative.

- (b) Let  $x, y, z \in \mathbb{Z}$ . Then

$$\begin{aligned} (x * y) * z &= \max(x, y) * z \\ &= \max(\max(x, y), z) \\ &= \max(x, y, z) \\ &= \max(x, \max(y, z)) \\ &= x * \max(y, z) \\ &= x * (y * z). \end{aligned}$$

Thus,  $*$  is associative.

- (c)  $(2 * (-3)) * 6 = |2 + (-3)| * 6 = 1 * 6 = |1 + 6| = 7$   
and  $2 * ((-3) * 6) = 2 * (|-3 + 6|) = 2 * 3 = |2 + 3| = 5$ . Hence,  $(2 * (-3)) * 6 \neq 2 * ((-3) * 6)$  and so  $*$  is not associative.

**Exercise 2:** Which of the following binary operations  $*$  on  $S = \{a, b, c\}$  are commutative?

*	a	b	c
a	c	b	c
b	b	a	a
c	c	a	b

*	a	b	c
a	c	b	a
b	b	a	a
c	c	b	b

**Solution:**

- (a) From the table we find that  $a * b = b = b * a$ ,  $a * c = c = c * a$ , and  $b * c = a = c * b$ . Hence, the operation  $*$  is commutative.  
(b)  $c * a = c$ ,  $a * c = a$ . Thus,  $c * a \neq a * c$ . Hence, the operation  $*$  is not commutative.

**Exercise 3:** Which of the following groupoids are semigroups?

- $(\mathbb{Z}, *)$ , where  $x * y = x$  for all  $x, y \in \mathbb{Z}$ .
- $(\mathbb{Z}, *)$ , where  $x * y = x + y + 2$  for all  $x, y \in \mathbb{Z}$ .
- $(\mathbb{N}, *)$ , where  $x * y = x^y$  for all  $x, y \in \mathbb{N}$ .

**Solution:**

- (a) Let  $x, y, z \in \mathbb{Z}$ . Then

$$(x * y) * z = x * z = x$$

and

$$x * (y * z) = x * y = x.$$

Thus,  $(x * y) * z = x * (y * z)$  for all  $x, y, z \in \mathbb{Z}$ . Hence,  $(\mathbb{Z}, *)$  is a semigroup.

- (b) Let  $x, y, z \in \mathbb{Z}$ . Then

$$\begin{aligned} (x * y) * z &= (x + y + 2) * z \\ &= (x + y + 2) + z + 2 = x + y + z + 4 \end{aligned}$$

and

$$\begin{aligned} x * (y * z) &= x * (y + z + 2) \\ &= x + (y + z + 2) + 2 = x + y + z + 4. \end{aligned}$$

Thus,  $(x * y) * z = x * (y * z)$  for all  $x, y, z \in \mathbb{Z}$ . Hence,  $(\mathbb{Z}, *)$  is a semigroup.

- (c)  $(2 * 3) * 4 = 2^3 * 4 = (2^3)^4 = 2^{12}$  and  $2 * (3 * 4) = 2 * 3^4 = 2^{3^4} = 2^{81}$ . Hence,  $(2 * 3) * 4 \neq 2 * (3 * 4)$ . Therefore,  $(\mathbb{N}, *)$  is not a semigroup.

**Exercise 4:** Which of the following groupoids are monoids?

- $(\mathbb{Z}, *)$ , where  $x * y = y$  for all  $x, y \in \mathbb{Z}$ .
- $(\mathbb{Z}, *)$ , where  $x * y = x + y - 3$  for all  $x, y \in \mathbb{Z}$ .

**Solution:**

- (a) Let  $x, y, z \in \mathbb{Z}$ . Then

$$(x * y) * z = y * z = z$$

and

$$x * (y * z) = x * z = z.$$

Thus,  $(x * y) * z = x * (y * z)$  for all  $x, y, z \in \mathbb{Z}$ . Hence,  $(\mathbb{Z}, *)$  is a semigroup.

Suppose  $(\mathbb{Z}, *)$  contains an identity  $e$ . Let  $x \in \mathbb{Z}$ . Then  $e * x = x = x * e$ . By the definition of  $*$ ,  $x * e = e$ , so  $x = x * e = e$ . This shows that  $\mathbb{Z}$  contains only one element, which is not true. Hence,  $(\mathbb{Z}, *)$  is not a monoid.

- (b) Let  $x, y, z \in \mathbb{Z}$ . Then

$$\begin{aligned} (x * y) * z &= (x + y - 3) * z \\ &= (x + y - 3) + z - 3 = x + y + z - 6 \end{aligned}$$

and

$$\begin{aligned} x * (y * z) &= x * (y + z - 3) \\ &= x + (y + z - 3) - 3 = x + y + z - 6. \end{aligned}$$

Thus,  $(x * y) * z = x * (y * z)$  for all  $x, y, z \in \mathbb{Z}$ . Hence,  $(\mathbb{Z}, *)$  is a semigroup.

In this semigroup, for all  $x \in \mathbb{Z}$

$$3 * x = 3 + x - 3 = x$$

and

$$x * 3 = x + 3 - 3 = x.$$

Hence, 3 is the identity element.

**Exercise 5:** Let  $(S, *)$  be a commutative idempotent semigroup. Define a relation  $R$  on  $S$  by  $a R b$  if and only if  $a * b = a$ . Show that  $R$  is a partial order relation.

**Solution:** Reflexive: For all  $a \in S$ ,  $a * a = a$  (because every element is idempotent)  $\Rightarrow a R a$ . Therefore,  $R$  is reflexive.

Antisymmetric: Let  $a, b \in S$  such that  $a R b$  and  $b R a$ . Then  $a * b = a$  and  $b * a = b$ . But  $a * b = b * a$ . Thus,  $a = b$ . Hence,  $R$  is antisymmetric.

Transitive: Let  $a, b, c \in S$  be such that  $a R b$  and  $b R c$ . Then  $a * b = a$  and  $b * c = b$ . Thus,

$$a * c = (a * b) * c = a * (b * c) = a * b = a$$

implies that  $a R c$ . Hence,  $R$  is transitive. Consequently,  $R$  is a partial order.

**Exercise 6:** Let  $S = \{1, 2, 3\}$  and let  $T$  denote the set of following six functions from  $S \rightarrow S$ .

$$\begin{array}{lll} f_1 : 1 \rightarrow 1 & f_2 : 1 \rightarrow 2 & f_3 : 1 \rightarrow 3 \\ 2 \rightarrow 2 & 2 \rightarrow 1 & 2 \rightarrow 2 \\ 3 \rightarrow 3 & 3 \rightarrow 3 & 3 \rightarrow 1 \end{array}$$

$$\begin{array}{lll} f_4 : 1 \rightarrow 1 & f_5 : 1 \rightarrow 2 & f_6 : 1 \rightarrow 3 \\ 2 \rightarrow 3 & 2 \rightarrow 3 & 2 \rightarrow 1 \\ 3 \rightarrow 2 & 3 \rightarrow 1 & 3 \rightarrow 2 \end{array}$$

Write the Cayley table for the binary operation  $*$  defined on  $T$  by the composition of functions.

**Solution:** Let us compute  $f_2 \circ f_3$ .

$$\begin{aligned} f_2 \circ f_3(1) &= f_2(f_3(1)) = f_2(3) = 3 = f_6(1), \\ f_2 \circ f_3(2) &= f_2(f_3(2)) = f_2(2) = 1 = f_6(2), \\ f_2 \circ f_3(3) &= f_2(f_3(3)) = f_2(1) = 2 = f_6(3). \end{aligned}$$

This implies that  $f_2 \circ f_3 = f_6$ . In a similar way, we can compute the other compositions and obtain the following table.

$\circ$	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$
$f_1$	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$
$f_2$	$f_2$	$f_1$	$f_6$	$f_5$	$f_4$	$f_3$
$f_3$	$f_3$	$f_5$	$f_1$	$f_6$	$f_2$	$f_4$
$f_4$	$f_4$	$f_6$	$f_5$	$f_1$	$f_3$	$f_2$
$f_5$	$f_5$	$f_3$	$f_4$	$f_2$	$f_6$	$f_1$
$f_6$	$f_6$	$f_4$	$f_2$	$f_3$	$f_1$	$f_5$

## SECTION REVIEW

### Key Terms

binary operation	mathematical system	idempotent
closed under	groupoid	idempotent semigroup
associative	identity	band
commutative	semigroup	free semigroup generated by
Cayley multiplication table	monoid	free monoid generated by
Cayley table	transformation semigroup	

### Some Key Definitions

- Let  $S$  be a nonempty set. A binary operation on  $S$  is a function from  $S \times S$  into  $S$ .
- Let  $S$  be a nonempty set and let  $*$  be a binary operation on  $S$ . Then
  - $*$  is called associative if for all  $x, y, z \in S$ ,  $x * (y * z) = (x * y) * z$ .
  - $*$  is called commutative if for all  $x, y \in S$ ,  $x * y = y * x$ .

3. A nonempty set  $S$  together with a finite number of binary operations is called a mathematical system.
4. Let  $(S, *)$  be a mathematical system. An element  $e \in S$  is called an identity of  $(S, *)$  if for all  $x \in S$ ,  $e * x = x = x * e$ .
5. A mathematical system  $(S, *)$  with associative binary operation  $*$  is called a semigroup.
6. A semigroup  $(S, *)$  is called a monoid if  $S$  contains an identity element  $e$ .
7. Let  $A$  be an alphabet. Let  $A^+$  be the set of all nonempty strings on the alphabet  $A$ . Then  $A^+$  is a semigroup with respect to the concatenation operation. This semigroup is called a free semigroup generated by  $A$ .
8. Let  $A^*$  denote the set of all words on the set  $A$  including the empty word  $\lambda$ . Then,  $A^*$  is a monoid with identity  $1 = \lambda$ . This monoid is called a free monoid generated by  $A$ . If  $A$  contains more than one element, then  $A^*$  is a noncommutative monoid.

## Some Key Results

1. An identity element (if it exists) of a mathematical system  $(S, *)$  is unique.
2. Let  $A$  be a nonempty set with more than two elements. The transformation semigroup  $T_A$  on  $A$  is a noncommutative monoid.

## EXERCISES

1. Which of the following are associative binary operations?
  - a.  $(\mathbb{N}, *)$ , where  $x * y = 2xy$  for all  $x, y \in \mathbb{N}$ .
  - b.  $(\mathbb{Z}, *)$ , where  $x * y = x + y + 1$  for all  $x, y \in \mathbb{Z}$ .
  - c.  $(\mathbb{N}, *)$ , where  $x * y = \gcd(x, y)$  for all  $x, y \in \mathbb{N}$ .
  - d.  $(\mathbb{R}, *)$ , where  $x * y = \min(x, y)$  for all  $x, y \in \mathbb{R}$ .
  - e.  $(\mathbb{R}, *)$ , where  $x * y = |x| + |y|$  for all  $x, y \in \mathbb{R}$ .
2. Let  $*$  be the binary operation on  $S = \{a, b, c, d\}$  defined by the following Cayley table.

*	a	b	c	d
a	a	c	a	a
b	c	b	a	d
c	c	b	a	b
d	c	d	b	b

Compute the following.

- (i)  $(a * b) * (c * d)$
- (ii)  $((((b * b) * c) * c) * d)$
- (iii)  $(d * a) * ((b * a) * (a * b))$
3. Which of the following binary operations  $*$  on  $S = \{a, b, c\}$  are commutative?

*	a	b	c
a	a	d	a
b	a	b	a
c	c	d	a
d	a	d	b

*	a	b	c
a	c	b	a
b	b	a	c
c	a	c	b

4. Let  $S = \{8, 3, 5, 4\}$ . A binary operation  $*$  is defined on  $S$  by  $x * y = \max\{x, y\}$ . Write the Cayley table for this operation.
5. Let  $B = \{0, 1\}$ . Write the Cayley table for the transformation monoid  $T_B$ . Is  $T_B$  a commutative monoid?
6. Which of the following groupoids are semigroups?
  - a.  $(\mathbb{Q} - \{0\}, *)$ , where  $x * y = \frac{x}{y}$  for all  $x, y \in \mathbb{Q} - \{0\}$ .
  - b.  $(\mathbb{Z}, *)$ , where  $x * y = |x|y$  for all  $x, y \in \mathbb{Z}$ .
  - c.  $(\mathbb{Z}, *)$ , where  $x * y = 5$  for all  $x, y \in \mathbb{Z}$ .
7. In Exercise 6, which of the operations are commutative?
8. Let  $S = \{a, b, c\}$  and  $\mathcal{P}(S)$  denote the set of all subsets of  $S$ . Write the Cayley table for the binary operation  $*$  defined by  $A * B = A \cap B$  for all  $A, B \in \mathcal{P}(S)$ .

9. Let  $S = \{1, -1\}$ . Write the Cayley table for the binary operation  $*$  defined by the usual multiplication.
10. Find all idempotent elements of the transformation semigroup  $T_B$  on  $B = \{0, 1\}$ .

## ► PROGRAMMING EXERCISES

---

1. Write a program to determine whether a relation from a finite set into a finite set is a function. If the relation is a function, then the program determines if the function is one-one, onto, and a one-to-one correspondence.
2. Write a program to find the domain and range of a function.
3. Write a program to determine if the elements of a sequence are in increasing order.
4. Write a program to find the sum of the finite number of terms of a sequence.
5. Write a program to find the product of the finite number of terms of a sequence.
6. Write a program to implement the nonrecursive algorithm to convert a number from base 2 to base 10.
7. Write a program to implement the nonrecursive algorithm to convert a number from base 10 to base 2.
8. Write a program that takes as input a string and returns the string in the reverse order. For example, the string *aabcdab* is returned as *bdcbaa*.
9. Write a program to implement the string concatenation operation as well as to find the length of the string. Store the string in a character array and do not use any built-in function of the programming language to implement these operations.
10. Write a program to determine if a string is a substring in a string.

## Congruences

The objectives of this chapter are to:

- Learn the basic properties of congruences
- Explore how congruences are used in the divisibility test
- Explore the applications of congruences in ISBNs, UPCs, and credit cards
- Learn about linear congruences
- Explore how linear congruences are used in round-robin tournaments and the design of hash functions
- Learn the special congruence theorems
- Explore the applications of special congruence theorems in cryptography

In Example 3.1.32, we introduced the equivalence relation congruence modulo, a fixed positive integer. We remarked that this relation was first studied by Carl Friedrich Gauss, who published his monumental book on number theory, *Disquisitiones Arithmeticae*, in 1801. In *Disquisitiones*, Gauss summarized previous work in a systematic way and solved some of the most difficult outstanding questions. He introduced the notion of congruence of integers modulo an integer  $n$ , ( $a \equiv b \pmod{n}$ ), extensively studied  $\mathbb{Z}_n$ , the set of equivalence classes modulo  $n$ , and obtained many of its important properties. It is difficult to exaggerate the revolutionary impact of congruences on number theory. Moreover, it has become an indispensable tool in computer science.

In Chapter 3, we mentioned that the notion of congruence modulo has many applications in various products we use daily. Our objective here is twofold: to further study the properties of congruence and to discuss its applications in the construction of ISBNs, UPCs, credit cards, and more.

It is amazing that the notion of congruence introduced by Gauss 200 years ago continues to have a deep impact on modern mathematics and modern life. In fact, Gauss's study of congruence is often regarded as the beginning of modern algebra.

## 6.1 CONGRUENCES

In this section, we discuss some basic properties of congruences.

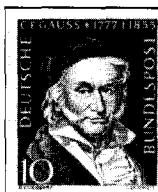
Let  $m$  be a (fixed) positive integer. In Example 3.1.32, we defined the relation  $R$  on  $\mathbb{Z}$  as follows: For all integers  $a$  and  $b$ ,  $a R b$  if and only if  $m$  divides  $a - b$ . It was shown that  $R$  is an equivalence relation.

Suppose  $m = 6$ ,  $a = 32$ , and  $b = 20$ . Now  $32 - 20 = 12 = 2 \cdot 6$ . This implies that 6 divides  $32 - 20$ , so  $32 R 20$ . Notice that  $32 = 5 \cdot 6 + 2$  and  $20 = 3 \cdot 6 + 2$ . That is, when 32 is divided by 6, the remainder is 2. Similarly, when 20 is divided by 6, the remainder is also 2. Let us generalize this observation in the following theorem.

**Theorem 6.1.1:** Let  $m$  be a positive integer and  $a$  and  $b$  be any integers.

Then  $m$  divides  $a - b$  if and only if the remainder of  $a$  on division by  $m$  and the remainder of  $b$  on division by  $m$  are the same.

**Proof:** First suppose that  $m$  divides  $a - b$ . Suppose that  $r_1$  is the remainder of  $a$  on division by  $m$  and  $r_2$  is the remainder of  $b$  on division by  $m$ . Then  $r_1, r_2 \in \mathbb{Z}$  and  $0 \leq r_1 < m$ ,  $0 \leq r_2 < m$ . There exist integers  $q_1$  and  $q_2$  such that  $a = q_1 m + r_1$



**Carl Friedrich Gauss**  
(1777–1855)

Gauss was born in Brunswick, Germany, and is considered to be one of the last mathematicians to know everything in his subject. Gauss's genius was revealed at a very early age. He was able to do long calculations in his head. He rediscovered the law of quadratic reciprocity, related the arithmetic-geometric mean to infinite series expansion, and conjectured the Prime Number Theorem. Before the age of 20, he showed that a regular polygon of 17 sides was constructible with

### Historical Notes

ruler and compass, an unsolved problem since Greek times. At the age of 20, he published the first proof of the fundamental theorem of algebra. He completed his Ph.D. at the University of Helmstedt when he was 22.

In *Disquisitiones Arithmeticae*, published when Gauss was 24, he laid the foundations of algebraic number theory. Besides being a mathematician, he was also a physicist and an astronomer. In January 1801, a new planet was briefly observed, but astronomers were unable to locate it again. Gauss calculated the position of the planet by using a more accurate orbit theory than

the usual circular approximation. At the end of the year, the planet was discovered at the precise location he had predicted. The methods he developed are still in use. They include the theory of least squares.

He was appointed director of the observatory at Göttingen and remained there for 40 years. Gauss disliked teaching and preferred his job at the observatory. Although he usually rejected students who sought his guidance those few he did accept included Dedekind, Dirichlet, Eisenstein, Riemann, and Kummer, all of whom became eminent mathematicians themselves.

and  $b = q_2m + r_2$ . (Actually,  $q_1$  and  $q_2$  are the quotients and  $r_1$  and  $r_2$  are the remainders, when  $a$  and  $b$  are divided by  $m$ , respectively.). Now

$$a - b = (q_1 - q_2)m + (r_1 - r_2).$$

This implies that

$$r_1 - r_2 = (a - b) - (q_1 - q_2)m.$$

Now  $m | (a - b)$  and  $m | (q_1 - q_2)m$ , so  $m | (r_1 - r_2)$ . However, both  $r_1$  and  $r_2$  are nonnegative and less than  $m$ . Therefore,  $m | (r_1 - r_2)$  implies  $r_1 - r_2 = 0$  so  $r_1 = r_2$ . Hence, the remainders are the same.

Conversely, suppose that the remainder of  $a$  on division by  $m$  and the remainder of  $b$  on division by  $m$  are the same. By the division algorithm, there exist  $q_1, q_2, r_1, r_2 \in \mathbb{Z}$  such that  $a = q_1m + r_1$  and  $b = q_2m + r_2$ . Then  $r_1$  and  $r_2$  are the remainders, when  $a$  and  $b$  are divided by  $m$ , respectively. Therefore, by the hypothesis,  $r_1 = r_2$ . This implies that

$$a - b = (q_1 - q_2)m + (r_1 - r_2) = (q_1 - q_2)m.$$

Hence,  $a - b$  is divisible by  $m$ . ■

It follows from Theorem 6.1.1 that the relation  $R$  of Example 3.1.32 can be defined by using the remainders; i.e.,  $a R b$  if and only if the remainder of  $a$  on division by  $m$  and the remainder of  $b$  on division by  $m$  are the same.

Let us again consider the integers 32, 20, and 6. Now  $32 \neq 20$ , but their remainders when divided by 6 are the same. In this case, we say that 32 is congruent to 20 modulo 6.

In general, let  $m$  be a positive integer. If  $a$  and  $b$  are integers, then by the division algorithm there exist integers  $q_1, q_2, r_1, r_2$  such that  $a = q_1m + r_1$  and  $b = q_2m + r_2$ , where  $0 \leq r_1 < m$  and  $0 \leq r_2 < m$ . In Chapter 2, we introduced the operator  $\text{mod}$ . Notice that  $r_1 = a \text{ mod } m$  and  $r_2 = b \text{ mod } m$ . If  $r_1 = r_2$ , i.e.,  $a \text{ mod } m = b \text{ mod } m$ , then we say that  $a$  is congruent to  $b$  modulo  $m$ . Notice that the integers  $a$  and  $b$  may not be equal. More formally, we have the following definition.

---

**DEFINITION 6.1.2** ▶ Let  $a$  and  $b$  be two integers and  $m$  be a positive integer. Then  $a$  is said to be **congruent to  $b$  modulo  $m$**  if  $a$  and  $b$  have the same remainder when  $m$  divides both  $a$  and  $b$ . If  $a$  is congruent to  $b$  modulo  $m$ , we write  $a \equiv b \pmod{m}$ .

If  $a$  is not congruent to  $b$  modulo  $m$ , then we write  $a \not\equiv b \pmod{m}$ .

---

**REMARK 6.1.3** ▶ Let  $m$  be a positive integer and  $a$  and  $b$  be any integers. Then by Theorem 6.1.1 and Definition 6.1.2, it follows that  $a \equiv b \pmod{m}$  if and only if  $m$  divides  $a - b$ . From this it follows that the relation  $R$  of Example 3.1.32 is nothing but congruence modulo  $m$ . In symbols, we can write that  $a R b$  if and only if  $a \equiv b \pmod{m}$ .

### EXAMPLE 6.1.4

Let  $a = 19$ ,  $b = 24$ , and  $m = 5$ . Then  $19 = 3 \cdot 5 + 4$  and  $24 = 4 \cdot 5 + 4$ . That is, the remainder of 19 on division by 5 is 4 and the remainder of 24 on division by 5 is 4. Thus,  $19 \equiv 24 \pmod{5}$ . We can also use the argument that  $19 - 24 = -5 = (-1)5$ , so 5 divides  $19 - 24$ . Therefore,  $19 \equiv 24 \pmod{5}$ .

However,  $19 - 24 = -5 \neq 6k$  for any integer  $k$ . Thus,  $19 \not\equiv 24 \pmod{6}$ .

Now,  $19 - 23 = -4 \neq 5k$  for any integer  $k$ . Thus,  $19 \not\equiv 23 \pmod{5}$ .

**EXAMPLE 6.1.5**

5 divides  $17 - 2$ . Hence,  $17 \equiv 2 \pmod{5}$ . But 5 does not divide  $11 - 3$ . Hence,  $11 \not\equiv 3 \pmod{5}$ .

**EXAMPLE 6.1.6**

Let  $m = 7$  and consider the integer 38. Now  $38 = 5 \cdot 7 + 3$ . This implies that  $38 - 3 = 5 \cdot 7$ , so 7 divides  $38 - 3$ . Hence,  $38 \equiv 3 \pmod{7}$ . Notice that 3 is the remainder when 38 is divided by 7.

Example 6.1.6 motivates the following result, which can be proved using the division algorithm.

**Theorem 6.1.7:** Let  $m$  be a positive integer and  $a$  be any integer. Let  $r$  be the remainder of  $a$  on division by  $m$ . Then  $a \equiv r \pmod{m}$ .

We quite often use the following result.

**Theorem 6.1.8:** Let  $a$  and  $b$  be two integers and  $m$  be a positive integer such that  $a \equiv b \pmod{m}$ . Then  $m$  divides  $a$  if and only if  $m$  divides  $b$ .

**Proof:** Because  $a \equiv b \pmod{m}$ , we have  $m$  divides  $a - b$ . Then  $a - b = mk$  for some integer  $k$ . Suppose  $m$  divides  $a$ . Then  $a = mt$  for some integer  $t$ . Now

$$m(t - k) = mt - mk = a - (a - b) = a - a + b = b.$$

This implies that  $m$  divides  $b$ . In a similar manner, we can show that if  $m$  divides  $b$ , then  $m$  divides  $a$ . ■

The following theorem lists the properties of congruence modulo  $m$ , proved in Example 3.1.32.

**Theorem 6.1.9:** Let  $a$ ,  $b$ , and  $c$  be integers and  $m$  be a positive integer. Then:

- (i)  $a \equiv a \pmod{m}$ .
- (ii) If  $a \equiv b \pmod{m}$ , then  $b \equiv a \pmod{m}$ .
- (iii) If  $a \equiv b \pmod{m}$  and  $b \equiv c \pmod{m}$ , then  $a \equiv c \pmod{m}$ .
- (iv) Congruence modulo  $m$  is an equivalent relation on  $\mathbb{Z}$ .

Now that congruence modulo  $m$  is an equivalence relation on  $\mathbb{Z}$ , where  $m$  is a positive integer, we can consider its equivalence classes.

**DEFINITION 6.1.10** ▶ Let  $m$  be a positive integer and  $a$  be an integer. Then the set

$$[a] := \{b \in \mathbb{Z} \mid b \equiv a \pmod{m}\}$$

is called the **congruence class modulo  $m$**  of the integer  $a$ .

**EXAMPLE 6.1.11**

Let  $m = 5$  and  $a = 3$ . Then the congruence class modulo 5 of 3 is the set

$$\begin{aligned}[3] &= \{b \in \mathbb{Z} \mid b \equiv 3 \pmod{5}\} \\ &= \{b \in \mathbb{Z} \mid 5 \text{ divides } b - 3\} \\ &= \{b \in \mathbb{Z} \mid b - 3 = 5k \text{ for some integer } k\} \\ &= \{b \in \mathbb{Z} \mid b = 3 + 5k \text{ for some integer } k\} \\ &= \{\dots, -12, -7, -2, 3, 8, 13, 18, \dots\}.\end{aligned}$$

The following theorem follows from Theorem 3.1.35.

**Theorem 6.1.12:** Let  $m$  be a positive integer. The congruence class modulo  $m$  satisfies the following:

- (i)  $[a] \neq \emptyset$  for any integers  $a$ .
- (ii) If  $b \in [a]$ , then  $[b] = [a]$  for all integers  $a, b$ .
- (iii) For all integers  $a, b$  either  $[a] \cap [b] = \emptyset$  or  $[b] = [a]$ .

Consider the positive integer 5 and the congruence classes modulo 5. Now  $8 \equiv 3 \pmod{5}$ . Therefore,  $8 \in [3]$ . This implies that  $[8] = [3]$ . Likewise  $[1] = [6]$ ,  $[4] = [9]$ , and so on.

---

**DEFINITION 6.1.13** ► For any positive integer  $m$ , let  $\mathbb{Z}_m$  denote the set of all congruence classes modulo  $m$ .

The following theorem shows that  $\mathbb{Z}_m$  is a finite set.

**Theorem 6.1.14:** The number of elements in  $\mathbb{Z}_m$  is finite. In fact,

$$\mathbb{Z}_m = \{[0], [1], [2], \dots, [m-1]\}$$

and  $|\mathbb{Z}_m| = m$ .

**Proof:** To prove the theorem, we do the following two things: (1) First we show that for any integer  $n$ , the class  $[n]$  is equal to one of the classes  $[0], [1], [2], \dots$ , or  $[m-1]$ . This will show that  $\mathbb{Z}_m = \{[0], [1], [2], \dots, [m-1]\}$ . (2) To show  $|\mathbb{Z}_m| = m$ , we show that for any integers  $k$  and  $t$  such that  $0 \leq k < m$ ,  $0 \leq t < m$ ,  $[k] = [t]$  if and only if  $k = t$ . This will show that the classes  $[0], [1], [2], \dots, [m-1]$  are distinct.

Let us prove (1).

Let  $n$  be any integer. By the division algorithm, there exist integers  $q$  and  $r$  such that  $n = qm + r$ , where  $0 \leq r \leq m-1$ . Then by Theorem 6.1.7,  $n \equiv r \pmod{m}$ . This implies that  $[n] = [r]$ , where  $0 \leq r \leq m-1$ . From this it follows that  $\mathbb{Z}_m = \{[0], [1], [2], \dots, [m-1]\}$ .

Let us now prove (2). Let  $k$  and  $t$  be integers such that  $0 \leq k < m$ ,  $0 \leq t < m$ . First note that if  $k = t$ , then  $[k] = [t]$ .

Suppose that  $[k] = [t]$  and  $k \neq t$ . Then  $k < t$  or  $k > t$ . To be specific, suppose  $k > t$ .

Now  $k \in [k] = [t]$ , so  $k \equiv t \pmod{m}$ . This implies that  $m$  divides  $k - t$ . Because  $0 \leq t < k \leq m - 1$  and  $m$  divides  $k - t$ , it follows that  $k - t = 0$ , i.e.,  $k = t$ . This is a contradiction to our assumption. Therefore,  $k = t$ . This proves (2).

We can now conclude that  $[0], [1], [2], \dots, [m - 1]$  are the  $m$  distinct congruence classes and any congruence class  $[n]$  equals one of these. Hence,  $|\mathbb{Z}_m| = m$ .

### EXAMPLE 6.1.15

Consider the integer 6 and the set  $\mathbb{Z}_6$  of all congruence classes modulo 6. By Theorem 6.1.14,

$$\mathbb{Z}_6 = \{[0], [1], [2], [3], [4], [5]\}.$$

Recall that each congruence class  $[k]$  is a subset of  $\mathbb{Z}$ . Also, it follows that the union of all the congruence classes modulo  $m$  is the set  $\mathbb{Z}$ . Corresponding to the integer 6,

$$\begin{aligned}[0] &= \{\dots, -18, -12, -6, 0, 6, 12, 18, 24, \dots\} \\ [1] &= \{\dots, -17, -11, -5, 1, 7, 13, 19, 25, \dots\} \\ [2] &= \{\dots, -16, -10, -4, 2, 8, 14, 20, 26, \dots\} \\ [3] &= \{\dots, -15, -9, -3, 3, 9, 15, 21, 27, \dots\} \\ [4] &= \{\dots, -14, -8, -2, 4, 10, 16, 22, 28, \dots\} \\ [5] &= \{\dots, -13, -7, -1, 5, 11, 17, 23, 29, \dots\}, \end{aligned}$$

and  $\mathbb{Z} = [0] \cup [1] \cup [2] \cup [3] \cup [4] \cup [5]$ .

The following theorem shows that we can add and multiply congruences in a similar way to those we use for equality.

**Theorem 6.1.16:** Let  $a, b, c$  and  $d$  be integers and  $m$  be a positive integer.

- (i) If  $a \equiv b \pmod{m}$  and  $c \equiv d \pmod{m}$ , then  $a + c \equiv (b + d) \pmod{m}$ .
- (ii) If  $a \equiv b \pmod{m}$  and  $c \equiv d \pmod{m}$ , then  $ac \equiv bd \pmod{m}$ .
- (iii) If  $a \equiv b \pmod{m}$ , then  $a^n \equiv b^n \pmod{m}$  for any positive integer  $n$ .

### Proof:

- (i) Suppose  $a \equiv b \pmod{m}$  and  $c \equiv d \pmod{m}$ . Then  $m$  divides  $a - b$  and  $c - d$ . Hence, there exist integers  $k$  and  $t$  such that  $a - b = mk$  and  $c - d = mt$ . Then

$$(a - b) + (c - d) = mk + mt = m(k + t).$$

This shows that  $(a + c) - (b + d) = m(k + t)$ , so  $m$  divides  $(a + c) - (b + d)$ . Hence, it follows that  $a + c \equiv (b + d) \pmod{m}$ .

- (ii) Suppose  $a \equiv b \pmod{m}$  and  $c \equiv d \pmod{m}$ . Then  $m$  divides  $a - b$  and  $c - d$ . Hence, there exist integers  $k$  and  $t$  such that  $a - b = mk$  and  $c - d = mt$ . Then  $ac - bc = mkc$  and  $bc - bd = mtb$ . This implies  $(ac - bc) + (bc - bd) = mkc + mtb$ , i.e.,  $ac - bd = m(kc + tb)$ . Therefore,  $m$  divides  $ac - bd$ , so  $ac \equiv bd \pmod{m}$ .
- (iii) We prove this result by induction on  $n$ .

*Basis step:* Suppose  $n = 1$ . Because  $a \equiv b \pmod{m}$ , it follows that  $a^n \equiv b^n \pmod{m}$  for  $n = 1$ .

*Inductive hypothesis:* Suppose  $a^k \equiv b^k \pmod{m}$  for some positive integer  $k \geq 1$ .

*Inductive step:* We want to show that  $a^{k+1} \equiv b^{k+1} \pmod{m}$ .

By the inductive hypothesis, we have  $a^k \equiv b^k \pmod{m}$ . Also, by Theorem 6.1.9(i),  $a \equiv a \pmod{m}$ . Thus, by part (ii) we find that

$$a^{k+1} = aa^k \equiv ab^k \pmod{m}.$$

Again,  $a \equiv b \pmod{m}$  and  $a^k \equiv b^k \pmod{m}$  imply that  $ab^k \equiv b^{k+1} \pmod{m}$ . Now  $a^{k+1} \equiv ab^k \pmod{m}$  and  $ab^k \equiv b^{k+1} \pmod{m}$  imply that  $a^{k+1} \equiv b^{k+1} \pmod{m}$  by Theorem 6.1.9(iii). Thus, the result is true for  $k+1$ .

Hence, by induction  $a^n \equiv b^n \pmod{m}$  for any positive integer  $n$ . ■

**Corollary 6.1.17:** Let  $a, b$ , and  $c$  be integers and  $m$  be a positive integer.

- (i) If  $a \equiv b \pmod{m}$ , then  $a+c \equiv b+c \pmod{m}$  for any integer  $c$ .
- (ii) If  $a \equiv b \pmod{m}$ , then  $ac \equiv bc \pmod{m}$  for any integer  $c$ .
- (iii) If  $a \equiv b \pmod{m}$ , then  $a-c \equiv b-c \pmod{m}$  for any integer  $c$ .

### Proof:

- (i) Suppose  $a \equiv b \pmod{m}$  and  $c \equiv c \pmod{m}$ . Then from Theorem 6.1.16(i) that  $a+c \equiv b+c \pmod{m}$ .
- (ii) This follows from Theorem 6.1.16(ii).
- (iii) Suppose  $a \equiv b \pmod{m}$ . Then from part (i),

$$a+(-c) \equiv b+(-c) \pmod{m},$$

i.e.,  $a-c \equiv b-c \pmod{m}$ . ■

From Theorem 6.1.16, we find that we can perform addition, subtraction, or multiplication to both sides of a congruence by an integer. Let us now see some elegant and important applications of this theorem as well as the use of congruence.

### EXAMPLE 6.1.18

Suppose we are asked to find the remainder when  $9^{342}$  is divided by 10. Notice that  $9^{342}$  is a very big number, so it is not easy to expand this number and then do the division. So how do we determine the remainder? Properties of congruences come to our rescue. From the congruence property, we know  $9^2 \equiv 1 \pmod{10}$ . Then by Theorem 6.1.16,  $(9^2)^{171} \equiv (1)^{171} \pmod{10}$ , i.e.,  $9^{342} \equiv 1 \pmod{10}$ . Therefore, the remainder is 1.

### EXAMPLE 6.1.19

Consider the integers  $7^{348}$  and  $25^{605}$ . We would like to know the remainder of  $7^{348} + 25^{605}$  when it is divided by 8. Again note that  $7^{348} + 25^{605}$  is a very big number. By the division algorithm, Theorem 2.1.6, we know that there exist quotient  $q$  and the remainder  $r$  such that

$$7^{348} + 25^{605} = q \cdot 8 + r, \quad 0 \leq r < 8.$$

This means that the remainder  $r$  satisfies the congruence  $7^{348} + 25^{605} \equiv r(\text{mod } 8)$ , where  $0 \leq r < 8$ . Now

$$\begin{aligned} 7 &\equiv -1(\text{mod } 8) & \text{and} & 25 \equiv 1(\text{mod } 8) \\ \Rightarrow 7^2 &\equiv (-1)^2(\text{mod } 8) & \Rightarrow (25)^{605} \equiv (1)^{605}(\text{mod } 8) \\ \Rightarrow 7^2 &\equiv 1(\text{mod } 8) & \Rightarrow 25^{605} \equiv 1(\text{mod } 8) \\ \Rightarrow (7^2)^{174} &\equiv (1)^{174}(\text{mod } 8) \\ \Rightarrow 7^{348} &\equiv 1(\text{mod } 8) \end{aligned}$$

Then by Theorem 6.1.16,

$$7^{348} + 25^{605} \equiv (1 + 1)(\text{mod } 8).$$

That is,  $7^{348} + 25^{605} \equiv 2(\text{mod } 8)$ . Hence, the remainder is 2.

Throughout this chapter, we will present more elegant applications of congruences.

From Theorem 6.1.16, we find that we can perform addition, subtraction, or multiplication to both sides of a congruence by an integer. However, this may not be true if we divide both sides of a congruence by an integer. For example, consider the congruence  $64 \equiv 22(\text{mod } 6)$ . Then  $32 \cdot 2 \equiv 11 \cdot 2(\text{mod } 6)$ . If we divide both sides of this congruence by 2, then we obtain  $32 \equiv 11(\text{mod } 6)$ . However, 6 does not divide  $32 - 11$  and so  $32 \not\equiv 11(\text{mod } 6)$ .

The following theorem tells when in a congruence the divisibility is possible.

**Theorem 6.1.20:** Let  $a$ ,  $b$ , and  $c$  be integers and  $m$  be a positive integer.

(i)  $ab \equiv ac(\text{mod } m)$  if and only if

$$b \equiv c \left( \text{mod } \frac{m}{\gcd(a, m)} \right).$$

(ii) If  $ab \equiv ac(\text{mod } m)$  and  $\gcd(a, m) = 1$ , then  $b \equiv c(\text{mod } m)$ .

### Proof:

(i) Let  $d = \gcd(a, m)$ . Because  $m > 0$ ,  $d \neq 0$ . Now  $d | a$  and  $d | m$ . Thus, there exist integers  $r$  and  $t$  such that  $a = dr$ ,  $m = dt$ . Because  $d = \gcd(a, m)$ , we must have  $\gcd(t, r) = 1$ .

Suppose  $ab \equiv ac(\text{mod } m)$ . Now

$$\begin{aligned} ab &\equiv ac(\text{mod } m) \\ \Rightarrow m &\mid (ab - ac) \\ \Rightarrow dt &\mid (drb - drc) && \text{because } a = dr \text{ and } m = dt \\ \Rightarrow dt &\mid d(rb - rc) \\ \Rightarrow t &\mid (rb - rc) \\ \Rightarrow t &\mid r(b - c). \end{aligned}$$

Because  $t$  and  $r$  are relatively prime, it follows that  $t$  divides  $b - c$ . Hence,  $b \equiv c(\text{mod } t)$ , i.e.,  $b \equiv c(\text{mod } \frac{m}{d})$ .

Conversely, assume that  $b \equiv c \pmod{\frac{m}{d}}$ . Then  $b - c = k\frac{m}{d}$  for some integer  $k$ . Hence,

$$ab - ac = a(b - c) = k\frac{m}{d}a = km\frac{a}{d} = kmr = mkr.$$

So we find that  $m$  divides  $ab - ac$ , i.e.,  $ab \equiv ac \pmod{m}$ .

- (ii) Part (ii) follows from part (i). ■

### EXAMPLE 6.1.21

$44 \equiv 24 \pmod{5}$ . Because  $\gcd(4, 5) = 1$ , we can divide both sides of this congruence by 4 (Theorem 6.1.20(ii)) and obtain  $11 \equiv 6 \pmod{5}$ .

**Theorem 6.1.22:** Let  $f(x) = a_nx^n + a_{n-1}x^{n-1} + \cdots + a_1x + a_0$  be a polynomial with integral coefficients. If  $u, v$ , and  $m$  are integers with  $m > 0$ ,  $u \equiv v \pmod{m}$ , then  $f(u) \equiv f(v) \pmod{m}$ .

**Proof:** Because  $u \equiv v \pmod{m}$ , it follows from Theorem 6.1.16 that for all  $i$

$$u^i \equiv v^i \pmod{m} \quad \text{and} \quad a_i u^i \equiv a_i v^i \pmod{m}.$$

Then  $a_n u^n + a_{n-1} u^{n-1} + \cdots + a_1 u + a_0 \equiv (a_n v^n + a_{n-1} v^{n-1} + \cdots + a_1 v + a_0) \pmod{m}$ . Hence,  $f(u) \equiv f(v) \pmod{m}$ . ■

In the following section, we will see some interesting applications of Theorem 6.1.22.

## Divisibility Tests

In this section, using the notion of congruences, we describe certain criteria that can be used to determine whether a given integer is divisible by another integer.

Let  $m$  be a positive integer. By Theorem 2.2.1,  $m$  can be written uniquely as

$$\begin{aligned} m &= a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10 + a_0 \\ &= (a_k a_{k-1} \cdots a_1 a_0)_{10}, \end{aligned}$$

where  $a_0, a_1, \dots, a_k$  are integers such that  $a_k \neq 0$  and  $0 \leq a_i < 10$  for all  $i = 0, 1, 2, \dots, k$ . The subscript 10 is merely to stress that this a base-10 representation of  $m$ . If no confusion arises, then we omit this subscript. Notice that  $a_0, a_1, \dots, a_k$  are the digits in the integer  $m$ .

For example, if  $m = 57609$ , then  $a_0 = 9, a_1 = 0, a_2 = 6, a_3 = 7, a_4 = 5$  and

$$\begin{aligned} m &= 57609 \\ &= 5 \cdot 10^4 + 7 \cdot 10^3 + 6 \cdot 10^2 + 0 \cdot 10 + 9 \\ &= a_4 10^4 + a_3 10^3 + a_2 10^2 + a_1 10 + a_0. \end{aligned}$$

In the discussion that follows, using the digits of  $m$ , we will be forming other integers. For example,  $a_1 a_0 = 09 = 9, a_2 a_1 a_0 = 609, a_3 a_2 a_1 = 760$ , and  $a_3 a_2 a_1 a_0 = 7609$ .

We first develop tests for divisibility by the powers of 2.

**Theorem 6.1.23:** Let  $m = a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10 + a_0$ , where  $a_0, a_1, \dots, a_k$  are integers such that  $a_k \neq 0$  and  $0 \leq a_i < 10$  for all  $i = 0, 1, 2, \dots, k$ .

- (i)  $m$  is divisible by 2 if and only if  $a_0$  is divisible by 2.
- (ii)  $m$  is divisible by  $2^2$  if and only if  $a_1 a_0$  is divisible by  $2^2$ .
- (iii)  $m$  is divisible by  $2^3$  if and only if  $a_2 a_1 a_0$  is divisible by  $2^3$ .
- (iv)  $m$  is divisible by  $2^4$  if and only if  $a_3 a_2 a_1 a_0$  is divisible by  $2^4$ .

**Proof:** Let

$$f(x) = a_k x^k + a_{k-1} x^{k-1} + \cdots + a_1 x + a_0.$$

(i) Now

$$f(10) = a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10 + a_0 = m$$

and  $f(0) = a_0$ . We observe that  $10 \equiv 0 \pmod{2}$ . Hence, by Theorem 6.1.22,  $f(10) \equiv f(0) \pmod{2}$ . This implies that  $m \equiv a_0 \pmod{2}$ . By Theorem 6.1.8, it follows that  $m$  is divisible by 2 if and only if  $a_0$  is divisible by 2.

(ii) Notice that  $10^2 \equiv 0 \pmod{4}$ . Moreover, we can show that  $10^i \equiv 0 \pmod{4}$  for all  $i \geq 2$ . Hence,

$$(a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_2 10^2) \equiv 0 \pmod{2^2}. \quad (6.1)$$

Also

$$(a_1 10 + a_0) \equiv (a_1 10 + a_0) \pmod{2^2}. \quad (6.2)$$

From (6.1) and (6.2), it follows that

$$(a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_2 10^2 + a_1 10 + a_0) \equiv (a_1 10 + a_0) \pmod{2^2},$$

i.e.,

$$m \equiv (a_1 10 + a_0) \pmod{2^2},$$

or

$$m \equiv (a_1 a_0)_{10} \pmod{2^2}.$$

Hence, By Theorem 6.1.8,  $m$  is divisible by  $2^2$  if and only if the number  $a_1 a_0$  is divisible by  $2^2$ .

(iii) and (iv) In a similar manner, we can prove that  $m$  is divisible by  $2^3$  if and only if  $a_2 a_1 a_0$  is divisible by  $2^3 = 8$  and  $m$  is divisible by  $2^4$  if and only if  $a_3 a_2 a_1 a_0$  is divisible by  $2^4 = 16$ . ■

#### EXAMPLE 6.1.24

From Theorem 6.1.23, we can say without performing the actual division that the integer 76954732 is divisible by 2 and also divisible by 4, but not divisible by 8.

Next, we develop divisibility tests for 3, 9, and 11.

**Theorem 6.1.25:** Let  $m = a_k 10^k + a_{k-1} 10^{k-1} + \dots + a_1 10 + a_0$ , where  $a_0, a_1, \dots, a_k$  are integers such that  $a_k \neq 0$  and  $0 \leq a_i < 10$  for all  $i = 0, 1, 2, \dots, k$ . Let

$$s = a_0 + a_1 + \dots + a_{k-1} + a_k, \quad \text{and}$$

$$t = a_0 - a_1 + a_2 + \dots + (-1)^k a_k.$$

Then

- (i)  $m$  is divisible by 3 if and only if  $s$  is divisible by 3.
- (ii)  $m$  is divisible by 9 if and only if  $s$  is divisible by 9.
- (iii)  $m$  is divisible by 11 if and only if  $t$  is divisible by 11.

**Proof:** Let

$$f(x) = a_k x^k + a_{k-1} x^{k-1} + \dots + a_1 x + a_0.$$

- (i) Now  $f(10) = m$  and  $f(1) = s$ . Because  $10 \equiv 1 \pmod{3}$ , by Theorem 6.1.22,  $f(10) \equiv f(1) \pmod{3}$ , i.e.,  $m \equiv s \pmod{3}$ . Hence, by Theorem 6.1.8,  $m$  is divisible by 3 if and only if  $s$  is divisible by 3.
- (ii)  $10 \equiv 1 \pmod{9}$ . Hence,  $f(10) \equiv f(1) \pmod{9}$ , i.e.,  $m \equiv s \pmod{9}$ . It follows that  $m$  is divisible by 9 if and only if  $s$  is divisible by 9.
- (iii) We have  $f(-1) = t$ . Also,  $10 \equiv -1 \pmod{11}$ . Hence,  $f(10) \equiv f(-1) \pmod{11}$ , i.e.,  $m \equiv t \pmod{11}$ . Consequently,  $m$  is divisible by 11 if and only if  $t$  is divisible by 11. ■

Example 6.1.26 illustrates Theorem 6.1.25.

### EXAMPLE 6.1.26

Consider the integer  $m = 609321$ . Here  $s = 6 + 0 + 9 + 3 + 2 + 1 = 21$ . Because 3 divides 21, it follows that 3 divides 609321. Now we find that 9 does not divide 21, so it follows that 9 does not divide 609321.

Let  $m = 27193257$ . Then  $s = 2 + 7 + 1 + 9 + 3 + 2 + 5 + 7 = 36$ . Because 36 is divisible by both 3 and 9, it follows that 27193257 is divisible by both 3 and 9.

Let  $m = 3375889$ . Now  $t = 9 - 8 + 8 - 5 + 7 - 3 + 3 = 11$ . Because 11 divides 11, it follows that 11 divides 3375889.

Next we develop the divisibility tests for 7 and 13.

**Theorem 6.1.27:** Let  $m = a_k 10^k + a_{k-1} 10^{k-1} + \dots + a_1 10 + a_0$ , where  $a_0, a_1, \dots, a_k$  are integers such that  $a_k \neq 0$  and  $0 \leq a_i < 10$  for all  $i = 0, 1, 2, \dots, k$ . Let

$$t = a_2 a_1 a_0 - a_5 a_4 a_3 + a_8 a_7 a_6 - \dots$$

Then

- (i)  $m$  is divisible by 7 if and only if  $t$  is divisible by 7.
- (ii)  $m$  is divisible by 13 if and only if  $t$  is divisible by 13.

**Proof:** Now,

$$\begin{aligned}
 m &= a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10 + a_0 \\
 &= a_0 + a_1 10 + a_2 10^2 + \cdots + a_{k-1} 10^{k-1} + a_k 10^k \\
 &= (a_0 + a_1 10 + a_2 10^2) + (a_3 10^3 + a_4 10^4 + a_5 10^5) \\
 &\quad + (a_6 10^6 + a_7 10^7 + a_8 10^8) + \cdots \\
 &= (a_2 10^2 + a_1 10 + a_0) + (a_5 10^5 + a_4 10^4 + a_3 10^3) \\
 &\quad + (a_8 10^8 + a_7 10^7 + a_6 10^6) + \cdots \\
 &= (a_2 10^2 + a_1 10 + a_0) + 10^3(a_5 10^2 + a_4 10^1 + a_3) \\
 &\quad + 10^6(a_8 10^2 + a_7 10^1 + a_6) + \cdots \\
 &= (a_2 10^2 + a_1 10 + a_0) + 1000(a_5 10^2 + a_4 10^1 + a_3) \\
 &\quad + (1000)^2(a_8 10^2 + a_7 10^1 + a_6) + \cdots \\
 &= a_2 a_1 a_0 + 1000 \cdot a_5 a_4 a_3 + 1000^2 \cdot a_8 a_7 a_6 + \cdots
 \end{aligned}$$

Now

$$1000 \equiv -1 \pmod{7} \text{ and } 1000 \equiv -1 \pmod{13}.$$

This implies that

$$(1000)^i \equiv \begin{cases} 1 \pmod{7} & \text{if } i \text{ is even,} \\ -1 \pmod{7} & \text{if } i \text{ is odd,} \end{cases}$$

and

$$(1000)^i \equiv \begin{cases} 1 \pmod{13} & \text{if } i \text{ is even,} \\ -1 \pmod{13} & \text{if } i \text{ is odd.} \end{cases}$$

Hence,

$$\begin{aligned}
 m &\equiv ((a_2 10^2 + a_1 10 + a_0) - (a_5 10^2 + a_4 10^1 + a_3) \\
 &\quad + (a_8 10^2 + a_7 10^1 + a_6) - \cdots) \pmod{7},
 \end{aligned}$$

i.e.,

$$m \equiv (a_2 a_1 a_0 - a_5 a_4 a_3 + a_8 a_7 a_6 - \cdots) \pmod{7}$$

or

$$m \equiv t \pmod{7}.$$

This implies that  $m$  is divisible by 7 if and only if  $t$  is divisible by 7.

In a similar manner, we can show that  $m \equiv t \pmod{13}$  and conclude that 13 divides  $m$  if and only if 13 divides  $t$ . ■

The following example illustrates the technique of the proof of Theorem 6.1.27.

### EXAMPLE 6.1.28

Let  $m = 83953861057105$ . Then

$$\begin{aligned}
 m &= 8 \cdot 10^{13} + 3 \cdot 10^{12} + 9 \cdot 10^{11} + 5 \cdot 10^{10} + 3 \cdot 10^9 + 8 \cdot 10^8 + 6 \cdot 10^7 \\
 &\quad + 1 \cdot 10^6 + 0 \cdot 10^5 + 5 \cdot 10^4 + 7 \cdot 10^3 + 1 \cdot 10^2 + 0 \cdot 10^1 + 5
 \end{aligned}$$

$$\begin{aligned}
&= (1 \cdot 10^2 + 0 \cdot 10^1 + 5) + 10^3(0 \cdot 10^2 + 5 \cdot 10^1 + 7) \\
&\quad + 10^6(8 \cdot 10^2 + 6 \cdot 10^1 + 1) + 10^9(9 \cdot 10^2 + 5 \cdot 10^1 + 3) + 10^{12}(8 \cdot 10 + 3) \\
&= 105 + 10^3 \cdot 57 + 10^6 \cdot 861 + 10^9 \cdot 953 + 10^{12} \cdot 83.
\end{aligned}$$

Now

$$\begin{aligned}
10^3 &\equiv -1 \pmod{7}, & 10^6 &\equiv 1 \pmod{7}, \\
10^9 &\equiv -1 \pmod{7}, & 10^{12} &\equiv 1 \pmod{7}.
\end{aligned}$$

Hence,

$$m \equiv (105 - 57 + 861 - 953 + 83) \pmod{7},$$

i.e.,

$$m \equiv 39 \pmod{7}.$$

This implies that  $m$  is divisible by 7 if and only if 39 is divisible by 7. But 7 does not divide 39. Hence,  $m$  is not divisible by 7.

Again,

$$\begin{aligned}
10^3 &\equiv -1 \pmod{13}, & 10^6 &\equiv 1 \pmod{13}, \\
10^9 &\equiv -1 \pmod{13}, & 10^{12} &\equiv 1 \pmod{13}.
\end{aligned}$$

Hence,

$$m \equiv (105 - 57 + 861 - 953 + 83) \pmod{13},$$

i.e.,

$$m \equiv 39 \pmod{13}.$$

$m$  is divisible by 13 if and only if 39 is divisible by 13. Because 13 divides 39, 13 divides  $m$ .

## Addition and Multiplication of Congruence Classes

Let  $m$  be a positive integer. Then by Theorem 6.1.14,  $\mathbb{Z}_m = \{[0], [1], [2], \dots, [m-1]\}$ .

We now define addition and multiplication of the congruence classes  $[a]$  and  $[b]$  in  $\mathbb{Z}_m$  by

$$[a] + [b] = [a + b] \quad \text{and} \quad [a] \cdot [b] = [ab].$$

Consider  $m = 6$ . In  $\mathbb{Z}_6$ , we find that

$$[2] + [3] = [5] \quad \text{and} \quad [2] \cdot [3] = [6] = [0].$$

Next observe that in  $\mathbb{Z}_6$ ,  $[2] = [20]$  and  $[3] = [9]$ . Now  $[20] + [9] = [29]$ . We have  $[2] + [3] = [5]$  and  $[2] + [3] = [20] + [9] = [29]$ . So what we see here is that  $[2] + [3]$  is  $[5]$  as well as  $[29]$ .

In the set  $\mathbb{Z}$  of all integers, we know that the sum of two integers is unique; i.e., for integers  $a$  and  $b$  there exists a unique integer  $c$  such that  $a + b = c$ .

But in  $\mathbb{Z}_6$ , the situation is  $[2] + [3] = [5]$  and  $[2] + [3] = [29]$ . It looks as if  $[2] + [3]$  gives two different classes,  $[5]$  and  $[29]$ , in  $\mathbb{Z}_6$ . However, observe that  $29 \equiv 5 \pmod{6}$ , so  $[5] = [29]$ .

We can also find other classes in  $\mathbb{Z}_6$  that represent the sum  $[2] + [3]$ . However, we can show that all such classes are the same.

In general, we show that the sum of any two congruence classes in  $\mathbb{Z}_m$  is unique for any positive integer  $m$ .

Let  $a, b, c$ , and  $d$  be integers such that  $[a] = [b]$  and  $[c] = [d]$  in  $\mathbb{Z}_m$ . Now  $[a] + [c] = [a + c]$  and  $[b] + [d] = [b + d]$ . We show that  $[a + c] = [b + d]$ .

Because  $[a] = [b]$ , we find that

$$a \equiv b \pmod{m} \quad (6.3)$$

Similarly,  $[c] = [d]$ , implies that

$$c \equiv d \pmod{m} \quad (6.4)$$

Hence, from (6.3) and (6.4),  $a + c \equiv (b + d) \pmod{m}$ . This implies that

$$[a + c] = [b + d].$$

So we conclude that the sum of two congruence classes  $[a]$  and  $[b]$  is unique.

Hence, addition of two congruence classes is well defined. This shows that  $+$  is a binary operation on  $\mathbb{Z}_m$ . Because  $\mathbb{Z}_m$  is a finite set, we can construct the Cayley table, also known as the addition table, of  $(\mathbb{Z}_m, +)$ .

Let  $m = 6$ . We know that:

$$\mathbb{Z}_6 = \{[0], [1], [2], [3], [4], [5]\}.$$

Here,  $[2] + [0] = [2]$ ,  $[2] + [1] = [3]$ ,  $[2] + [2] = [4]$ ,  $[2] + [3] = [5]$ ,  $[2] + [4] = [6] = [0]$ ,  $[2] + [5] = [7] = [1]$ , and so on. Using these facts, we can construct the following addition table for  $\mathbb{Z}_6$ .

$+$	[0]	[1]	[2]	[3]	[4]	[5]
[0]	[0]	[1]	[2]	[3]	[4]	[5]
[1]	[1]	[2]	[3]	[4]	[5]	[0]
[2]	[2]	[3]	[4]	[5]	[0]	[1]
[3]	[3]	[4]	[5]	[0]	[1]	[2]
[4]	[4]	[5]	[0]	[1]	[2]	[3]
[5]	[5]	[0]	[1]	[2]	[3]	[4]

As in the case of addition, we can prove that multiplication of two congruence classes is well defined.

Notice that  $[3] \cdot [5] = [15] = [3]$  and  $[2] \cdot [5] = [10] = [4]$  in  $\mathbb{Z}_6$ . Similarly, we can multiply other congruence classes in  $\mathbb{Z}_6$  and obtain the following multiplication table.

$\cdot$	[0]	[1]	[2]	[3]	[4]	[5]
[0]	[0]	[0]	[0]	[0]	[0]	[0]
[1]	[0]	[1]	[2]	[3]	[4]	[5]
[2]	[0]	[2]	[4]	[0]	[2]	[4]
[3]	[0]	[3]	[0]	[3]	[0]	[3]
[4]	[0]	[4]	[2]	[0]	[4]	[2]
[5]	[0]	[5]	[4]	[3]	[2]	[1]

**Theorem 6.1.29:** Let  $m$  be a positive integer. Define  $+$  and  $\cdot$  on  $\mathbb{Z}_m$  by:  
For all  $[a], [b], [c], [d] \in \mathbb{Z}_m$ ,

$$[a] + [b] = [a + b],$$

$$[a] \cdot [b] = [ab].$$

Let  $[a], [b], [c], [d] \in \mathbb{Z}_m$  such that  $[a] = [c]$  and  $[b] = [d]$ . Then the following assertions hold:

- (i)  $[a] + [b] = [a + b] = [c + d] = [c] + [d]$ ; that is,  $+$  is well defined on  $\mathbb{Z}_m$ .
- (ii)  $[a] \cdot [b] = [ab] = [cd] = [c] \cdot [d]$ ; that is,  $\cdot$  is well defined on  $\mathbb{Z}_m$ .

## WORKED-OUT EXERCISES

**Exercise 1:** Find the remainder when  $107 + 23 + 109 + 35 + 93$  is divided by 5.

**Solution:** Now

$$\begin{aligned} 107 &\equiv 2 \pmod{5}, & 23 &\equiv 3 \pmod{5}, & 109 &\equiv 4 \pmod{5}, \\ 35 &\equiv 0 \pmod{5}, & 93 &\equiv 3 \pmod{5}. \end{aligned}$$

This implies that

$$107 + 23 + 109 + 35 + 93 \equiv (2 + 3 + 4 + 0 + 3) \pmod{5},$$

i.e.,

$$107 + 23 + 109 + 35 + 93 \equiv 12 \pmod{5}.$$

But  $12 \equiv 2 \pmod{5}$ . Therefore, by the transitive property,

$$107 + 23 + 109 + 35 + 93 \equiv 2 \pmod{5}$$

Hence, the remainder is 2.

If there is no confusion about congruences, we also do the computation as follows:

$$\begin{aligned} 107 + 23 + 109 + 35 + 93 &\equiv (2 + 3 + 4 + 0 + 3) \pmod{5} \\ &\equiv 12 \pmod{5} \equiv 2 \pmod{5} \end{aligned}$$

Hence, the remainder is 2.

Here is another way to do the computation.

$$\begin{aligned} 107 + 23 + 109 + 35 + 93 &\equiv (2 - 2 - 1 + 0 + 3) \pmod{5} \\ &\equiv 2 \pmod{5} \end{aligned}$$

Because  $23 \equiv -2 \pmod{5}$ ,  $109 \equiv -1 \pmod{5}$ .

**Exercise 2:** What is the remainder when  $6^{248}$  is divided by 7?

**Solution:** By the division algorithm, Theorem 2.1.6, we know that there exist quotient  $q$  and remainder  $r$  such that  $6^{248} = q \cdot 7 + r$ ,  $0 \leq r < 7$ . This means that the remainder  $r$  satisfies the congruence  $6^{248} \equiv r \pmod{7}$ , where  $0 \leq r < 7$ .

Now

$$6 \equiv -1 \pmod{7}$$

$$\Rightarrow 6^2 \equiv (-1)^2 \pmod{7} \quad \text{by Theorem 6.1.16(iii)}$$

$$\Rightarrow 6^2 \equiv 1 \pmod{7}$$

$$\Rightarrow 6^{2 \cdot 124} \equiv 1^{124} \pmod{7} \quad \text{by Theorem 6.1.16(iii)}$$

$$\Rightarrow 6^{248} \equiv 1 \pmod{7}.$$

Hence, the remainder is 1.

**Exercise 3:** What is the remainder when  $16 \cdot 5^{32} + 89$  is divided by 6?

**Solution:**

$$5^2 \equiv 1 \pmod{6}$$

$$\Rightarrow (5^2)^{16} \equiv 1^{16} \pmod{6} \quad \text{by Theorem 6.1.16(iii)}$$

$$\Rightarrow 5^{32} \equiv 1 \pmod{6}$$

$$\Rightarrow 16 \cdot 5^{32} \equiv 16 \cdot 1 \pmod{6} \equiv 4 \pmod{6} \quad \text{by Corollary 6.1.17(ii).}$$

Also,  $89 \equiv 5 \pmod{6}$ . Hence,  $16 \cdot 5^{32} + 89 \equiv (4 + 5) \pmod{6} \equiv 3 \pmod{6}$ . Hence, the remainder is 3.

**Exercise 4:** What is the smallest number that when divided by 10, 9, 8, 7, 6 leaves the remainder 9, 8, 7, 6, 5, respectively?

**Solution:** Let  $x$  be the required number. Then

$$x \equiv 9 \pmod{10}$$

$$\Rightarrow x + 1 \equiv 10 \pmod{10}$$

$$\Rightarrow x + 1 \equiv 0 \pmod{10}.$$

Similarly,  $x + 1 \equiv 0 \pmod{9}$ ,  $x + 1 \equiv 0 \pmod{8}$ ,  $x + 1 \equiv 0 \pmod{7}$ ,  $x + 1 \equiv 0 \pmod{6}$ . Thus,  $x + 1$  is the least common multiple of 10, 9, 8, 7, and 6. Now the lcm of 10, 9, 8, 7, and 6 is 2520. Hence,  $x + 1 = 2520$ . This implies that  $x = 2520 - 1 = 2519$ .

**Exercise 5:** Let  $a$  and  $b$  be integers and  $m$  be a positive integer. Prove that  $a \equiv b \pmod{m}$  if and only if  $a = km + b$  for some integer  $k$ .

**Solution:** Suppose  $a \equiv b \pmod{m}$ . Then  $m$  divides  $a - b$ . Hence,  $a - b = km$  for some integer  $k$ ; i.e.,  $a = km + b$  for some integer  $k$ . Conversely, suppose that  $a = km + b$  for some integer  $k$ . Then  $a - b = km$  for some integer  $k$ . Hence,  $m$  divides  $a - b$ , i.e.,  $a \equiv b \pmod{m}$ .

### Exercise 6:

- Find all integers  $k \geq 3$  such that  $11 \equiv k^2 \pmod{k}$ .
- Find all integers  $k \geq 3$  such that  $7 \equiv k \pmod{k^2}$ .

### Solution:

- Given  $11 \equiv k^2 \pmod{k}$ . Then  $k$  divides  $11 - k^2$ . Because  $k$  divides  $11 - k^2$  and  $k$  divides  $k^2$ ,  $k$  divides  $11$ . Now  $k$  divides  $11$  and  $k \geq 3$ . Therefore, we find that  $k = 11$ .
- Given  $7 \equiv k \pmod{k^2}$ . Then  $k^2$  divides  $7 - k$ . Hence, there exists an integer  $t$  such that  $7 - k = k^2t$ . Then  $7 = k + k^2t = k(1 + kt)$ . So we find that  $k$  divides  $7$ . But  $k \geq 3$ . Hence,  $k = 7$ .

**Exercise 7:** What is the remainder when  $1! + 2! + 3! + \dots + 99! + 100!$  is divided by 18?

**Solution:** We have to find an integer  $r$  such that  $0 \leq r < 18$  and  $1! + 2! + 3! + \dots + 99! + 100! \equiv r \pmod{18}$ . Now

$$\begin{aligned} 1! &\equiv 1 \pmod{18} \\ 2! &\equiv 2 \pmod{18} \\ 3! &\equiv 6 \pmod{18} \\ 4! &\equiv 6 \pmod{18} \\ 5! &\equiv 12 \pmod{18} \\ 6! &= 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \equiv 0 \pmod{18} \\ n! &\equiv 0 \pmod{18} \text{ for all } n \geq 6. \end{aligned}$$

Thus,  $1! + 2! + 3! + \dots + 99! + 100! \equiv (1 + 2 + 6 + 6 + 12) \pmod{18} \equiv 9 \pmod{18}$ . Hence,

$$1! + 2! + 3! + \dots + 99! + 100! \equiv 9 \pmod{18}.$$

So the remainder is 9.

**Exercise 8:** Without performing the long division, determine whether 361905102 is divisible by 9, 11, or 3.

**Solution:** Let  $m = 361905102$ . Now  $s = 3 + 6 + 1 + 9 + 0 + 5 + 1 + 0 + 2 = 27$ . Because 27 is divisible by 9, by Theorem 6.1.25, 361905102 is divisible by 9. It also follows that 361905102 is divisible by 3.

Let  $t = 2 - 0 + 1 - 5 + 0 - 9 + 1 - 6 + 3 = -13$ . Because 11 does not divide  $-13$ , it follows from Theorem 6.1.25 that 11 does not divide 361905102.

**Exercise 9:** Which of the following integers are divisible by 13?

- 501121301
- 2711111202201

### Solution:

- Let  $m = 501121301$ . For this integer  $t = 301 - 121 + 501 = 681$ . Because 13 does not divide 681, by Theorem 6.1.25, 13 does not divide 501121301.
- Let  $m = 2711111202201$ . For this integer  $t = 201 - 202 + 111 - 111 + 27 = 26$ . Because 13 divides 26, by Theorem 6.1.25, 13 divides 2711111202201.

**Exercise 10:** Find the highest power of 2 dividing each of the following integers.

- $(1101000)_2$
- $(110111010)_2$

### Solution:

- Let  $m = (1101000)_2$ . Then  $m = 1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 2^3(2^3 + 2^2 + 1)$ . Hence,  $m$  is divisible by  $2^3$ , but  $2^4$  does not divide  $m$ . Therefore, the highest power of 2 that divides  $m$  is 3.
- Let  $m = (110111010)_2$ . Then  $m = 1 \cdot 2^8 + 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 2^8 + 2^7 + 2^5 + 2^4 + 2^3 + 2^1 + 0 = 2(2^7 + 2^6 + 2^4 + 2^3 + 2^2 + 1)$ . Hence, the highest power of 2 that divides  $m$  is 1.

**Exercise 11:** Let  $n = (55F05A7)_{16}$ . Show that 17 divides  $n$ .

**Solution:** Now  $n = 5 \cdot 16^6 + 5 \cdot 16^5 + F \cdot 16^4 + 0 \cdot 16^3 + 5 \cdot 16^2 + A \cdot 16^1 + 7$ . Now

$$\begin{aligned} 16 &\equiv -1 \pmod{17}, & 16^2 &\equiv 1 \pmod{17}, \\ 16^3 &\equiv -1 \pmod{17}, & 16^4 &\equiv 1 \pmod{17}, \\ 16^5 &\equiv -1 \pmod{17}, & 16^6 &\equiv 1 \pmod{17}. \end{aligned}$$

Hence,

$$n \equiv (5 - 5 + F - 0 + 5 - A + 7) \pmod{17},$$

i.e.,

$$n \equiv (5 - 5 + 15 - 0 + 5 - 10 + 7) \pmod{17}$$

or

$$n \equiv 17 \pmod{17}.$$

Because 17 divides 17, by Theorem 6.1.8, it follows that 17 divides  $n$ .

## SECTION REVIEW

---

### Key Terms

congruent

congruence class modulo  $m$

### Some Key Definitions

1. Let  $a$  and  $b$  be two integers and  $m$  be a positive integer. Then  $a$  is said to be congruent to  $b$  modulo  $m$  if  $a$  and  $b$  have the same remainder when  $m$  divides both  $a$  and  $b$ . If  $a$  is congruent to  $b$  modulo  $m$ , we write  $a \equiv b \pmod{m}$ .
2. Let  $m$  be a positive integer and  $a$  be an integer. Then the set  $[a] = \{b \in \mathbb{Z} \mid b \equiv a \pmod{m}\}$  is called the congruence class modulo  $m$  of the integer  $a$ .

### Some Key Results

1. Let  $a, b, c$  and  $d$  be integers and  $m$  be a positive integer.
  - (i) If  $a \equiv b \pmod{m}$  and  $c \equiv d \pmod{m}$ , then  $a + c \equiv (b + d) \pmod{m}$ .
  - (ii) If  $a \equiv b \pmod{m}$  and  $c \equiv d \pmod{m}$ , then  $ac \equiv bd \pmod{m}$ .
  - (iii) If  $a \equiv b \pmod{m}$ , then  $a^n \equiv b^n \pmod{m}$  for any positive integer  $n$ .
2. Let  $a, b, c$  and  $d$  be integers and  $m$  be a positive integer.
  - (i) If  $a \equiv b \pmod{m}$ , then  $a + c \equiv (b + c) \pmod{m}$  for any integer  $c$ .
  - (ii) If  $a \equiv b \pmod{m}$ , then  $ac \equiv bc \pmod{m}$  for any integer  $c$ .
  - (iii) If  $a \equiv b \pmod{m}$ , then  $a - c \equiv (b - c) \pmod{m}$  for any integer  $c$ .
3. Let  $a, b$ , and  $c$  be integers and  $m$  be a positive integer.
  - (i)  $ab \equiv ac \pmod{m}$  if and only if  $b \equiv c \pmod{\frac{m}{\gcd(a, m)}}$ .
  - (ii) If  $ab \equiv ac \pmod{m}$  and  $\gcd(a, m) = 1$ , then  $b \equiv c \pmod{m}$ .
4. Let  $m = a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10 + a_0$ , where  $a_0, a_1, \dots, a_k$  are integers such that  $a_k \neq 0$  and  $0 \leq a_i < 10$  for all  $i = 0, 1, 2, \dots, k$ .
  - (i)  $m$  is divisible by 2 if and only if  $a_0$  is divisible by 2.
  - (ii)  $m$  is divisible by  $2^2$  if and only if  $a_1 a_0$  is divisible by  $2^2$ .
  - (iii)  $m$  is divisible by  $2^3$  if and only if  $a_2 a_1 a_0$  is divisible by  $2^3$ .
  - (iv)  $m$  is divisible by  $2^4$  if and only if  $a_3 a_2 a_1 a_0$  is divisible by  $2^4$ .
5. Let  $m = a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10 + a_0$ , where  $a_0, a_1, \dots, a_k$  are integers such that  $a_k \neq 0$  and  $0 \leq a_i < 10$  for all  $i = 0, 1, 2, \dots, k$ . Let  $s = a_0 + a_1 + \cdots + a_{k-1} + a_k$ , and  $t = a_0 - a_1 + a_2 - \cdots + (-1)^k a_k$ . Then
  - (i)  $m$  is divisible by 3 if and only if  $s$  is divisible by 3.
  - (ii)  $m$  is divisible by 9 if and only if  $s$  is divisible by 9.
  - (iii)  $m$  is divisible by 11 if and only if  $t$  is divisible by 11.

6. Let  $m = a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10 + a_0$ , where  $a_0, a_1, a_k$  are integers such that  $a_k \neq 0$  and  $0 \leq a_i < 10$  for all  $i = 0, 1, 2, \dots, k$ . Let  $t = (a_2 a_1 a_0)_{10} - (a_5 a_4 a_3)_{10} + (a_8 a_7 a_6)_{10} - \cdots$ . Then
- $m$  is divisible by 7 if and only if  $t$  is divisible by 7.
  - $m$  is divisible by 13 if and only if  $t$  is divisible by 13.

## EXERCISES

---

- Determine whether each case is true or false.
  - $44 \equiv 13 \pmod{3}$
  - $125 \equiv 25 \pmod{4}$
  - $75 \equiv 32 \pmod{5}$
  - $-55 \equiv 5 \pmod{4}$
  - $8^4 \equiv 3 \pmod{11}$
  - $7^4 \equiv 1 \pmod{10}$
- What is the remainder when  $27 + 98 + 123 + 97 + 351$  is divided by 5?
- What is the remainder when  $1! + 2! + 3! + \cdots + 99! + 100!$  is divided by 20?
- Find all integers  $k$  making each of the following true.
  - $6 \equiv 3k \pmod{7}$
  - $3k \equiv 9 \pmod{14}$
- What is the remainder when  $7^{30}$  is divided by 4?
- What is the remainder when  $6 \cdot 7^{32} + 17 \cdot 9^{45}$  is divided by 5?
- What is the smallest number that when divided by 10, 9, 8, 7, 6 leaves the remainder 8, 7, 6, 5, 4, respectively?
- Find all integers  $k \geq 3$  such that
  - $-4 \equiv 11 \pmod{k}$ .
  - $9 \equiv -5 \pmod{k}$ .
  - $5 \equiv k^2 \pmod{k}$ .
  - $k^2 \equiv 7k \pmod{21}$ ,  $2 < k \leq 50$ .
  - $k^2 \equiv 5k \pmod{15}$ ,  $2 < k \leq 50$ .
- Let  $a, b, c$ , and  $d$  be integers and  $m$  be a positive integer. If  $a \equiv b \pmod{m}$  and  $c \equiv d \pmod{m}$ , then prove that
 
$$ax + cy \equiv (bx + dy) \pmod{m}.$$
- If  $a, b, n$ , and  $m$  are positive integers such that  $a \equiv b \pmod{n}$  and  $m$  divides  $n$ , then prove that  $a \equiv b \pmod{m}$ .
- Which of the following integers are divisible by 16?
  - 590012047
  - 1799212
  - 4902016
- Determine the highest power of 2 dividing each of the following integers.
  - 12132
  - 5799212
  - 90205632
  - $(101010100)_2$
  - $(11001001)_2$
- Which of the following integers are divisible by 3 or 9?
  - 5412351
  - 9093939
  - $(10020101)_3$
  - 4760181
- Which of the following integers are divisible by 7 or 13?
  - 1890561
  - 90030005
  - 23170187005
- Which of the following integers are divisible by 66?
  - 5030622
  - 50392036
  - 3030324
- Let  $n = (a_k a_{k-1} \cdots a_1 a_0)_b$  be an integer in the base  $b$ . If  $d$  is a positive integer such that  $d$  divides  $b - 1$ , prove that  $d$  divides  $n$  if and only if  $d$  divides  $a_k + a_{k-1} + \cdots + a_1 + a_0$ .
- Let  $n = (53F35A6)_{16}$ . Show that 17 does not divide  $n$ .
- Which of the following integers are divisible by 3 and which are divisible by 5?
  - $(6A3E5)_{16}$
  - $(A2CDEF)_{16}$
  - $(3910A2F3)_{16}$
- Construct the addition and multiplication tables for  $\mathbb{Z}_5$  and  $\mathbb{Z}_7$

## 6.2 CHECK DIGITS

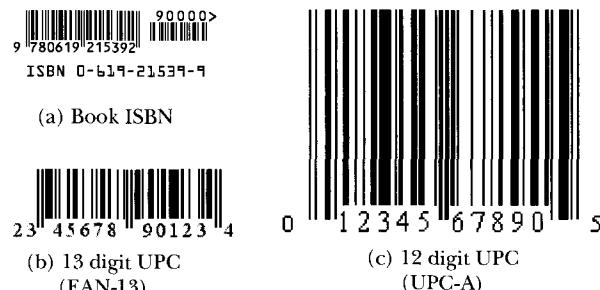
---

In the preceding section, we described the basic properties of congruences and applied those results in divisibility. This section is solely devoted to the applications of congruences.

A new young-adult mystery novel is being released and is in great demand. Shelly and her friends head to the bookstore where they wait in long lines to enter. As soon as the store opens, everyone rushes in. Shelly and her friends each manage to get a copy. After waiting in line to check out, Shelly watches as the cashier passes her copy of the book over a glass that covers a blinking red light. The checkout screen instantly shows the amount due. Shelly and her friends joke about the long waits but marvel at how the computers can determine the prices so quickly. They start to wonder about how the computer is able to determine the

price. They check the back of the book that the cashier passed over the optical scanner. On the back corner of the book they find a series of black and white stripes. On top of the stripes is a number that begins with the letters ISBN. For Shelly and her friends the meaning of the ISBN number is a mystery, not unlike the book they just purchased.

While Shelly and her friends may have been baffled by the meaning and use of the ISBN code found on books, most people are familiar with their purpose, even if they don't understand the specific meaning of the numbers. Also, not only books but many other products that we use daily come with packaging numbers that assist in pricing as well as inventory control and management. Typically, the package contains an identification number that consists of some numeric digits. Figure 6.1 shows some examples of identification numbers.



**FIGURE 6.1** Identification numbers of various products

In this section, we discuss how these identification numbers are assigned as well as various ways to check the validity of the identification numbers.

## ISBN

Modern-day books published worldwide are identified by a string of digits known as the **International Standard Book Number**, or ISBN. This is a relatively recent practice. Before 1970, in some of the English-speaking countries a Standard Book Number (SBN) was used to identify books. In 1970, this SBN was transformed into the International Standard Book Number (ISBN). At present, ISBNs are used on books published in 159 countries and territories.

The ISBN of a book is usually found on the last cover page. It appears as a string of ten digits. The first nine digits identify the book, and the last digit is called the **check digit**. For example, the ISBN of the book *C++ Programming: From Problem Analysis to Program Design* is 0-619-06213-4. As we can see, the first nine digits are divided into three parts separated by hyphens. The first digit, 0, which we call the leading digit, means that the book is published in the English-speaking countries (the U.K., the United States, Australia, New Zealand, and Canada). The next group of digits, 619, identifies the publisher, which in this case is Course Technology. The third set of digits, 06213, is the number assigned to the book by the publisher (in this case, the particular book published by Course Technology). The last digit, 4, is the check digit.

Formally, we describe an ISBN in the following way:

An ISBN is an expression

$$x_1 x_2 x_3 \cdots x_9 x_{10}$$

of ten digits divided into four blocks. The first block indicates the language or country of origin where it is published, the second block specifies the publishing

company, the third block is the number assigned to the book by the publisher, and the final block, consisting of only one digit, is the check digit.

The size of each of the first three blocks is not fixed. For example, the ISBN of the text *Abstract Algebra* by I. N. Herstein is 0-02-353820-1. Here the second block consists of two digits and the third block consists of six digits. The leading digit 0 means that the book is published in the English-speaking world. The second block, which consists of the digits 02, identifies the publisher, in this case MacMillan Publishing Company. Here the check digit is 1.

A very small publisher may have a large block for the publisher's number and a large publisher may have a small block, leaving much more space for book numbers. Typically when a book is reprinted, the reprint receives the same ISBN, unless the material is sufficiently revised to receive a new ISBN.

For all  $i = 1, 2, \dots, 9$ ,  $x_i$  is one of the digits 0, 1, 2, 3, ..., 9. However, for the check digit  $x_{10}$  there are eleven possible values: 0, 1, 2, ..., 10. If the check digit is a 10, then it is denoted by the roman numeral X. Therefore,

$$x_{10} \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, X\}.$$

As it turns out, for a publisher keeping track of books, using ISBNs is easier than using titles, authors, or editions. Two (or more) different publishers may have the same title for a particular subject. For example, most books on calculus are titled *Calculus*. However, because different publishers assign different ISBNs to their books, not only the publishers, but also bookstores, which sell books from a variety of publishers, can easily identify the books and subsequently place orders using ISBNs. Moreover, the ISBNs make it easier for publishers to computerize their inventories and billing procedures.

Because ISBNs are a string of numbers, practical experience shows that they are vulnerable to human errors, such as entering a wrong single digit or interchanging two digits. It is in order to detect such errors that the mathematical means of using the check digit were devised.

Our interest in ISBNs centers around check digits because they help us determine the validity of ISBNs. Besides that, they provide an interesting application of congruences.

**REMARK 6.2.1** ► Before we begin our discussion of ISBNs and their check digits, we would like to point out that the apex international body of the publishers has already decided to expand the ISBN from a 10-digit code to a 13-digit code in order to make it efficient in handling a lot more books. The new ISBN will come into effect in January 2005, whereas the deadline for this transition worldwide is scheduled for January 2007. The appropriate congruence scheme for the *check digit* of these new numbers is presently at the research level, undertaken by the Society of Motion Pictures and Television Engineers (SMPTE) in USA.

We now explain how congruences are used to assign a check digit to a particular book.

Suppose the first nine digits  $x_1, x_2, x_3, \dots, x_9$  of ISBN have been chosen. The check digit  $x_{10}$ , which is one of 0, 1, 2, 3, ..., 9, X, is determined by the following congruence:

$$1x_1 + 2x_2 + \cdots + 8x_8 + 9x_9 + 10x_{10} \equiv 0 \pmod{11}.$$

That is,

$$1x_1 + 2x_2 + \cdots + 8x_8 + 9x_9 + 10x_{10} + x_{10} \equiv x_{10} \pmod{11}$$

or

$$1x_1 + 2x_2 + \cdots + 8x_8 + 9x_9 + 11x_{10} \equiv x_{10} \pmod{11}$$

or

$$1x_1 + 2x_2 + \cdots + 8x_8 + 9x_9 \equiv x_{10} \pmod{11} \quad \text{because } 11 \equiv 0 \pmod{11}.$$

### EXAMPLE 6.2.2

The first nine digits of the book *C++ Programming: From Problem Analysis to Program Design* are 0-619-06213. Thus the check digit  $x_{10}$  of this book is determined by the congruence

$$1 \cdot 0 + 2 \cdot 6 + 3 \cdot 1 + 4 \cdot 9 + 5 \cdot 0 + 6 \cdot 6 + 7 \cdot 2 + 8 \cdot 1 + 9 \cdot 3 \equiv x_{10} \pmod{11},$$

i.e.,

$$136 \equiv x_{10} \pmod{11}.$$

Now  $136 = 12 \cdot 11 + 4 = 132 + 4$ . Thus we write  $136 \equiv x_{10} \pmod{11}$  as

$$132 + 4 \equiv x_{10} \pmod{11}.$$

This implies that

$$4 \equiv x_{10} \pmod{11}.$$

Because  $0 \leq x_{10} < 11$ , it follows that  $x_{10} = 4$ . Hence, the check digit for the ISBN of this book is 4.

### EXAMPLE 6.2.3

In this example, we consider the ISBN 81-203-1147- $x_{10}$ .

The check digit  $x_{10}$  is one of the digits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, X and it satisfies

$$1 \cdot 8 + 2 \cdot 1 + 3 \cdot 2 + 4 \cdot 0 + 5 \cdot 3 + 6 \cdot 1 + 7 \cdot 1 + 8 \cdot 4 + 9 \cdot 7 \equiv x_{10} \pmod{11}.$$

or

$$8 + 2 + 6 + 0 + 15 + 6 + 7 + 32 + 63 \equiv x_{10} \pmod{11}. \quad (6.5)$$

Now  $15 \equiv 4 \pmod{11}$ ,  $32 \equiv -1 \pmod{11}$ ,  $63 \equiv -3 \pmod{11}$ . Therefore, we can write

$$\begin{aligned} & 8 + 2 + 6 + 0 + 15 + 6 + 7 + 32 + 63 \\ & \equiv (8 + 2 + 6 + 4 + 6 + 7 - 1 - 3) \pmod{11} \\ & \equiv 29 \pmod{11} \\ & \equiv 7 \pmod{11}. \end{aligned}$$

or

$$7 \equiv x_{10} \pmod{11}.$$

This implies  $x_{10} = 7$ . Thus, the ISBN of this book is 81-203-1147-7.

As we said earlier, ISBNs make it easier for publishers to computerize their inventories and billing procedures. A serial number of standardized length and format is far easier to handle by computer than the alternative identification by title, author, and edition. Orders received by ISBN can be processed more easily than the orders received by title, author, edition, and so on. With the help of ISBNs, we can overcome language barriers—for example, a Chinese buyer can

place an order by fax with a German publisher without specifying the book's title. The ISBN system serves the needs of all parties in the book distribution system.

### Single-Error Detection in an ISBN

It is natural for errors to occur when people enter numerical data into a computer. Transmission over telephone or microwave channels can also lead to errors. We now look at the use of the check digits in identifying certain errors. First we show that a single error in the ISBN can be detected.

#### EXAMPLE 6.2.4

Let

$$x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9 x_{10}$$

be the correct ISBN of a book. During the billing procedure, suppose that a single error has been made; in the  $i$ th place  $y_i$  is printed instead of  $x_i$ , where  $x_i \neq y_i$ .

For example, the correct ISBN of a particular book is 0-534-93189-8. During the billing procedure, suppose it is printed as 0-534-43189-8. That is, in the fifth place a 4 appears instead of 9. This is a single-digit error. We now show that this kind of error can be detected.

A correct ISBN,  $x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9 x_{10}$ , must satisfy the congruence  $\sum_{i=1}^{10} ix_i \equiv 0 \pmod{11}$ .

Let us see what happens with the ISBN 0-534-43189-8. Now

$$\begin{aligned} & 1 \cdot 0 + 2 \cdot 5 + 3 \cdot 3 + 4 \cdot 4 + 5 \cdot 4 + 6 \cdot 3 + 7 \cdot 1 + 8 \cdot 8 + 9 \cdot 9 + 10 \cdot 8 \\ & = 10 + 9 + 16 + 20 + 18 + 7 + 64 + 81 + 80 \\ & \equiv (-1 - 2 + 5 - 2 - 4 + 7 - 2 + 4 + 3) \pmod{11} \\ & \equiv 8 \pmod{11} \\ & \not\equiv 0 \pmod{11}. \end{aligned}$$

Hence, 0-534-43189-8 is not a correct ISBN.

The preceding example shows that a single error in an ISBN can be detected. Let us prove, in general, that a single error can be detected by the check digits.

Suppose the changed ISBN is

$$y_1 y_2 y_3 y_4 y_5 y_6 y_7 y_8 y_9 y_{10}$$

where  $y_k = x_k$  for all  $k \neq i$  and  $y_i \neq x_i$ . Suppose  $x_i > y_i$ . Because  $0 \leq x_i \leq 10$  and  $0 \leq y_i \leq 10$ , there exists an integer  $a$  such that  $x_i = y_i + a$ , where  $0 < a \leq 10$ . Now

$$\begin{aligned} & 1y_1 + 2y_2 + 3y_3 + 4y_4 + 5y_5 + 6y_6 + 7y_7 + 8y_8 + 9y_9 + 10y_{10} \\ & = 1x_1 + 2x_2 + 3x_3 + \cdots + (i-1)x_{i-1} + iy_i + (i+1)x_{i+1} + \cdots + 9x_9 + 10x_{10} \\ & = 1x_1 + 2x_2 + 3x_3 + \cdots + (i-1)x_{i-1} + ix_i + i(-a) + (i+1)x_{i+1} + \cdots + 9x_9 + 10x_{10} \\ & = \sum_{i=1}^{10} ix_i + i(-a) \\ & \equiv i(-a) \pmod{11} \quad \text{because } \sum_{i=1}^{10} ix_i \equiv 0 \pmod{11}. \end{aligned}$$

That is,

$$\sum_{i=1}^{10} iy_i \equiv i(-a) \pmod{11}. \tag{6.6}$$

Suppose  $y_1y_2y_3y_4y_5y_6y_7y_8y_9y_{10}$  is a correct ISBN. Then  $\sum_{i=1}^{10} iy_i \equiv 0 \pmod{11}$ . Therefore, by (6.6),

$$i(-a) \equiv 0 \pmod{11}$$

i.e.,

$$ia \equiv 0 \pmod{11}.$$

This implies that 11 divides  $ia$ . Because 11 is prime, it follows that 11 divides  $i$  or 11 divides  $a$ . However,  $1 \leq i \leq 10$  and  $0 < a \leq 10$ . Therefore, 11 cannot divide  $i$  or  $a$ . Consequently,  $y_1y_2y_3y_4y_5y_6y_7y_8y_9y_{10}$  is not a correct ISBN.

If  $y_i > x_i$ , in a similar manner, we can prove that  $y_1y_2y_3y_4y_5y_6y_7y_8y_9y_{10}$  is not a correct ISBN.

Hence, a single error can be detected.

### Detection of Errors Due to Unequal-Digits Interchange in an ISBN

We now consider an error of the form

$$x_1x_2 \cdots x_i x_{i+1} \cdots x_j x_{j+1} \cdots x_9 x_{10} \rightarrow x_1 x_2 \cdots x_j x_{i+1} \cdots x_i x_{j+1} \cdots x_9 x_{10},$$

where  $x_i \neq x_j$ . That is, while typing the ISBN two unequal digits are interchanged. We show that this type of error can also be detected.

Let us first illustrate this with the help of the following example.

#### EXAMPLE 6.2.5

The correct ISBN of a book is 0-534-93189-8. While typing the ISBN the digit 3 in the third place and the digit 9 in the 9th place are interchanged. In other words, the entered ISBN is 0-594-93183-8. If it is a correct ISBN, then it must satisfy the congruence:

$$1 \cdot 0 + 2 \cdot 5 + 3 \cdot 9 + 4 \cdot 4 + 5 \cdot 9 + 6 \cdot 3 + 7 \cdot 1 + 8 \cdot 8 + 9 \cdot 3 + 10 \cdot 8 \equiv 0 \pmod{11}.$$

However,

$$\begin{aligned} & 1 \cdot 0 + 2 \cdot 5 + 3 \cdot 9 + 4 \cdot 4 + 5 \cdot 9 + 6 \cdot 3 + 7 \cdot 1 + 8 \cdot 8 + 9 \cdot 3 + 10 \cdot 8 \\ & = 10 + 27 + 16 + 45 + 18 + 7 + 64 + 27 + 80 \\ & \equiv (-1 + 5 + 5 + 1 - 4 + 7 - 2 + 5 - 8) \pmod{11} \\ & \equiv 8 \pmod{11} \\ & \not\equiv 0 \pmod{11}. \end{aligned}$$

Hence, 0-594-93183-8 is not a correct ISBN.

Let us in fact prove that the error of interchanging two unequal digits can be detected.

Because  $x_1x_2 \cdots x_i x_{i+1} \cdots x_j x_{j+1} \cdots x_9 x_{10}$  is a correct ISBN, we have

$$\begin{aligned} & 1x_1 + 2x_2 + \cdots + ix_i + (i+1)x_{i+1} + \cdots + jx_j \\ & \quad + (j+1)x_{j+1} + \cdots + 10x_{10} \equiv 0 \pmod{11}. \end{aligned}$$

Now the changed ISBN is

$$y_1y_2y_3y_4y_5y_6y_7y_8y_9y_{10},$$

where

$$\begin{aligned} y_k &= x_k, \quad \text{for all } k = 1, 2, \dots, i-1, \\ y_i &= x_j \\ y_t &= x_t, \quad \text{for all } t = i+1, \dots, j-1, \\ y_j &= x_i \\ y_k &= x_k, \quad \text{for all } k = j+1, \dots, 10. \end{aligned}$$

Now

$$\begin{aligned} &\sum_{i=1}^{10} iy_i \\ &= 1y_1 + 2y_2 + \dots + iy_i + \dots + jy_j + \dots + 10x_{10} \\ &= 1x_1 + 2x_2 + \dots + (i-1)x_{i-1} + ix_j + (i+1)x_{i+1} + \dots + (j-1)x_{j-1} \\ &\quad + jx_i + (j+1)x_{j+1} + \dots + 10x_{10} \\ &= 1x_1 + 2x_2 + \dots + (i-1)x_{i-1} + ix_i + (i+1)x_{i+1} + \dots + (j-1)x_{j-1} \\ &\quad + jx_j + (j+1)x_{j+1} + \dots + 10x_{10} - ix_i + ix_j - jx_j + jx_i \\ &= \left( \sum_{i=1}^{10} ix_i \right) - ix_i + ix_j - jx_j + jx_i \\ &\equiv (0 - ix_i + ix_j - jx_j + jx_i) \pmod{11} \\ &\equiv (i(x_j - x_i) - j(x_j - x_i)) \pmod{11} \\ &\equiv (i-j)(x_j - x_i) \pmod{11}. \end{aligned}$$

Hence,

$$\sum_{i=1}^{10} iy_i \equiv (i-j)(x_j - x_i) \pmod{11}. \quad (6.7)$$

Suppose  $y_1y_2y_3y_4y_5y_6y_7y_8y_9y_{10}$  is a correct ISBN. Then

$$\sum_{i=1}^{10} iy_i \equiv 0 \pmod{11}.$$

Thus, (6.7) implies

$$0 \equiv (i-j)(x_j - x_i) \pmod{11},$$

i.e., 11 divides  $(i-j)(x_j - x_i)$ . Now 11 is prime, so 11 divides  $i-j$  or  $x_j - x_i$ . Because  $1 \leq i \leq 10$ ,  $1 \leq j \leq 10$ , and  $i \neq j$ , 11 cannot divide  $i-j$ .

Also,  $x_i \neq x_j$  and  $1 \leq x_i \leq 10$ ,  $1 \leq x_j \leq 10$ . Therefore, 11 cannot divide  $x_j - x_i$ .

So we can conclude that  $y_1y_2y_3y_4y_5y_6y_7y_8y_9y_{10}$  is not a correct ISBN. Consequently, an error that is due to interchanging two unequal digits of an ISBN can also be detected.

Now a few words on the International Standard Serial Number (ISSN). The ISSN is used to label series publications, (magazine, newspapers) in the same way that an ISBN labels books. The ISSN is an eight-digit numerical label. The last digit is the check digit. For example, the ISSN of the journal *Proceedings of American Mathematical Society* is 0002-9939. Here the last digit, 9, is the check digit.

Let us see how to find the check digit of an ISSN, whose first seven digits are  $x_1, x_2, x_3, x_4, x_5, x_6$ , and  $x_7$ .

If  $x_8$  is the check digit, then  $x_8$  satisfies the congruence

$$8x_1 + 7x_2 + 6x_3 + 5x_4 + 4x_5 + 3x_6 + 2x_7 + x_8 \equiv 0 \pmod{11}.$$

Here  $0 \leq x_i < 10$  for  $i = 1, 2, \dots, 7$  and  $x_8$  is one of  $0, 1, 2, \dots, 9, X$ .

For the ISSN 0002-9939, we see that

$$8 \cdot 0 + 7 \cdot 0 + 6 \cdot 0 + 5 \cdot 2 + 4 \cdot 9 + 3 \cdot 9 + 2 \cdot 3 + x_8 \equiv 0 \pmod{11}.$$

Hence,

$$10 + 36 + 27 + 6 + x_8 \equiv 0 \pmod{11},$$

i.e.,

$$79 + x_8 \equiv 0 \pmod{11}.$$

Because  $0 \leq x_8 \leq 10$ , we find that  $x_8 = 9$ .

Recall that in an ISBN, the first block of numbers identifies the language or the country where the book is published. Here are the codes for some of the languages and countries.

- 0 (English) U.K., United States, Australia, New Zealand, Canada
- 1 (English) South Africa, Zimbabwe
- 2 (French) France, Belgium, Canada, Switzerland
- 3 (German) Germany, Austria, Switzerland
- 4 Japan
- 5 Russia
- 7 China
- 80 Czech Republic, and Slovakia
- 81 India

## UPC-A and EAN-13

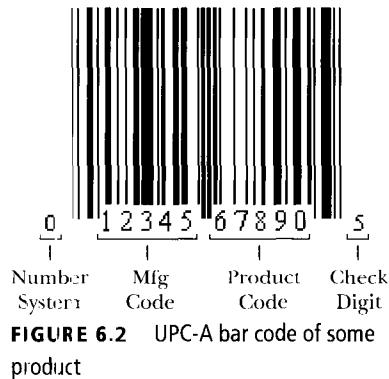
Many of the products sold in supermarkets contain identification numbers coded with bars which are read by optical scanners. In a supermarket, the cashier at the checkout counter scans the bar code of the item on the scanner and the cash register retrieves the price of the item. This bar code is called the **Universal Product Code (UPC)**.

There are different versions of UPC. The **UPC-A bar code** is by far the most common and best-known symbology, at least in the United States. A UPC-A bar code is the bar code we find on every consumer good on supermarket shelves. It also appears on books, magazines, and newspapers. It is commonly called a **UPC bar code** or **UPC symbol**.

A UPC-A consists of 11 digits (each digit in the range 0 through 9), message data along with a trailing check digit, for a total of 12 digits of bar code data. An example of a typical UPC-A bar code is given in Figure 6.2.

The digits of a UPC-A bar code are divided into four blocks:

1. The number system
2. The manufacturer code
3. The product code
4. The check digit



**FIGURE 6.2** UPC-A bar code of some product

The number-system digit is usually printed just to the left of the bar code, the check digit just to the right of the bar code, and the manufacturer and product codes just below the bar code, as shown in Figure 6.2. The number system is a single digit which identifies the type of product. For example, 3 is used for national drug or health-related products, 0 is used for groceries, and so on.

The manufacturer code, which consists of five digits, is assigned by the **Uniform Code Council (UCC)** to each manufacturer or company that distributes goods with a UPC-A identification number. A company may not randomly or without consulting the UCC choose its manufacturer number because this could quickly result in multiple manufacturers using the same code.

The product code is assigned to the product by the manufacturer. Unlike the manufacturer code, which must be assigned by the UCC, manufacturers are free to assign product codes to each of their products without consulting any other organization.

The check digit is an additional digit that is very useful in troubleshooting. It can be used, for example, to verify that a bar code is being scanned correctly. A scanner can produce incorrect data due to inconsistent scanning speed, print imperfections, or a host of other problems. The check digit helps the operator to verify whether the rest of the data in the bar code have been correctly interpreted. The check digit is calculated based on the first 11 digits of the bar code. The method of calculating the check digit will be discussed later in this section.

The International Article Numbering Association EAN assigns identification numbers to products manufactured in European countries. This identification number consists of 13 digits and is known as **EAN-13**. A typical EAN-13 bar code is shown in Figure 6.3.



**FIGURE 6.3** EAN-13 bar code of some product

An EAN-13 bar code is also divided into four blocks:

1. The number system
2. The manufacturer code

3. The product code
4. The check digit

The number system consists of two or three digits which identify the country or economic region. Normally the first digit of the number system is printed just to the left of the bar code.

The manufacturer code is assigned by the EAN to each manufacturer or company that distributes goods with an EAN-13 identification number. EAN-13 uses variable-length manufacturer codes. EAN may issue a manufacturer code whose length may be more than five digits.

The product code is assigned to the product by the manufacturer. The manufacturer and product codes are printed just below the bar code.

The check digit is an additional digit used to verify the validity of the bar code. The check digit is printed as the last digit on the right-hand side just below the bar code.

A UPC-A bar code is an EAN-13 bar code with the first EAN-13 number-system digit set to "0." This implies that any hardware or software capable of reading EAN-13 is automatically capable of reading a UPC-A.

---

**REMARK 6.2.6** ► The Uniform Code Council has announced that as of January 1, 2005, all products in the United States and Canada must be labeled with EAN-13, a big task for Information Technology professionals. All the 12-digit identification numbers used in the United States and Canada are to be replaced by 13-digit identification numbers.

---

**REMARK 6.2.7** ► George J. Laurer developed the UPC in 1973 and the EAN after that. Interested readers can read the history at his Web site.

Whether the UPC is of length 12 or 13, we are mainly interested in the check digits, which ensure the validity of the UPC. The methods of determining the check digits for UPCs of lengths of 12 or 13 are the same. Considering the recent developments, however, our discussion will focus on UPC codes of length 13, i.e., EAN-13. (We will also indicate how the check digit of UPC strings of length 12 is determined.)

A UPC of 13 digits is of the form

$$x_1 x_2 \cdots x_{12} x_{13},$$

where  $x_i$ 's are integers such that  $0 \leq x_i < 10$ . The 13th digit,  $x_{13}$ , is the check digit.

### Historical Notes

**George Laurer**  
(b. 1925)

Laurer was raised and attended school in Maryland. At a young age he contracted polio, from which he eventually recovered. As a result of his illness, however, he was still in high school when he was drafted to serve with the U.S. Army during World War II.

Upon returning home, he still lacked a high school diploma, and enrolled in a TV and radio repair school. At the insistence of one of his instructors, he earned his GED and began studies at the University of Maryland. He graduated in 1951.

Laurer began work at IBM upon graduation and had a very successful career. In 1970, some of the major elec-

tronics corporations were approached to help standardize the labeling of grocery products. In 1973, while still working for IBM, Laurer patented the UPC bar code which is still in use in grocery stores today. Laurer holds more than 25 patents, and even though he is retired, he still continues to work as a consultant.

Consider the following UPC for some product

$$8 \ 901023 \ 000904 \quad (6.8)$$

Here the 13th digit, 4, is the check digit. We now explain how this check digit is obtained.

In a UPC of length 13, the check digit,  $x_{13}$ , is an integer such that  $0 \leq x_{13} < 10$  and it satisfies the congruence:

$$x_{13} \equiv -(1x_1 + 3x_2 + 1x_3 + 3x_4 + \cdots + 1x_{11} + 3x_{12}) \pmod{10}$$

i.e.,

$$1x_1 + 3x_2 + 1x_3 + 3x_4 + \cdots + 1x_{11} + 3x_{12} + x_{13} \equiv 0 \pmod{10}. \quad (6.9)$$

Consider the UPC given in (6.8). For this UPC, the check digit must satisfy

$$\begin{aligned} x_{13} &\equiv -(1 \cdot 8 + 3 \cdot 9 + 1 \cdot 0 + 3 \cdot 1 + 1 \cdot 0 + 3 \cdot 2 + 1 \cdot 3 \\ &\quad + 3 \cdot 0 + 1 \cdot 0 + 3 \cdot 0 + 1 \cdot 9 + 3 \cdot 0) \pmod{10} \\ &= -56 \pmod{10} \\ &\equiv 4 \pmod{10}. \end{aligned}$$

This implies that  $x_{13} = 4$  because  $0 \leq x_{13} < 10$ . Hence, the check digit is 4.

### Dot Product Notation

It may be convenient if we use dot product notation, explained next, to express the congruences such as in (6.9).

If  $(a_1, a_2, \dots, a_k)$  and  $(b_1, b_2, \dots, b_k)$  are two  $k$ -tuples, then the *dot product* of these  $k$ -tuples is defined by

$$(a_1, a_2, \dots, a_k) \cdot (b_1, b_2, \dots, b_k) = a_1 b_1 + a_2 b_2 + \cdots + a_k b_k.$$

With the help of the dot product, the congruence in (6.9) can be written as

$$(1, 3, 1, 3, 1, \dots, 3, 1) \cdot (x_1, x_2, x_3, x_4, x_5, \dots, x_{12}, x_{13}) \equiv 0 \pmod{10}.$$

Let us determine the check digit of a product whose first six digits are 344207 and the next six digits are 000210. Using the dot product notation, the check digit,  $x_{13}$ , satisfies the congruence

$$(1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1) \cdot (3, 4, 4, 2, 0, 7, 0, 0, 0, 2, 1, 0, x_{13}) \equiv 0 \pmod{10}.$$

This implies that

$$3 + 12 + 4 + 6 + 0 + 21 + 0 + 0 + 0 + 6 + 1 + 0 + x_{13} \equiv 0 \pmod{10},$$

i.e.,

$$3 + 2 + 4 + 6 + 0 + 1 + 0 + 0 + 0 + 6 + 1 + x_{13} \equiv 0 \pmod{10}$$

because  $12 \equiv 2 \pmod{10}$ ,  $21 \equiv 1 \pmod{10}$ . Therefore,

$$23 + x_{13} \equiv 0 \pmod{10}.$$

Now  $0 \leq x_{13} < 10$ . Hence, the check digit,  $x_{13}$ , is 7. So we find that the UPC of this product is 344207 000210 7.

---

**REMARK 6.2.8** ► The UPC of products made in the United States starts with 00, 01, 02, ..., 09, 10, 11, 12, 13.

As we mentioned at the beginning of this section, some products in the United States and other countries use UPC-A bar coding, the 12-digit identification number. For example, in the United States, the UPC of some dishwashing soaps is 0 35000 46728 7. For this UPC, the last digit is the check digit.

A UPC  $x_1 x_2 x_3 x_4 x_5 \cdots x_{12}$  of 12 digits satisfies the following congruence

$$(3, 1, 3, 1, \dots, 3, 1) \cdot (x_1, x_2, x_3, x_4, x_5, \dots, x_{12}) \equiv 0 \pmod{10}.$$

i.e.,

$$3x_1 + 1x_2 + 3x_3 + 1x_4 + \cdots + 3x_{11} + 1x_{12} \equiv 0 \pmod{10}.$$

Let us verify that 0 35000 46728 7 is a correct 12-digit UPC. For this we compute

$$\begin{aligned} & (3, 1, 3, 1, \dots, 3, 1) \cdot (0, 3, 5, 0, 0, 0, 4, 6, 7, 2, 8, 7) \\ &= 3 \cdot 0 + 1 \cdot 3 + 3 \cdot 5 + 1 \cdot 0 + 3 \cdot 0 + 1 \cdot 0 + 3 \cdot 4 \\ &\quad + 1 \cdot 6 + 3 \cdot 7 + 1 \cdot 2 + 3 \cdot 8 + 1 \cdot 7 \\ &= 0 + 3 + 15 + 0 + 0 + 0 + 12 + 6 + 21 + 2 + 24 + 7 \\ &\equiv (3 + 5 + 2 + 6 + 1 + 2 + 4 + 7) \pmod{10} \quad \text{because } 15 \equiv 5 \pmod{10}, 12 \equiv 2 \pmod{10}, \\ &\quad \text{and } 21 \equiv 1 \pmod{10} \\ &\equiv 0 \pmod{10}. \end{aligned}$$

Hence, 0 35000 46728 7 is a correct 12-digit UPC.

### Single-Error Detection in an UPC

We can show that if a single error is made in entering the UPC in a computer, then this error can be detected.

Let  $x_1 x_2 \cdots x_{12} x_{13}$  be a 13-digit UPC. Suppose the typist entering this number into a computer types  $y_1 y_2 \cdots y_{12} y_{13}$  instead of  $x_1 x_2 \cdots x_{12} x_{13}$ , where  $y_k = x_k$  for all  $k \neq i$ , but  $y_i \neq x_i$ ; i.e., the  $i$ th digit is incorrect.

Suppose  $y_1 y_2 \cdots y_{12} y_{13}$  is a correct UPC. Then

$$\begin{aligned} & (1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1) \cdot (y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8, y_9, y_{10}, y_{11}, y_{12}, y_{13}) \\ &= 1y_1 + 3y_2 + 1y_3 + 3y_4 + 1y_5 + 3y_6 + 1y_7 + 3y_8 + 1y_9 + 3y_{10} + 1y_{11} + 3y_{12} + 1y_{13}. \end{aligned}$$

We have two cases to consider: (1) when the subscript  $i$  of  $y_i$  is odd; (2) when the subscript  $i$  of  $y_i$  is even.

First suppose the subscript  $i$  of  $y_i$  is odd.

Then

$$\begin{aligned} & (1, 3, 1, 3, 1, \dots, 3, 1) \cdot (y_1, y_2, \dots, y_{12}, y_{13}) \\ &= 1y_1 + 3y_2 + 1y_3 + \cdots + 3y_{i-1} + y_i + 3y_{i+1} + \cdots + 1y_{11} + 3y_{12} + 1y_{13} \\ &= 1x_1 + 3x_2 + 1x_3 + \cdots + 3x_{i-1} + x_i + 3x_{i+1} + \cdots + 1x_{11} + 3x_{12} + 1x_{13} + y_i - x_i \\ &\quad \text{add and subtract } x_i \\ &\equiv (0 + y_i - x_i) \pmod{10}. \end{aligned}$$

Because  $y_i \neq x_i$  and  $0 \leq x_i \leq 9$  and  $0 \leq y_i \leq 9$ , we find that

$$y_i - x_i \not\equiv 0 \pmod{10}.$$

Similarly, if the subscript  $i$  of  $y_i$  is even, then

$$\begin{aligned} & (1, 3, 1, 3, 1, \dots, 3, 1) \cdot (y_1, y_2, \dots, y_{12}, y_{13}) \equiv (0 + 3(y_i - x_i)) \pmod{10}. \\ &\quad \not\equiv 0 \pmod{10}. \end{aligned}$$

Hence,  $y_1 y_2 \cdots y_{12} y_{13}$  is not a correct UPC.

We now consider the following example.

**EXAMPLE 6.2.9**

Suppose a correct UPC for some product is 0 023942 874102. Suppose that while typing in a computer, the following number

$$0 \ 023942 \ 874012$$

is entered as a UPC (the 11th and 12th digits are interchanged). Now

$$\begin{aligned} & 1 \cdot 0 + 3 \cdot 0 + 1 \cdot 2 + 3 \cdot 3 + 1 \cdot 9 + 3 \cdot 4 + 1 \cdot 2 \\ & + 3 \cdot 8 + 1 \cdot 7 + 3 \cdot 4 + 1 \cdot 0 + 3 \cdot 1 + 1 \cdot 2 \\ & = 0 + 0 + 2 - 9 + 9 + 12 + 2 + 24 + 7 + 12 + 3 + 2 \\ & = 82 \\ & \not\equiv 0 \pmod{10} \end{aligned}$$

This implies that the UPC 0 023942 874012 is incorrect.

Notice that in this case, the error is due to the interchange of the adjacent digits 0 and 1. Also note that  $|0 - 1| \neq 5$ .

Suppose now the following number is typed:

$$0 \ 023492 \ 874102.$$

Because this is not a correct UPC, it should not satisfy the congruence

$$\begin{aligned} & 1 \cdot 0 + 3 \cdot 0 + 1 \cdot 2 + 3 \cdot 3 + 1 \cdot 4 + 3 \cdot 9 + 1 \cdot 2 \\ & + 3 \cdot 8 + 1 \cdot 7 + 3 \cdot 4 + 1 \cdot 1 + 3 \cdot 0 + 1 \cdot 2 \\ & \equiv 0 \pmod{10}. \end{aligned}$$

However,

$$\begin{aligned} & 1 \cdot 0 + 3 \cdot 0 + 1 \cdot 2 + 3 \cdot 3 + 1 \cdot 4 + 3 \cdot 9 + 1 \cdot 2 \\ & + 3 \cdot 8 + 1 \cdot 7 + 3 \cdot 4 + 1 \cdot 1 + 3 \cdot 0 + 1 \cdot 2 \\ & = 0 + 0 + 2 + 9 + 4 + 27 + 2 + 24 + 7 + 12 + 1 + 2 \\ & = 90 \\ & \equiv 0 \pmod{10}. \end{aligned}$$

So the error is not detected here.

Observe that in this case, the error is due to the interchange of the adjacent digits 9 and 4, and  $|9 - 4| = 5$ .

From Example 6.2.9, we observe that the undetected errors due to the interchange of adjacent digits  $a_i$  and  $a_{i+1}$  arise when  $|a_i - a_{i+1}| = 5$ . To verify this, we can prove the following: In a UPC, a transposition error of the form

$$a_1 a_2 \cdots a_i a_{i+1} \cdots a_{12} a_{13} \rightarrow a_1 a_2 \cdots a_{i+1} a_i \cdots a_{12} a_{13}$$

is undetected if and only if  $|a_i - a_{i+1}| = 5$ .

We have already discussed the ISBN, which is an identification number for a book, and the UPC, which is an identification number for products sold in the market. We have seen that a correct ISBN  $a_1 a_2 a_3 \cdots a_{10}$  of some book satisfies the congruence

$$(a_1, a_2, \dots, a_{10}) \cdot (w_1, w_2, \dots, w_{10}) \equiv 0 \pmod{11},$$

where  $(w_1, w_2, \dots, w_{10}) = (1, 2, \dots, 10)$ .

The 10-tuple  $(1, 2, \dots, 10)$  is called the **weighting vector** of the ISBN, and the weighting vector for the UPC is the 13-tuple  $(1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1)$ . The following theorem describes the relationship between the weighting vector and its ability to detect errors.

**Theorem 6.2.10:** Suppose an identification number  $a_1 a_2 \cdots a_k$  satisfies

$$(a_1, a_2, \dots, a_k) \cdot (w_1, w_2, \dots, w_k) \equiv 0 \pmod{n},$$

where  $0 \leq a_i < n$  for each  $i$ . Then all single-digit errors in the  $i$ th place are detected if and only if  $w_i$  is relatively prime to  $n$ .

**Theorem 6.2.11:** Suppose that an identification number  $a_1 a_2 \cdots a_k$  satisfies

$$(a_1, a_2, \dots, a_k) \cdot (w_1, w_2, \dots, w_k) \equiv 0 \pmod{n},$$

where  $0 \leq a_i < n$  for each  $i$ . Then all errors of the form

$$\cdots a_i a_{i+1} \cdots a_j a_{j+1} \cdots \rightarrow \cdots a_j a_{i+1} \cdots a_i a_{j+1} \cdots$$

(the digits in the  $i$ th and  $j$ th positions are different and interchanged) are detected if  $w_i - w_j$  is relatively prime to  $n$ .

---

**REMARK 6.2.12** ► The UPC (EAN-13) of products made in France starts with 30, 31, ..., 36, 37; the UPC of products made in Japan starts with 45, 49; the UPC of products made in Taiwan starts with 471; the UPC of products made in India starts with 890; the UPC of products made in Sri Lanka starts with 479; the UPC of products made in China starts with 690, 691, or 692; the UPC of products made in Singapore starts with 888, and so on.

## Check Digit in Credit Cards

This section describes how check digits are assigned in credit cards and shows how they are used to verify the validity of credit cards.

We first note that the identification numbers of different credit cards have different lengths and different prefixes. For a Master Card, the identification number consists of 16 digits and the number starts with 51, 52, 53, 54, or 55. For a Visa card, the identification number consists of 13 or 16 digits and the number starts with the digit 4.

Both credit cards use congruence mod 10 to determine the check digit, and in all the cases the check digit is the rightmost digit in the number.

Consider a credit card with the following identification number:

5548 3742 7983 0696

Here the first two digits indicate that this is a Master Card.

In the Master Card,

1. If the digit in the second place is a 1, then the digits from the 2nd place to the 3rd place represent the bank number.
2. If the digit in the second place is a 2, then the digits from the 2nd place to the 4th place represent the bank number.
3. If the digit in the second place is a 3, then the digits from the 2nd place to the 5th place represent the bank number.
4. If the digit in the second place is any digit other than 1, 2, or 3, then the digits from 2nd place to 6th place represent the bank number.

For example, in the Master Card number 5548 3742 7983 0696, the second digit is 5. Hence, the digits from 2nd place to 6th place denote the bank number; i.e., the bank number is 54837.

The digits after the bank number up to 15th place are the account number of the card holder. For the above card, the account number of the card holder is 42 7983 069.

Finally, the digit in the 16th place is the check digit.

In the case of Visa cards, digits from 2nd place to 6th place denote the identification number of the bank, digits from 7th place to 12th place or 7th place to 15th place denote the account number, and the digit in the 13th or 16th place is the check digit.

The check digit  $a_k$  of the identification number  $a_1 a_2 \cdots a_{k-1} a_k$ , where  $a_i$ 's are integers such that  $0 \leq a_i < 10$ , of the Master Card or the Visa card is obtained from the following congruence:

If  $k$  is even, then

$$((a_1, a_2, \dots, a_{k-1}) \cdot (2, 1, 2, 1, 2, \dots, 2) + r) + a_k \equiv 0 \pmod{10}.$$

If  $k$  is odd, then

$$((a_1, a_2, \dots, a_{k-1}) \cdot (1, 2, 1, 2, 1, 2, \dots, 2) + r) + a_k \equiv 0 \pmod{10},$$

where  $r$  is the number of terms in the dot product greater than or equal to 10.

To determine the check digit for Master Card and Visa, we use the following algorithm.

- Step 1.** Starting from the second digit from the *right* and moving toward the *left*, multiply every alternate digit by 2.
- Step 2.** Add the individual digits comprising the products obtained in step 1.
- Step 3.** Add all of the digits of the card not multiplied by 2 in step 1.
- Step 4.** Add the results obtained in steps 2 and 3.
- Step 5.** If  $s$  is the sum obtained in step 4, then solve the congruence  $s \equiv 0 \pmod{10}$  to obtain the check digit  $a_k$ , where  $0 \leq a_k < 10$ .

We can determine if the check digit of a given credit card is valid by using steps 1–5. If the credit card number is valid, then the sum  $s$  obtained in step 4 must satisfy  $s \equiv 0 \pmod{10}$ .

Let us explain this algorithm with the help of the following example.

#### EXAMPLE 6.2.13

Consider a credit card with the following identification number:

5546 1997 2335 5004

Here the first two digits indicate that this is a Master Card. We now check the validity of this card.

**Step 1.** Starting from the second digit from the right and moving toward the left, multiply every alternate digit by 2. (The digits to be multiplied by 2 are underlined 5546 1997 2335 504.)

$$\begin{array}{llll} 0 \cdot 2 = 0, & 5 \cdot 2 = 10, & 3 \cdot 2 = 6, & 2 \cdot 2 = 4, \\ 9 \cdot 2 = 18, & 1 \cdot 2 = 2, & 4 \cdot 2 = 8, & 5 \cdot 2 = 10. \end{array}$$

**Step 2.** Add the individual digits comprising the products obtained in step 1. (For example, the product  $0 \cdot 2 = 0$ , so the sum of the digits of the product 0 is 0. The product  $5 \cdot 2 = 10$ , so the sum of the digits of the product 10 is  $1 + 0 = 1$ . Similarly, the product  $9 \cdot 2 = 18$ , so the sum of the digits in this product is  $1 + 8 = 9$ .)

$$0 + 1 + 6 + 4 + 9 + 2 + 8 + 1 = 31$$

**Step 3.** Add all of the digits of the card not multiplied by 2 in step 1.

$$4 + 0 + 5 + 3 + 7 + 9 + 6 + 5 = 39$$

**Step 4.** Add the results from steps 2 and 3.

$$31 + 39 = 70$$

Now  $70 \equiv 0 \pmod{10}$ . Hence, the identification number of this card is valid.

## WORKED-OUT EXERCISES

**Exercise 1:** Let  $0-669-19496-x_{10}$  be the ISBN of a book. Find the check digit.

**Solution:** If  $x_1 x_2 \dots x_{10}$  is an ISBN, then

$$\sum_{i=1}^9 i x_i \equiv x_{10} \pmod{11}.$$

Hence,

$$\begin{aligned} & 1 \cdot 0 + 2 \cdot 6 + 3 \cdot 6 + 4 \cdot 9 + 5 \cdot 1 \\ & + 6 \cdot 9 + 7 \cdot 4 + 8 \cdot 9 + 9 \cdot 6 \\ & \equiv x_{10} \pmod{11}, \end{aligned}$$

i.e.,

$$\begin{aligned} & 12 + 18 + 36 + 5 + 54 + 28 + 72 + 54 \\ & \equiv x_{10} \pmod{11}. \end{aligned} \tag{6.10}$$

Now  $12 \equiv 1 \pmod{11}$ ,  $18 \equiv 7 \pmod{11}$ ,  $36 \equiv 3 \pmod{11}$ ,  $54 \equiv -1 \pmod{11}$ ,  $28 \equiv 6 \pmod{11}$ ,  $72 \equiv -5 \pmod{11}$ . Hence, from (6.10)

$$1 + 7 + 3 + 5 - 1 + 6 - 5 - 1 \equiv x_{10} \pmod{11},$$

i.e.,

$$\begin{aligned} & 15 \equiv x_{10} \pmod{11} \\ & \Rightarrow 4 \equiv x_{10} \pmod{11}. \end{aligned}$$

Because  $0 \leq x_{10} \leq 10$ , we find that the check digit is 4.

**Exercise 2:** Let  $3-540-05329-x_{10}$  be the ISBN of a book. Find the check digit.

**Solution:** The check digit,  $x_{10}$ , is one of the digits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, X and it satisfies

$$\begin{aligned} & 1 \cdot 3 + 2 \cdot 5 + 3 \cdot 4 + 4 \cdot 0 + 5 \cdot 0 \\ & + 6 \cdot 5 + 7 \cdot 3 + 8 \cdot 2 + 9 \cdot 9 \\ & \equiv x_{10} \pmod{11}. \end{aligned}$$

or

$$3 + 10 + 12 + 0 + 0 + 30 + 21 + 16 + 81 \equiv x_{10} \pmod{11}.$$

Now  $10 \equiv -1 \pmod{11}$ ,  $12 \equiv 1 \pmod{11}$ ,  $30 \equiv -3 \pmod{11}$ ,  $21 \equiv -1 \pmod{11}$ ,  $16 \equiv 5 \pmod{11}$ , and  $81 \equiv 4 \pmod{11}$ .

$$\begin{aligned} & 3 + 10 + 12 + 0 + 0 + 30 + 21 + 16 + 81 \\ & \equiv 3 - 1 + 1 - 3 - 1 + 5 + 4 \\ & \equiv 8 \pmod{11}. \end{aligned}$$

Therefore,  $x_{10} = 8$ . Hence, the ISBN of this book is 3-540-05329-8.

**Exercise 3:** Determine whether the following ISBNs are valid.

$$(a) 3-540-19102-X \quad (b) 0-201-15668-3$$

**Solution:**

(a) The correct ISBN  $x_1x_2 \cdots x_9x_{10}$  must satisfy

$$\sum_{i=1}^{10} ix_i \equiv 0 \pmod{11}.$$

Here

$$\begin{aligned} & 1 \cdot 3 + 2 \cdot 5 + 3 \cdot 4 + 4 \cdot 0 + 5 \cdot 1 + 6 \cdot 9 \\ & + 7 \cdot 1 + 8 \cdot 0 + 9 \cdot 2 + 10 \cdot 10 \\ = & 3 + 10 + 12 + 0 + 5 + 54 + 7 + 0 + 18 + 100 \\ \equiv & (3 - 1 + 1 + 0 + 5 - 1 + 7 + 0 + 7 + 1) \pmod{11} \\ & \text{because } 10 \equiv -1 \pmod{11}, \\ & 12 \equiv 1 \pmod{11}, 54 \equiv -1 \pmod{11}, \\ & 18 \equiv 7 \pmod{11}, \text{ and } 100 \equiv 1 \pmod{11} \\ \equiv & 22 \pmod{11} \\ \equiv & 0 \pmod{11} \quad \text{because } 22 \equiv 0 \pmod{11} \end{aligned}$$

Hence, 3-540-19102-X is a valid ISBN.

(b) If 0-201-15668-3 is a correct ISBN, the following congruence must hold:

$$\begin{aligned} & 1 \cdot 0 + 2 \cdot 2 + 3 \cdot 0 + 4 \cdot 1 + 5 \cdot 1 \\ & + 6 \cdot 5 + 7 \cdot 6 + 8 \cdot 6 + 9 \cdot 8 + 10 \cdot 3 \\ \equiv & 0 \pmod{11}. \end{aligned}$$

Let us evaluate the left side. We have

$$\begin{aligned} & 1 \cdot 0 + 2 \cdot 2 + 3 \cdot 0 + 4 \cdot 1 + 5 \cdot 1 \\ & + 6 \cdot 5 + 7 \cdot 6 + 8 \cdot 6 + 9 \cdot 8 + 10 \cdot 3 \\ = & 4 + 4 + 5 + 30 + 42 + 48 + 72 + 30 \\ \equiv & (4 + 4 + 5 - 3 - 2 + 4 - 5 - 3) \pmod{11} \\ & \text{because } 30 \equiv -3 \pmod{11}, 42 \equiv -2 \pmod{11}, \\ & 48 \equiv 4 \pmod{11}, 72 \equiv -5 \pmod{11} \\ \equiv & 4 \pmod{11}. \end{aligned}$$

This implies that the given ISBN is not valid.

**Exercise 4:** Suppose that one digit, indicated with a question mark in the ISBN 80-203-0?71-9, cannot be read. Find this missing digit.

**Solution:** Let  $x_7$  be the missing digit. Then

$$\begin{aligned} & 1 \cdot 8 + 2 \cdot 0 + 3 \cdot 2 + 4 \cdot 0 + 5 \cdot 3 \\ & + 6 \cdot 0 + 7 \cdot x_7 + 8 \cdot 7 + 9 \cdot 1 + 10 \cdot 9 \\ \equiv & 0 \pmod{11}. \end{aligned}$$

This implies that

$$\begin{aligned} & (8 + 6 + 15 + 7 \cdot x_7 + 56 + 9 + 90) \equiv 0 \pmod{11} \\ \Rightarrow & 8 + 6 + 4 + 7 \cdot x_7 + 1 + 9 + 2 \equiv 0 \pmod{11} \\ \Rightarrow & 30 + 7 \cdot x_7 \equiv 0 \pmod{11} \\ \Rightarrow & 7 \cdot x_7 \equiv -30 \pmod{11}. \end{aligned}$$

Because  $x_7$  is an integer,  $0 \leq x_7 \leq 9$ , and  $7 \cdot x_7 \equiv -30 \pmod{11} \equiv 3 \pmod{11}$ , we find that  $x_7 = 2$ .

**Exercise 5:** The identification of a company in India is 890103. The identification number of some product of this company is 114519. Write the UPC of this product.

**Solution:** Let  $x_{13}$  be the check digit. Then the UPC of the product is 890103 114519  $x_{13}$ . Now the check digit satisfies the congruence

$$\begin{aligned} & (1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1) \\ & \cdot (8, 9, 0, 1, 0, 3, 1, 1, 4, 5, 1, 9, x_{13}) \\ \equiv & 0 \pmod{10}. \end{aligned}$$

Now

$$\begin{aligned} & (1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1) \\ & \cdot (8, 9, 0, 1, 0, 3, 1, 1, 4, 5, 1, 9, x_{13}) \\ = & 8 + 27 + 0 + 3 + 0 + 9 + 1 + 3 + 4 + 15 + 1 + 27 + x_{13} \\ \equiv & (8 + 7 + 3 + 9 + 1 + 3 + 4 + 5 + 1 + 7 + x_{13}) \pmod{10} \\ & \text{because } 27 \equiv 7 \pmod{10} \text{ and } 15 \equiv 5 \pmod{10} \\ \equiv & (48 + x_{13}) \pmod{10}. \end{aligned}$$

Because  $x_{13}$  is an integer,  $0 \leq x_{13} \leq 9$ , we find that  $x_{13} = 2$ . Hence, the UPC is 890103 114519 2.

**Exercise 6:** Is 0 023492 874102 a correct UPC for some product?

**Solution:** 0 023492 874102 is a correct UPC if it satisfies the congruence

$$\begin{aligned} & (1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1) \\ & \cdot (0, 0, 2, 3, 4, 9, 2, 8, 7, 4, 1, 0, 2) \\ \equiv & 0 \pmod{10}. \end{aligned}$$

Now

$$\begin{aligned} & (1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1) \\ & \cdot (0, 0, 2, 3, 4, 9, 2, 8, 7, 4, 1, 0, 2) \\ = & 1 \cdot 0 + 3 \cdot 0 + 1 \cdot 2 + 3 \cdot 3 + 1 \cdot 4 + 3 \cdot 9 + 1 \cdot 2 \\ & + 3 \cdot 8 + 1 \cdot 7 + 3 \cdot 4 + 1 \cdot 1 + 3 \cdot 0 + 1 \cdot 2 \\ \equiv & (2 + 9 + 4 + 7 + 2 + 4 + 7 + 2 + 1 + 0 + 2) \pmod{10} \\ & \text{because } 27 \equiv 7 \pmod{10}, \\ & 24 \equiv 4 \pmod{10} \text{ and } 12 \equiv 2 \pmod{10} \\ \equiv & 40 \pmod{10} \\ \equiv & 0 \pmod{10}. \end{aligned}$$

Hence, the given identification number is a correct UPC.

**Exercise 7:** In the 12-digit UPC 0724 8903 004-?, of a product, what is the 12th digit, i.e., the check digit?

**Solution:** For the given product, let  $x_{12}$  be the check digit. Then the UPC of the product is

$$07248903004x_{12}.$$

Now this UPC must satisfy the congruence

$$\begin{aligned} & 3 \cdot 0 + 1 \cdot 7 + 3 \cdot 2 + 1 \cdot 4 + 3 \cdot 8 + 1 \cdot 9 \\ & + 3 \cdot 0 + 1 \cdot 3 + 3 \cdot 0 + 1 \cdot 0 + 3 \cdot 0 + 1 \cdot x_{12} \\ & \equiv 0 \pmod{10}. \end{aligned}$$

This implies

7 + 6 + 4 + 24 + 9 + 3 + x\_{12} \equiv 0 \pmod{10}
$$\Rightarrow 7 + 6 + 4 + 4 + 9 + 3 + x_{12} \equiv 0 \pmod{10}$$

because  $24 \equiv 4 \pmod{10}$

$$\Rightarrow 33 + x_{12} \equiv 0 \pmod{10}.$$

This implies  $x_{12} = 7$ . Hence, the UPC of the product is 072489030047.

**Exercise 8:** The identification number of Mr. Raj's credit card is the following:

$$5368 \underline{2}358 \underline{9}683 \underline{1}135.$$

- (a) Is this credit card a Master Card or a Visa card?
- (b) What is the bank number of this card?
- (c) What is the account number of this card?
- (d) What is the check digit of this card?
- (e) Is this identification number valid?

**Solution:**

- (a) The first digit of this number is 5, and the second digit is 3. Hence, this is a Master Card.
- (b) The second digit is 3. Hence, the bank number is 3682.
- (c) The account number of this card is 3589683113.
- (d) The check digit is 5.

**Step 1.** Starting from the second digit from the right and moving toward the left, multiply every alternate digit by 2. (We have underlined the digits to be multiplied by 2 : 5368 2358 9683 1135.)

$$\begin{aligned} 2 \cdot 3 &= 6, & 2 \cdot 1 &= 2, & 2 \cdot 8 &= 16, \\ 2 \cdot 9 &= 18, & 2 \cdot 5 &= 10, & 2 \cdot 2 &= 4, \\ 2 \cdot 6 &= 12, & 2 \cdot 5 &= 10 & & \end{aligned}$$

**Step 2.** Add the individual digits in each product obtained in step 1:

$$\begin{aligned} 6 + 2 + (1 + 6) + (1 + 8) + (1 + 0) \\ + 4 + (1 + 2) + (1 + 0) = 33 \end{aligned}$$

**Step 3.** Add all of the digits of the card not multiplied by 2 in step 1:

$$3 + 8 + 3 + 8 + 6 + 3 + 1 + 5 = 37$$

**Step 4.** Add the results of steps 2 and 3:

$$37 + 33 = 70 \equiv 0 \pmod{10}.$$

Hence, the identification number of the card is valid.

**Exercise 9:** The first 15 digits of a Visa card are 4550 3891 6078 001. Find the check digit for this card.

**Solution:** To determine the check digit, we follow the algorithm.

Let  $a_{16}$  be the check digit of this card. Then the identification number of the card is 4550 3891 6078 001  $a_{16}$ .

**Step 1.** Starting from the second digit from the right and moving toward the left, multiply every alternate digit by 2. (We have underlined the digits to be multiplied by 2 : 4550 3891 6078 001  $a_{16}$ .)

$$\begin{aligned} 2 \cdot 1 &= 2, & 0 \cdot 2 &= 0, & 7 \cdot 2 &= 14, & 6 \cdot 2 &= 12, \\ 9 \cdot 2 &= 18, & 3 \cdot 2 &= 6, & 5 \cdot 2 &= 10, & 4 \cdot 2 &= 8 \end{aligned}$$

**Step 2.** Add the individual digits of each product obtained in step 1.

$$\begin{aligned} 2 + 0 + (1 + 4) + (1 + 2) \\ + (1 + 8) + 6 + (1 + 0) + 8 = 34 \end{aligned}$$

**Step 3.** Add all of the digits of the card not multiplied by 2 in step 1.

$$5 + 0 + 8 + 1 + 0 + 8 + 0 + a_{16} = 22 + a_{16}$$

**Step 4.** Add the results of steps 2 and 3.

$$34 + 22 + a_{16}$$

**Step 5.**

$$34 + 22 + a_{16} \equiv 0 \pmod{10}.$$

Then,

$$56 + a_{16} = 0 \pmod{10}.$$

This implies that  $a_{16} = 4$ . Hence, the check digit is 4.

## SECTION REVIEW

---

### Key Terms

International Standard Book Number

UPC symbol

check digit

Uniform Code Council (UCC)

Universal Product Code (UPC)

EAN-13

UPC-A bar code

weighting vector

UPC bar code

### Some Key Definitions

1. The ISBN of a book is a string of 10 digits,  $x_1x_2x_3 \cdots x_9x_{10}$ , where  $x_i \in \{0, 1, 2, \dots, 9\}$ ,  $1 \leq i \leq 9$ , and  $x_{10} \in \{0, 1, 2, \dots, 9, X\}$ .
2. The check digit  $x_{10}$ , which is one of  $0, 1, 2, 3, \dots, 9, X$ , is determined by the following congruence

$$1x_1 + 2x_2 + \cdots + 8x_8 + 9x_9 \equiv x_{10} \pmod{11}.$$

3. A UPC of 13 digits is of the form  $x_1x_2 \cdots x_{12}x_{13}$ , where  $x_i$ 's are integers such that  $0 \leq x_i < 10$ . The 13th digit,  $x_{13}$ , is the check digit.
4. In a UPC of length 13, the check digit,  $x_{13}$ , is an integer such that  $0 \leq x_{13} < 10$ , and it satisfies the congruence:

$$1x_1 + 3x_2 + 1x_3 + 3x_4 + \cdots + 1x_{11} + 3x_{12} + x_{13} \equiv 0 \pmod{10}.$$

5. If  $(a_1, a_2, \dots, a_k)$  and  $(b_1, b_2, \dots, b_k)$  are two  $k$ -tuples, then the dot product of these  $k$ -tuples is defined by

$$(a_1, a_2, \dots, a_k) \cdot (b_1, b_2, \dots, b_k) = a_1b_1 + a_2b_2 + \cdots + a_kb_k.$$

6. A UPC  $x_1x_2x_3x_4x_5 \cdots x_{12}$  of 12 digits satisfies the following congruence:

$$3x_1 + 1x_2 - 3x_3 + 1x_4 + \cdots + 3x_{11} + 1x_{12} \equiv 0 \pmod{10}.$$

7. In a Master Card, the identification number consists of 16 digits, and the number starts with 51, 52, 53, 54, or 55.
8. A Visa card is of length 13 or 16, and the identification number starts with the digit 4.

### Some Key Results

1. A single error in an ISBN can be detected.
2. The error in an ISBN due to the interchange of two unequal digits can also be detected.
3. A UPC-A consists of 11 digits (each digit in the range 0 through 9), message data along with a trailing check digit, for a total of 12 digits of bar code data.
4. Suppose an identification number  $a_1a_2 \cdots a_k$  satisfies

$$(a_1, a_2, \dots, a_k) \cdot (w_1, w_2, \dots, w_k) \equiv 0 \pmod{n},$$

where  $0 \leq a_i < n$  for each  $i$ . Then all single-digit errors in the  $i$ th place are detected if and only if  $w_i$  is relatively prime to  $n$ .

5. Suppose that an identification number  $a_1 a_2 \cdots a_k$  satisfies

$$(a_1, a_2, \dots, a_k) \cdot (w_1, w_2, \dots, w_k) \equiv 0 \pmod{n},$$

where  $0 \leq a_i < n$  for each  $i$ . Then all errors of the form

$$\cdots a_i a_{i+1} \cdots a_j a_{j+1} \cdots \rightarrow \cdots a_j a_{i+1} \cdots a_i a_{j+1} \cdots$$

(the digits in the  $i$ th and  $j$ th positions are different and interchanged) are detected if and only if  $w_i - w_j$  is relatively prime to  $n$ .

6. To determine the check digit for Master Card and Visa we use the following algorithm.

**Step 1.** Starting from the second digit from the *right* and moving toward the *left*, multiply every alternate digit by 2.

**Step 2.** Add the individual digits comprising the products obtained in step 1.

**Step 3.** Add all of the digits of the card not multiplied by 2 in step 1.

**Step 4.** Add the results obtained in steps 2 and 3.

**Step 5.** If  $s$  is the sum obtained in step 4, then solve the congruence  $s \equiv 0 \pmod{10}$  to obtain the check digit  $x_k$ , where  $0 \leq a_k < 10$ .

## EXERCISES

---

1. Let  $0-85312-612-x_{10}$  be the ISBN of a book. Find the check digit.
  2. Let  $3-540-05329-x_{10}$  be the ISBN of a book. Find the check digit.
  3. Let  $0-201-06561-x_{10}$  be the ISBN of a book. Find the check digit.
  4. Find the correct check digit, indicated with ?, for each of the following incomplete ISBNs.
    - a. 0-201-15768-? b. 0-02-360721-?
    - c. 0-07-040035-? d. 81-7319-077-?
    - e. 0-534-00837-? f. 0-669-19496-?
    - g. 2-512-43005-? h. 0-8273-5727-?
    - i. 3-540-19102-? j. 3-540-78053-?
  5. Determine whether the following ISBNs are valid.
    - a. 0-619-015919-7 b. 0-201-55540-8
    - c. 0-201-70982-1 d. 3-240-19102-X
    - e. 81-3719-077-1 f. 0-07-115468-X
  6. Suppose that the digit indicated with a ? in the following ISBNs cannot be read. Find the correct digit.
    - a. 3-41?-27840-5 b. 81-7319-?-13-4
    - c. 0-14?-13474-3 d. 0-0?-112690-2
    - e. 8?-224-0399-9 f. 0-07-115468-?
  7. Show that  $x_1 x_2 \cdots x_9 x_{10}$  is a correct ISBN with the check digit  $x_{10}$  if and only if
- $10x_1 + 9x_2 + 8x_3 + \cdots + 2x_9 + x_{10} \equiv 0 \pmod{11}.$
8. Offer two alternative correct ISBNs for the following incorrect one without changing the first block and the check digit: 0-13-122234-3.
  9. Is 9 780023 627219 a correct UPC for some product?
  10. Find the correct check digit for each of the following incomplete UPCs.
    - a. 6 902744 21142
    - b. 8 901023 00003
    - c. 3 423470 31136
  11. Find the correct check digit for each of the following 12-digit incomplete UPCs.
    - a. 0246 0001 101
    - b. 0808 7801 122
    - c. 0741 0161 120
  12. Show that 3 523470 311236 is not a valid UPC.
  13. Show that 0307 3760 0081 is not a valid 12-digit UPC.
  14. The identification of a company in India is 890102. The identification of a product of that company is 300090. What is the UPC of this product?
  15. The identification of a company in France is 344207. The company gives the identification number 000310 to a particular product of this company. What is the UPC of this product?

16. Find the check digits of the following UPC.

	Company Code	Product Code	Check Digit
(a)	890106	321107	—
(b)	890103	013660	—
(c)	890151	210050	—
(d)	344207	000320	—

17. The identification number of Heather's credit card is the following.

4563 9810 3862 5408

- a. Is this credit card a Master Card or a Visa card?
- b. What is the bank number of this card?

- c. What is the account number of this card?
  - d. What is the check digit of this card?
  - e. Is this identification number valid?
18. The first 15 digits of a Master Card are 5548 3742 7983 069. Find the check digit for this card.
19. Check the validity of the following identification numbers of the Master Card.
- a. 5549 3742 7983 0696
  - b. 5431 8620 0022 9428
  - c. 5546 1992 3697 5009
20. Check the validity of the following identification numbers of the Visa card.
- a. 4129 0370 8108 3000
  - b. 4563 9811 3862 5408
  - c. 4520 3891 6078 0014

## 6.3 LINEAR CONGRUENCES

Students learn in an algebra course that the equation  $ax = b$ ,  $a, b \in \mathbb{Z}$  and  $a \neq 0$ , has a solution in  $\mathbb{Z}$  if and only if  $a$  divides  $b$  in  $\mathbb{Z}$ . In this section, we discuss the solution of congruence  $ax \equiv b \pmod{m}$  in  $\mathbb{Z}$ .

**DEFINITION 6.3.1** ▶ A congruence of the form

$$ax \equiv b \pmod{m}, \quad (6.11)$$

where  $a$  and  $b$  are integers,  $m$  is a positive integer, and  $x$  is an unknown integer, is called a **linear congruence in one variable  $x$** . An integer  $x_0$  is called a solution of (6.11) if  $ax_0 \equiv b \pmod{m}$ .

### EXAMPLE 6.3.2

Consider the linear congruence  $4x \equiv 1 \pmod{9}$  in one variable. Now  $4 \cdot 7 \equiv 1 \pmod{9}$ , so it follows that 7 is a solution of this congruence. Notice that  $16 \equiv 7 \pmod{9}$ . Therefore, 16 is also a solution of  $4x \equiv 1 \pmod{9}$ .

In fact, we can show that if  $x_0$  is an integer such that  $x_0 \equiv 7 \pmod{9}$ , i.e.,  $x_0$  is an element of the congruence class  $[7]$ , then  $x_0$  is solution of the congruence. Notice that  $[x_0] = [7]$ . The following theorem proves such as a result.

**Theorem 6.3.3:** Let  $a$ ,  $b$ , and  $m$  be integers with  $m > 0$ . Suppose that  $x_0$  is a solution of the linear congruence  $ax \equiv b \pmod{m}$ . Then any member of the class  $[x_0]$  is a solution of this linear congruence.

**Proof:** Now  $x_0$  is a solution of  $ax \equiv b \pmod{m}$ , so  $ax_0 \equiv b \pmod{m}$ . Suppose  $x_1 \in [x_0]$ . Then  $x_1 \equiv x_0 \pmod{m}$ . This implies that  $ax_1 \equiv ax_0 \pmod{m}$  by Corollary 6.1.17(ii).

From  $ax_0 \equiv b \pmod{m}$  and  $ax_1 \equiv ax_0 \pmod{m}$ , we can conclude that  $ax_1 \equiv b \pmod{m}$ . So we find that  $x_1$  is also a solution of this congruence. Hence, any member of the congruence class  $[x_0]$  is a solution. ■

It is not necessary that every linear congruence has a solution. Consider the following example.

**EXAMPLE 6.3.4**

Consider the linear congruence  $7x \equiv 4 \pmod{14}$ . Suppose that this linear congruence has a solution, say  $x' \in \mathbb{Z}$ . This implies that  $7x' \equiv 4 \pmod{14}$ . Now

$$\begin{aligned} 7x' &\equiv 4 \pmod{14} \\ \Rightarrow 14 &\text{ divides } 7x' - 4 \\ \Rightarrow 7x' - 4 &= 14t \quad \text{for some integer } t \\ \Rightarrow -4 &= 14t - 7x' \\ \Rightarrow -\frac{4}{7} &= 2t - x' \in \mathbb{Z}, \end{aligned}$$

which is a contradiction. Hence, the linear congruence  $7x \equiv 4 \pmod{14}$  has no integral solutions.

**EXAMPLE 6.3.5**

Consider the congruence  $15x \equiv 7 \pmod{8}$ .

Now  $\gcd(15, 8) = 1$ . Thus, by Theorem 2.1.19, there exist integers  $u$  and  $t$  such that  $15u + 8t = 1$ . This implies that  $15 \cdot 7u + 8 \cdot 7t = 7$ . Thus,  $15 \cdot 7u - 7 = 8(-7t)$ , so 8 divides  $15 \cdot 7u - 7$ . Hence,  $15 \cdot 7u \equiv 7 \pmod{8}$ . This implies that  $7u$  is a solution of this congruence. (Notice that in  $15u + 8t = 1$ , we can take  $u = -1$  and  $t = 2$ ).

By Theorem 6.3.3, the solutions of the given congruence are precisely the elements of class  $[7u]$  modulo 8. Here, we can take  $u = -1$ . So the solutions set is the class  $[-7] = [1]$ .

The preceding example indicates that in the congruence  $ax \equiv b \pmod{m}$  if  $\gcd(a, m) = 1$ , then the congruence has a solution. Indeed, this is the case as proved in the following theorem.

**Theorem 6.3.6:** Let  $a, b$ , and  $m$  be integers with  $m > 0$  and  $\gcd(a, m) = 1$ .

Then the congruence  $ax \equiv b \pmod{m}$  has a solution. Further, if  $x_0$  is a solution, then the set of all solutions is precisely the equivalence class  $[x_0]$ .

Thus, we say that the solution is **unique modulo  $m$** .

**Proof:** Because  $\gcd(a, m) = 1$ , by Theorem 2.1.19, there exist integers  $u$  and  $t$  such that  $au + mt = 1$ . This implies that  $aub + mtb = b$ . Thus,  $aub - b = m(-tb)$ , so  $m$  divides  $aub - b$ . So we find that

$$aub \equiv b \pmod{m}.$$

This implies that  $x_0 = ub$  is a solution of  $ax \equiv b \pmod{m}$ .

Suppose  $x_0$  is a solution of the given congruence. Let  $y_0$  be another solution of  $ax \equiv b \pmod{m}$ . Then  $ay_0 \equiv b \pmod{m}$ .

Now  $ax_0 \equiv b \pmod{m}$  and  $ay_0 \equiv b \pmod{m}$ , i.e.,  $ax_0 \equiv b \pmod{m}$  and  $b \equiv ay_0 \pmod{m}$ . Hence,  $ax_0 \equiv ay_0 \pmod{m}$ . Because  $\gcd(a, m) = 1$ , by Theorem 6.1.20(ii), it follows that  $x_0 \equiv y_0 \pmod{m}$ . This implies that  $y_0 \in [x_0]$ .

Also, it is easy to see that any member of the class  $[x_0]$  is a solution of the given congruence. Thus, the set of all solutions is precisely the equivalence class  $[x_0]$ . ■

**Corollary 6.3.7:** Let  $a$ ,  $b$ , and  $p$  be integers such that  $p$  is prime and  $p \nmid a$ . Then the congruence  $ax \equiv b \pmod{p}$  has a solution that is unique modulo  $p$ .

**Proof:** Because  $p \nmid a$ ,  $\gcd(a, p) = 1$ . Hence, the corollary follows from Theorem 6.3.6. ■

**EXAMPLE 6.3.8**

In this example, we solve the linear congruence  $18x \equiv 5 \pmod{7}$ .

Now  $\gcd(18, 7) = 1$ , so by Theorem 6.3.6, this congruence has a unique solution modulo 7. Notice that  $x = 3$  is a solution of this congruence. Thus, by Theorem 6.3.6, the solution set is the class [3].

**DEFINITION 6.3.9** ▶ Let  $m$  be a positive integer. For any integer  $a$  with  $\gcd(a, m) = 1$ , an integer  $b$  is called an **inverse** of  $a$  modulo  $m$  if

$$ab \equiv 1 \pmod{m}.$$

From Definition 6.3.9, it follows that  $b$  is an inverse of  $a$  modulo  $m$  if and only if  $b$  is a solution of  $ax \equiv 1 \pmod{m}$ .

**EXAMPLE 6.3.10**

4 is an inverse of 2 modulo 7, because  $4 \cdot 2 \equiv 1 \pmod{7}$ .

We now extend Theorem 6.3.6 to linear congruence  $ax \equiv b \pmod{m}$ , where  $\gcd(a, m) = d \geq 1$ .

**Theorem 6.3.11:** Let  $a$ ,  $b$ , and  $m$  be integers with  $m > 0$  and  $\gcd(a, m) = d$ . Then  $ax \equiv b \pmod{m}$  has no solutions when  $d$  does not divide  $b$ ; but if  $d$  divides  $b$ , then there are exactly  $d$  solutions modulo  $m$ .

(Note: The words “exactly  $d$  solutions modulo  $m$ ” mean that there are  $d$  distinct integers  $a_1, a_2, \dots, a_d$  such that  $aa_i \equiv b \pmod{m}$  for  $i = 1, 2, \dots, d$ ,  $a_i \not\equiv a_j \pmod{m}$  if  $i \neq j$ , and if  $u$  is an integer such that  $au \equiv b \pmod{m}$ , then  $u \equiv a_i \pmod{m}$  for some  $i$ , where  $1 \leq i \leq d$ .)

**Proof:** Consider the congruence

$$ax \equiv b \pmod{m}. \quad (6.12)$$

Suppose  $d$  does not divide  $b$  and, if possible, (6.12) has a solution  $x_0$ . Then  $ax_0 \equiv b \pmod{m}$ , i.e.,  $ax_0 - b = mk$  for some integer  $k$ . Hence,  $ax_0 - mk = b$ . Now  $\gcd(a, m) = d$ . This implies that  $d$  divides  $a$  and  $m$ . Hence, the equality  $ax_0 - mk = b$  shows that  $d$  divides  $b$ , a contradiction. This contradiction implies that if  $d$  does not divide  $b$ , (6.12) has no solutions.

Suppose now  $d$  divides  $b$ . Because  $\gcd(a, m) = d$ , there are integers  $a_1$  and  $m_1$  such that  $aa_1 + mm_1 = d$ . Also, there is an integer  $k$  such that  $b = dk$ . Hence,  $aa_1k + mm_1k = dk = b$ . Then  $aa_1k - b = m(-m_1k)$ . This shows that  $a(a_1k) \equiv b \pmod{m}$ . Hence,  $x_0 = a_1k$  is a solution of (6.12).

Now we show that  $x_0 = a_1 k$  is a solution of (6.12) if and only if  $x_0 = a_1 k$  and  $y_0 = m_1 k$  is a solution of the Diophantine equation

$$ax + my = b. \quad (6.13)$$

If  $(x_0, y_0)$  is an integral solution of (6.13), then all the integral solutions of (6.13) are

$$\begin{aligned} y &= y_0 - \frac{a}{d}n \\ x &= x_0 + \frac{m}{d}n, \end{aligned}$$

for any integer  $n$ .

Let  $x_i = x_0 + \frac{m}{d}i$  for  $i = 0, 1, 2, \dots, d-1$ . Now  $x_0, x_1, \dots, x_{d-1}$  are distinct integers. Then  $(x_i, y_i)$ , where  $y_i = y_0 - \frac{a}{d}i$  is an integral solution of (6.13). Because  $ax_i + my_i = b$ , it follows that

$$ax_i \equiv b \pmod{m},$$

i.e., the integers  $x_0, x_1, \dots, x_{d-1}$  are solutions of the congruence (6.12).

Suppose  $x_i \neq x_j$ , where  $0 \leq i \leq d-1, 0 \leq j \leq d-1$ .

We show that  $x_i \not\equiv x_j \pmod{m}$ . Suppose that  $x_i \equiv x_j \pmod{m}$ . Then

$$\left( x_0 + \frac{m}{d}i \right) \equiv \left( x_0 + \frac{m}{d}j \right) \pmod{m},$$

i.e.,

$$\frac{m}{d}i \equiv \frac{m}{d}j \pmod{m}.$$

Now

$$\gcd\left(m, \frac{m}{d}\right) = 1.$$

Hence, by Theorem 6.1.20, we obtain

$$i \equiv j \pmod{m} \quad (6.14)$$

Because  $0 \leq i \leq d-1, 0 \leq j \leq d-1$ , (6.14) holds if and only if  $i = j$ . This implies  $x_i = x_j$ , a contradiction.

Hence,  $x_i \neq x_j$  implies  $x_i \not\equiv x_j \pmod{m}$ .

Finally, let  $x_t$  be a solution of the congruence (6.12). Then there exists an integer  $n$  such that  $x_t = x_0 + \frac{m}{d}n$ , where  $ax_0 \equiv b \pmod{m}$ . Now for the integers  $n$  and  $d$ , there exist integers  $q$  and  $r$  such that  $n = qd + r$ , where  $0 \leq r \leq d-1$ .

Hence,

$$\begin{aligned} x_t &= x_0 + \frac{m}{d}n \\ &= x_0 + \frac{m}{d}(qd + r) \\ &= x_0 + \frac{mr}{d} + mq \\ &\equiv \left( x_0 + \frac{mr}{d} \right) \pmod{m}. \end{aligned}$$

Thus, we find that any solution  $x_t$  of (6.12) is congruent to one of  $x_0, x_1, \dots, x_{d-1}$  modulo  $m$ . So there exist only  $d$  solutions of (6.12). ■

In the proof of Theorem 6.3.11, we find that if  $x_0$  is a solution of  $ax \equiv b \pmod{m}$ , then all the  $d$  solutions of this congruence are given by

$$x \equiv \left( x_0 + \frac{m}{d} i \right) \pmod{m}$$

where  $i = 0, 1, 2, \dots, d - 1$ .

**EXAMPLE 6.3.12**

Consider the congruence.

$$8x \equiv 4 \pmod{12}. \quad (6.15)$$

The  $\gcd(8, 12) = 4$ , which divides 4, hence from the theorem the congruence has a solution and has exactly four solutions modulo 12.

To solve this congruence we first solve the Diophantine equation

$$8x + 12y = 4 \quad (6.16)$$

Now  $8 \cdot (-1) + 12 \cdot (1) = 4$ . Thus, the equation has an integral solution  $(-1, 1)$ . Hence, all solutions of  $8x \equiv 4 \pmod{12}$  are

$$x \equiv \left( -1 + \frac{12}{4} i \right) \pmod{m}$$

i.e.,

$$x \equiv (-1 + 3i) \pmod{12},$$

where  $i = 0, 1, 2, 3$ . Hence, the four solutions of the congruence (6.15) are as follows:  $x \equiv -1 \pmod{12}$ ,  $x \equiv 2 \pmod{12}$ ,  $x \equiv 5 \pmod{12}$ , and  $x \equiv 8 \pmod{12}$ .

## The Chinese Remainder Theorem

This theorem deals with the solution of simultaneous congruences. The Chinese mathematician Sun-tsi considered this theorem in the first century A.D.

**Theorem 6.3.13: Chinese Remainder Theorem.** Let  $m_1, m_2, \dots, m_k$  be positive integers such that  $\gcd(m_i, m_j) = 1$  for  $i \neq j$ . Then for any integers  $a_1, a_2, \dots, a_k$ , the system of congruences

$$x \equiv a_1 \pmod{m_1}$$

$$x \equiv a_2 \pmod{m_2}$$

⋮

$$x \equiv a_k \pmod{m_k}$$

has a solution. Furthermore, any two solutions of the system are congruent modulo  $m_1 m_2 \cdots m_k$ .

**Proof:** Let  $M = m_1 m_2 \cdots m_k$  and  $N_i = \frac{M}{m_i}$ , where  $i = 1, 2, \dots, k$ . Because  $\gcd(m_i, m_j) = 1$  for  $i \neq j$ , we find that  $\gcd(N_i, m_i) = 1$ .

Then by Theorem 6.3.6, the linear congruence  $N_i x \equiv 1 \pmod{m_i}$  has a unique solution modulo  $m_i$ , say  $b_i$ . Now, we show that the integer

$$x_0 = a_1 b_1 N_1 + a_2 b_2 N_2 + \cdots + a_k b_k N_k$$

is a solution of the given system of congruences.

We first observe that  $N_i = \frac{M}{m_i} = m_1 m_2 \cdots m_{i-1} m_{i+1} \cdots m_k \equiv 0 \pmod{m_j}$  for  $j = 1, 2, \dots, i-1, i+1, \dots, k$ . Hence,

$$x_0 \equiv a_i b_i N_i \pmod{m_i}$$

for  $i = 1, 2, \dots, k$ . But  $N_i b_i \equiv 1 \pmod{m_i}$  and this implies that  $N_i b_i a_i \equiv a_i \pmod{m_i}$ .

Hence,  $x_0 \equiv a_i \pmod{m_i}$ . This shows that  $x_0$  is a solution of the given system of congruences.

Next, we show that if  $x'$  is a solution of the given system, then

$$x' \equiv x_0 \pmod{m_1 m_2 \cdots m_k}.$$

So suppose  $x' \equiv a_i \pmod{m_i}$  for  $i = 1, 2, \dots, k$ . Because  $\gcd(m_i, m_j) = 1$  for  $i \neq j$ , it follows that  $m_1 m_2 \cdots m_k$  divides  $x' - x_0$ . Thus, we find that  $x' \equiv x_0 \pmod{m_1 m_2 \cdots m_k}$ . This completes the proof. ■

### EXAMPLE 6.3.14

Consider the following system of congruences.

$$x \equiv 3 \pmod{7},$$

$$x \equiv 5 \pmod{9},$$

$$x \equiv 4 \pmod{5}.$$

We use Theorem 6.3.13, to solve this system of congruences.

Let  $M = 7 \cdot 9 \cdot 5$ . Consider the congruences

$$\frac{M}{7} x \equiv 1 \pmod{7},$$

$$\frac{M}{9} x \equiv 1 \pmod{9},$$

$$\frac{M}{5} x \equiv 1 \pmod{5},$$

i.e.,

$$45x \equiv 1 \pmod{7}$$

$$35x \equiv 1 \pmod{9}$$

$$63x \equiv 1 \pmod{5},$$

i.e.,

$$(42 + 3)x \equiv 1 \pmod{7},$$

$$(36 - 1)x \equiv 1 \pmod{9},$$

$$(60 + 3)x \equiv 1 \pmod{5}.$$

Now consider the congruences

$$3x \equiv 1 \pmod{7}, \tag{6.17}$$

$$-1x \equiv 1 \pmod{9}, \tag{6.18}$$

$$3x \equiv 1 \pmod{5}. \tag{6.19}$$

Notice that  $x = 5$  is a solution of (6.17),  $x = 8$  is a solution of (6.18), and  $x = 2$  is a solution of (6.19). Hence,

$45x \equiv 1 \pmod{7}$  is satisfied by  $x = 5$ ,

$35x \equiv 1 \pmod{9}$  is satisfied by  $x = 8$ ,

$63x \equiv 1 \pmod{5}$  is satisfied by  $x = 2$ .

Hence, a solution of the given system is given by

$$x_0 = 3 \cdot 5 \cdot 45 + 5 \cdot 8 \cdot 35 + 4 \cdot 2 \cdot 63 = 2579$$

and the unique solution is given by

$$x \equiv 2579 \pmod{7 \cdot 9 \cdot 5},$$

i.e.,

$$x \equiv 2579 \pmod{315},$$

i.e.,

$$x \equiv 59 \pmod{315}.$$

## Modular (Residue) Representation of Integers

In this section, we describe a technique that is useful in the multiplication of larger integers.

Let  $m_1, m_2, \dots, m_{k-1}, m_k$  be pairwise relatively prime integers greater than or equal to 2. Let  $m = m_1 m_2 \cdots m_{k-1} m_k$ . If  $a, a_1, a_2, \dots, a_k$ , are positive integers such that  $0 < a < m$ ,  $0 \leq a_i < m_i$  and

$$a \equiv a_1 \pmod{m_1}$$

$$a \equiv a_2 \pmod{m_2}$$

⋮

$$a \equiv a_k \pmod{m_k},$$

then by the Chinese Remainder Theorem (Theorem 6.3.13), it follows that  $a$  is the unique integer  $< m$  such that  $a$  satisfies the above system of congruences. If  $0 < b < m$  and  $b$  is a solution of

$$x \equiv a_1 \pmod{m_1}$$

$$x \equiv a_2 \pmod{m_2}$$

⋮

$$x \equiv a_k \pmod{m_k}$$

then by the Chinese Remainder Theorem (Theorem 6.3.13),  $b \equiv a \pmod{m}$ . Because  $0 < b < m$ ,  $0 < a < m$ , it follows that  $b = a$ .

The  $k$ -tuple

$$(a_1, a_2, \dots, a_k) =: (x \pmod{m_1}, x \pmod{m_2}, \dots, x \pmod{m_k})$$

is called the **modular (or residue) representation** of the integer  $a$  with respect to moduli  $m_1, m_2, \dots, m_{k-1}, m_k$ . We often write this as

$$a \Leftrightarrow (a \pmod{m_1}, a \pmod{m_2}, \dots, a \pmod{m_k})$$

### EXAMPLE 6.3.15

Let  $m_1 = 2$ ,  $m_2 = 3$ ,  $m_3 = 5$ . Then  $m = 30$ . Let  $a = 25$ . Now

$$25 \equiv 1 \pmod{2},$$

$$25 \equiv 1 \pmod{3},$$

$$25 \equiv 0 \pmod{5}.$$

Hence, the 3-tuples  $(25 \pmod{2}, 25 \pmod{3}, 25 \pmod{5}) = (1, 1, 0)$  is the modular representation of 25 with respect to the moduli 2, 3, 5, respectively.

**EXAMPLE 6.3.16**

Suppose we have the modular representation of an integer  $0 < a < 105$  as follows:

$$a \Leftrightarrow (a \pmod{3}, a \pmod{5}, a \pmod{7}) = (2, 1, 5).$$

Then  $a$  is a solution of the system of congruences

$$\begin{aligned}x &\equiv 2 \pmod{3}, \\x &\equiv 1 \pmod{5}, \\x &\equiv 5 \pmod{7}.\end{aligned}$$

By the Chinese Remainder Theorem, we get  $a = 26$ .

**Sum and Product of Modular Representations**

Let

$$a \Leftrightarrow (a \pmod{m_1}, a \pmod{m_2}, \dots, a \pmod{m_k}) = (a_1, a_2, \dots, a_k),$$

and

$$b \Leftrightarrow (b \pmod{m_1}, b \pmod{m_2}, \dots, b \pmod{m_k}) = (b_1, b_2, \dots, b_k)$$

be the modular representations of  $a$  and  $b$ , respectively, with respect to the moduli  $m_1, m_2, \dots, m_{k-1}, m_k$ .

If  $a + b < m$ , then  $a + b$  has the modular representation  $(c_1, c_2, \dots, c_k)$ , where  $0 \leq c_i < m_i$  and

$$\begin{aligned}a_1 + b_1 &\equiv c_1 \pmod{m_1}, \\a_2 + b_2 &\equiv c_2 \pmod{m_2}, \\&\vdots \\a_k + b_k &\equiv c_k \pmod{m_k}.\end{aligned}$$

The following example illustrates the preceding discussion.

**EXAMPLE 6.3.17**

Let  $m_1 = 2, m_2 = 3, m_3 = 5$ . Then  $m = 30$ .

Now  $11 \Leftrightarrow (1, 2, 1)$  and  $13 \Leftrightarrow (1, 1, 3)$  are the modular representations of 11 and 13, respectively, with respect to the moduli 2, 3, 5. We show that  $((1+1) \pmod{2}, (1+2) \pmod{3}, (1+3) \pmod{5})$  is the modular representation of  $24 = 11 + 13$ .

Notice that  $((1+1) \pmod{2}, (1+2) \pmod{3}, (1+3) \pmod{5}) = (0, 0, 4)$ .

Consider the system of congruences

$$\begin{aligned}x &\equiv 0 \pmod{2}, \\x &\equiv 0 \pmod{3}, \\x &\equiv 4 \pmod{5}.\end{aligned}$$

We solve this system. For this we consider the system

$$\begin{aligned}15x &\equiv 1 \pmod{2}, \\10x &\equiv 1 \pmod{3}, \\6x &\equiv 1 \pmod{5}.\end{aligned}$$

We find that

$$\begin{aligned}15 \cdot 1 &\equiv 1 \pmod{2}, \\10 \cdot 1 &\equiv 1 \pmod{3}, \\6 \cdot 1 &\equiv 1 \pmod{5}.\end{aligned}$$

Therefore, the solution of the above system is

$$x \equiv (15 \cdot 1 \cdot 0 + 10 \cdot 1 \cdot 0 + 6 \cdot 1 \cdot 4) (\text{mod } 30) \equiv 24 (\text{mod } 30).$$

Hence,  $24 \Leftrightarrow (0, 0, 4)$ .

A similar result holds for multiplication also.

Let

$$a \Leftrightarrow (a(\text{mod } m_1), a(\text{mod } m_2), \dots, a(\text{mod } m_k)) = (a_1, a_2, \dots, a_k),$$

and

$$b \Leftrightarrow (b(\text{mod } m_1), b(\text{mod } m_2), \dots, b(\text{mod } m_k)) = (b_1, b_2, \dots, b_k)$$

be the modular representations of  $a$  and  $b$ , respectively, with respect to the moduli  $m_1, m_2, \dots, m_{k-1}, m_k$ .

If  $a \cdot b < m$ , then  $a \cdot b$  has the modular representation  $(d_1, d_2, \dots, d_k)$ , where  $0 \leq d_k < m_i$  and

$$a_1 \cdot b_1 \equiv d_1 (\text{mod } m_1),$$

$$a_2 \cdot b_2 \equiv d_2 (\text{mod } m_2),$$

⋮

$$a_k \cdot b_k \equiv d_k (\text{mod } m_k).$$

Generally, this technique is used to break up calculations that involve large integers into small integers. One of the reasons for this is that in computer memory the size of an integer is fixed. For example, if 32 bits are used to store both positive and negative integers, then the largest integer that can be stored in computer memory, using the built-in data type, is 2147483647. However, using the technique described above we can break down large integers involved in addition and multiplication into smaller numbers. This technique is also very useful on parallel processing computers, which can run several programs simultaneously.

### EXAMPLE 6.3.18

Suppose we want to multiply the integers  $a = 8473$  and  $b = 1959$ . For this we consider the integers  $m_1 = 95$ ,  $m_2 = 97$ ,  $m_3 = 98$ , and  $m_4 = 99$ . These integers are pairwise relatively prime. Note that  $a = 8473 < 9215 = 95 \cdot 97$  and  $b = 1959 < 9702 = 98 \cdot 99$ . Thus,  $a \cdot b < m_1 m_2 m_3 m_4$ .

Now the modular representation of

$$a \Leftrightarrow (8473(\text{mod } 95), 8473(\text{mod } 97), 8473(\text{mod } 98), 8473(\text{mod } 99))$$

$$= (18, 34, 45, 58).$$

$$b \Leftrightarrow (1959(\text{mod } 95), 1959(\text{mod } 97), 1959(\text{mod } 98), 1959(\text{mod } 99))$$

$$= (59, 19, 97, 78).$$

Let

$$t = 8473 \cdot 1959$$

$$= a \cdot b \Leftrightarrow (18 \cdot 59(\text{mod } 95), 34 \cdot 19(\text{mod } 97), 45 \cdot 97(\text{mod } 98), 58 \cdot 78(\text{mod } 99))$$

$$= (17, 64, 53, 69).$$

From the Chinese Remainder Theorem, it follows that the integer  $t$  is the unique solution of the following system of congruences:

$$\begin{aligned}x &\equiv 17 \pmod{95}, \\x &\equiv 64 \pmod{97}, \\x &\equiv 53 \pmod{98}, \\x &\equiv 69 \pmod{99}.\end{aligned}$$

We now solve this system.

Note that  $m = 95 \cdot 97 \cdot 98 \cdot 99 = 89403930$  and  $N_1 = \frac{m}{95} = 941094$ ,  $N_2 = \frac{m}{97} = 921690$ ,  $N_3 = \frac{m}{98} = 912285$ ,  $N_4 = \frac{m}{99} = 903070$ .

So, we consider the following system of congruences:

$$\begin{aligned}941094x &\equiv 1 \pmod{95}, \\921690x &\equiv 1 \pmod{97}, \\912285x &\equiv 1 \pmod{98}, \\903070x &\equiv 1 \pmod{99}.\end{aligned}$$

Now

$$\begin{aligned}941094x &\equiv 24x \pmod{95}, \\921690x &\equiv 93x \pmod{97}, \\912285x &\equiv 3x \pmod{98}, \\903070x &\equiv 91x \pmod{99}.\end{aligned}$$

So, we consider the system of congruences:

$$\begin{aligned}24x &\equiv 1 \pmod{95}, \\93x &\equiv 1 \pmod{97}, \\3x &\equiv 1 \pmod{98}, \\91x &\equiv 1 \pmod{99}.\end{aligned}$$

Now

$$\begin{aligned}24 \cdot 4 &\equiv 1 \pmod{95}, \\93 \cdot 24 &\equiv 1 \pmod{97}, \\3 \cdot 33 &\equiv 1 \pmod{98}, \\91 \cdot 37 &\equiv 1 \pmod{99}.\end{aligned}$$

Then

$$\begin{aligned}x &\equiv (941094 \cdot 17 \cdot 4 + 921690 \cdot 64 \cdot 24 + 912285 \cdot 53 \cdot 33 \\&\quad + 903070 \cdot 69 \cdot 37) \pmod{95 \cdot 97 \cdot 98 \cdot 99}\end{aligned}$$

is the unique solution of the given system of congruences. Now

$$\begin{aligned}&941094 \cdot 17 \cdot 4 + 921690 \cdot 64 \cdot 24 + 912285 \cdot 53 \cdot 33 + 903070 \cdot 69 \cdot 37 \\&= 5380834407 \pmod{89403930} \\&\equiv 5380834407 \pmod{89403930} \\&\equiv 16598607 \pmod{89403930}.\end{aligned}$$

Hence,  $8473 \cdot 1959 = 16598607$ .

## Round-Robin Tournaments

We've shown how congruences exist in the marketplace through the use of ISBNs and UPCs. Congruences also have a place in the world of sports, specifically in the scheduling of round-robin tournaments.

**DEFINITION 6.3.19** ► A tournament of  $n$  different teams in which each team plays against each other team exactly once is called a **round-robin tournament**.

Let us make a schedule for a round-robin tournament for  $n$  teams.

Suppose that teams are labeled  $1, 2, 3, \dots, n$ . Let  $i$  and  $j$  be two teams. We use the following rule to make the schedule for the round-robin tournament.

Team  $i$  plays against Team  $j$  in the  $k$ th round if

$$i + j \equiv k \pmod{n}.$$

The following example illustrates the use of this rule.

**EXAMPLE 6.3.20**

Suppose eight teams participate in a round-robin tournament. We want to make a schedule for the tournament.

Let us label the teams 1, 2, 3, 4, 5, 6, 7, and 8. We use the following rule to determine whether Team  $i$  plays against Team  $j$  in the  $k$ th round: Team  $i$  plays against Team  $j$  in the  $k$ th round if

$$i + j \equiv k \pmod{8}.$$

For  $k = 1$ , i.e., for the first round to determine which team plays against which other team, we use the congruence

$$i + j \equiv 1 \pmod{8}.$$

Now

$$\begin{aligned} 1 + 8 &\equiv 1 \pmod{8}, & 2 + 7 &\equiv 1 \pmod{8}, \\ 3 + 6 &\equiv 1 \pmod{8}, & 4 + 5 &\equiv 1 \pmod{8}. \end{aligned}$$

Thus, in the first round

- Team 1 plays against Team 8,
- Team 2 plays against Team 7,
- Team 3 plays against Team 6,
- Team 4 plays against Team 5.

For the second round, we consider the congruence

$$i + j \equiv 2 \pmod{8}.$$

Now

$$\begin{aligned} 1 + 1 &\equiv 2 \pmod{8}, & 2 + 8 &\equiv 2 \pmod{8}, \\ 3 + 7 &\equiv 2 \pmod{8}, & 4 + 6 &\equiv 2 \pmod{8}, \\ 5 + 5 &\equiv 2 \pmod{8}. \end{aligned}$$

Thus, in the second round

- Team 1 plays against Team 1,
- Team 2 plays against Team 8,
- Team 3 plays against Team 7,
- Team 4 plays against Team 6
- Team 5 plays against Team 5

Here “Team 1 plays against Team 1” means that Team 1 gets a bye in the second round; similarly, Team 5 gets a bye in the second round.

For the third round, we consider the congruence  $i + j \equiv 3 \pmod{8}$ . We have

$$\begin{aligned} 1 + 2 &\equiv 3 \pmod{8}, & 3 + 8 &\equiv 3 \pmod{8}, \\ 4 + 7 &\equiv 3 \pmod{8}, & 5 + 6 &\equiv 3 \pmod{8}. \end{aligned}$$

For  $k = 4$  (the fourth round), we consider the congruence  $i + j \equiv 4 \pmod{8}$ . We have

$$\begin{aligned} 1 + 3 &\equiv 4 \pmod{8}, & 2 + 2 &\equiv 4 \pmod{8}, \\ 4 + 8 &\equiv 4 \pmod{8}, & 5 + 7 &\equiv 4 \pmod{8}, \\ 6 + 6 &\equiv 4 \pmod{8}. \end{aligned}$$

For  $k = 5$  (the fifth round), we consider the congruence  $i + j \equiv 5 \pmod{8}$ . We have

$$\begin{aligned} 1 + 4 &\equiv 5 \pmod{8}, & 2 + 3 &\equiv 5 \pmod{8}, \\ 5 + 8 &\equiv 5 \pmod{8}, & 6 + 7 &\equiv 5 \pmod{8}. \end{aligned}$$

For  $k = 6$  (the sixth round), we consider the congruence  $i + j \equiv 6 \pmod{8}$ . We have

$$\begin{aligned} 1 + 5 &\equiv 6 \pmod{8}, & 2 + 4 &\equiv 6 \pmod{8}, & 3 + 3 &\equiv 6 \pmod{8}, \\ 6 + 8 &\equiv 6 \pmod{8}, & 7 + 7 &\equiv 6 \pmod{8}. \end{aligned}$$

For  $k = 7$  (the seventh round), we consider the congruence  $i + j \equiv 7 \pmod{8}$ . We have

$$\begin{aligned} 1 + 6 &\equiv 7 \pmod{8}, & 2 + 5 &\equiv 7 \pmod{8}, \\ 3 + 4 &\equiv 7 \pmod{8}, & 8 + 7 &\equiv 7 \pmod{8}. \end{aligned}$$

For  $k = 8$  (for the eighth round), we consider the congruence  $i + j \equiv 8 \pmod{8}$ . We have

$$\begin{aligned} 1 + 7 &\equiv 8 \pmod{8}, & 2 + 6 &\equiv 8 \pmod{8}, \\ 3 + 5 &\equiv 8 \pmod{8}, & 4 + 4 &\equiv 8 \pmod{8}, \\ 8 + 8 &\equiv 8 \pmod{8}. \end{aligned}$$

Thus, we obtain the following schedule for the tournament.

Round ↓ Team →	1	2	3	4	5	6	7	8
1	8	7	6	5	4	3	2	1
2	bye	8	7	6	bye	4	3	2
3	2	1	8	7	6	5	4	3
4	3	bye	1	8	7	bye	5	4
5	4	3	2	1	8	7	6	5
6	5	4	bye	2	1	8	bye	6
7	6	5	4	3	2	1	8	7
8	7	6	5	bye	3	2	1	bye

From this table, we find that in the first round Team 1 plays against Team 8, Team 2 plays against Team 7, Team 3 plays against Team 6, and Team 4 plays against Team 5. In the second round Teams 1 and 5 get a bye, and Team 2 plays against Team 8, Team 3 plays against Team 7, and so on.

From the table, we also find that each team plays against other teams exactly once. Thus, the table satisfies the condition of the round-robin tournament.

The following theorem clarifies the validity of the congruence used to create the schedule for a round-robin tournament.

**Theorem 6.3.21:** Let  $n \geq 2$  be the number of teams in a round-robin tournament. The teams are labeled  $1, 2, 3, \dots, n$ . Suppose that in the  $k$ th round two teams  $i$  and  $j$  play against each other if  $i + j \equiv k \pmod{n}$ . Then the whole tournament is completed in  $n$  rounds and each team will play against each other team exactly once.

**Proof:** Let  $i$  and  $j$  be any two teams. Then  $0 < i \leq n$  and  $0 < j \leq n$ . By the division algorithm,

$$i + j = qn + k$$

for some integers  $q$  and  $k$  such that  $0 \leq k < n$ . This implies that

$$i + j \equiv k \pmod{n}.$$

Suppose  $k = 0$ . Now  $n \equiv 0 \pmod{n}$ . It now follows that

$$i + j \equiv k \pmod{n}$$

where  $1 \leq k \leq n$ . Thus, there exists a round  $k$ ,  $1 \leq k \leq n$ , such that the teams  $i$  and  $j$  play against each other in the  $k$ th round. Hence, the tournament will be completed by  $n$  rounds.

Let  $i$  and  $j$  be two teams such that  $i + j \equiv k \pmod{n}$ . If  $t$  is another team such that  $i + t \equiv k \pmod{n}$ , then

$$\begin{aligned} i + j &\equiv (i + t) \pmod{n} \\ \Rightarrow j &\equiv t \pmod{n} \\ \Rightarrow j &= t \quad \text{because } 0 < j \leq n \text{ and } 0 < t \leq n. \end{aligned}$$

Hence, in the  $k$ th round if Team  $i$  plays against Team  $j$ , then Team  $i$  does not play against any other team in that round.

Suppose now that  $i + j \equiv k \pmod{n}$  and  $i + j \equiv k_1 \pmod{n}$ , where  $0 < k \leq n$  and  $0 < k_1 \leq n$ . Then it follows that  $k \equiv k_1 \pmod{n}$ . This shows that Team  $i$  does not play against Team  $j$  in any other round. ■

## Hashing Functions

In this section, we discuss another application of congruence, an application that is very common in the design of a database.

When we organize data into computer memory one of our highest priorities is to do it in such a way that data retrieval is efficient. Typically, the most common operation for retrieving data is to use a search algorithm. Using the search algorithm, we can:

- determine whether a particular item is in the list.

- find the location in the list where a new item can be inserted, if the data are specially organized (for example, sorted).
- find the location of an item to be deleted.

The search algorithm's performance, therefore, is crucial. If the search is slow, it takes a large amount of computer time to accomplish your task; if the search is fast, you can accomplish your task quickly. Before considering an example, let us make the following observations.

Associated with each item in a data set is a special member that uniquely identifies the item in the data set. For example, if we have a data set consisting of students' records, then the student ID uniquely identifies each student in a particular school. This unique member of the item is called the *key* of the item. The keys of the items in the data set are used in such operations as searching, sorting, inserting, and deleting. For instance, when we search the data set for a particular item, we compare the key of the item for which we are searching with the keys of the items in the data set.

Let us now consider the following scenario.

The computer science department wants to store each computer science major's data in a computer. One way to do this is to create an array and sequentially store each student's data. During data retrieval we can sequentially search the array starting at the first array element and continue the search until either the data is found or we have searched the entire array. However, as we said before, a sequential search is not efficient for large amounts of data.

Now for each student's data, the key is the student ID. Because we are organizing the data into an array, one way of storing the data is by selecting the largest possible student ID and creating an array composed of that many elements. The data for each student are stored in the array at the position specified by the student's ID. During data retrieval, we look at the student ID and because an array is a random access data structure, using the student ID, we can directly access the array location containing the data. This technique is efficient, but it has a severe drawback. Suppose that the student ID is the student's social security number. Because a social security number is nine digits long, the largest possible social security number is 999,999,999. It follows that we would need to create a very large array just to manage a few students' data.

An efficient technique for organizing data is called **hashing**. In hashing, the data are organized with the help of a table called a **hash table**, denoted by *HashTable*, stored in an array. Suppose *HashTable* is of the size  $m$ , indexed  $0, 1, \dots, m - 1$ . Consider an item with key, say  $X$ , in the data set. To store this item in *HashTable*, we apply a function  $h$ , called a **hash function**, and compute  $h(X)$  such that  $0 \leq h(X) \leq m - 1$ . Then the item with key  $X$  is, usually, stored in *HashTable*[ $h(X)$ ]. Notice that  $h(X)$ , called a **hash address**, gives the index in *HashTable* of the item with key  $X$ . During retrieval, for the item with key  $X$ , we again compute  $h(X)$  and check the item in *HashTable* at the position *HashTable*[ $h(X)$ ]. Because the position of the items is computed with the help of a function, it follows that in *HashTable*, the items are stored in no particular order.

The following example illustrates hashing.

#### EXAMPLE 6.3.22

Suppose there are six students  $\{a_1, a_2, a_3, a_4, a_5, a_6\}$  in the discrete structures class and their IDs are

$$\begin{array}{lll} a_1 : 197354863, & a_2 : 933185952, & a_3 : 132489973, \\ a_4 : 134152056, & a_5 : 216500306, & a_6 : 106500306. \end{array}$$

Let

$$\begin{aligned} k_1 &= 197354863, & k_2 &= 933185952, & k_3 &= 132489973, \\ k_4 &= 134152056, & k_5 &= 216500306, & k_6 &= 106500306. \end{aligned}$$

Suppose *HashTable* is of the size 13 indexed 0, 1, 2, ..., 12.

Define the function  $h : \{k_1, k_2, k_3, k_4, k_5, k_6\} \rightarrow \{0, 1, 2, \dots, 12\}$  by

$$h(k_i) = k_i \pmod{13}. \quad (6.20)$$

Now  $k_i \pmod{13}$  denotes the integer  $m_i$  such that  $0 \leq m_i < 13$  and  $k_i \equiv m_i \pmod{13}$ . We store the data of the student with ID  $k$  into the array position  $h(k)$ .

Now

$$h(k_1) = h(197354863) = 197354863 \pmod{13} = 4.$$

So the data of the student with ID 197354863 is stored in *HashTable*[4]. Also,

$$\begin{aligned} 933185952 &\equiv 10 \pmod{13}, & 216500306 &\equiv 9 \pmod{13}, \\ 132489973 &\equiv 5 \pmod{13}, & 106500306 &\equiv 3 \pmod{13}, \\ 134152056 &\equiv 12 \pmod{13}. \end{aligned}$$

Hence,

$$\begin{aligned} h(933185952) &= 10, & h(216500306) &= 9, \\ h(132489973) &= 5, & h(106500306) &= 3, \\ h(134152056) &= 12, \end{aligned}$$

Suppose  $\text{HashTable}[b] \leftarrow a$  means “store the data of the student with ID  $a$  into  $\text{HashTable}[b]$ .” Then

$$\begin{aligned} \text{HashTable}[4] &\leftarrow 197354863, & \text{HashTable}[10] &\leftarrow 933185952, \\ \text{HashTable}[5] &\leftarrow 132489973, & \text{HashTable}[12] &\leftarrow 134152056, \\ \text{HashTable}[9] &\leftarrow 216500306, & \text{HashTable}[3] &\leftarrow 106500306. \end{aligned}$$

The function  $h$  defined in (6.20) is called a **hashing function**.

Next, we consider a slight variation of Example 6.3.22.

### EXAMPLE 6.3.23

Suppose there are eight students in the class in a college and their IDs are 197354864, 933185952, 132489973, 134152056, 216500306, 106500306, 216510306, 197354865. We want to store each student's data into *HashTable* in this order. Let

$$\begin{aligned} k_1 &= 197354864, & k_2 &= 933185952, & k_3 &= 132489973, & k_4 &= 134152056, \\ k_5 &= 216500306, & k_6 &= 106500306, & k_7 &= 216510306, & k_8 &= 197354865. \end{aligned}$$

Suppose *HashTable* is of the size 13 indexed 0, 1, 2, ..., 12. Define the function  $h : \{k_1, k_2, k_3, k_4, k_5, k_6, k_7, k_8\} \rightarrow \{0, 1, 2, \dots, 12\}$  by

$$h(k_i) = k_i \pmod{13}.$$

Now

$$h(k_1) = h(197354864) = 197354864 \pmod{13} = 5.$$

So the data of the student with ID 197354864 is stored in  $\text{HashTable}[5]$ . Also,

$$\begin{aligned} 933185952 &\equiv 10 \pmod{13}, & 132489973 &\equiv 5 \pmod{13}, \\ 134152056 &\equiv 12 \pmod{13}, & 216500306 &\equiv 9 \pmod{13}, \\ 106500306 &\equiv 3 \pmod{13}, & 216510306 &\equiv 12 \pmod{13}, \\ 197354865 &\equiv 6 \pmod{13}. \end{aligned}$$

Hence,

$$\begin{aligned} h(933185952) &= 10, & h(132489973) &= 5, \\ h(134152056) &= 12, & h(216500306) &= 9, \\ h(106500306) &= 3, & h(216510306) &= 12, \\ h(197354865) &= 6. \end{aligned}$$

As in the previous example, suppose  $\text{HashTable}[b] \leftarrow a$  means “store the data of the student with ID  $a$  into  $\text{HashTable}[b]$ .” Then

$$\begin{aligned} \text{HashTable}[5] &\leftarrow 197354864, & \text{HashTable}[10] &\leftarrow 933185952, \\ \text{HashTable}[5] &\leftarrow 132489973, & \text{HashTable}[12] &\leftarrow 134152056, \\ \text{HashTable}[9] &\leftarrow 216500306, & \text{HashTable}[3] &\leftarrow 106500306, \\ \text{HashTable}[12] &\leftarrow 216510306, & \text{HashTable}[6] &\leftarrow 197354865. \end{aligned}$$

It follows that the data of the student with ID 132489973 is to be stored in  $\text{HashTable}[5]$ . However,  $\text{HashTable}[5]$  is already occupied by the data of the student with ID 197354864. In such a situation, we say that a *collision* has occurred. Next we discuss some ways to handle collisions.

As we can see from Example 6.3.23, a hash function may give the same hash address for distinct keys. That is, for keys  $X_1$  and  $X_2$ ,  $X_1 \neq X_2$ ,  $h(X_1) = h(X_2)$ . In this case, we say that a **collision** has occurred. It follows that to implement hashing successfully, collisions, which cannot be avoided, must be resolved. In fact, collision resolution is a major concern in hashing. Below we describe two collision resolution techniques.

The first collision resolution technique that we describe is called **linear probing**. In linear probing, when a collision occurs for the key, say  $X$ , then starting with our original hashing function  $h = h_0$ , we construct a sequence of array indices  $h_j(X)$ , for  $j = 0, 1, 2, 3, \dots$ , such that

$$h_j(X) = (h_0(X) + j) \pmod{m},$$

where  $m$  is the size of the hash table.

Consider the student with ID  $k$ . To store its data in  $\text{HashTable}$ , first we look at the position specified by  $h_0(k)$ , i.e., check  $\text{HashTable}[h_0(k)]$ . If this position is already occupied, then we look at the position specified by  $h_1(k)$ . If this position is also occupied, then we look at the array position specified by  $h_2(k)$ , and so on. The sequence  $\{h_0(k), h_1(k), h_2(k), \dots\}$  is called a **probe sequence**.

Now for Example 6.3.23, we construct the following table, which shows the array position where each student's data are stored.

ID	$h_0(k)$	$h_1(k)$	$h_2(k)$
197354864	5		
933185952	10		
132489973	5	6	
134152056	12		
216500306	9		
106500306	3		
216510306	12	0	
197354865	6	7	

Now if  $\text{HashTable}[b] \leftarrow a$  means “store the data of the student with ID  $a$  into  $\text{HashTable}[b]$ ,” then

$$\begin{aligned} \text{HashTable}[5] &\leftarrow 197354864, & \text{HashTable}[10] &\leftarrow 933185952, \\ \text{HashTable}[6] &\leftarrow 132489973, & \text{HashTable}[12] &\leftarrow 134152056, \\ \text{HashTable}[9] &\leftarrow 216500306, & \text{HashTable}[3] &\leftarrow 106500306, \\ \text{HashTable}[0] &\leftarrow 216510306, & \text{HashTable}[7] &\leftarrow 197354865. \end{aligned}$$

Essentially, in linear probing, when a collision occurs for the key, say  $X$ , starting at the initial hash index,  $h_0(X) = h(X)$ , we sequentially search the hash table until an empty position is found. As we can see, linear probing does resolve collisions. However, it also causes clustering; i.e., more and more keys are likely to map to the positions that are already occupied leading to more and more collisions. (For more information on linear probing, see a standard book on data structures listed in the References.)

Another method for collision resolution policy is known as **double hashing**, which we describe next.

As before, we define the hash function

$$h(k) = k \pmod{m},$$

where  $m$  is the size of the hash table. Then  $0 \leq h(k) < m$ . We also define a second hash function  $g$  such that  $1 \leq g(k) < m$  and  $g(k)$  is relatively prime to  $m$ .

For example, suppose that the size of the hash table is a prime, which we denote by  $p$ . Then we can define

$$g(k) = 1 + (k \pmod{p-2}).$$

Notice that  $1 \leq g(k) \leq p-2$  and  $g(k)$  is relatively prime to  $p = m$ .

Next we define the functions  $h_j(k)$ ,  $j = 0, 1, 2, 3, \dots$ , by

$$h_j(k) = (h(k) + jg(k)) \pmod{p}.$$

Notice that  $0 \leq h_j(k) < p$ .

Now to store data of the student with ID  $k$  in  $\text{HashTable}$ , we look first at the position  $h_0(k) = h(k)$ ; i.e., look at  $\text{HashTable}[h_0(k)]$ . If this position is already occupied, then we check the array position  $h_1(k)$ . If this position is also assigned, then we check the array position  $h_2(k)$ , and so on. Here the probe sequence  $\{h_0(k), h_1(k), h_2(k), \dots\}$  is generated by two functions,  $h$  and  $g$ .

The following example explains how double hashing works.

**EXAMPLE 6.3.24**

Suppose there are six students in the discrete structures class and their IDs are 115, 153, 586, 206, 985, 111, respectively. We want to store each student's data in this order. Suppose *HashTable* is of the size 19 indexed 0, 1, 2, 3, ..., 18. Consider the prime number  $p = 19$ . Then  $p - 2 = 17$ . For the ID  $k$ , we define the following hashing function:

$$h(k) = k \pmod{19}. \quad (6.21)$$

Let  $k = 115$ . Now

$$115 \pmod{19} \equiv 1.$$

Thus,  $h_0(115) = h(115) = 1$ . So the data of the student with ID 115 are stored in *HashTable*[1].

Next consider  $k = 153$ . Now  $153 \pmod{19} \equiv 1$ . Thus,  $h_0(153) = 1$ . However, *HashTable*[1] is already occupied. So we find  $h_1(153)$ . For this we first calculate  $g(153)$ . Now

$$g(153) \equiv 1 + (153 \pmod{17}) = 1 + 0 = 1.$$

Thus,  $h(153) = 1$  and  $g(153) = 1$ . Hence,

$$h_1(153) = (h(153) + 1 \cdot g(153)) \pmod{19} = 2 \pmod{19} = 2.$$

This implies that  $h_1(153) = 2$ . So the data of the student with ID 153 are stored in *HashTable*[2].

Consider  $k = 586$ . Now  $586 \pmod{19} \equiv 16$ . Therefore,  $h_0(586) = 16$ . Because *HashTable*[16] is empty, we store the data of the student with 586 in *HashTable*[16].

Consider  $k = 206$ . Now  $206 \pmod{19} \equiv 16$ . Because *HashTable*[16] is already occupied, we compute  $h_1(206)$ . For this we compute  $g(206)$ . Now

$$g(206) \equiv 1 + (206 \pmod{17}) = 1 + 2 = 3.$$

We have  $h(206) = 16$  and  $g(206) = 3$ . Therefore,

$$h_1(206) = (16 + 3) \pmod{19} = 19 \pmod{19} \equiv 0.$$

This implies that  $h_1(206) = 0$ . So the data of the student with ID 206 are stored in *HashTable*[0].

Consider  $k = 985$ . Now  $985 \pmod{19} \equiv 16$ . Because *HashTable*[16] is already occupied, we compute  $h_1(985)$ . For this we calculate  $g(985)$ . Now

$$g(985) \equiv 1 + (985 \pmod{17}) = 1 + 16 = 17.$$

We have  $h(985) = 16$  and  $g(985) = 17$ . Therefore,

$$h_1(985) = (16 + 17) \pmod{19} = 33 \pmod{19} \equiv 14.$$

This implies that  $h_1(985) = 14$ . Because *HashTable*[14] is empty. So the data of the student with ID 985 are stored in *HashTable*[14].

Finally, consider  $k = 111$ . Now  $111 \pmod{19} \equiv 16$ . Because *HashTable*[16] is already occupied, we compute  $h_1(111)$ . For this we calculate  $g(111)$ .

$$g(111) \equiv 1 + (111 \pmod{17}) = 1 + 9 = 10.$$

We have  $h(111) = 16$  and  $g(111) = 10$ . Therefore,

$$h_1(111) = (16 + 10) \pmod{19} = 26 \pmod{19} \equiv 7.$$

This implies that  $h_1(111) = 7$ . So the data of the student with ID 111 are stored in *HashTable*[7].

The following table shows the memory location for the corresponding IDs.

ID	$h_0(k)$	$h_1(k)$	$h_2(k)$
115	1		
153	1	2	
586	16		
206	16	0	
985	16	14	
111	16	7	

Thus,

$$\begin{aligned} \text{HashTable}[1] &\leftarrow 115, & \text{HashTable}[2] &\leftarrow 153, & \text{HashTable}[16] &\leftarrow 586, \\ \text{HashTable}[0] &\leftarrow 206, & \text{HashTable}[14] &\leftarrow 985, & \text{HashTable}[7] &\leftarrow 111. \end{aligned}$$

We close this section with the following remarks. When choosing a hash function, the main objectives are to:

- choose a hash function that is easy to compute and
- minimize the number of collisions.

For more information on hashing, see [8] in the References.

## WORKED-OUT EXERCISES

**Exercise 1:** Find the solutions, if any, of the congruence  $8x \equiv 6 \pmod{4}$ ?

**Solution:** Let

$$8x \equiv 6 \pmod{4}. \quad (6.22)$$

Here  $\gcd(8, 4) = 4$  and 4 does not divide 6. Hence, (6.22) has no solutions.

**Exercise 2:** Find all solutions of the congruence  $17x \equiv 4 \pmod{36}$ .

**Solution:** Let

$$17x \equiv 4 \pmod{36}. \quad (6.23)$$

Here  $\gcd(17, 36) = 1$ . Therefore, (6.23) has a unique solution. We express  $1 = 17u + 36v$ .

Now

$$36 = 2 \cdot 17 + 2$$

$$17 = 2 \cdot 8 + 1$$

$$8 = 8 \cdot 1 + 0$$

Then

$$\begin{aligned} 2 &= 36 - 2 \cdot 17 \\ 1 &= 17 - 2 \cdot 8 \\ &= 17 - (36 - 2 \cdot 17) \cdot 8 \\ &= 17 \cdot 17 - 36 \cdot 8. \end{aligned}$$

Hence,  $4 = 17 \cdot 68 + 36 \cdot (-32)$ . This implies

$$17 \cdot 68 \equiv 4 \pmod{36}.$$

Hence,  $x_0 = 68$  is a solution of (6.23). Now

$$68 \equiv 32 \pmod{36}.$$

Therefore, the given congruence has the solution  $x \equiv 32 \pmod{36}$ .

**Exercise 3:** Find an inverse of 14 modulo 17, if it exists.

**Solution:** Consider the congruence  $14x \equiv 1 \pmod{17}$ . Because  $\gcd(14, 17) = 1$ , it follows that the congruence  $14x \equiv 1 \pmod{17}$  has a solution. Hence, there exists an inverse of

14 modulo 17. Now,

$$\begin{aligned} 17 &= 14 \cdot 1 + 3 \\ 14 &= 4 \cdot 3 + 2 \\ 3 &= 1 \cdot 2 + 1. \end{aligned}$$

Hence,

$$\begin{aligned} 3 &= 17 - 14 \cdot 1 \\ 2 &= 14 - 4 \cdot 3 = 14 - 4 \cdot (17 - 14 \cdot 1) = 14 \cdot 5 + 17(-4) \\ 1 &= 3 - 1 \cdot 2 = 17 - 14 \cdot 1 - (14 \cdot 5 + 17(-4)) \\ &= 17 \cdot 5 + 14 \cdot (-6). \end{aligned}$$

This implies that  $14(-6) \equiv 1 \pmod{17}$ . Now  $-6 \equiv 11 \pmod{17}$  and therefore  $14(-6) \equiv (14 \cdot 11) \pmod{17} \equiv 1 \pmod{17}$ . Thus, we find that 11 is a solution of  $14x \equiv 1 \pmod{17}$ . This shows that 11 is an inverse of 14 modulo 17.

**Exercise 4:** Solve the congruence  $6x \equiv 9 \pmod{15}$ .

**Solution:** Because  $\gcd(6, 15) = 3$ , and 3 divides 9, the congruence

$$6x \equiv 9 \pmod{15} \quad (6.24)$$

has exactly three solution.

Now  $3 = 6(-2) + 15(1)$ . Thus,  $9 = 6(-6) + 15(3)$ . Accordingly, we have  $6(-6) \equiv 9 \pmod{15}$ , and hence  $x_0 = -6$  is a solution of  $6x \equiv 9 \pmod{15}$ . Therefore, the three solutions of the congruence (6.24) are given by

$$x \equiv \left( -6 + \frac{15}{3}j \right) \pmod{15},$$

i.e.,

$$x \equiv (-6 + 5j) \pmod{15}$$

where  $j = 0, 1$ , and 2.

**Exercise 5:** Solve the following system of congruences

$$\begin{aligned} x &\equiv 2 \pmod{5} \\ x &\equiv 9 \pmod{11} \\ x &\equiv 7 \pmod{12}. \end{aligned}$$

**Solution:** Let  $M = 5 \cdot 11 \cdot 12$ . Now consider the congruences

$$\begin{aligned} 11 \cdot 12x &\equiv 1 \pmod{5} \\ 12 \cdot 5x &\equiv 1 \pmod{11} \\ 5 \cdot 11x &\equiv 1 \pmod{12}. \end{aligned}$$

That is,

$$\begin{aligned} 132x &\equiv 1 \pmod{5} \\ 60x &\equiv 1 \pmod{11} \\ 55x &\equiv 1 \pmod{12}, \end{aligned}$$

or

$$\begin{aligned} (26 \cdot 5 + 2)x &\equiv 1 \pmod{5} \\ (5 \cdot 11 + 5)x &\equiv 1 \pmod{11} \\ (4 \cdot 12 + 7)x &\equiv 1 \pmod{12}. \end{aligned}$$

Hence, consider the congruences

$$\begin{aligned} 2x &\equiv 1 \pmod{5} \\ 5x &\equiv 1 \pmod{11} \\ 7x &\equiv 1 \pmod{12}. \end{aligned}$$

Now  $x = 3$  is a solution of  $2x \equiv 1 \pmod{5}$ . Hence,  $x = 3$  is a solution of  $132x \equiv 1 \pmod{5}$ .

Similarly,  $x = 9$  is a solution of  $60x \equiv 1 \pmod{11}$  and  $x = 7$  is a solution of  $55x \equiv 1 \pmod{12}$ .

Hence, a solution of (6.25) is given by

$$\begin{aligned} x_0 &= 2 \cdot 3 \cdot 132 + 9 \cdot 9 \cdot 60 + 7 \cdot 7 \cdot 55 \\ &= 792 + 4860 + 2695 = 8347 \end{aligned}$$

and the unique solution is given by

$$x \equiv 8347 \pmod{5 \cdot 11 \cdot 12},$$

i.e.,

$$x \equiv 8347 \pmod{660}.$$

This implies that

$$x \equiv 427 \pmod{660}.$$

**Exercise 6:** A certain integer between 1 and 1000 leaves the remainder 1, 2, 7 when divided by 8, 11, 15, respectively. Find the integer.

**Solution:** The required integer is a solution of the following system of linear congruences.

$$\begin{aligned} x &\equiv 1 \pmod{8} \\ x &\equiv 2 \pmod{11} \\ x &\equiv 7 \pmod{15}. \end{aligned} \quad (6.25)$$

Let  $M = 8 \cdot 11 \cdot 15$ . Consider the congruences

$$\begin{aligned} 15 \cdot 11x &\equiv 1 \pmod{8} \\ 8 \cdot 15x &\equiv 1 \pmod{11} \\ 8 \cdot 11x &\equiv 1 \pmod{15}. \end{aligned}$$

That is,

$$\begin{aligned} 165x &\equiv 1 \pmod{8} \\ 120x &\equiv 1 \pmod{11} \\ 88x &\equiv 1 \pmod{15}, \end{aligned}$$

or

$$\begin{aligned} (160 + 5)x &\equiv 1 \pmod{8} \\ (110 + 10)x &\equiv 1 \pmod{11} \\ (90 - 2)x &\equiv 1 \pmod{15}. \end{aligned}$$

Hence, consider the congruences

$$\begin{aligned} 5x &\equiv 1 \pmod{8} \\ 10x &\equiv 1 \pmod{11} \\ -2x &\equiv 1 \pmod{15}. \end{aligned}$$

$x = -3$  is a solution of  $5x \equiv 1 \pmod{8}$ . Hence,  $x = -3$  is a solution of  $165x \equiv 1 \pmod{8}$ .

Similarly,  $x = -1$  is a solution of  $120x \equiv 1 \pmod{11}$  and  $x = 7$  is a solution of  $88x \equiv 1 \pmod{15}$ .

Hence, a solution of (6.25) is given by

$$x_0 = 165 \cdot (-3) \cdot 1 + 120 \cdot (-1) \cdot 2 + 88 \cdot 7 \cdot 7 = 3577$$

and the unique solution is given by

$$x \equiv 3577 \pmod{1320},$$

i.e.,

$$x \equiv 937 \pmod{1320}.$$

Hence, the required integer is 937.

**Exercise 7:** Find the positive integer  $m < 105$  that has the following modular representation  $m \Leftrightarrow (0, 1, 5)$  with respect to the moduli 3, 5, 7.

**Solution:**  $a \Leftrightarrow ((a \pmod{3}), (a \pmod{5}), (a \pmod{7})) = (0, 1, 5)$ . Then  $a$  is a solution of the system of congruences

$$x \equiv 0 \pmod{3}$$

$$x \equiv 1 \pmod{5}$$

$$x \equiv 5 \pmod{7}.$$

To solve this system we consider the following system of congruences:

$$35x \equiv 1 \pmod{3}$$

$$21x \equiv 1 \pmod{5}$$

$$15x \equiv 1 \pmod{7}.$$

Again, to solve this system we consider the following system:

$$2x \equiv 1 \pmod{3} \quad 2 \cdot 2 \equiv 1 \pmod{3} \quad 35 \cdot 2 \equiv 1 \pmod{3}$$

$$x \equiv 1 \pmod{5} \Rightarrow 1 \equiv 1 \pmod{5} \Rightarrow 21 \cdot 1 \equiv 1 \pmod{5}$$

$$x \equiv 1 \pmod{7} \quad 1 \equiv 1 \pmod{7} \quad 15 \cdot 1 \equiv 1 \pmod{7}$$

Hence, by the Chinese Remainder Theorem, the unique solution of the system of congruences is given by  $x \equiv (35 \cdot 2 \cdot 0 + 21 \cdot 1 \cdot 1 + 15 \cdot 1 \cdot 5) \pmod{3 \cdot 5 \cdot 7}$ , i.e.,  $x \equiv 96 \pmod{105}$ . Hence,  $a = 96$ .

**Exercise 8:** Set up a round-robin tournament for seven teams.

**Solution:** We labeled the teams as 1, 2, 3, 4, 5, 6, and 7. We take the following rule: Team  $i$  will play against Team  $j$  in the  $k$ th round if

$$i + j \equiv k \pmod{7}.$$

## SECTION REVIEW

### Key Terms

linear congruence in one variable  $x$   
unique modulo

inverse  
modular representation

residue representation  
round-robin tournament

Suppose  $k = 1$ . Then

$$\begin{aligned} 1 + 7 &\equiv 1 \pmod{7}, & 2 + 6 &\equiv 1 \pmod{7}, \\ 3 + 5 &\equiv 1 \pmod{7}, & 4 + 4 &\equiv 1 \pmod{7}. \end{aligned}$$

Thus, in the first round

- Team 1 plays against Team 7,
- Team 2 plays against Team 6,
- Team 3 plays against Team 5,
- Team 4 plays against Team 4.

Here “Team 4 plays against Team 4” means that Team 4 draws a bye in the first round.

For the second round, consider the congruence

$$i + j \equiv 2 \pmod{7}.$$

Then

$$\begin{aligned} 1 + 1 &\equiv 2 \pmod{7}, & 2 + 7 &\equiv 2 \pmod{7}, \\ 3 + 6 &\equiv 2 \pmod{7}, & 4 + 5 &\equiv 2 \pmod{7}. \end{aligned}$$

Hence, in the second round

- Team 1 plays against Team 1,
- Team 2 plays against Team 7,
- Team 3 plays against Team 6,
- Team 4 plays against Team 5.

Here Team 1 gets a bye in the second round.

Following this rule given by the congruence, we obtain the following schedule for the tournament, which satisfies the conditions of the round-robin tournament.

Round ↓ Team →	1	2	3	4	5	6	7
1	7	6	5	bye	3	2	1
2	bye	7	6	5	4	3	2
3	2	1	7	6	bye	4	3
4	3	bye	1	7	6	5	4
5	4	3	2	1	7	bye	5
6	5	4	bye	2	1	7	6
7	6	5	4	3	2	1	bye

This table shows that each team plays against each other team exactly once. Hence, the above table is a schedule for a round-robin tournament of seven teams.

hashing	hash address	linear probing
hash table	hashing function	probe sequence
hash function	collision	double hashing

## Some Key Definitions

1. A congruence of the form  $ax \equiv b(\text{mod } m)$ , where  $a$  and  $b$  are integers,  $m$  is a positive integer, and  $x$  is an unknown integer, is called a linear congruence in one variable  $x$ . An integer  $x_0$  is called a solution of this congruence if  $ax_0 \equiv b(\text{mod } m)$ .
2. Let  $m$  be a positive integer. For any integer  $a$  with  $\gcd(a, m) = 1$ , an integer  $b$  is called an inverse of  $a$  modulo  $m$  if  $ab \equiv 1(\text{mod } m)$ .
3. A tournament of  $n$  different teams in which each team plays against each other team exactly once is called a round-robin tournament.

## Some Key Results

1. Let  $a, b$ , and  $m$  be integers with  $m > 0$ . Suppose that  $x_0$  is a solution of the linear congruence  $ax \equiv b(\text{mod } m)$ . Then any member of the class  $[x_0]$  is a solution of this linear congruence.
2. Let  $a, b$ , and  $m$  be integers with  $m > 0$  and  $\gcd(a, m) = 1$ . Then the congruence  $ax \equiv b(\text{mod } m)$  has a solution. Further, if  $x_0$  is a solution, then the set of all solutions is precisely the equivalence class  $[x_0]$ .
3. Let  $a, b$ , and  $p$  be integers such that  $p$  is prime and  $p \nmid a$ . Then the congruence  $ax \equiv b(\text{mod } p)$  has a solution, which is unique modulo  $p$ .
4. Let  $a, b$ , and  $m$  be integers with  $m > 0$  and  $\gcd(a, m) = d$ . Then  $ax \equiv b(\text{mod } m)$  has no solutions when  $d$  does not divide  $b$ ; but if  $d$  divides  $b$ , then there are exactly  $d$  solutions modulo  $m$ .
5. Let  $m_1, m_2, \dots, m_k$  be positive integers such that  $\gcd(m_i, m_j) = 1$  for  $i \neq j$ . Then for any integers  $a_1, a_2, \dots, a_k$ , the system of congruences

$$x \equiv a_1(\text{mod } m_1)$$

$$x \equiv a_2(\text{mod } m_2)$$

⋮

$$x \equiv a_k(\text{mod } m_k)$$

has a solution. Furthermore, any two solutions of the system are congruent modulo  $m_1 m_2 \cdots m_k$ .

6. Suppose there are  $n$  teams labeled  $1, 2, 3, \dots, n$ . Let  $i$  and  $j$  be two teams. In a round-robin tournament, Team  $i$  plays against Team  $j$  in the  $k$ th round if  $i + j \equiv k(\text{mod } n)$ .
7. Let  $n \geq 2$  be the number of teams in a round-robin tournament and the teams are labeled as  $1, 2, 3, \dots, n$ . Suppose that in the  $k$ th round two teams  $i$  and  $j$  play against each other if  $i + j \equiv k(\text{mod } n)$ . Then the whole tournament is completed in  $n$  rounds and each team plays against each other team exactly once.
8. In linear probing, when a collision occurs for the key, say  $X$ , then starting with our original hashing function  $h = h_0$ , we construct a sequence of array

indices  $h_j(X)$ , for  $j = 0, 1, 2, 3, \dots$ , such that  $h_j(X) = (h_0(X) + j)(\text{mod } p)$ . The sequence  $\{h_0(k), h_1(k), h_2(k), \dots\}$  is called a probe sequence. Essentially, in linear probing, when a collision occurs for the key, say  $X$ , starting at the initial hash index,  $h_0(X) = h(X)$ , we sequentially search the hash table until an empty position is found.

9. In double hashing, when a collision occurs, a second hash function is used to generate the probe sequence.

## EXERCISES

---

1. Find all solutions of the following linear congruences.
  - a.  $7x \equiv 3(\text{mod } 9)$
  - b.  $15x \equiv 3(\text{mod } 26)$
  - c.  $12x \equiv 9(\text{mod } 15)$
  - d.  $6x \equiv 5(\text{mod } 19)$
  - e.  $6x \equiv 3(\text{mod } 9)$
  - f.  $72x \equiv 18(\text{mod } 42)$
2. Find an inverse, if it exists:
  - a. of 12 modulo 15.
  - b. of 6 modulo 11.
  - c. of 12 modulo 13.
3. Solve the following system of congruences:
  - a.  $x \equiv 3(\text{mod } 17)$
  - b.  $x \equiv 2(\text{mod } 7)$
  - $x \equiv 4(\text{mod } 9).$
  - $x \equiv 5(\text{mod } 19)$
  - $x \equiv 4(\text{mod } 5).$
  - c.  $x \equiv 1(\text{mod } 5)$
  - d.  $x \equiv 4(\text{mod } 13)$
  - $x \equiv 2(\text{mod } 6)$
  - $x \equiv 5(\text{mod } 11)$
  - $x \equiv 3(\text{mod } 7).$
  - $x \equiv 11(\text{mod } 15).$
4. A certain integer between 1 and 1000 leaves the remainders 1, 2, 6 when divided by 9, 11, and 13, respectively. Find the integer.
5. Find all positive integers that leave the remainders 1, 2, 3 when divided by 2, 3, and 5, respectively.
6. Find the smallest integer, if any, greater than 30 that leaves the remainders 2, 3, 2 when divided by 7, 9, and 11, respectively.
7. Find the smallest integer greater than 100 that is divisible by 3 but leaves the remainders 1, 5 when divided by 4 and 7, respectively.
8. Find the positive integer  $m < 105$  that has the modular representation  $m \Leftrightarrow (2, 1, 3)$  with respect to the moduli 3, 5, 7.
9. Let  $m_1 = 3$ ,  $m_2 = 5$ , and  $m_3 = 7$ . Then  $m = 30$ . If  $a \Leftrightarrow (2, 2, 1)$  and  $b \Leftrightarrow (1, 1, 3)$  are modular representations of  $a$  and  $b$  with respect to the moduli 3, 5, 7, then find  $a + b$  and  $a \cdot b$ .
10. **Ancient Hindu Problem.** If eggs are taken out from a basket, two, three, four, five, and six at a time there are leftovers, respectively, one, two, three, four, and five eggs. If they are taken out seven at a time, there are no eggs left over. How many eggs are there in the basket?
11. Find a round-robin tournament for
  - a. 10 teams.
  - b. 12 teams.
12. In a round-robin tournament for  $n$  teams:
  - a. If  $n > 1$  and  $n$  is odd, show that in each round one and only one team gets a bye.
  - b. If  $n > 1$  and  $n$  is even, show that there are rounds where no team will get a bye and in other rounds exactly two teams get a bye.
13. Suppose there are eight students in a class and their IDs are 907354864, 193318595, 132489973, 134052056, 316500307, 106500306, 116510307, and 107354865. Suppose  $\text{HashTable}$  is of the size 13, indexed 0, 1, 2, ..., 12. Show how these students' IDs, in the order given, are inserted in  $\text{HashTable}$ , using the hashing function  $h(k) = k(\text{mod } 13)$ , where  $k$  is a student ID.
14. Suppose there are eight teachers in the computer science department and their IDs are 2720, 1396, 2718, 1528, 1991, 2088, 2155, and 1850. Suppose  $\text{HashTable}$  is of the size 13, indexed 0, 1, 2, ..., 12. Show how these IDs are inserted in  $\text{HashTable}$ , using the hashing function  $h(k) = k(\text{mod } 13)$ , where  $k$  is an ID.
15. Suppose there are eight students in a class and their IDs are 197354864, 933185952, 132489973, 134152056, 216500306, 106500306, 216510306, and 197354865. Suppose  $\text{HashTable}$  is of the size 19, indexed 0, 1, 2, ..., 18. Show how these students' IDs, in the order given, are inserted in  $\text{HashTable}$ , using the hashing function  $h(k) = k(\text{mod } 19)$ , with the probe sequence  $h_j(k) = (h(k) + j)(\text{mod } 19)$ ,  $0 \leq j \leq 18$ , where  $k$  denotes an ID.
16. Suppose there are six workers in a workshop and their IDs are 134, 156, 567, 203, 987, and 111, respectively. Suppose  $\text{HashTable}$  is of the size 13, indexed 0, 1, 2, ..., 12. Show how these workers' IDs, in the order given, are inserted in  $\text{HashTable}$ , using the hashing function  $h(k) = k(\text{mod } 13)$ , with the probe sequence  $h_j(k) = (h(k) + j)(\text{mod } 13)$ ,  $0 \leq j \leq 12$ , where  $k$  denotes the identification number.
17. Suppose there are five workers in a shop and their IDs are 902, 192, 650, 109, and 143. Suppose  $\text{HashTable}$  is of the size 7, indexed 0, 1, 2, ..., 6. Show how these workers' IDs, in the order given, are inserted in  $\text{HashTable}$ , using the hashing function  $h(k) \equiv k(\text{mod } 7)$  with the probe sequence  $h_j(k) \equiv (h(k) + j)(\text{mod } 7)$ ,  $0 \leq j \leq 6$ , where  $k$  denotes an ID.
18. Suppose there are seven students in a class, and their IDs are 5701, 9302, 4210, 9015, 1553, 9902, and 2104. Suppose  $\text{HashTable}$  is of the size 19, indexed 0, 1, 2, ..., 18. Show how these students' IDs, in the order given, are inserted in  $\text{HashTable}$ , using the hashing function  $h(k) = k(\text{mod } 19)$ , with the probe sequence  $h_j(k) \equiv (h(k) + jg(k))(\text{mod } 19)$ ,  $0 \leq j \leq 18$ , where  $k$  is an ID and  $g$  is the second hash function defined by  $g(k) = (k + 1)(\text{mod } 17)$ .

## 6.4 SPECIAL CONGRUENCE THEOREMS

In the preceding sections, we discussed basic properties of congruences and described their applications in various products used in everyday life. In this section, we continue this trend. First we discuss some special theorems relating to congruences and then we apply some of the results obtained in cryptography.

In fact, some of the results that we describe in this section, such as Fermat's Little Theorem and Euler's generalization of Fermat's theorem were obtained long before the notion of congruence was introduced by Gauss. Moreover, these theorems are the basic theorems for congruences.

Fermat was a lawyer by profession and is perhaps the most famous amateur mathematician in history. Although he published almost none of his mathematical discoveries, he did correspond with contemporary mathematicians about these discoveries. In Chapter 2, we described how Fermat's Last Theorem baffled mathematician for more than 350 years.

Euler's interest in number theory seems to have been stimulated by Christian Goldbach, another amateur mathematician fascinated by number theory as well as a man with broad intellectual interests. He is remembered in mathematics for his still unsolved conjecture, now known as Goldbach's conjecture: "Every even integer greater than 4 is the sum of two odd primes."

Euler and Goldbach carried on an extensive correspondence for several decades, with Euler often discussing his newest discoveries in number theory.

In 1729, Goldbach, in response to Euler's first letter, mentioned Fermat's conjecture that all numbers of the form  $2^{2^n} + 1$  are primes. In 1732, Euler, using his computation ability, showed that  $2^{2^5} + 1 = 4294967297$  is factorizable into  $6700417 \cdot 641$ , i.e.,

$$2^{2^5} + 1 = 4294967297 = 6700417 \cdot 641.$$

Just as Euler, by the means of a counterexample, disproved one of Fermat's conjectures, mathematicians in the twentieth century disapproved a conjecture made by Euler. In 1769, Euler conjectured that no  $n$ th power could be represented as the sum of fewer than  $n$   $n$ th powers; i.e.,

$$x_1^n + x_2^n + x_3^n + \cdots + x_k^n = x^n$$

has nontrivial integral solutions if and only if  $k = n$ . In 1968, it was shown that the sum of only four fifth powers can be a fifth power, for example,

$$27^5 + 84^5 + 110^5 + 133^5 = 144^5.$$

It should be noted that in the latter case, it took almost two centuries and the services of a high-speed computer to construct the counterexample.

Euler was the first to publish a proof of Fermat's Little Theorem. It is stated next.

**Theorem 6.4.1: Fermat's Little Theorem.** If  $p$  is a prime and  $a$  is an integer such that  $p$  does not divide  $a$ , then

$$a^{p-1} \equiv 1 \pmod{p}.$$

**Proof:** Consider the  $(p - 1)$  integers

$$a, 2a, 3a, \dots, (p - 1)a. \quad (6.26)$$

We show that no two distinct members of these  $(p - 1)$  integers are congruent to each other modulo  $p$ . Suppose, if possible,

$$ra \equiv sa \pmod{p},$$

where  $1 \leq s < r \leq p - 1$ . This implies that  $p$  divides  $ra - sa = (r - s)a$ .

Because  $p$  is a prime, either  $p$  divides  $r - s$  or  $p$  divides  $a$ . Now  $1 \leq s < r \leq p - 1$ . Therefore,  $p$  does not divide  $r - s$ . Also, from the hypothesis, we find that  $p$  does not divide  $a$ . So we have a contradiction. Hence,  $ra \not\equiv sa \pmod{p}$ .

We now show that  $p$  does not divide any of the numbers listed in (6.26).

Let  $1 \leq r < p$ . Then  $p$  does not divide  $r$ . Because  $p \nmid r$ ,  $p \nmid a$ , and  $p$  is a prime, it follows that  $p \nmid ra$ ,  $1 \leq r < p$ .

From this it follows that  $ra \not\equiv 0 \pmod{p}$  for  $r = 1, 2, 3, \dots, p - 1$ .

Consider the integer  $ra$ ,  $1 \leq r < p$ , and the integer  $p$ . By the division algorithm, we can write  $ra = pt + i$ , for some integers  $t$  and  $i$ ,  $0 \leq i < p$ . This implies that  $ra \equiv i \pmod{p}$ , for some integer  $i$  such that  $0 < i < p$  (notice that  $i \neq 0$ ).

Notice that the number of elements listed in (6.26) is  $p - 1$ . Now no two elements of this list are congruent to each other modulo  $p$  and each element of this list is congruent to some integer  $i$  modulo  $p$ ,  $1 \leq i < p$ . From this it follows that the  $p - 1$  integers  $a, 2a, 3a, \dots, (p - 1)a$  must be congruent modulo  $p$  to  $1, 2, 3, \dots, (p - 1)$ , taken in some order. Hence,

$$a \cdot 2a \cdot 3a \cdots (p - 1)a \equiv 1 \cdot 2 \cdot 3 \cdots (p - 1) \pmod{p}.$$

Then

$$(1 \cdot 2 \cdot 3 \cdots (p - 1))a^{p-1} \equiv 1 \cdot 2 \cdot 3 \cdots (p - 1) \pmod{p},$$

i.e.,

$$(p - 1)!a^{p-1} \equiv (p - 1)! \pmod{p}. \quad (6.27)$$

Because  $\gcd(p, (p - 1)!) = 1$ , by Theorem 6.1.20(ii), we can cancel  $(p - 1)!$  from both sides of (6.27) and obtain

$$a^{p-1} \equiv 1 \pmod{p}. \quad \blacksquare$$

The following corollary is an immediate consequence of Fermat's little theorem.

**Corollary 6.4.2:** If  $p$  is a prime and  $a$  is any integer, then

$$a^p \equiv a \pmod{p}.$$

**Proof:** If  $p$  divides  $a$ , then  $p$  divides  $a^p$  and so  $p$  divides  $a^p - a$ . Hence,  $a^p \equiv a \pmod{p}$ .

Suppose  $p$  does not divide  $a$ . By Theorem 6.4.1, we get

$$a^{p-1} \equiv 1 \pmod{p}.$$

By Corollary 6.1.17(ii), we can multiply both sides by  $a$  to get  $a^p \equiv a \pmod{p}$ . ■

### EXAMPLE 6.4.3

5 is a prime number and 5 does not divide 9. Then by Theorem 6.4.1 (Fermat's Little Theorem),  $9^4 \equiv 1 \pmod{5}$ . This implies that if we divide  $9^4$  by 5, then the remainder is 1.

**DEFINITION 6.4.4** ▶ Let  $n$  be an integer. The number of positive integers not exceeding  $n$  and relatively prime to  $n$  is denoted by  $\phi(n)$ . This number  $\phi(n)$  is called the **Euler phi-function**.

**EXAMPLE 6.4.5**

Let  $n = 15$ . The positive integers that do not exceed 15 and that are relatively prime to 15 are the following:

$$1, 2, 4, 7, 8, 11, 13, 14$$

Hence,  $\phi(15) = 8$ . Similarly,

$$\begin{aligned}\phi(1) &= 1, & \phi(4) &= 2, & \phi(7) &= 6, & \phi(10) &= 4, & \phi(13) &= 12, \\ \phi(2) &= 1, & \phi(5) &= 4, & \phi(8) &= 4, & \phi(11) &= 10, & \phi(14) &= 6, \\ \phi(3) &= 2, & \phi(6) &= 2, & \phi(9) &= 6, & \phi(12) &= 4.\end{aligned}$$

**EXAMPLE 6.4.6**

Let  $p$  be a prime integer. If  $n$  is a positive integer such that  $n < p$ , then  $\gcd(n, p) = 1$ . Hence, the number of positive integers not exceeding  $p$  and relatively prime to  $p$  is  $p - 1$ , i.e.,  $\phi(p) = p - 1$ .

**Theorem 6.4.7:** If  $p$  is a prime and  $n$  is a positive integer, then

$$\phi(p^n) = p^n \left(1 - \frac{1}{p}\right).$$

**Proof:** Let us arrange the integers from 1 to  $p^n$  in the following way:

1,	2,	3,	...	$(p-1)$ ,	$p$ ,
$(p+1)$ ,	$(p+2)$ ,	$(p+3)$ ,	...	$p+(p-1)$ ,	$2p$ ,
$(2p+1)$ ,	$(2p+2)$ ,	$(2p+3)$ ,	...	$2p+(p-1)$ ,	$3p$ ,
⋮					
$(p^{n-1}-1)p+1$ ,	$(p^{n-1}-1)p+2$ ,	$(p^{n-1}-1)p+3$ ,	...	$(p^{n-1}-1)p+(p-1)$ ,	$p^{n-1}p$ .

From this table notice that if  $q \leq p^n$  is a positive integer, then  $\gcd(q, p^n) \neq 1$  if and only if  $q$  is one of

$$p, 2p, 3p, \dots, p^{n-1}p.$$

The number of such integers is  $p^{n-1}$  (the numbers in the last column).

If  $q$  is not equal to any one of  $p, 2p, 3p, \dots, p^{n-1}p$ , then

$$\gcd(q, p^n) = 1.$$

Now there are  $p^n$  integers from 1 to  $p^n$ . Among these integers there are  $p^{n-1}$  integers that are not relatively prime to  $p^n$ . So,

$$\phi(p^n) = p^n - p^{n-1} = p^n \left(1 - \frac{1}{p}\right).$$

This completes the proof. ■

We leave the proof of the following theorem as an exercise.

**Theorem 6.4.8:** If  $a$  and  $b$  are positive integers such that  $\gcd(a, b) = 1$ , then  $\phi(ab) = \phi(a)\phi(b)$ .

The following theorem shows how to determine  $\phi(n)$  for any positive integer  $n$ .

**Theorem 6.4.9:** Let  $n$  be an integer greater than 1 such that  $n = p_1^{r_1} p_2^{r_2} \cdots p_k^{r_k}$ , where  $p_1, p_2, \dots, p_k$  are distinct prime integers. Then

$$\phi(n) = n \left(1 - \frac{1}{p_1}\right) \left(1 - \frac{1}{p_2}\right) \cdots \left(1 - \frac{1}{p_k}\right).$$

**Proof:** From Theorem 6.4.8, we have

$$\begin{aligned}\phi(n) &= \phi(p_1^{r_1} p_2^{r_2} \cdots p_k^{r_k}) \\ &= \phi(p_1^{r_1})\phi(p_2^{r_2}) \cdots \phi(p_k^{r_k}),\end{aligned}$$

because  $\gcd(p_1^{r_1}, p_2^{r_2} \cdots p_k^{r_k}) = 1$ . Proceeding this way, we obtain

$$\phi(n) = \phi(p_1^{r_1})\phi(p_2^{r_2}) \cdots \phi(p_k^{r_k}).$$

By Theorem 6.4.7, we have

$$\phi(p_i^{r_i}) = p_i^{r_i} \left(1 - \frac{1}{p_i}\right).$$

Hence,

$$\begin{aligned}\phi(n) &= p_1^{r_1} \left(1 - \frac{1}{p_1}\right) p_2^{r_2} \left(1 - \frac{1}{p_2}\right) \cdots p_k^{r_k} \left(1 - \frac{1}{p_k}\right) \\ &= p_1^{r_1} p_2^{r_2} \cdots p_k^{r_k} \left(1 - \frac{1}{p_1}\right) \left(1 - \frac{1}{p_2}\right) \cdots \left(1 - \frac{1}{p_k}\right) \\ &= n \left(1 - \frac{1}{p_1}\right) \left(1 - \frac{1}{p_2}\right) \cdots \left(1 - \frac{1}{p_k}\right). \blacksquare\end{aligned}$$

We now discuss Euler's generalization of Fermat's Little Theorem.

If  $p$  is a prime, then  $\phi(p) = p - 1$ . Therefore, Fermat's Little Theorem can be written as follows:

If  $a$  is a positive integer and  $p$  is a prime such that  $\gcd(a, p) = 1$ , then

$$a^{\phi(p)} \equiv 1 \pmod{p}.$$

Euler generalized this result from the case of a prime to an arbitrary integer  $n$ . The following theorem is the Euler's generalization of Fermat's Little Theorem. The proof of this theorem can be found in a standard book on number theory, and we therefore leave the proof as an exercise.

**Theorem 6.4.10: Euler.** Let  $a$  and  $n$  be integers such that  $n > 0$  and  $\gcd(a, n) = 1$ . Then

$$a^{\phi(n)} \equiv 1 \pmod{n}.$$

**EXAMPLE 6.4.11**

In this example we verify Theorem 6.4.10. Let  $n = 12$  and  $a = 7$ . Then  $\gcd(12, 7) = 1$ .

Now  $\phi(n) = \phi(12) = 4$ . Also

$$7^4 = 2401 \equiv 1 \pmod{12},$$

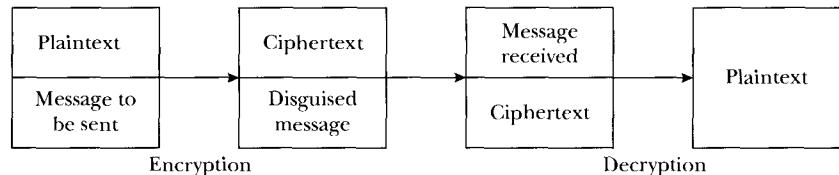
i.e.,

$$7^{\phi(12)} \equiv 1 \pmod{12}.$$

## Cryptography

Cryptography (from the Greek words *Kryptos*, meaning hidden, and *graphein*, meaning to write) is the study of sending and receiving secret messages.

The message to be sent is called **plaintext**. The disguised message is called **ciphertext**. The process of converting from plaintext to ciphertext is called encryption, while the reverse process of changing from ciphertext back to plaintext is called decryption. We describe this in Figure 6.4.



**FIGURE 6.4** Encryption and decryption

One of the earliest cryptographic systems was used by the Roman emperor Julius Caesar around 50 B.C. He made messages secret by shifting each letter three letters forward in the alphabet and sending the last three letters,  $X, Y, Z$  to the letters  $A, B, C$ , respectively. We first discuss this system using congruences.

First we digitize the letters of the alphabet. Consider the uppercase letters of the English alphabet. Translate the letters of this alphabet into the integers  $0, 1, 2, \dots, 25$  as shown in Table 6.1:

**Table 6.1** Coding of uppercase English alphabet

$A \leftrightarrow 00$	$B \leftrightarrow 01$	$C \leftrightarrow 02$	$D \leftrightarrow 03$	$E \leftrightarrow 04$	$F \leftrightarrow 05$	$G \leftrightarrow 06$
$H \leftrightarrow 07$	$I \leftrightarrow 08$	$J \leftrightarrow 09$	$K \leftrightarrow 10$	$L \leftrightarrow 11$	$M \leftrightarrow 12$	$N \leftrightarrow 13$
$O \leftrightarrow 14$	$P \leftrightarrow 15$	$Q \leftrightarrow 16$	$R \leftrightarrow 17$	$S \leftrightarrow 18$	$T \leftrightarrow 19$	$U \leftrightarrow 20$
$V \leftrightarrow 21$	$W \leftrightarrow 22$	$X \leftrightarrow 23$	$Y \leftrightarrow 24$	$Z \leftrightarrow 25$		

Now consider the function  $f : \{0, 1, 2, 3, \dots, 24, 25\} \rightarrow \{0, 1, 2, 3, \dots, 24, 25\}$  defined by

$$f(n) = (n + 3) \pmod{26}.$$

Suppose that the general wants to send the message *ATTACK* to the commander.

Here we have

plaintext: *ATTACK*

To do the encryption, the general translates the letters of the message into its numerical equivalent using Table 6.1. Therefore,

$$A \rightarrow 0, T \rightarrow 19, T \rightarrow 19, A \rightarrow 0, C \rightarrow 2, K \rightarrow 10$$

Then the general uses the function  $f$  and changes the digits 0, 19, 19, 0, 2, and 10 as follows:

$$f : 0 \rightarrow 3, 19 \rightarrow 22, 19 \rightarrow 22, 0 \rightarrow 3, 2 \rightarrow 5, 10 \rightarrow 13.$$

Next the general changes the numerals 3, 22, 22, 3, 5, 13 into letters using Table 6.1. Therefore,

$$3 \rightarrow D, 22 \rightarrow W, 22 \rightarrow W, 3 \rightarrow D, 5 \rightarrow F, 13 \rightarrow N.$$

Thus, the message *ATTACK* is encrypted to *DWWDFN*. That is,

ciphertext: *DWWDFN*

The general sends this disguised message to the commander. The commander receives the message *DWWDFN*.

We assume that the commander knows the function  $f$ .

Because  $f$  is a one-one and onto function,  $f^{-1}$  exists. Moreover,  $f^{-1}$  is defined as follows: For any integer  $n \in \{0, 1, 2, 3, \dots, 24, 25\}$ ,  $f^{-1}(n) \in \{0, 1, 2, 3, \dots, 24, 25\}$  such that

$$f^{-1}(n) \equiv (n - 3)(\text{mod } 26).$$

After receiving the message, the commander changes the letters of the message into their numerical equivalent using Table 6.1. Therefore,

$$D \rightarrow 3, W \rightarrow 22, W \rightarrow 22, D \rightarrow 3, F \rightarrow 5, N \rightarrow 13.$$

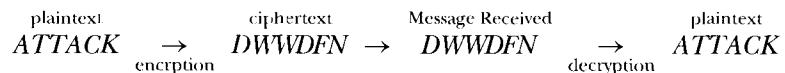
Next, the commander changes the digits 3, 22, 22, 3, 5, 13 using  $f^{-1}$ , i.e., finds the images of these numbers under  $f^{-1}$ . Now  $f^{-1}(3) \equiv (3 - 3)(\text{mod } 26) \equiv 0(\text{mod } 26)$ ,  $f^{-1}(22) \equiv (22 - 3)(\text{mod } 26) \equiv 19(\text{mod } 26)$ , and so on. Hence,

$$f^{-1} : 3 \rightarrow 0, 22 \rightarrow 19, 22 \rightarrow 19, 3 \rightarrow 0, 5 \rightarrow 2, 13 \rightarrow 10.$$

Next, the commander again uses Table 6.1 to change the digits 0, 19, 19, 0, 2, 10 into the corresponding letters. Therefore,

$$0 \rightarrow A, 19 \rightarrow T, 19 \rightarrow T, 0 \rightarrow A, 2 \rightarrow C, 10 \rightarrow K.$$

Hence, the commander decrypts the received message, *DWWDFN*, as *ATTACK*. That is,



The function used in the process of encryption and decryption is called an **encryption function**.

Moreover, in the above cryptosystem the function  $f$  is called the **encryption key** and  $f^{-1}$  is called the **decryption key**. Ideally, only the general and the commander know these two keys.

Of course, if  $f$  is known then  $f^{-1}$  is known, so we can talk about only one key and both the general and the commander have this key.

## RSA Cryptosystem

In 1977, Rivest, Shamir, and Adleman proposed a public key cryptosystem, known as the RSA cryptosystem. We explain this using the following example.

Suppose Allan wants to send a message *COME* to Bonny. For this, A (Allan) will use the public key number of B (Bonny).

B chooses two big primes,  $p$  and  $q$ . To explain the system, let us take  $p = 89$  and  $q = 97$ . Then B computes

$$n = pq = 8633$$

and

$$\phi(n) = \phi(pq) = \phi(p)\phi(q) = (89 - 1)(97 - 1) = 8448.$$

Let

$$m = (p - 1)(q - 1) = 8448.$$

Now B chooses a positive integer  $k > 1$  such that  $\gcd(k, m) = 1$ .

For this B considers

$$\phi(n) + 1 = 8449 = 7 \cdot 1207.$$

Because  $\gcd(8448 + 1, 8448) = 1$ , it follows that  $\gcd(7, 8448) = 1$ . Hence, B chooses  $k = 7$ . Now the congruence

$$7x \equiv 1 \pmod{8448}$$

has a solution  $x = 1207$ . B writes  $d = 1207$  and makes the pair

$$(n, k) = (8633, 7)$$

public so that anyone can communicate with her in this system by using this pair.

This pair is the public key for B and B keeps the pair  $(n, d) = (8633, 1207)$  secret. Notice that B does not make public the prime numbers  $p, q$  and also keeps the pair  $(n, d) = (8633, 1207)$  secret.

This pair  $(n, d)$  is the decryption key for B, and the pair  $(n, k)$  is the encryption key for anyone who wants to send the message to B. A will encrypt the message by using this encryption key.



**The Development of RSA Key Encryption** In the 1970s, three well-educated scientists working together in an MIT lab created the RSA algorithm for key encryption that has become the standard for security in passing information over the Internet and via e-mail. Ronald Rivest received his undergraduate degree from Yale as well as a Ph.D from Stanford, Adi Shamir received a B.S. from Tel Aviv University and a Ph.D from Weizmann Institute of Science in Israel, and Leonard Adelman received his B.S. and Ph.D from the University of California, Berkeley.

### Historical Notes

In 1976, they set out to create an algorithm that would allow for a secure public key exchange. Familiar with the work of Merkle and Hellman and with the limitations of their findings, Rivest, Shamir, and Adelman set out to create a multi-user cryptography system. They functioned in a unique way with Rivest developing the cryptography and Adelman and Shamir trying to break the code. Because there is no mathematical method of proving the security of encryption, the only way to prove it is secure is by continuously trying to prove that it is insecure and to fail in each attempt.

In his search for ideas, Rivest stumbled upon the notion that while it is difficult to find the prime factors of very large numbers, it is quite easy to

multiply primes. This thinking evolved into the RSA key exchange algorithm. Rivest, Shamir, and Adelman published a paper on their findings and offered an award of \$100 in *Scientific American* to challenge anyone to break a sample of the encryption code created with their algorithm. The code was eventually cracked by an international team of scientists, supercomputers, and mainframes in 1994, which pretty much ensured that a single hacker would never be able to crack the encryption. In 2002, the three were presented with the Turing Award for their work in public key encryption. Today Rivest is still at MIT, Shamir teaches at Weizmann, and Adelman is at the University of Southern California.

In order to send the message to B, A changes the letters of the word *COME* into digits by using Table 6.2.

**Table 6.2** Coding of uppercase English alphabet, digits, and some punctuation marks

$A \leftrightarrow 01$	$B \leftrightarrow 02$	$C \leftrightarrow 03$	$D \leftrightarrow 04$	$E \leftrightarrow 05$	$F \leftrightarrow 06$	$G \leftrightarrow 07$
$H \leftrightarrow 08$	$I \leftrightarrow 09$	$J \leftrightarrow 10$	$K \leftrightarrow 11$	$L \leftrightarrow 12$	$M \leftrightarrow 13$	$N \leftrightarrow 14$
$O \leftrightarrow 15$	$P \leftrightarrow 16$	$Q \leftrightarrow 17$	$R \leftrightarrow 18$	$S \leftrightarrow 19$	$T \leftrightarrow 20$	$U \leftrightarrow 21$
$V \leftrightarrow 22$	$W \leftrightarrow 23$	$X \leftrightarrow 24$	$Y \leftrightarrow 25$	$Z \leftrightarrow 26$	$, \leftrightarrow 27$	$. \leftrightarrow 28$
? $\leftrightarrow 29$	0 $\leftrightarrow 30$	1 $\leftrightarrow 31$	2 $\leftrightarrow 32$	3 $\leftrightarrow 33$	4 $\leftrightarrow 34$	5 $\leftrightarrow 35$
6 $\leftrightarrow 36$	7 $\leftrightarrow 37$	8 $\leftrightarrow 38$	9 $\leftrightarrow 39$	! $\leftrightarrow 40$		

The word *COME* is transformed into the string

$$m = 03151305$$

If the number  $m > n$ , then we divide the string into smaller blocks whose numerical values are smaller than  $n$ . Here  $m > n$ . So we divide  $m = 03151305$  into two blocks,  $m_1 = 0315$  and  $m_2 = 1305$ .

First A sends the message  $m_1$  and then  $m_2$ . Now  $m_1$  and  $m_2$  are plaintexts. He wants to change it to ciphertext. The rule for doing so is the following:

A computes  $r_1$ ,  $0 < r_1 < n$ , such that  $m_1^k \equiv r_1 \pmod{n}$ .

Here

$$r_1 = m_1^k \pmod{n} = 0315^7 \pmod{8633} = 2285.$$

Plaintext:  $m_1 = 0315$

Ciphertext:  $r_1 = 2285$

A sends the ciphertext 2285 to B.

After receiving this ciphertext 2285, B decrypts the message using the decryption key

$$(n, d) = (8633, 1207).$$

The rule is the following: Calculate

$$r_1^d \pmod{n}.$$

In this case, B computes

$$2285^{1207} \pmod{8633}.$$

Computing this, B finds that

$$2285^{1207} \pmod{8633} = 315 = 0315.$$

Similarly, B will recover the other block and then, using the conversion table, B will get the message *COME*.

In summary, this method of encryption and decryption is as follows:

*Set-up:* B chooses two primes  $p, q$  of, say 200 digits each, and computes  $n = pq$ . Next B determines  $\phi(n) = m$  and selects  $k$  such that  $1 < k < \phi(n)$  and  $\gcd(k, \phi(n)) = 1$ . Then B finds  $d$  such that

$$kd \equiv 1 \pmod{\phi(n)},$$

and makes the pair  $(n, k)$  (the encryption key) public. B keeps  $(n, d)$  in secret as the decryption key.

*Encryption:* A does the following: A writes the message using the alphabet  $\{A, B, \dots\}$  of the conversion table given in Table 6.2 and translates the message into numerical equivalents and obtains a string of digits. Next A divides the string into the smaller blocks  $m_1, m_2, m_3, \dots, m_k$  so that each  $m_i < n$ . For each  $m_i$ , using the encryption key  $(n, k)$ , A computes  $r_i$ ,  $0 < r_i < n$ , such that  $m_i^k \equiv r_i \pmod{n}$  and sends  $r_i$  to B.

*Decryption:* After receiving the message  $r_i$ , B does the following: B uses her decryption key  $(n, d)$  and computes  $r_i^d \pmod{n}$ . By Euler's theorem,  $r_i^d \pmod{n}$  is  $m_i$ .

Let us verify that  $r_i^d \pmod{n}$  is  $m_i$ . Now

$$m_i^k \equiv r_i \pmod{n} \Rightarrow m_i^{kd} \equiv r_i^d \pmod{n}.$$

**Case 1:** Suppose  $\gcd(m_i, n) = 1$ . Then by Euler's theorem  $m_i^{\phi(n)} \equiv 1 \pmod{n}$ . From  $kd \equiv 1 \pmod{\phi(n)}$ , we find that  $kd - 1 = \phi(n)t$ , i.e.,  $kd = 1 + \phi(n)t$ . Hence,

$$r_i^d \equiv m_i^{kd} \pmod{n} \equiv m_i^{1+\phi(n)t} \pmod{n} \equiv m_i \cdot m_i^{\phi(n)t} \pmod{n} \equiv m_i \pmod{n}$$

because  $m_i^{\phi(n)} \equiv 1 \pmod{n}$ .

**Case 2:** Suppose  $\gcd(m_i, n) > 1$ , where  $n = p \cdot q$ . Because  $p$  and  $q$  are primes, either  $p \mid m_i$  or  $q \mid m_i$ . But  $m_i < n$ . Hence, if  $p \mid m_i$ , then  $q \nmid m_i$  or if  $q \mid m_i$ , then  $p \nmid m_i$ . Assume  $p \mid m_i$  and  $q \nmid m_i$ . Now  $q \nmid m_i$  implies  $\gcd(q, m_i) = 1$ . Then by Fermat's theorem  $m_i^{q-1} \equiv 1 \pmod{q}$ . This implies that  $m_i^{(p-1)(q-1)} \equiv 1 \pmod{q}$ , i.e.,  $m_i^{\phi(n)} \equiv 1 \pmod{q}$ . Hence,

$$m_i^{1+\phi(n)t} \equiv m_i \cdot m_i^{\phi(n)t} \equiv m_i \pmod{q}$$

because  $m_i^{\phi(n)} \equiv 1 \pmod{q}$  and

$$m_i^{1+\phi(n)t} \equiv m_i \cdot m_i^{\phi(n)t} \equiv m_i \pmod{p}$$

because  $p \mid m_i$ . Therefore,  $m_i^{kd} \equiv m_i \pmod{pq}$  because  $p$  and  $q$  are both primes, i.e.,  $m_i^{kd} \equiv m_i \pmod{n}$ . Therefore,  $r_i^d \equiv m_i \pmod{n}$ .

**REMARK 6.4.12** ▶ For the RSA cryptosystem, if you know the decryption key  $(n, d)$ , then you can find the original message from the secret message. Notice that for the decryption key  $(n, d)$ ,  $n$  is known to everyone. To find  $d$  we must know  $\phi(n)$ , which depends on the prime factors of  $n$ . Hence, the RSA cryptosystem is based on the difficulty of factoring large numbers. It is not difficult to choose two large primes of 200 digits, and also it is not hard to multiply these two primes, but factoring an integer of 400 digits could take more than 50 million years using the fastest algorithms currently known. So it is an open question whether or not the RSA cryptosystem can be broken.

## WORKED-OUT EXERCISES

**Exercise 1:** Find the remainder when  $10^{907}$  is divided by 13.

**Solution:** If we find an integer  $r$  such that  $0 \leq r < 13$  and  $10^{907} \equiv r \pmod{13}$ , then  $r$  will be the required remainder.

Now, by Fermat's theorem  $10^{13-1} \equiv 1 \pmod{13}$ , i.e.,

Notice that  $907 = 75 \cdot 12 + 7$ . Hence, from the congruence (6.28),  $(10^{12})^{75} \equiv 1^{75} \pmod{13}$ , i.e.,

$$10^{75 \cdot 12} \equiv 1 \pmod{13}. \quad (6.29)$$

$$10^{12} \equiv 1 \pmod{13}. \quad (6.28)$$

Again,  $10^2 \equiv 9 \pmod{13}$ . Then

$$\begin{aligned} 10^3 &\equiv 90 \pmod{13} \\ &\equiv -1 \pmod{13}. \\ \Rightarrow (10^3)^2 &\equiv (-1)^2 \pmod{13} \\ &\equiv 1 \pmod{13} \quad (6.30) \\ \Rightarrow 10^6 &\equiv 1 \pmod{13} \\ \Rightarrow 10^7 &\equiv 10 \pmod{13}. \end{aligned}$$

Hence, from (6.29),  $10^{75-12} \cdot 10^7 \equiv 10 \pmod{13}$ , i.e.,  $10^{907} \equiv 10 \pmod{13}$ . Therefore, the remainder is 10.

**Exercise 2:** Show that for any positive integer  $n$ ,  $\frac{n^{19}}{19} + \frac{n^7}{7} + \frac{107n}{133}$  is an integer.

**Solution:** We have  $n^{19} \equiv n \pmod{19}$  and  $n^7 \equiv n \pmod{7}$ . Hence, there exist integers  $r$  and  $t$  such that

$$n^{19} - n = 19r \quad \text{and} \quad n^7 - n = 7t.$$

Hence,

$$\begin{aligned} \frac{n^{19}}{19} + \frac{n^7}{7} + \frac{107n}{133} &= \frac{19r+n}{19} + \frac{7t+n}{7} + \frac{107n}{133} \\ &= (r+t) + \frac{n}{19} + \frac{n}{7} + \frac{107n}{133} \\ &= (r+t) + \frac{7n+19n+107n}{133} \\ &= (r+t) + \frac{133n}{133} \\ &= (r+t) + n, \end{aligned}$$

which is an integer.

**Exercise 3:** Find the integer in the unit place of  $3^{15}$ .

**Solution:**  $3^{15}$  can be expressed uniquely as

$$3^{15} = a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10 + a_0,$$

where  $a_i$ 's are integers such that  $a_k \neq 0$  and  $0 \leq a_i < 10$ .

Now  $a_0$  is the digit in the unit place of  $3^{15}$  and  $a_0$  satisfies

$$3^{15} \equiv a_0 \pmod{10}.$$

Now  $3^{15} = (3^5)^3 = 81^3$  and  $81 \equiv 1 \pmod{10}$ . Hence,  $81^3 \equiv 1^3 \pmod{10}$ . This shows that  $3^{15} \equiv 1 \pmod{10}$ . Therefore, the integer in the unit place of  $3^{15}$  is 1.

**Exercise 4:** Find the integer in the unit place of  $32^{631}$ .

**Solution:**  $32^{631}$  can be expressed uniquely as

$$32^{631} = a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10 + a_0, \quad (6.31)$$

where  $a_i$ 's are integers such that  $a_k \neq 0$  and  $0 \leq a_i < 10$ .

Here  $a_0$  is the digit in the unit place. From (6.31), we find that the integer  $a_0$  satisfies the congruence

$$32^{631} \equiv a_0 \pmod{10}.$$

Now,

$$32 \equiv 2 \pmod{10}.$$

Hence,

$$(32)^5 \equiv 2^5 \pmod{10},$$

and

$$2^5 = 32 \equiv 2 \pmod{10}. \quad (6.32)$$

Then

$$(32)^{5 \cdot 126} \equiv 2^{126} \pmod{10}. \quad (6.33)$$

Notice that  $631 = 5 \cdot 126 + 1$ . Hence, from (6.33)

$$(32)^{5 \cdot 126} \equiv 2^{126} \pmod{10}. \quad (6.34)$$

Then from (6.32),

$$(2^5)^5 \equiv 2^5 \pmod{10} \equiv 2 \pmod{10},$$

i.e.,

$$2^{25} \equiv 2 \pmod{10}.$$

Thus,

$$(2^{25})^5 \equiv 2^5 \pmod{10} \equiv 2 \pmod{10},$$

i.e.,

$$2^{125} \equiv 2 \pmod{10}.$$

This implies that

$$2^{125} \cdot 2^1 \equiv 2 \cdot 2 \pmod{10},$$

i.e.,

$$2^{126} \equiv 4 \pmod{10}.$$

Hence, from (6.34),

$$(32)^{5 \cdot 126} \equiv 4 \pmod{10}.$$

Thus,

$$(32)^{5 \cdot 126} \cdot 32 \equiv 4 \cdot 32 \pmod{10},$$

i.e.,

$$(32)^{5 \cdot 126 + 1} \equiv 128 \pmod{10}.$$

Because  $128 \equiv 8 \pmod{10}$ ,  $(32)^{5 \cdot 126 + 1} \equiv 8 \pmod{10}$ . Therefore, the integer in the unit place of  $32^{631}$  is 8.

**Exercise 5:** Determine the integer in the unit place of  $7^{7^{14}}$ .

**Solution:**  $7^{7^{14}}$  can be expressed uniquely as

$$7^{7^{14}} = a_k 10^k + a_{k-1} 10^{k-1} + \cdots + a_1 10 + a_0, \quad (6.35)$$

where  $a_i$ 's are integers such that  $a_k \neq 0$  and  $0 \leq a_i < 10$ .

Here  $a_0$  is the digit in the unit place. From (6.35), we find that the integer  $a_0$  satisfies the congruence

$$7^{7^{14}} \equiv a_0 \pmod{10}.$$

We first note that  $7 \equiv -1 \pmod{4}$ . Hence,  $7^{14} \equiv (-1)^{14} \pmod{4}$  or  $7^{14} \equiv 1 \pmod{4}$ . This implies that  $7^{14} = 4m + 1$  for some integer  $m$ .

Again,  $7^2 \equiv -1 \pmod{10} \Rightarrow 7^4 \equiv 1 \pmod{10} \Rightarrow$

$$7^{4m} \equiv 1 \pmod{10}.$$

Thus,

$$7^{4m+1} \equiv 7 \pmod{10}.$$

This implies that

$$7^{7^{14}} \equiv 7 \pmod{10}.$$

Hence, the digit in the unit place of  $7^{7^{14}}$  is 7.

**Exercise 6:** Find the remainder when  $3^{1000000}$  is divided by 17.

**Solution:** Suppose  $r$  is the remainder when  $3^{1000000}$  is divided by 17. Then  $0 \leq r < 17$  and

$$3^{1000000} \equiv r \pmod{17}.$$

Now 17 is a prime and  $\gcd(17, 3) = 1$ . Hence,

$$3^{17-1} \equiv 1 \pmod{17},$$

i.e.,

$$3^{16} \equiv 1 \pmod{17}. \quad (6.36)$$

Notice that  $1000000 = 62500 \cdot 16$ . Hence, (6.36) implies that

$$(3^{16})^{62500} \equiv 1 \pmod{17},$$

i.e.,

$$3^{1000000} \equiv 1 \pmod{17}.$$

Therefore, the remainder is 1.

**Exercise 7:** Find the remainder when  $53^{49}$  is divided by 36.

**Solution:** Suppose  $r$  is the remainder when  $53^{49}$  is divided by 36. Hence,  $0 \leq r < 36$  and  $53^{49} \equiv r \pmod{36}$ .

Because  $\gcd(53, 36) = 1$ , by Euler's theorem

$$53^{\phi(36)} \equiv 1 \pmod{36}.$$

Now

$$\phi(36) = \phi(2^2 \cdot 3^2) = 36 \left(1 - \frac{1}{2}\right) \left(1 - \frac{1}{3}\right) = 36 \cdot \frac{1}{2} \cdot \frac{2}{3} = 12.$$

This implies that

$$53^{12} \equiv 1 \pmod{36}.$$

Hence,

$$(53^{12})^4 \equiv 1^4 \pmod{36}$$

and so

$$53^{48} \equiv 1 \pmod{36}.$$

It now follows that

$$53^{48} \cdot 53 \equiv 53 \pmod{36},$$

i.e.,

$$53^{49} \equiv 53 \pmod{36}.$$

Because  $53 \equiv 17 \pmod{36}$ , we have  $53^{49} \equiv 17 \pmod{36}$ . Therefore, the remainder is 17.

**Exercise 8:** Find  $\phi(173)$ .

**Solution:** We first examine whether 173 is a prime or not. To do this, find all primes  $p$  such that  $p^2 \leq 173$ . These primes are 2, 3, 5, 7, 11, 13. But none of these primes divides 173. Hence, 173 is a prime. Thus,  $\phi(173) = 173 - 1 = 172$ .

**Exercise 9:** Find  $\phi(2400)$ .

**Solution:**  $2400 = 2^5 \cdot 5^2 \cdot 3$ . Hence,

$$\begin{aligned} \phi(2400) &= \phi(2^5 \cdot 5^2 \cdot 3) \\ &= 2400 \left(1 - \frac{1}{2}\right) \left(1 - \frac{1}{5}\right) \left(1 - \frac{1}{3}\right) \\ &= 2400 \cdot \frac{1}{2} \cdot \frac{4}{5} \cdot \frac{2}{3} \\ &= 640. \end{aligned}$$

**Exercise 10:** Let  $n > 2$  be an integer. Show that  $\phi(n)$  is even.

**Solution:** Let  $n = p_1^{r_1} p_2^{r_2} \cdots p_k^{r_k}$ , where  $p_1, p_2, \dots, p_k$  are distinct primes and  $r_1, r_2, \dots, r_k$  are positive integers. Now

$$\phi(p_i^{r_i}) = p_i^{r_i} \left(1 - \frac{1}{p_i}\right) = p_i^{r_i-1} (p_i - 1).$$

Hence, if  $p_i$  is an odd prime, then  $p_i - 1$  is even. This implies that  $\phi(p_i^{r_i})$  is even. So we find that if one of  $p_i$ 's is odd, then  $\phi(n)$  is even. Suppose there is no odd prime factor of  $n$ . Then  $n = 2^r$ . Because  $n > 2$ , we find that  $r > 1$ . Hence,

$$\phi(n) = \phi(2^r) = 2^r \left(1 - \frac{1}{2}\right) = 2^{r-1}.$$

Because  $r > 1$ ,  $r - 1 > 0$ . Hence,  $\phi(n)$  is even. Thus, for any integer  $n > 2$ ,  $\phi(n)$  is even.

**Exercise 11:** Let  $n$  be a positive integer such that  $\gcd(n, 27) = 1$ . Prove that 27 divides  $n^{54} - 1$ .

**Solution:** Because  $\gcd(n, 27) = 1$ , by Euler's theorem, we have

$$n^{\phi(27)} \equiv 1 \pmod{27}.$$

Now  $\phi(27) = \phi(3^3) = 3^3 \left(1 - \frac{1}{3}\right) = 18$ . Hence,  $n^{18} \equiv 1 \pmod{27}$ . Then

$$(n^{18})^3 \equiv 1^3 \pmod{27},$$

i.e.,

$$n^{54} \equiv 1 \pmod{27}.$$

Hence, 27 divides  $n^{54} - 1$ .

## SECTION REVIEW

---

### Key Terms

Euler phi-function

ciphertext

encryption key

plaintext

encryption function

decryption key

### Key Definition

- Let  $n$  be an integer. The number of positive integers not exceeding  $n$  and relatively prime to  $n$  is denoted by  $\phi(n)$ . This number  $\phi(n)$  is called the Euler phi-function.

### Some Key Results

- If  $p$  is a prime and  $a$  is an integer such that  $p$  does not divide  $a$ , then  $a^{p-1} \equiv 1 \pmod{p}$ .
- If  $p$  is a prime and  $a$  is any integer, then  $a^p \equiv a \pmod{p}$ .
- If  $p$  is a prime and  $n$  is a positive integer, then  $\phi(p^n) = p^n(1 - \frac{1}{p})$ .
- If  $a$  and  $b$  are positive integers such that  $\gcd(a, b) = 1$ , then  $\phi(ab) = \phi(a)\phi(b)$ .
- Let  $n$  be an integer greater than 1 such that  $n = p_1^{n_1} p_2^{n_2} \cdots p_k^{n_k}$ , where  $p_1, p_2, \dots, p_k$  are distinct prime integers. Then

$$\phi(n) = n \left(1 - \frac{1}{p_1}\right) \left(1 - \frac{1}{p_2}\right) \cdots \left(1 - \frac{1}{p_k}\right).$$

- Let  $a$  and  $n$  be integers such that  $n > 0$  and  $\gcd(a, n) = 1$ . Then  $a^{\phi(n)} \equiv 1 \pmod{n}$ .

## EXERCISES

---

- Find the remainder when
  - $10^{241}$  is divided by 7.
  - $7^{350}$  is divided by 11.
  - $3^{495}$  is divided by 13.
  - $5^{904}$  is divided by 19.
  - $2^{20}$  is divided by 7.
  - $2^{16} \cdot 3^{12}$  is divided by 11.
- Find the remainder when  $10^{515}$  is divided by 7.
- Find the integer in the unit place of  $2^{15}$ .
- Find the remainder when  $2^{1000000}$  is divided by 17.
- Find the remainder when  $35^{33}$  is divided by 24.
- Find the digit in the unit place of
  - $32^{631}$ .
  - $37^{3113}$ .
  - $7^{2000}$ .
- Determine the integer in the unit place of  $17^{17^{17}}$ .
- Show that for any positive integer  $n$ ,
  - $\frac{n^{11}}{11} + \frac{n^3}{3} + \frac{19n}{33}$  is an integer.
  - $\frac{n^7}{7} + \frac{n^3}{3} + \frac{11n}{21}$  is an integer.
- If  $p$  is a prime and  $a$  and  $b$  are positive integers, prove that  $(a+b)^p \equiv (a+b) \pmod{p}$ .
- If  $a$  is a positive integer such that  $\gcd(a, 429) = 1$ , then prove that  $a^{480} \equiv 1 \pmod{429}$ .
- For any positive integer  $n$ , show that  $n^7 - n$  is divisible by 42.
- Prove that  $n^{16} - a^{16}$  is divisible by 17, where  $n$  and  $a$  are positive integers such that  $\gcd(n, 17) = 1 = \gcd(a, 17)$ .
- Find
  - $\phi(1999)$ ,
  - $\phi(101)$ ,
  - $\phi(243)$ ,
  - $\phi(1998)$ .

14. Prove that  $\phi(5n) = 5\phi(n)$  if and only if 5 divides  $n$ .
15. Show that if  $p$  and  $q$  are distinct primes, then  $p^{q-1} + q^{p-1} \equiv 1 \pmod{pq}$ .
16. If  $n$  is a positive integer such that  $(n-1)! \equiv -1 \pmod{n}$ , then prove that  $n$  is a prime.
17. If  $m$  and  $n$  are relatively prime positive integers, prove that  $m^{\phi(n)} + n^{\phi(m)} \equiv 1 \pmod{mn}$ .
18. Prove that 42 divides  $7^2 + 6^6 - 1$ .
19. Prove that
 
$$1 + 20 + 20^2 + 20^3 + \cdots + 20^{21} \equiv 0 \pmod{23}.$$
20. Prove that  $2^{145} \cdot 14^{10} + 1$  is divisible by 11.
21. If  $f(x) = 14x^5 - 9x^4 + 7x^2 - 3$ , find the remainder when  $f(16)$  is divided by 5.
22. Encrypt the message *CONGRATULATIONS* by changing the letters into the equivalent numerals, then changing the numerals using the encryption function  $f(n) = (n+5) \pmod{26}$ , and finally changing the numerals into letters of the alphabet, find the ciphertext.
23. Verify the RSA cryptosystem with primes  $p = 23$  and  $q = 29$ . Suppose Bob chooses these two prime numbers. Find (i) a public key for Bob, (ii) the decryption key of Bob. Allan wants to send the message *DO* to Bob. Find the ciphertext for this message. Now show how Bob decrypts the message.

## PROGRAMMING EXERCISES

1. Write a program to determine whether a positive integer is divisible by 2, 3, 4, 8, 7, 9, 11, or 13.
2. Write a program to do the following.
  - a. Determine the check digit of an ISBN.
  - b. Determine a missing single digit in an ISBN.
  - c. Determine whether an ISBN is valid.
3. Write a program to do the following.
  - a. Determine the check digit of a UPC.
  - b. Determine a missing single digit in a UPC.
  - c. Determine whether a UPC is valid.
4. Write a program to do the following. Assume that Visa card has 16 digits.
  - a. Determine the check digit of a Master or Visa card.
  - b. Determine a missing single digit in a Master or Visa card.
  - c. Determine whether a Master or Visa card is valid.
5. Write a program to solve a linear congruence in one variable.
6. Write a program to solve a system of linear congruences (Chinese Remainder Theorem).
7. Write a program to print the schedule of up to 20 teams in a round-robin tournament.
8. Write a program to implement hashing. Use double hashing to resolve collisions.
9. Write a program to implement the RSA cryptosystem using three- or four-digit prime numbers.

## Counting Principles

**The objectives of this chapter are to:**

- Learn the basic counting principles—multiplication and addition
- Explore the pigeonhole principle
- Learn about permutations
- Learn about combinations
- Explore generalized permutations and combinations
- Learn about binomial coefficients and explore the algorithm to compute them
- Discover the algorithms to generate permutations and combinations
- Become familiar with discrete probability

We often face problems of counting objects having certain properties. In order to meet our daily needs we must count certain objects or ascertain the number of possible ways of doing a particular job. Counting problems occur throughout mathematics and computer science. For example, in computer science, we count the number statements executed in algorithm analysis.

In this chapter, we will discuss some basic techniques of counting. The ideas of union, intersection, and the Cartesian product of sets will be useful to understand these basic techniques.

## 7.1 BASIC COUNTING PRINCIPLES

This section describes the basic counting principles known as the addition and multiplication principles. Before describing them, let us present two problems, which we will answer later with the help of the basic counting principles.

1. There are three boxes containing books (see Figure 7.1). The first box contains 15 mathematics books by different authors, the second box contains 12 chemistry books by different authors, and the third box contains 10 computer science books by different authors. A student wants to take a book from one of the three boxes. In how many ways can the student do this?

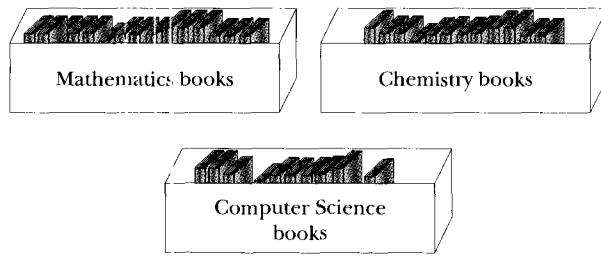


FIGURE 7.1 Boxes of books

2. Morgan is a lead actor in a new movie. She needs to shoot a scene in the morning in studio A and an afternoon scene in studio C. She looks at the map and finds that there is no direct route from studio A to studio C. On further checking she finds that studio B is located between studios A and C. Morgan realizes that her friends Brad and Jennifer are shooting a movie in studio B and she has not seen them for a long time. While going from studio A to studio C, she decides to stop at studio B have lunch with Brad and Jennifer and then go on to studio C to shoot the afternoon scene. There are three roads, say  $A_1$ ,  $A_2$ , and  $A_3$ , from studio A to studio B and four roads, say  $B_1$ ,  $B_2$ ,  $B_3$ , and  $B_4$ , from studio B to studio C, as shown in Figure 7.2. In how many ways can Morgan go from studio A to studio C and have lunch with Brad and Jennifer at Studio B?

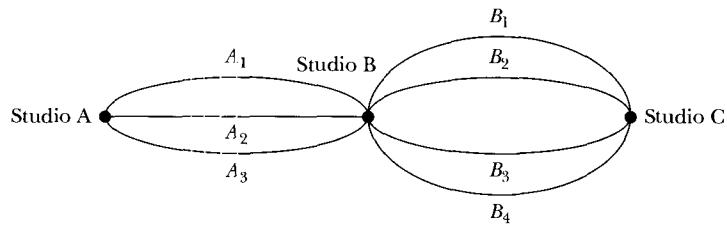


FIGURE 7.2 Routes from studio A to studio C

Problems such as these can be answered by using some counting principles, which we describe next.

## Addition Principle

Suppose that we want to find the number of integers between 4 and 100 that end with 3 or 5. Let  $T$  denote this task. We divide  $T$  into the following tasks.

$T_1$  : Find all integers between 4 and 100 that end with 3.

$T_2$  : Find all integers between 4 and 100 that end with 5.

Now 13, 23, 33, 43, 53, 63, 73, 83, and 93 are 9 integers between 4 and 100 that end with 3; and 5, 15, 25, 35, 45, 55, 65, 75, 85, and 95 are 10 integers between 4 and 100 that end with 5. Hence, tasks  $T_1$  and  $T_2$  can be done in 9 and 10 ways, respectively. Both tasks are independent of each other; i.e., they can be completed in any order because their outcomes do not depend on each other. Therefore, the number of ways to do one of these tasks is  $9 + 10 = 19$ . Hence, the number of integers between 4 and 100 that end with 3 or 5 is 19. That is, task  $T$  can be completed in 19 ways.

If we look closely at the preceding counting method, we see that it is connected with the union of two disjoint sets. For example, task  $T$  is the union of tasks  $T_1$  and  $T_2$ .

Let  $X = \{x_1, x_2, \dots, x_n\}$  be a set with  $n$  distinct elements and  $Y = \{y_1, y_2, \dots, y_m\}$  be a set with  $m$  distinct elements. Suppose that  $X \cap Y = \emptyset$ . Then  $X \cup Y = \{x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m\}$ . Because  $x_i \neq y_j$  for all  $i = 1, 2, \dots, n, j = 1, 2, \dots, m$ , the elements of  $X$  and  $Y$  are distinct, we can establish a one-to-one correspondence between the sets  $I_{n+m} = \{1, 2, \dots, n, n+1, n+2, \dots, n+m\}$  and  $\{x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m\} = X \cup Y$ . Hence, it follows that the number of elements in  $X \cup Y$  is  $n+m$ , i.e.,

$$|X \cup Y| = n+m = |X| + |Y|.$$

We can prove the following theorem by mathematical induction.

**Theorem 7.1.1:** Let  $X_1, X_2, \dots, X_k$  be sets such that the number of elements in  $X_i$  is  $n_i$ , that is,  $|X_i| = n_i, i = 1, 2, \dots, k, k \geq 2$ . Suppose that for any two sets  $X_i$  and  $X_j$ ,  $X_i \cap X_j = \emptyset, i = 1, 2, \dots, k, j = 1, 2, \dots, k, i \neq j$ ; that is, the sets  $X_1, X_2, \dots, X_k$  are pairwise disjoint. Then  $|X_1 \cup X_2 \cup \dots \cup X_k| = n_1 + n_2 + \dots + n_k$ .

**DEFINITION 7.1.2** ► Suppose a task  $T$  is a collection or a sequence of tasks  $T_1, T_2, \dots, T_k$ . Tasks  $T_1, T_2, \dots, T_k$  are called *independent* if the outcome of any task, say  $T_i$ , does not influence the outcome of any other task in the collection or sequence.

Following Theorem 7.1.1, we formulate the following counting rule, commonly known as the **addition principle**.

**Addition Principle:** Suppose that tasks  $T_1, T_2, \dots, T_k$  can be done in  $n_1, n_2, \dots, n_k$  ways, respectively. If all these tasks are independent of each other, then the number of ways to do one of these tasks is  $n_1 + n_2 + \dots + n_k$ .

The following example further illustrates the addition principle.

**EXAMPLE 7.1.3**

In this example, we answer the first problem we posed at the beginning of this section. To do this, we return to Figure 7.1 and the student who wants to take a book from one of three boxes. If the student wants to take a mathematics book, then he/she can choose one mathematics book from 15 different mathematics books and he/she can do this in exactly 15 ways. Similarly, the student can choose one chemistry book in exactly 12 ways and one computer science book in exactly 10 ways. Suppose tasks  $T_1$ ,  $T_2$ , and  $T_3$  are as follows:

- $T_1$  : Choose a mathematics book.
- $T_2$  : Choose a chemistry book.
- $T_3$  : Choose a computer science book.

Then tasks  $T_1$ ,  $T_2$ , and  $T_3$  can be done in 15, 12, and 10 ways, respectively. All of these tasks are independent of each other. Hence, the number of ways to do one of these tasks is  $15 + 12 + 10 = 37$ .

## Multiplication Principle

Let us consider the following problem. Suppose we want to find the number of words of length 3 that can be written using the letters  $A$ ,  $B$ ,  $C$ ,  $D$ , and  $E$  such that no repetition of letters in a word is allowed. For example,  $BAD$ ,  $ACB$ ,  $AEB$ ,  $BAE$ ,  $BCE$ , and  $EDA$  are some of the words of length 3 that do not contain repetition of letters.

Let  $T$  be the task of constructing such a word and let  $s$  denote such a word. Then  $T$  can be completed in three successive steps,  $T_1$ ,  $T_2$ , and  $T_3$ , in which

- $T_1$  : Choose the first letter.
- $T_2$  : Choose the second letter.
- $T_3$  : Choose the third letter.

The first letter of  $s$  can be any one of the letters  $A$ ,  $B$ ,  $C$ ,  $D$ , or  $E$ . Therefore, the first letter of  $s$  can be chosen in 5 different ways. Thus, step  $T_1$  can be completed in 5 different ways.

Once the first letter of  $s$  is chosen, the number of remaining letters in the given set is 4. Therefore the second letter of  $s$  can be any one of the remaining 4 letters, so the second letter of  $s$  can be chosen in 4 ways. Thus, step  $T_2$  can be completed in 4 different ways.

Suppose we choose  $A$  as the first letter of  $s$ .

$$\underline{A}$$

Because repetitions of a letter are not allowed, in the second place we can put any one of the letters  $B$ ,  $C$ ,  $D$ , or  $E$ .

$$\underline{A} \underline{B}, \quad \underline{A} \underline{C}, \quad \underline{A} \underline{D}, \quad \underline{A} \underline{E}$$

It follows that for each choice of completing step  $T_1$ , step  $T_2$  can be completed in 4 ways. Hence, we find that steps  $T_1$  and  $T_2$  can be completed in  $5 \cdot 4$  different ways.

After choosing the first and the second letters of  $s$ , the number of remaining letters in the given set is 3. Therefore, the third letter of  $s$  can be any one of these 3 letters. Thus, the third letter can be chosen in 3 ways; i.e., step  $T_3$  can be completed in 3 ways.

Suppose we have chosen  $A$  in step  $T_1$  and  $B$  in step  $T_2$ .

$$\underline{A} \underline{B}$$

Then we can complete step  $T_3$  in three different ways.

$$\underline{A} \underline{B} \underline{C}, \quad \underline{A} \underline{B} \underline{D}, \quad \underline{A} \underline{B} \underline{E}$$

Because steps  $T_1$  and  $T_2$  can be completed in  $5 \cdot 4$  ways, for each of these choices of  $5 \cdot 4$  different ways, we can complete step  $T_3$  in 3 different ways. Consequently, steps  $T_1$ ,  $T_2$ , and  $T_3$  can be completed in  $5 \cdot 4 \cdot 3$  different ways. Hence, there are 60 different words of length 3 such that no word contains a repetition of letters.

Let us now consider a variation of the preceding problem. Suppose we want to find the number of words of length 3 that can be written by using the letters  $A$ ,  $B$ ,  $C$ ,  $D$ , and  $E$  such that repetition of letters in a word is allowed.  $BAB$ ,  $ACB$ ,  $AEE$ ,  $BAA$ ,  $BCE$ , and  $DDA$  are examples of such words of length 3.

As before, let  $T$  be the task of constructing such a word of length 3 and let  $t$  be such a word. The task  $T$  can be completed in three successive steps,  $T_1$ ,  $T_2$ , and  $T_3$ .

$T_1$  : Choose the first letter.

$T_2$  : Choose the second letter.

$T_3$  : Choose the third letter.

The first letter of  $t$  can be any one of the letters  $A$ ,  $B$ ,  $C$ ,  $D$ , or  $E$ . Therefore, the first letter of  $t$  can be chosen in 5 different ways. This implies that step  $T_1$  can be completed in 5 different ways.

Because repetition of a letter is allowed, the second letter of  $t$  can be any one of the letters  $A$ ,  $B$ ,  $C$ ,  $D$ , or  $E$ . Therefore, the second letter of  $t$  can be chosen in 5 different ways. That is, step  $T_2$  can be completed in 5 different ways.

Suppose we choose  $A$  as the first letter.

$$\underline{A}$$

Because repetition is allowed, in the second place we can write any one of the letters  $A$ ,  $B$ ,  $C$ ,  $D$ , or  $E$ .

$$\underline{A} \underline{A}, \quad \underline{A} \underline{B}, \quad \underline{A} \underline{C}, \quad \underline{A} \underline{D}, \quad \underline{A} \underline{E}$$

It follows that steps  $T_1$  and  $T_2$  can be completed in  $5 \cdot 5$  different ways.

Next, the third letter of  $t$  can be any one of  $A$ ,  $B$ ,  $C$ ,  $D$ , or  $E$ . Therefore, the third letter can be chosen in 5 ways. Suppose we chose  $A$  in the first step,  $T_1$ , and  $B$  in the second step,  $T_2$ .

$$\underline{A} \underline{B}$$

Then we can complete the third step,  $T_3$ , in 5 different ways

$$\underline{A} \underline{B} \underline{A}, \quad \underline{A} \underline{B} \underline{B}, \quad \underline{A} \underline{B} \underline{C}, \quad \underline{A} \underline{B} \underline{D}, \quad \underline{A} \underline{B} \underline{E}$$

We find that for each of the  $5 \cdot 5$  different ways of completing steps  $T_1$  and  $T_2$ , we can complete step  $T_3$  in 5 different ways.

Consequently, steps  $T_1$ ,  $T_2$ , and  $T_3$  can be completed in  $5 \cdot 5 \cdot 5$  different ways. Hence, there are 125 different words of length 3 such that a word may contain repetition of letters.

Let us consider the subtasks  $T_1$  and  $T_2$ . Suppose that  $T_1$  has five choices, say  $a_1, a_2, a_3, a_4$ , and  $a_5$ ; and  $T_2$  has four choices, say  $b_1, b_2, b_3$ , and  $b_4$ . Then the choices

for the task  $T_1 T_2$  can be viewed as  $(a_1, b_1), (a_1, b_2), (a_1, b_3), (a_1, b_4), (a_2, b_1), (a_2, b_2), (a_2, b_3), (a_2, b_4), (a_3, b_1), (a_3, b_2), (a_3, b_3), (a_3, b_4), (a_4, b_1), (a_4, b_2), (a_4, b_3), (a_4, b_4), (a_5, b_1), (a_5, b_2), (a_5, b_3)$ , and  $(a_5, b_4)$ . In other words, the numbers of choices for task  $T_1 T_2$  can be determined by considering the ordered pairs of the set  $T_1 \times T_2$ . So we see a connection between the preceding way of determining the number of ways a task can be completed and the Cartesian product of sets.

Notice that to find the number of choices for the task  $T = T_1 T_2 T_3$ , we can consider the Cartesian product  $T_1 \times T_2 \times T_3$ .

Recall that in Chapter 1, we introduced the Cartesian product  $A_1 \times A_2$  of the sets  $A_1, A_2$ . We know that if  $A_1$  contains  $n_1$  elements and  $A_2$  contains  $n_2$  elements, then  $A_1 \times A_2$  contains  $n_1 n_2$  elements. By mathematical induction we can extend this result to  $k$  ( $\geq 2$ ) finite sets  $A_1, A_2, \dots, A_k$  with  $n_1, n_2, \dots, n_k$  elements, respectively, and prove that  $A_1 \times A_2 \times \dots \times A_k$  contains  $n_1 n_2 \dots n_k$  elements. We use this result to form another counting principle, commonly known as the **multiplication principle**. This is another very useful counting technique.

**Multiplication Principle:** Suppose that a task  $T$  can be completed in  $k$  successive steps. Suppose step 1 can be completed in  $n_1$  different ways, step 2 can be completed in  $n_2$  different ways, and in general, no matter how the preceding steps are completed, step  $k$  can be completed in  $n_k$  different ways. Then the task  $T$  can be completed in

$$n_1 n_2 \dots n_k$$

different ways.

#### EXAMPLE 7.1.4

In this example, we answer the second problem we posed at the beginning of this section. We asked how many ways Morgan can go from studio A to studio C and have lunch with Brad and Jennifer at Studio B. We refer again to Figure 7.2 and see that there are 3 ways to go from studio A to studio B and 4 ways to go from studio B to studio C. The number of ways to go from studio A to studio C via studio B is  $3 \cdot 4 = 12$ .

**REMARK 7.1.5** ▶ In this section, we described the addition and multiplication principles. The following helps in clarifying when to apply which principle.

Suppose our task  $T$  is to find the number of integers between 4 and 100 that end with 3 or 5. We divide task  $T$  into the following tasks.

$T_1$  : Find all integers between 4 and 100 that end with 3.

$T_2$  : Find all integers between 4 and 100 that end with 5.

Each member of  $T$  is a member of either  $T_1$  or  $T_2$  and each member of  $T_1$  or  $T_2$  is a member of  $T$ . That is, task  $T$  is the union of tasks  $T_1$  and  $T_2$ . Moreover, tasks  $T_1$  and  $T_2$  are independent of each other. So we apply the addition principle.

Now consider task  $T$  : Find a word of length 3. Here we divide task  $T$  into three successive steps,  $T_1$ ,  $T_2$ , and  $T_3$ , as follows:

$T_1$  : Choose the first letter.

$T_2$  : Choose the second letter.

$T_3$  : Choose the third letter.

Each member of  $T_i$  is not a member of  $T$ , because each member of  $T_i$  gives us a word of length 1. But if we complete these three steps, then we create a member of  $T$ . Here we apply the multiplication principle.

So, to complete task  $T$ , if we can determine whether to take the union or the Cartesian product of the subtasks, we can determine which principle to apply.

### EXAMPLE 7.1.6

In this example, we find the number of license plates that can be written with 3 uppercase letters followed by 3 digits.

There are 26 uppercase letters and 10 digits. Now a license plate consists of 3 letters followed by 3 digits:

$$\overline{l_1} \overline{l_2} \overline{l_3} \overline{d_1} \overline{d_2} \overline{d_3}$$

This counting problem can be viewed as a task consisting of six successive steps. In each of steps 1, 2, and 3, choose an uppercase letter; in each of steps 4, 5, and 6, choose a digit. Because there are 26 letters, each of steps 1, 2, and 3 can be completed in 26 ways. Similarly, because there are 10 digits, each of steps 4, 5, and 6 can be completed in 10 different ways. Thus, a license plate can be formed in:

$$26 \cdot 26 \cdot 26 \cdot 10 \cdot 10 \cdot 10 = 17,576,000$$

ways. Hence, there are 17,576,000 different license plates with 3 uppercase letters followed by 3 digits.

### EXAMPLE 7.1.7

Two 6-faced red and blue dice are rolled. We want to determine the number of possible outcomes. We can view this as a sequence of two steps. In the first step a red die is rolled, and in the second step a blue die is rolled. Now the red die can be rolled in 6 different ways, and the blue die can be rolled in 6 different ways. Thus, by the multiplication principle, the two dice can be rolled in  $6 \cdot 6 = 36$  different ways. Hence, there are 36 different outcomes.

### EXAMPLE 7.1.8

In Worked-Out Exercise 9 (Chapter 2, page 144), we used induction to show that if  $A$  is a set with  $n$  distinct elements,  $n > 0$ , then the number of subsets of  $A$  is  $2^n$ . In this example, we show this result using the multiplication principle.

Let  $B$  be a subset of  $A$  and let  $A = \{x_1, x_2, \dots, x_n\}$ . Now for each element  $x_i \in A$ , either  $x_i \in B$  or  $x_i \notin B$ . Therefore,  $B$  can be constructed by making a decision on each element of  $A$ . Because  $A$  has  $n$  elements, we make a sequence of  $n$  successive decisions. In step 1 decide whether  $x_1 \in B$  or  $x_1 \notin B$ , in step 2 decide whether  $x_2 \in B$  or  $x_2 \notin B$ , and in step  $n$  decide whether  $x_n \in B$  or  $x_n \notin B$ . Because each of these steps has two choices, the number of ways  $B$  can be constructed is:

$$\underbrace{2 \cdot 2 \cdots 2}_{n \text{ times}} = 2^n.$$

Hence, the number of subsets of  $A$  is  $2^n$ .

### REMARK 7.1.9

In Chapter 1, we discussed how to represent finite sets in computer memory using bit strings. For example, suppose  $A$  is a nonempty set of  $n$  elements, say  $A = \{x_1, x_2, \dots, x_n\}$ . Then any subset of  $A$  including  $A$  can be represented as a bit string of length  $n$ . We consider the listing of the elements of  $A$  in the ordering  $x_1, x_2, \dots, x_n$ . The bit string corresponding to  $A$  consists of all 1's. Suppose  $B$  is a subset of  $A$ . Let  $b = b_1 b_2 \cdots b_n$  be the bit string of length  $n$  such that for each  $i$ ,

$b_i = 1$  if  $x_i \in B$  and  $b_i = 0$  if  $x_i \notin B$ . This shows that for the subset  $B$  we can associate the bit string  $b_1 b_2 \cdots b_n$  of length  $n$ . Conversely, given a bit string  $b_1 b_2 \cdots b_n$  of length  $n$  we can associate a subset  $B$  of  $A$ , defined by  $x_i \in B$  if and only if  $b_i = 1$ . Thus, we find a one-to-one correspondence between the set of all subsets of  $A$  and the set of all bit strings of length  $n$ . Hence, counting the number of subsets of  $A$  is the same as counting the number of bit strings of length  $n$ . Let  $b = b_1 b_2 \cdots b_n$  be a bit string of length  $n$ . Now  $b_1$  is 0 or 1,  $b_2$  is 0 or 1, and so on. Thus, for each  $i$ ,  $b_i$  has two choices. Hence, by the multiplication principle the number of choices for the string  $b$  is

$$\underbrace{2 \cdot 2 \cdots 2}_{n \text{ times}} = 2^n.$$

Hence, the number of subsets of  $A$  is  $2^n$ .

## Simultaneously Using Addition and Multiplication

The counting problems that we have considered so far involved *either* the addition principle *or* the multiplication principle. Sometimes, however, we need to use *both* of these counting principles to solve a particular problem. The following example shows one such instance.

### EXAMPLE 7.1.10

Let us determine the number of license plates that can be formed with 3 uppercase letters followed by 3 digits such that the first letter is either  $A$  or  $B$ .

Let  $X_1$  be the set of all license plates with 3 uppercase letters followed by 3 digits such that the first letter is  $A$ . Let  $X_2$  be the set of all license plates with 3 uppercase letters followed by 3 digits such that the first letter is  $B$ . Then  $X_1 \cup X_2$  is the set of all license plates with 3 uppercase letters followed by 3 digits that can be formed such that the first letter is either  $A$  or  $B$ . Because the license plates in the set  $X_1$  begin with  $A$  and the license plates in the set  $X_2$  begin with  $B$ , it follows that  $X_1 \cap X_2 = \emptyset$ . By the addition principle,

$$|X_1 \cup X_2| = |X_1| + |X_2|.$$

Next we determine the number of elements in  $X_1$  and the number of elements in  $X_2$ . Let  $L$  be a license plate in  $X_1$ . Then  $L$  is of the form:

$$\begin{array}{c} A \\ \hline l_1 \quad l_2 \quad l_3 \quad d_1 \quad d_2 \quad d_3 \end{array}$$

The first letter can be chosen in only one way, each of the second and third letters can be chosen in 26 ways, and each of the digits can be chosen in 10 ways. Thus, by the multiplication principle, the license plate  $L$  can be formed in

$$1 \cdot 26 \cdot 26 \cdot 10 \cdot 10 \cdot 10 = 676,000$$

ways. Hence,  $|X_1| = 676,000$ .

A license plate in the set  $X_2$  must begin with the uppercase letter  $B$ . By arguments similar to those determining the number of elements in  $X_1$ , the number of elements in  $X_2$  is 676,000; that is,  $|X_2| = 676,000$ . It now follows that

$$|X_1 \cup X_2| = |X_1| + |X_2| = 676,000 + 676,000 = 1,352,000.$$

Therefore, the number of license plates with 3 uppercase letters followed by 3 digits that can be formed such that the first letter is either  $A$  or  $B$  is 1,352,000.

**REMARK 7.1.11** ▶ In Example 7.1.10, we can also determine the number of licence plates using the multiplication principle as follows: There are 2 choices for the first letter, 26 choices for each of the second and third letters, and 10 choices for each of the digits. Hence, the number of license plates with 3 uppercase letters followed by 3 digits that can be formed such that the first letter is either  $A$  or  $B$  is  $2 \cdot 26 \cdot 26 \cdot 10 \cdot 10 \cdot 10 = 1,352,000$ .

### EXAMPLE 7.1.12

The following items are available for breakfast: 4 types of cereal, 2 types of juice, and 3 types of bread. Let us determine the number of ways a breakfast can be prepared if exactly two items are selected from two different groups.

Because exactly two items must be selected from two different groups, a breakfast can be prepared in either of the following three ways:

- (i) a cereal and a juice, or
- (ii) a cereal and a bread, or
- (iii) a juice and a bread.

Let

$X$  be the set of breakfasts that consist of a cereal and a juice;

$Y$  be the set of breakfasts that consist of a cereal and a bread; and

$Z$  be the set of all breakfasts that consist of a juice and a bread.

Then  $X \cup Y \cup Z$  is the set of all breakfasts that consist of exactly two items from two different groups.

Notice that  $X \cap Y = \emptyset$ ,  $X \cap Z = \emptyset$ , and  $Y \cap Z = \emptyset$ ; that is, the sets  $X$ ,  $Y$ , and  $Z$  are pairwise disjoint.

Here some confusion may arise regarding  $X \cap Y = \emptyset$ . To clarify, suppose the set of cereals  $C = \{c_1, c_2, c_3, c_4\}$ , the set of juices  $J = \{j_1, j_2\}$ , and the set of breads  $B = \{b_1, b_2, b_3\}$ . This implies that  $X = \{\{c_1, j_1\}, \{c_1, j_2\}, \{c_2, j_1\}, \{c_2, j_2\}, \{c_3, j_1\}, \{c_3, j_2\}, \{c_4, j_1\}, \{c_4, j_2\}\} = \{\{c, j\} \mid c \text{ is a cereal, } j \text{ is a juice}\}$ . Similarly,  $Y = \{\{c, b\} \mid c \text{ is a cereal, } b \text{ is a bread}\}$ , and  $Z = \{\{j, b\} \mid j \text{ is a juice, } b \text{ is a bread}\}$ . It now follows that  $X \cap Y = \emptyset$ ,  $X \cap Z = \emptyset$ , and  $Y \cap Z = \emptyset$ .

We can now apply the addition principle to conclude that

$$|X \cup Y \cup Z| = |X| + |Y| + |Z|.$$

Next we determine,  $|X|$ , and  $|Y|$ , and  $|Z|$ .

A breakfast in set  $X$  consists of a cereal and a juice. Thus, we can first select a cereal and then a juice. Now a cereal can be selected in 4 ways and a juice can be selected in 2 ways. Thus, by the multiplication principle, a breakfast consisting of a cereal and a juice can be prepared in  $4 \cdot 2 = 8$  ways. Therefore,  $|X| = 8$ . In a similar manner, we can show that  $|Y| = 4 \cdot 3 = 12$  and  $|Y| = 2 \cdot 3 = 6$ . Consequently,

$$|X \cup Y \cup Z| = |X| + |Y| + |Z| = 8 + 12 + 6 = 26.$$

## The Principle of Inclusion-Exclusion

In the previous section, to apply the addition principle we used the fact that the number of elements in the set  $X_1 \cup X_2 \cup \dots \cup X_k$  is given by

$$|X_1 \cup X_2 \cup \dots \cup X_k| = |X_1| + |X_2| + \dots + |X_k|,$$

when  $X_i \cap X_j = \emptyset$  for all  $i$  and  $j$ ,  $i \neq j$ , and  $X_i$ 's are finite sets. In this section, we

derive a formula that will enable us to count the number of elements in  $X_1 \cup X_2 \cup \dots \cup X_k$  whether  $X_i \cap X_j = \emptyset$  or  $X_i \cap X_j \neq \emptyset$  for all  $i$  and  $j$ ,  $i \neq j$ .

First let us consider two finite sets,  $X_1$  and  $X_2$ . To count the number of elements in  $X_1 \cup X_2$ , we first count the number of elements in  $X_1$  and then add the number of elements in  $X_2$ . However, in this process the elements that are common to  $X_1$  and  $X_2$  are counted twice—once to determine  $|X_1|$  and the second time to determine  $|X_2|$ . Therefore, to determine the number of elements in  $X_1 \cup X_2$ , we must subtract  $|X_1 \cap X_2|$  from  $|X_1| + |X_2|$ , i.e.,

$$|X_1 \cup X_2| = |X_1| + |X_2| - |X_1 \cap X_2|. \quad (7.1)$$

### EXAMPLE 7.1.13

Let  $A$  be the set of positive integers that are  $\leq 30$  and multiples of 4. Let  $B$  be the set of positive integers that are  $\leq 30$  and multiples of 6. We want to determine the number of distinct elements in  $A \cup B$ .

Now  $A = \{4, 8, 12, 16, 20, 24, 28\}$  and  $B = \{6, 12, 18, 24, 30\}$ . Then  $A \cap B = \{12, 24\}$ . Also,  $|A| = 7$ ,  $|B| = 5$ , and  $|A \cap B| = 2$ . Hence,

$$|A \cup B| = |A| + |B| - |A \cap B| = 7 + 5 - 2 = 10.$$

A direct calculation, i.e., writing the elements of  $A \cup B$ , also shows that  $|A \cup B| = 10$ .

Another way to find the number of elements in  $A$  is as follows: Because  $A$  is the set of positive integers that are  $\leq 30$  and multiples of 4, it follows that  $|A| = \lfloor \frac{30}{4} \rfloor = 7$ .

### EXAMPLE 7.1.14

Let  $A$  denote the set of bit strings of length 6 that begin with 101 and  $B$  denote the set of bit strings of length 6 that terminate at 00. We determine the number of elements in  $A \cup B$ .

An element of  $A$  is of the form  $101a_4a_5a_6$ , where each  $a_i$  is 0 or 1. Therefore, by the multiplication principle, the number of elements in  $A$ ,  $|A| = 2 \cdot 2 \cdot 2 = 8$ .

An element of  $B$  is of the form  $b_1b_2b_3b_400$ , where each  $b_i$  is 0 or 1. It follows that the number of elements in  $B$  is  $|B| = 2 \cdot 2 \cdot 2 \cdot 2 = 16$ .

Let us now count the number of elements in  $A \cap B$ . An element of  $A \cap B$  is a bit string of length 6 that starts with 101 and ends with 00. Thus, an element of  $A \cap B$  is of the form  $101c_400$ , where  $c_4$  is 0 or 1. It follows that the number of elements in  $A \cap B$ ,  $|A \cap B| = 2$ .

Consequently, by formula (7.1),

$$|A \cup B| = |A| + |B| - |A \cap B| = 8 + 16 - 2 = 22.$$

Next we extend the formula given in (7.1) to three finite sets,  $A$ ,  $B$ , and  $C$ .

We have by Theorem 1.1.29(vi),

$$(A \cup B) \cap C = (A \cap C) \cup (B \cap C).$$

Now

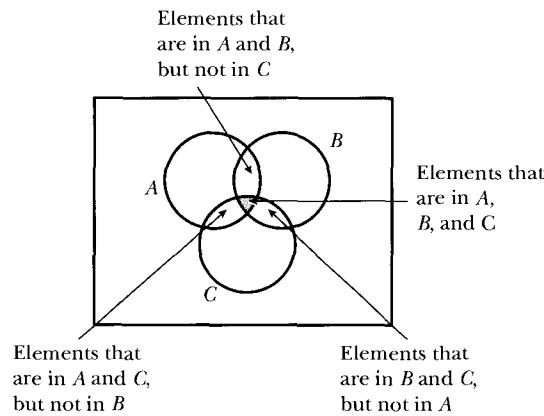
$$\begin{aligned} & |A \cup B \cup C| \\ &= |(A \cup B) \cup C| \\ &= |A \cup B| + |C| - |(A \cup B) \cap C| \quad \text{by (7.1)} \\ &= (|A| + |B| - |A \cap B|) + |C| - |(A \cap C) \cup (B \cap C)| \end{aligned}$$

$$\begin{aligned}
 &= |A| + |B| + |C| - |A \cap B| - \{|A \cap C| + |B \cap C| - |A \cap C \cap B \cap C|\} \\
 &\quad \text{by applying (7.1) to } (A \cap C) \cup (B \cap C) \\
 &= |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C| \\
 &\quad \text{because } A \cap C \cap B \cap C = A \cap B \cap C.
 \end{aligned}$$

We thus have

$$|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C|, \quad (7.2)$$

where  $A$ ,  $B$ , and  $C$  are finite sets. We can visualize the formula from the Venn diagram given in Figure 7.3.



**FIGURE 7.3** Venn diagram

As remarked earlier, the formula given in (7.1) is called the *inclusion-exclusion rule for two finite sets*. The formula given in (7.2) is called the *inclusion-exclusion rule for three finite sets*. Each of these formulas is also called the **inclusion-exclusion principle**.

The next example illustrates how to use the inclusion-exclusion rule for three sets.

#### EXAMPLE 7.1.15

In this example, we determine all positive integers less than 2102 that are divisible by at least one of the primes 2, 3, and 5.

Let  $A$  be the set of all positive integers that are less than or equal to 2102 and divisible by 2, i.e.,

$$\begin{aligned}
 A &= \{2n \mid n \in \mathbb{Z}, 0 < 2n \leq 2102\} \\
 &= \left\{2n \mid n \in \mathbb{Z}, 0 < n \leq \frac{2102}{2}\right\} \\
 &= \{2n \mid n \in \mathbb{Z}, 1 \leq n \leq 1051\}.
 \end{aligned}$$

This implies  $|A| = 1051$ . Similarly, let

$$\begin{aligned}
 B &= \{3n \mid n \in \mathbb{Z}, 0 < 3n \leq 2102\} \\
 &= \left\{3n \mid n \in \mathbb{Z}, 0 < n \leq \frac{2102}{3}\right\} \\
 &= \{3n \mid n \in \mathbb{Z}, 1 \leq n \leq 700\}.
 \end{aligned}$$

This implies  $|B| = 700$ .

Let

$$\begin{aligned} C &= \{5n \mid n \in \mathbb{Z}, 0 < 5n \leq 2102\} \\ &= \left\{5n \mid n \in \mathbb{Z}, 0 < n \leq \frac{2102}{5}\right\} \\ &= \{5n \mid n \in \mathbb{Z}, 1 \leq n \leq 420\}. \end{aligned}$$

This implies  $|C| = 420$ .

Now  $A \cap B$  is the set of all positive integers that are divisible by 2 and 3. Because 2 and 3 are relatively prime, we can show that an integer is divisible by 2 and 3 if and only if it is divisible by 6. Thus,

$$\begin{aligned} A \cap B &= \{6n \mid n \in \mathbb{Z}, 0 < 6n \leq 2102\} \\ &= \left\{6n \mid n \in \mathbb{Z}, 0 < n \leq \frac{2102}{6}\right\} \\ &= \{6n \mid n \in \mathbb{Z}, 1 \leq n \leq 350\}. \end{aligned}$$

Thus,  $|A \cap B| = 350$ .

Next  $A \cap C$  is the set of all positive integers that are divisible by 2 and 5. Because 2 and 5 are relatively prime, we can show that an integer is divisible by 2 and 5 if and only if it is divisible by 10. Thus,

$$\begin{aligned} A \cap C &= \{10n \mid n \in \mathbb{Z}, 0 < 10n \leq 2102\} \\ &= \left\{10n \mid n \in \mathbb{Z}, 0 < n \leq \frac{2102}{10}\right\} \\ &= \{10n \mid n \in \mathbb{Z}, 1 \leq n \leq 210\}. \end{aligned}$$

Thus,  $|A \cap C| = 210$ .

$B \cap C$  is the set of all positive integers that are divisible by 3 and 5. Because 3 and 5 are relatively prime, we can show that an integer is divisible by 3 and 5 if and only if it is divisible by 15. Thus,

$$\begin{aligned} B \cap C &= \{15n \mid n \in \mathbb{Z}, 0 < 15n \leq 2102\} \\ &= \left\{15n \mid n \in \mathbb{Z}, 0 < n \leq \frac{2102}{15}\right\} \\ &= \{15n \mid n \in \mathbb{Z}, 1 \leq n \leq 140\}. \end{aligned}$$

Thus,  $|B \cap C| = 140$ .

$A \cap B \cap C$  is the set of all positive integers that are divisible by 2, 3, and 5. Because 2, 3, and 5 are relatively prime to each other, we can show that an integer is divisible by 2, 3, and 5 if and only if it is divisible by 30. Thus,

$$\begin{aligned} A \cap B \cap C &= \{30n \mid n \in \mathbb{Z}, 0 < 30n \leq 2102\} \\ &= \left\{30n \mid n \in \mathbb{Z}, 0 < n \leq \frac{2102}{30}\right\} \\ &= \{30n \mid n \in \mathbb{Z}, 1 \leq n \leq 70\}. \end{aligned}$$

Thus,  $|A \cap B \cap C| = 70$ .

Consequently, by the inclusion-exclusion principle,

$$\begin{aligned} |A \cup B \cup C| &= |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C| \\ &= 1051 + 700 + 420 - 350 - 210 - 140 + 70 \\ &= 1541. \end{aligned}$$

The principle of inclusion-exclusion can be generalized to a finite number of sets as described in the next theorem. This theorem can be proved by using induction, so we leave the proof as an exercise.

**Theorem 7.1.16:** Let  $A_1, A_2, \dots, A_m$  be finite sets. Let

$$\begin{aligned} n_1 &= |A_1| + |A_2| + \cdots + |A_m| = \sum_{i=1}^m |A_i|, \\ n_2 &= |A_1 \cap A_2| + |A_1 \cap A_3| + \cdots + |A_1 \cap A_m| \\ &\quad + |A_2 \cap A_3| + |A_2 \cap A_4| + \cdots + |A_2 \cap A_m| \\ &\quad + |A_3 \cap A_4| + |A_3 \cap A_5| + \cdots + |A_3 \cap A_m| \\ &\quad + \cdots + |A_{m-1} \cap A_m| \\ &= \sum_{1 \leq i < j \leq m} |A_i \cap A_j|, \\ n_3 &= \sum_{1 \leq i < j < k \leq m} |A_i \cap A_j \cap A_k|, \\ n_t &= \sum_{1 \leq i_1 < i_2 < \cdots < i_t \leq m} |A_{i_1} \cap A_{i_2} \cap \cdots \cap A_{i_t}|, \quad t = 1, 2, \dots, m \\ n_m &= |A_1 \cap A_2 \cap \cdots \cap A_m|. \end{aligned}$$

Then

$$|A_1 \cup A_2 \cup \cdots \cup A_m| = n_1 - n_2 + n_3 - n_4 + \cdots + (-1)^{m-1} n_m.$$



## WORKED-OUT EXERCISES

**Exercise 1:** Find the number of integers between 4 and 100 that end with 3 or 5 or 7.

**Solution:** We divide the task into the following tasks.

$T_1$  : Find all integers between 4 and 100 that end with 3.

$T_2$  : Find all integers between 4 and 100 that end with 5.

$T_3$  : Find all integers between 4 and 100 that end with 7.

Now 13, 23, 33, 43, 53, 63, 73, 83, and 93 are the 9 integers between 4 and 100 that end with 3. There are 10 integers, 5, 15, 25, 35, 45, 55, 65, 75, 85, and 95, between 4 and 100 that end with 5. Also there are 10 integers, 7, 17, 27, 37, 47, 57, 67, 77, 87, and 97, between 4 and 100 that end with 7.

Hence, the tasks  $T_1$ ,  $T_2$ , and  $T_3$  can be completed in 9, 10, and 10 ways, respectively. The tasks are all independent of each other. Therefore, the number of ways to do one of these tasks is  $9 + 10 + 10 = 29$ . Hence, the number of integers between 4 and 100 that end with 3 or 5 or 7 is 29.

**Exercise 2:** Find the number of words of length 3 using the letters  $A, B, C, D$ , and  $E$  that start with the letter  $C$  such that no word contains a repetition of letters.

**Solution:** A word of length 3 can be constructed in three successive steps. Choose the first letter, then choose the second letter, and then choose the third letter. In the first step we must choose the letter  $C$ . Therefore, the first letter can be chosen in 1 way. Because no word contains a repetition of letters, once the first letter is chosen, the number of remaining letters is 4. The second letter can be any one of the remaining 4 letters. Therefore, the second letter can be chosen in 4 ways. After choosing the first and the second letters, the number of remaining letters is 3. The third letter can be any one of these 3 letters. Therefore, the third letter can be chosen in 3 ways.

Consequently, by the multiplication principle, there are  $1 \cdot 4 \cdot 3 = 12$  different words of length 3 that start with the letter  $C$  and do not contain a repetition of letters.

**Exercise 3:** Find the number of bit strings of length 8 that begin with 1011.

**Solution:** A bit string of length 8 that begins with 1101 is of the form  $1101a_5a_6a_7a_8$ , where each  $a_i$  is 0 or 1,  $i = 5, 6, 7, 8$ . These strings can be constructed in four successive steps. In

the first step, we choose  $a_5$ , which is either 0 or 1. Therefore,  $a_5$  can be selected in 2 different ways. Next we choose  $a_6$ , which can be chosen in 2 different ways. Similarly, each of  $a_7$  and  $a_8$  can be chosen in 2 different ways. Therefore, by the multiplication principle, the number of bit strings of length 8 that begin with 1101 is  $2 \cdot 2 \cdot 2 \cdot 2 = 16$ .

**Exercise 4:** Let  $A$  be a set with 8 elements and  $B$  be a set with 6 elements. Find the number of functions from  $A$  into  $B$ .

**Solution:** Let  $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8$  be the elements of  $A$ . Let  $T$  be the task of constructing a function, say  $f : A \rightarrow B$ . The function  $f$  is completely determined if we know the images  $f(a_i)$  for  $i = 1, 2, \dots, 8$ . Hence, to find the number of ways  $T$  can be completed, i.e., the number of functions from  $A$  into  $B$ , we divide  $T$  into the following eight successive steps  $T_1, T_2, \dots, T_8$ .

$T_i$  : Choose the image of  $a_i$  for  $i = 1, 2, \dots, 8$ .

Because the image of  $a_1$  can be any member of  $B$ , step  $T_1$  can be completed in 6 different ways. Similarly, each of steps  $T_i$  for  $i = 2, \dots, 8$  can be completed in 6 different ways. Hence, by the multiplication principle  $T$  can be completed in  $6 \cdot 6 = 6^8$  ways. Hence, the number functions from  $A$  into set  $B$  is  $6^8$ .

**Exercise 5:** Let  $A$  be a set with 5 elements and  $B$  be a set with 6 elements. Find the number of one-one functions from  $A$  into a set  $B$ .

**Solution:** Let  $a_1, a_2, a_3, a_4, a_5$  be the elements of  $A$ . Let  $T$  be the task of constructing a one-one function, say  $f : A \rightarrow B$ . The function  $f$  is completely determined if we know the images  $f(a_i)$  for  $i = 1, 2, \dots, 5$ . Hence, to find the number of ways  $T$  can be completed, i.e., the number of functions from  $A$  into  $B$ , we divide  $T$  into following five successive steps  $T_1, T_2, \dots, T_5$ .

$T_i$  : Choose the image of  $a_i$  for  $i = 1, 2, \dots, 5$ .

Because the image of  $a_1$  can be any member of  $B$ , the step  $T_1$  can be completed in 6 different ways. Now the function  $f$  must be one-one. Hence,  $f(a_2)$ , the image of  $a_2$ , must be different from  $f(a_1)$ , the image of  $a_1$ . Hence, after making a choice for  $f(a_1)$ , only 5 elements are left in  $B$  to make a choice for  $f(a_2)$ . It follows that there are 5 choices for  $f(a_2)$ ; that is, the image of  $a_2$  has 5 choices, so  $T_2$  can be completed in 5 different ways. Similarly, the image of  $a_3$  can be chosen in 4 different ways, the image of  $a_4$  can be chosen in 3 different ways, and the image of  $a_5$  can be chosen in 2 different ways. Hence, by the multiplication principle task  $T$  can be completed in  $6 \cdot 5 \cdot 4 \cdot 3 \cdot 2$  ways. Hence, the number of one-one functions from  $A$  into set  $B$  is  $6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 = 720$ .

**Exercise 6:** How many license plates of 4 letters from the English alphabet,  $A$  to  $Z$ , followed by 3 digits from 0 to 9 can be made, if repetition of letters is not allowed?

**Solution:** A license plate consisting of 4 letters followed by 3 digits is of the form

$$a_1 a_2 a_3 a_4 d_1 d_2 d_3,$$

where  $a_i \in \{A, B, C, \dots, Z\}$ ,  $i = 1, 2, 3, 4$  and  $d_j \in \{0, 1, 2, \dots, 9\}$ ,  $j = 1, 2, 3$ .

The license plates can be built in seven successive steps. In each of the first four steps, we choose a letter, and in each of the remaining three steps, we choose a digit.

Now  $a_1$  can be chosen in 26 different ways. Because the repetition of letters is not allowed, after choosing  $a_1$ , only 25 letters are left. Therefore, in the second step,  $a_2$  can be chosen in 25 different ways. Similarly,  $a_3$  can be chosen in 24 ways and  $a_4$  can be chosen in 23 ways.

Now consider the digits. Because the repetition of digits is allowed, in the remaining three steps, each of  $d_1, d_2$ , and  $d_3$  can be selected in 10 different ways.

Hence, by the multiplication principle, the number of license plates is

$$26 \cdot 25 \cdot 24 \cdot 23 \cdot 10 \cdot 10 \cdot 10 = 358,800,000.$$

**Exercise 7:** Suppose there are eight different books on algebra, four different books on discrete structures, five different books on computer science, and two different books on history. Determine the number of ways three different books can be selected from three different categories.

**Solution:** Let  $A$  denote an algebra book,  $D$  denote a discrete structures book,  $C$  denote a computer science book, and  $H$  denote a history book. Our task,  $T$ , is to select three different books from different categories. We divide the task  $T$  into tasks  $T_1, T_2, T_3$ , and  $T_4$ :

$T_1$  : Choose an algebra book, a discrete structures book, and a computer science book.

$T_2$  : Choose an algebra book, a discrete structures book, and a history book.

$T_3$  : Choose an algebra book, a computer science book, and a history book.

$T_4$  : Choose a discrete structures book, a computer science book, and a history book.

All these tasks are independent of each other. Hence, if the tasks  $T_1, T_2, T_3, T_4$  can be done in  $n_1, n_2, n_3, n_4$  ways, respectively, then by the addition principle task  $T$  can be done in  $n_1 + n_2 + n_3 + n_4$  ways.

Now consider task  $T_1$  : to choose an algebra book, a discrete structures book and a computer science book. Consider the triple  $(A, D, C)$ ,  $A$  for algebra book,  $D$  for discrete structures book, and  $C$  for computer science book. So task  $T_1$  is to choose the triple  $(A, D, C)$ .

There are eight ways an algebra book can be chosen, four ways a discrete structures book can be chosen, and five ways a computer science book can be chosen. Therefore, by the multiplication principle, task  $T_1$  can be done in  $8 \cdot 5 \cdot 4 = 160$  different ways. Hence,  $n_1 = 160$ .

In a similar manner, we can show that  $n_2 = 8 \cdot 4 \cdot 2 = 64$ ,  $n_3 = 8 \cdot 5 \cdot 2 = 80$ , and  $n_4 = 4 \cdot 5 \cdot 2 = 40$ . It now follows that

$$n_1 + n_2 + n_3 + n_4 = 160 + 64 + 80 + 40 = 344.$$

Therefore, we can select three books from three different categories in 344 ways.

## SECTION REVIEW

---

### Key Terms

addition principle

inclusion-exclusion principle

multiplication principle

### Some Key Results

1. Let  $X_1, X_2, \dots, X_k$  be sets such that the number of elements in  $X_i$  is  $n_i$ , that is,  $|X_i| = n_i$ ,  $i = 1, 2, \dots, k$ ,  $k \geq 2$ . Suppose that for any two sets  $X_i$  and  $X_j$ ,  $X_i \cap X_j = \emptyset$ ,  $i = 1, 2, \dots, k$ ,  $j = 1, 2, \dots, k$ ,  $i \neq j$ . That is, the sets  $X_1, X_2, \dots, X_k$  are pairwise disjoint. Then  $|X_1 \cup X_2 \cup \dots \cup X_k| = n_1 + n_2 + \dots + n_k$ .
2. Suppose that tasks  $T_1, T_2, \dots, T_k$  can be done in  $n_1, n_2, \dots, n_k$  ways, respectively. If all these tasks are independent of each other, then the number of ways to do one of these tasks is  $n_1 + n_2 + \dots + n_k$ .
3. Suppose that a task  $T$  can be completed in  $k$  successive steps. Suppose step 1 can be completed in  $n_1$  different ways, step 2 can be completed in  $n_2$  different ways, and in general, no matter how the preceding steps are completed, step  $k$  can be completed in  $n_k$  different ways. Then task  $T$  can be completed in  $n_1 n_2 \dots n_k$  different ways.
4. Let  $X_1$  and  $X_2$  be finite sets. Then  $|X_1 \cup X_2| = |X_1| + |X_2| - |X_1 \cap X_2|$ .

## EXERCISES

---

1. There are four routes from New York to Chicago, five routes from Chicago to Denver, and three routes from Denver to Los Angeles. Find the number of different routes from New York to Los Angeles via Chicago and Denver.
2. For a show a clown has four types of wigs, six types of dresses, and five types of shoes. How many ways can the clown dress up for the show?
3. For a birthday dinner, there are four types of soft drinks, three types of desserts, and five types of pizzas. A guest can choose one item from each group. How many ways can the dinner be served?
4. Two six-faced red and blue dice are thrown. What is the number of outcomes if the first die must show a 1?
5. Two six-faced distinct dice are thrown. What is the number of outcomes if the sum of the digits shown is 8?
6. Two six-faced distinct dice are thrown. What is the number of outcomes if the sum of the digits is not 5?
7. Two six-faced distinct dice are thrown. What is the number of outcomes if the sum of the digits shown is 3 or 6?
8. Two six-faced distinct dice are thrown. What is the number of outcomes if the sum of the digits shown is odd?
9. How many license plates consisting of three letters followed by four digits can be prepared if repetitions are allowed?
10. How many license plates consisting of three letters followed by four digits can be prepared if repetitions are not allowed?
11. How many license plates consisting of three letters followed by four digits can be prepared if the licence plates contain the letter A and repetitions are allowed?
12. How many license plates consisting of three letters followed by four digits can be prepared if the license plates contain the letter A and repetitions are not allowed?
13. How many strings of 0's and 1's of length 10 are there?
14. How many strings of 0's and 1's of length 6 that begin with 1 are there?
15. How many strings of 0's and 1's of length 8 that end with 10 are there?
16. How many strings of 0's and 1's of length 12 that begin with 11 and end with 00 are there?
17. How many strings of 0's and 1's of length 6 in which the second bit is 1 and the fifth bit is 0 are there?
18. How many strings of 0's and 1's of length 7 and containing exactly one 1 are there?
19. How many strings of 0's and 1's of length 6 and containing at least one 1 are there?
20. How many strings of 0's and 1's of length less than or equal to 5 are there?
21. How many strings of 0's and 1's of length less than or equal to 7 and that start with 1 or end with 0 are there?
22. Suppose there is set of four distinct mystery novels, five distinct romance novels, and three distinct poetry books.
  - a. In how many ways can these books be arranged on a shelf?

- b. In how many ways can these books be arranged on a shelf if all mystery novels are arranged first?  
 c. In how many ways can these books be arranged on a shelf if all poetry books stay together?
23. Three coins are tossed and the outcomes are placed in a row.
- How many outcomes are there?
  - How many outcomes contain at least two consecutive heads?
  - How many outcomes do not contain at least two consecutive heads?
  - How many outcomes do not contain exactly two heads?
24. Find the number of three-digit even numbers.
25. Find the number of three-digit even numbers divisible by 5.
26. Find the number of four-digit numbers divisible by 3 or 7.
27. Find the number of five-digit numbers with distinct digits.
28. Find the number of integers that are greater than or equal to 2 and less than or equal to 500 and
  - have distinct digits.
  - have distinct digits and are divisible by 5.
  - contain the digit 3.
29. Find the number of positive integers less than or equal to 500 and are
  - divisible by 3.
  - divisible by 5.
  - divisible by 7.
  - divisible by 3 and 5.
  - divisible by 3 and 7 or 5.
  - neither divisible by 3 nor divisible by 7.
30. Let  $A$  be a set with 10 elements. Find the number functions from  $A$  into a set  $B$  such that the number of elements in  $B$  is
  - 5.
  - 7.
  - 10.
  - 15.
31. Let  $A$  be a set with 8 elements. Find the number of one-one functions from  $A$  into a set  $B$  such that the number of elements in  $B$  is
  - 4.
  - 8.
  - 10.
  - 20.
32. Let  $A = \{a_1, a_2, \dots, a_n\}$  and  $B = \{x, y\}$  be sets.
  - Find the number of functions from  $A$  into  $B$  if each element of  $A$  is mapped to  $x$ .
  - Find the number of functions from  $A$  into  $B$  if each even subscripted element of  $A$  is mapped to  $x$  and each odd subscripted element of  $A$  is mapped to  $y$ .
  - Find the number of functions from  $A$  into  $B$  if each even subscripted element of  $A$  is mapped to  $x$ .
33. Let  $S$  be a set with 100 elements.
  - Find the number of subsets of  $S$  that have exactly one element.
  - Find the number of subsets of  $S$  that have at least two elements.
34. Let  $S$  be a set with  $n$  elements,  $n > 0$ .
  - Find the number of subsets of  $S$  that have exactly  $n - 1$  elements.
  - Find the number of subsets of  $S$  that do not have  $n - 1$  elements.
35. A palindrome is a string that reads the same forward and backward. For example, madam is a palindrome. Let  $A$  be a set of lowercase English letters.
  - Find the number of palindromes over the set  $A$  of length 10.
  - Find the number of palindromes over the set  $A$  of length 11.
  - Find the number of palindromes over the set  $A$  of length  $n$ ,  $n > 0$ .
36. Suppose Brad uses a string  $n_1 n_2 n_3 n_4$  of four digits from the digits 1, 2, 3, 4, 5, 6 as the identification number on the books in his collection. Find the maximum number of distinct identification numbers.
37. Consider an ISBN  $x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9 x_{10}$  for books  $x_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$  for  $1 \leq i \leq 9$ . Find the maximum number of expressions  $x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9$  if not all  $x_i$ 's are 0 at the same time.
38. Consider the following nested loops.
- ```
for i := 1 to 10 do
  for j := 1 to 20 do
    print "Hello";
```
- How many times is the word Hello printed?
  - How many times does the inner loop (**for** j := 1 **to** 20 **do**) execute? What is the number of iterations of this loop?
  - How many times does the outer (**for** i := 1 **to** 10 **do**) loop execute? What is the number of iterations of this loop?
39. Consider the following nested loops.
- ```
for i := 1 to 10 do
  for j := i to 10 do
    print "Hello";
```
- How many time is the word Hello printed?
  - How many times does the inner loop (**for** j := i **to** 10 **do**) execute? What is the number of iterations of this loop?
  - How many times does the outer (**for** i := 1 **to** 10 **do**) loop execute? What is the number of iterations of this loop?
40. Consider the following nested loops.
- ```
for i := 1 to 10 do
  for j := 1 to 20 do
    for k := 1 to 15 do
      print "Hello";
```
- How many times is the word Hello printed?
  - How many times does the innermost loop (**for** k := 1 **to** 15 **do**) execute? What is the number of iterations of this loop?

- c. How many times does the inner loop (`for j := 1 to 20 do`) execute? What is the number of iterations of this loop?
- d. How many times does the outer (`for i := 1 to 10 do`) loop execute? What is the number of iterations of this loop?

## 7.2 PIGEONHOLE PRINCIPLE

This chapter is about counting principles and in this section we describe another counting principle.

Let us consider the following problems.

- There are 13 people in a room. At least 2 of these 13 people must be born in the same month.
- Sometimes airlines, hoping for cancellations, overbook flights. If 101 people are booked for a trip and the plane has only 100 seats, then at least 2 people must be assigned the same seat.

There are many other problems like these in which an object with certain properties needs to be determined. For example, in the birthday problem, we want to show that there is a month with the property that at least two people are born in the same month. Such problems can be answered by applying the principle commonly known as the **pigeonhole principle**.

**The Pigeonhole Principle:** Suppose there are  $n$  pigeons,  $k$  pigeonholes, and  $n > k$ . If these  $n$  pigeons fly into these  $k$  pigeonholes, then some pigeonhole must contain at least two pigeons.

Figure 7.4 illustrates the pigeonhole principle, where each dot represents a pigeon.

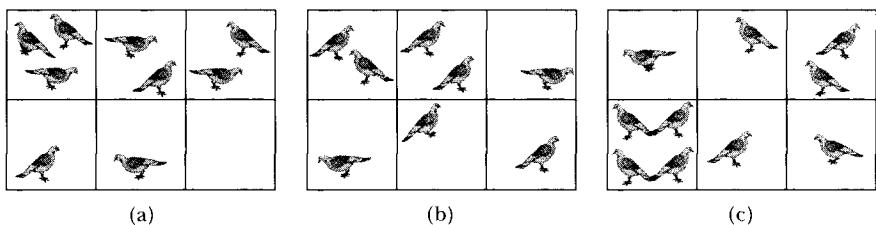


FIGURE 7.4 Pigeons in the pigeonholes

Notice that the pigeonhole principle only tells us that an object with the desired property exists. It does not tell us which object has the desired property or how to find that object.

The pigeonhole principle is also known as the *Dirichlet drawer principle*, or the *shoebox principle*. This principle was first formally stated by Peter Gustave Lejeune Dirichlet (1805–1859).

In order to apply the pigeonhole principle, we need to specify which objects are pigeons and which objects are pigeonholes.

### EXAMPLE 7.2.1

In this example, we answer the first problem posed at the beginning of this section: There are 13 people in a room. At least 2 of these 13 people must be born in the same month.

Here we can think of months as pigeonholes and people in the room as pigeons. Then, because there are 13 people,  $n = 13$ . Similarly, because there are 12 months in a year,  $k = 12$ . Therefore,  $n = 13 > 12 = k$ . Now a person's birthday is in one of the 12 months. Therefore, each of the 13 people must have their birthday in one of the 12 months. Because  $n > k$ , by the pigeonhole principle, at least 2 people must be born in the same month.

**EXAMPLE 7.2.2**

In this example, we answer the second problem posed at the beginning of this section: Sometimes airlines, hoping for cancellations, overbook flights. If 101 people are booked for a trip and the plane has only 100 seats, then at least 2 people must be assigned the same seat.

Here we can think of the seats as pigeonholes and the people holding seats as pigeons. Then  $n = 101$  and  $k = 100$ . Because,  $n > k$ , by the pigeonhole principle, at least 2 people must be assigned the same seat.

**EXAMPLE 7.2.3**

Students are registering for the coming semester. Due to popular demand, only three elective classes remain open in the computer science department. To graduate Ashley, Jeremy, Salma, Michael, and Carlos must take at least one of these classes. Then at least two of these students must be in the same class.

In this problem, a student must be assigned one of the three elective classes. Therefore, we can think of students as pigeons and elective classes as pigeonholes. Then  $n = 5$  and  $k = 3$ . Because  $n > k$ , by the pigeonhole principle, at least two students must be in the same elective class.

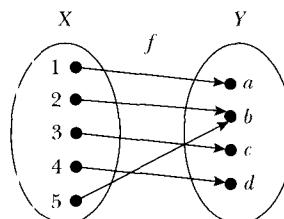
**Theorem 7.2.4:** Let  $X = \{x_1, x_2, \dots, x_n\}$  be a set with  $n$  distinct elements, and  $Y = \{y_1, y_2, \dots, y_k\}$  be a set with  $k$  distinct elements. Let

$$f : X \rightarrow Y$$

be a function. Suppose  $n > k$ . Then there exist  $x_i, x_j \in X$ ,  $i \neq j$  and  $y_t \in Y$  such that

$$f(x_i) = y_t \quad \text{and} \quad f(x_j) = y_t.$$

That is, at least two distinct elements of  $X$  must be mapped to the same element of  $Y$ . For example, see Figure 7.5 for a case in which when  $n = 5$  and  $k = 4$ .



**FIGURE 7.5** Function  $f$   
when  $n = 5$  and  $k = 4$

**Proof:** Because  $f$  is a function, each element  $x$  of  $X$  is mapped to an element of  $Y$ . Therefore, for each element  $y_i$  of  $Y$ , we consider the set  $A_{y_i}$  of all those elements of  $X$  that are mapped to  $y_i$ , i.e.,  $A_{y_i} = \{x \in X \mid f(x) = y_i\}$ . Then  $A_{y_i}$  is a subset of  $X$ . It is possible that for some  $y_i \in Y$ , the set  $A_{y_i}$  is empty. For example, if all elements of  $X$  are mapped to, say  $y_1$ , then  $A_{y_2} = \emptyset$ . Now  $f$  is a function from  $X$  into  $Y$ , and so every element of  $X$  must be mapped to only one element of  $Y$ . From this it follows that  $A_{y_i} \cap A_{y_j} = \emptyset$  if  $i \neq j$ . Thus,

$$|X| = |A_{y_1}| + |A_{y_2}| + \cdots + |A_{y_{k-1}}| + |A_{y_k}|.$$

Suppose no  $A_{y_i} = \{x \in X \mid f(x) = y_i\}$  contains at least two elements. Then

$$|X| \leq 1 + 1 + \cdots + 1 + 1 = k < n.$$

This contradicts our assumption that  $|X| = n$ . Hence, there exists some  $A_{y_i}$  that contains at least two distinct elements. That is, at least two distinct elements of  $X$  must be mapped to the same element of  $Y$ . ■

**REMARK 7.2.5** ► Theorem 7.2.4 is nothing but an equivalent statement of the pigeonhole principle. In fact, this theorem is known as the function form of the pigeonhole principle. Here you can think of the elements of  $X$  as pigeons and the elements of  $Y$  as pigeonholes.

### EXAMPLE 7.2.6

Let  $A = \{1, 2, 3, 4, 5, 6\}$ . We show that if we choose any four distinct members of  $A$ , then for at least one pair of these four integers their sum is 7.

Notice that  $\{1, 6\}$ ,  $\{2, 5\}$ , and  $\{3, 4\}$  are the only pairs of distinct integers such that  $1 + 6 = 7$ ,  $2 + 5 = 7$ ,  $3 + 4 = 7$ . Let  $X = \{x_1, x_2, x_3, x_4\}$  be any subset of four distinct elements of  $A$  and  $Y = \{\{1, 6\}, \{2, 5\}, \{3, 4\}\}$ . Now  $Y$  is a set of three distinct elements,  $y_1 = \{1, 6\}$ ,  $y_2 = \{2, 5\}$ , and  $y_3 = \{3, 4\}$ . Moreover, notice that  $\{\{1, 6\}, \{2, 5\}, \{3, 4\}\}$  is a partition of  $A$ .

Define  $f : X \rightarrow Y$  by  $f(a) = y_i$  if  $a \in y_i$ . (See Figure 7.6(a).) For example, if  $a = 1 \in X$ , then  $f(1) = \{1, 6\}$ .

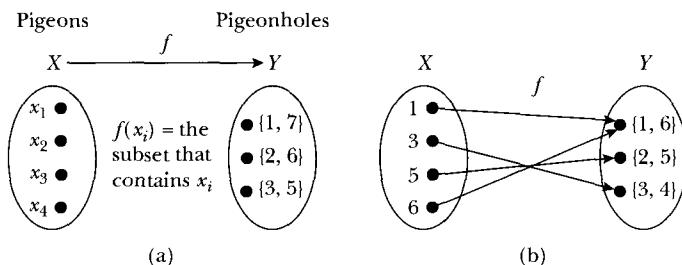


FIGURE 7.6 Function  $f$

For  $X = \{1, 3, 5, 6\}$ , see Figure 7.6(b).

Now  $|X| = 4$  and  $|Y| = 3$ . Then at least two distinct elements of  $X$  must be mapped to the same element of  $Y$ . Hence, if we choose any four distinct members of  $A$ , then for at least one pair of these four integers their sum is 7.

### EXAMPLE 7.2.7

Using instant messaging, every Sunday evening 10 friends communicate with each other. Instant messaging allows a person to open a separate window for each

person he or she is communicating with. Then at any time at least 2 of these 10 friends must be communicating with the same number of friends.

Let

$$X = \{x_1, x_2, \dots, x_{10}\}$$

be the set of 10 friends. For each  $x_i$ , let  $n_i$  be the number of friends they are communicating with,  $i = 1, 2, \dots, 10$ . Now a person may not be communicating with any person or may be communicating with as many as 9 people. Thus,  $0 \leq n_i \leq 9$ ,  $i = 1, 2, \dots, 10$ . Notice that if we take

$$Y = \{0, 1, 2, \dots, 9\},$$

then we cannot apply the pigeonhole principle because the number of elements in  $X$  and the number of elements in the set  $Y$  are the same.

Now a person is either communicating with at least one other person or not communicating with anyone. Suppose that one of the friends, say  $x_i$ , is not communicating with any other friend. Then  $n_i = 0$ . From this it follows that the remaining people can communicate with at most 8 other people. Thus, in this case,  $0 \leq n_i \leq 8$ ,  $i = 1, 2, \dots, 10$ ; i.e., the number of people with whom another person can communicate is in the set

$$Y = \{0, 1, \dots, 8\}.$$

Now each element of  $X$  is assigned a value in the set  $Y$ . We can think of the elements of  $X$  as pigeons and the elements of  $Y$  as pigeonholes. Then  $n = 10$  and  $k = 9$ . Because  $n > k$ , by the pigeonhole principle, at least two elements of  $X$  must be assigned the same value in the set  $Y$ . Thus, at least 2 of the friends are communicating with the same number of friends.

Now suppose that  $n_i \neq 0$  for all  $i = 1, 2, \dots, 10$ . Then a person can communicate with at most 9 other people. It follows that  $1 \leq n_i \leq 9$ ,  $i = 1, 2, \dots, 10$ . Let

$$Y = \{1, 2, \dots, 9\}.$$

Now  $|X| = 10$  and  $|Y| = 9$ . As in the previous case, we can conclude that at least 2 of the friends are communicating with the same number of friends.

**Generalized Pigeonhole Principle:** Suppose that there are  $n$  pigeons,  $k$  pigeonholes,  $n > k$ , and  $m = \lceil \frac{n}{k} \rceil$ . If these  $n$  pigeons fly into these  $k$  pigeonholes, then some pigeonhole must contain at least  $m$  pigeons.

### EXAMPLE 7.2.8

Suppose there are 50 people in a room. Then at least 5 of these people must have their birthday in the same month.

We can think of the people as pigeons and the months as pigeonholes. Then  $n = 50$ ,  $k = 12$ , and

$$m = \left\lceil \frac{50}{12} \right\rceil = 5.$$

By the generalized pigeonhole principle, at least 5 people must have their birthday in the same month.

As in Theorem 7.2.4, the generalized pigeonhole principle can also be stated in the following form: Let  $X = \{x_1, x_2, \dots, x_n\}$  be a set with  $n$  distinct elements and

$Y = \{y_1, y_2, \dots, y_k\}$  be a set with  $k$  distinct elements. Let

$$f : X \rightarrow Y$$

be a function. Suppose  $n > k$ . Let  $m = \lceil \frac{n}{k} \rceil$ . Then there exist  $m$  distinct elements, say  $a_1, a_2, \dots, a_m$  in  $X$ , such that

$$f(a_1) = f(a_2) = \dots = f(a_m).$$

The proof of this result is almost the same as the proof of Theorem 7.2.4. The only difference is that we have to replace the line “Suppose no  $A_{y_i} = \{x \in X \mid f(x) = y_i\}$  contains at least two elements” with “Suppose no  $A_{y_i} = \{x \in X \mid f(x) = y_i\}$  contains at least  $m$  elements.”

## WORKED-OUT EXERCISES

**Exercise 1:** A box that contains eight green balls and six red balls is kept in a completely dark room. What is the least number of balls one must take out from the box so that at least two balls will be the same color?

**Solution:** Let  $X$  be the set of all balls in the box and  $Y = \{G, R\}$ , where  $G$  indicates green and  $R$  indicates red color. Define a function  $f : X \rightarrow Y$  by  $f(b) = G$ , if the color of the ball is green and  $f(b) = R$ , if the color of the ball is red. (See Figure 7.7.)

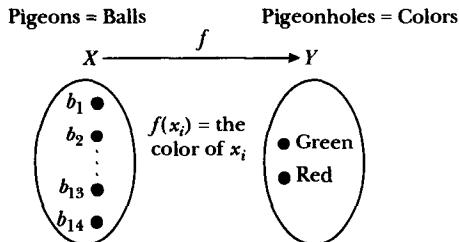


FIGURE 7.7 Function  $f$  from the set of balls to the set of colors

If we take a subset  $A$  of three balls of  $X$ , then we find that  $|A| > |Y|$ , so by the pigeonhole principle, at least two elements of  $A$  must be assigned the same value in  $Y$ . Therefore, at least two of the balls of  $A$  must have the same color. On the other hand, there may exist a subset  $B = \{b, c\}$  with two balls only such that  $f(b) = G$  and  $f(c) = R$ . Hence, the least number of balls that one must take out of the box so that at least two balls are the same color is three.

**Exercise 2:** Let  $X = \{x_1, x_2, \dots, x_{100}\}$  be a set of 100 distinct positive integers. If these positive integers are divided by 75, then show that at least two of the remainders must be the same.

**Solution:** Let  $r_i$  be the remainder when  $x_i$  is divided by 75,  $i = 1, 2, \dots, 100$ . Let

$$R = \{r_1, r_2, \dots, r_{100}\},$$

i.e.,  $R$  is the set of remainders. Then  $|R| = 100$ . If a positive integer  $x$  is divided by 75, then the remainder  $r$  is such that

$$0 \leq r \leq 74.$$

$$Let S = \{0, 1, \dots, 74\}.$$

For each element of  $R$  there is a corresponding element in  $S$ ; i.e., each element of  $R$  is assigned a value from the set  $S$ .

We can think of the elements of  $R$  as pigeons and the elements of  $S$  as pigeonholes. Then  $n = 100$  and  $k = 75$ . Because  $n > k$ , by the pigeonhole principle, at least two elements of  $R$  must be assigned the same value in  $S$ . Therefore, at least two of the  $r_i$ 's must be the same. Hence, at least two of the remainders must be the same.

**Exercise 3:** From the integers in the set  $\{1, 2, \dots, 30\}$ , what is the least number of integers that must be chosen so that at least one of them is divisible by 3 or 5?

**Solution:** The numbers in the given range that are divisible by 3 or 5 are 3, 5, 6, 9, 10, 12, 15, 18, 20, 21, 24, 25, 27, and 30. Thus, there are 14 numbers in the given range that are divisible by 3 or 5. This implies that there are 16 numbers that are not divisible by 3 or 5. It follows that we must choose at least 17 numbers from the given set to ensure that at least one of them is divisible by 3 or 5.

**Exercise 4:** Let  $\{a_i\}_{i=1}^n$  be a finite sequence of length  $n$ ; i.e., it has  $n$  elements.

- (i)  $\{a_i\}_{i=1}^n$  is called *strictly increasing* if  $a_1 < a_2 < \dots < a_n$ , i.e.,  $a_i < a_{i+1}$  for all  $i = 1, 2, \dots, n - 1$ .
- (ii)  $\{a_i\}_{i=1}^n$  is called *strictly decreasing* if  $a_1 > a_2 > \dots > a_n$ , i.e.,  $a_i > a_{i+1}$  for all  $i = 1, 2, \dots, n - 1$ .
- (iii) A *subsequence* of  $\{a_n\}$  is a sequence of the form  $a_{i_1}, a_{i_2}, \dots, a_{i_k}$ , where  $1 \leq i_1 < i_2 < \dots < i_k \leq n$ .

For example, 2, 6, 8, 7, 10 is a subsequence of 1, 2, 9, 6, 3, 8, 12, 7, 10, 15, 18.

Let  $a_1, a_2, a_3, \dots, a_{n^2+1}$  be a sequence of distinct  $n^2 + 1$  real numbers. Show that this sequence has a strictly increasing or strictly decreasing subsequence of  $n + 1$  elements.

**Solution:** Consider the element  $a_k$  of the sequence, where  $1 \leq k \leq n^2 + 1$ . Now starting at  $a_k$  we can construct a strictly increasing or strictly decreasing sequence.

For example, in the sequence 1, 2, 9, 6, 3, 8, 12, 7, 10, 15, starting at 2 we can construct the strictly increasing sequences 2, 9, 12 and 2, 8, 12, 15. Similarly, starting at 9, we can construct the strictly decreasing subsequence 9, 8, 7.

Because  $a_1, a_2, a_3, \dots, a_{n^2+1}$  is a finite sequence, the number of strictly increasing and strictly decreasing subsequences is finite. Among the strictly increasing subsequences, we can choose the increasing subsequence of the greatest length. Similarly, among the decreasing subsequences, we can choose the decreasing subsequence of the greatest length. Let  $i_k$  be the length of the largest strictly increasing subsequence starting at  $a_k$  and  $d_k$  be the length of the largest decreasing subsequence starting at  $a_k$ . We associate the pair  $(i_k, d_k)$  with the element  $a_k$ . Consider the set  $A = \{(i_k, d_k) \mid k = 1, 2, \dots, n^2 + 1\}$ .

If possible, suppose there is no strictly increasing or strictly decreasing subsequence of length  $n + 1$  in  $a_1, a_2, a_3, \dots, a_{n^2+1}$ . It follows that  $1 \leq i_k \leq n$  and  $1 \leq d_k \leq n$  for all  $k = 1, 2, \dots, n^2 + 1$ .

For each  $k$ ,  $i_k$  has  $n$  possible choices and  $d_k$  has  $n$  possible choices. Hence, for each  $k$ , the pair  $(i_k, d_k)$  has  $n^2$  possible choices.

The set  $A$  has  $n^2 + 1$  elements and each pair has  $n^2$  choices. It follows by the pigeonhole principle that at least two elements in the set  $A$  must be the same. That is, there exist integers  $u$  and  $v$ ,  $1 \leq u < v \leq n^2 + 1$  such that  $(i_u, d_u) = (i_v, d_v)$ , i.e.,  $i_u = i_v$  and  $d_u = d_v$ .

Consider  $a_u$  and  $a_v$ . Because all elements of the sequence  $a_1, a_2, a_3, \dots, a_{n^2+1}$  are distinct, it follows that  $a_u < a_v$  or  $a_u > a_v$ .

Suppose  $a_u < a_v$ . Now because  $i_u = i_v$ , there is a strictly increasing subsequence, say  $a_{i_1}, a_{i_2}, \dots, a_{i_{i_u}}$  of length  $i_u$  starting at  $a_v$ . Then  $a_u, a_{i_1}, a_{i_2}, \dots, a_{i_{i_u}}$  is a strictly increasing subsequence of length  $i_u + 1$  starting at  $a_u$ . This is a contradiction to the fact that the largest strictly increasing subsequence starting at  $a_u$  is of the length  $i_u$ .

Similarly, if  $a_u > a_v$ , then using the fact that  $d_u = d_v$ , we can show that  $a_v$  has a strictly decreasing subsequence of length  $d_v + 1$ .

Thus, our assumption is false. Consequently, the sequence  $a_1, a_2, a_3, \dots, a_{n^2+1}$  has a strictly increasing or strictly decreasing subsequence of  $n + 1$  elements.

**Exercise 5:** Let  $A$  be square such that each side is of length 2 inches. Show that if 5 points are placed inside  $A$ , then the distance between at least two of the points is  $\leq \sqrt{2}$ .

**Solution:** Let  $A$  be as shown in Figure 7.8.

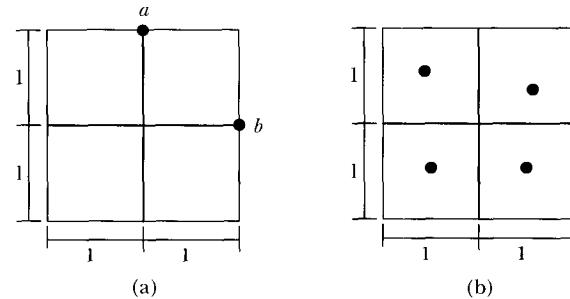


FIGURE 7.8 Square of size 2 inches

We have decomposed the square  $A$  into four smaller squares, each side of length 1 inch, as shown in Figure 7.8.

Let  $a$  and  $b$  be two points in a smaller square (see Figure 7.8(a)). The greatest distance between points  $a$  and  $b$  would occur only when they are on opposite corners of the square, in which case the distance between them is  $\sqrt{2}$ .

Now consider Figure 7.8(b). There are four squares, each side of length 1 inch. We need to place 5 points in the square  $A$ . By the pigeonhole principle, at least two of the points must be either inside a smaller square or on the boundaries of the smaller square. As observed in Figure 7.8(a), the greatest distance between these two points is  $\sqrt{2}$ . Thus, the distance between at least two of the points is  $\leq \sqrt{2}$ .

## SECTION REVIEW

### Key Terms

pigeonhole principle

generalized pigeonhole principle

### Some Key Definitions

1. Suppose that there are  $n$  pigeons,  $k$  pigeonholes, and  $n > k$ . If these  $n$  pigeons fly into these  $k$  pigeonholes, then some pigeonhole must contain at least two pigeons.
2. Let  $X = \{x_1, x_2, \dots, x_n\}$  be a set with  $n$  distinct elements and  $Y = \{y_1, y_2, \dots, y_k\}$  be a set with  $k$  distinct elements. Let  $f : X \rightarrow Y$  be a function. Suppose  $n > k$ . Then there exist  $x_i, x_j \in X$ ,  $i \neq j$  and  $y_t \in Y$  such that  $f(x_i) = y_t$  and  $f(x_j) = y_t$ .

$f(x_i) = y_i$ . That is, at least two distinct elements of  $X$  must be mapped to the same element of  $Y$ .

3. Suppose that there are  $n$  pigeons,  $k$  pigeonholes,  $n > k$ , and  $m = \lceil \frac{n}{k} \rceil$ . If these  $n$  pigeons fly into these  $k$  pigeonholes, then some pigeonhole must contain at least  $m$  pigeons.

## EXERCISES

---

1. There are 400 students in a programming class. Show that at least two of them were born on the same day of a month.
2. Let  $A = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7\}$  be a set of seven integers. Show that if these numbers are divided by 6, then at least two of them must have the same remainder.
3. Let  $A = \{1, 2, 3, 4, 5, 6, 7, 8\}$ . Show that if you choose any five distinct members of  $A$ , then there will be two integers such that their sum is 9.
4. Let  $A = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ . Is it true that if we choose any five distinct members of  $A$ , then the sum of two of the numbers chosen is 11? Justify your answer.
5. Let  $A = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ . Is it true that if we choose any six distinct members of  $A$ , then the sum of two of the numbers chosen is 11? Justify your answer.
6. Let  $n$  be a positive integer. Show that if a set of  $n + 1$  distinct integers are divided by  $n$ , then at least two of the remainders must be the same.
7. Suppose there is a group of 10 senators. Each senator must serve in one of eight committees. Show that there is at least one committee with more than one senator.
8. Let  $A$  be the set of integers 1, 2, 3, 4, 5. Show that if 4 numbers are selected from  $A$ , then at least two must add up to 6.
9. Let  $A$  be a set of  $2n$  integers,  $1, 2, 3, \dots, 2n - 1, 2n$ .
  - a. Show that if  $n + 1$  numbers are chosen from  $A$ , then at least one of them is divisible by 2.
  - b. Show that if  $n + 1$  are chosen from  $A$ , then the multiplication of at least two of them must be even.
10. Suppose that we have a deck of 52 cards.
  - a. How many cards must be picked up so that at least one of them is red?
  - b. How many cards must be picked up so that at least one of them is a heart?
  - c. How many cards must be picked up so that at least one of them is an ace?
11. From the integers in the set  $\{1, 2, \dots, 20\}$ , what is the least number of integers that must be chosen so that at least one of them is divisible by 4?
12. From a set of 50 integers, how many must be chosen so that at least two of them have the same remainder when divided by 9?
13. Let  $f : A \rightarrow B$  be a function from set  $A$  into set  $B$  and  $|A| = 10$  and  $|B| = 8$ . Show that  $f$  is not one-one.
14. Let  $a$  and  $b$  be integers such that  $b \neq 0$ . Show that when  $\frac{a}{b}$  is written as a decimal number, then either the expansion stops or a certain set of digits repeats.
15. In a group of 38 people, at least how many must have been born in the same month?
16. A group of 40 students in a class must share a set of 15 computers. To avoid conflict, each student is assigned only 1 computer. Moreover, no computer is assigned to more than 4 students. Prove that at least 2 computers are assigned to 3 or more students.
17. In a quiz taken by 70 students, the scores range from 60 to 88. At least how many students must have the same score?
18. Let  $a_1, a_2, a_3, \dots, a_{10}$  be a sequence of 10 elements. Show that this sequence has a strictly increasing or decreasing subsequence of 4 elements.
19. Let  $A = \{1, 2, \dots, 2n - 1, 2n\}$  be a set of integers. Let  $S \subseteq A$  such that  $S$  has  $n + 1$  elements. Show that  $S$  contains two elements  $a$  and  $b$  such that  $a | b$ .
20. Let  $A = \{a_1, a_2, a_3, a_4, a_5\}$  such that for all  $i$ ,  $1 \leq a_i \leq 7$ . Show that  $A$  has subsets  $S_1$  and  $S_2$  such that the sum of the elements of  $S_1$  and the sum of the elements of  $S_2$  are the same.
21. Suppose that we have the list of integers 100 to 500.
  - a. What is the maximum number of integers that can be chosen from this list so that the digits in each number are distinct?
  - b. What is the least number of integers that must be chosen so that at least one number has distinct digits?
  - c. What is the least number of integers that must be chosen so that at least two numbers have a digit in common?
22. Let  $A$  be a set of 101 integers such that none is divisible by 100. Show that there exists  $a, b \in A$  such that  $a - b$  is divisible by 100.
23. Let  $A$  be a set of  $n + 1$  integers,  $n \geq 1$ , such that none is divisible by  $n$ . Show that there exists  $a, b \in A$  such that  $a - b$  is divisible by  $n$ .
24. Let  $A$  be a square such that each side is of length 4 inches.
  - a. Show that if 5 points are placed inside  $A$ , then the distance between at least two of the points is  $\leq 2\sqrt{2}$ .
  - b. Show that if 17 points are placed inside  $A$ , then the distance between at least two of the points is  $\leq \sqrt{2}$ .

## 7.3 PERMUTATIONS

---

Suppose that you are given four colored blocks—Red, Green, Blue, and Purple. How many ways can two out of these four blocks be arranged in a row? We work with the blocks and come up with the following arrangements—

**RG, RB, RP, GR, GB, GP, BR, BG, BP, PR, PG, and PB,**

which is a total of 12 different arrangements. These are called 2-permutations of the set of blocks. Notice that the arrangement **RG** is different than the arrangement **GR**.

If we are only interested in knowing the number of arrangements, we could have argued as follows: We need to arrange two blocks out of four in a row. This activity can be completed in two steps. In the first step we select the first block and in the second step we select the second block. For the first step all four blocks are available, so the first step can be completed in 4 different ways. Because the first step has already used one of the blocks, for the second step only three blocks are available. Therefore, the second step can be completed in 3 different ways. By the multiplication principle the activity can be completed in

$$4 \cdot 3 = 12$$

ways. This way of counting arrangements is very useful if we are dealing with a large set. Next we formally define permutations.

---

**DEFINITION 7.3.1** ▶ Let  $S$  be a set with  $n$  distinct elements,  $n > 0$ . Let  $0 < r \leq n$ . An ordered arrangement of  $r$  elements of  $S$  is called an  **$r$ -permutation** of  $S$ . An  **$n$ -permutation** of  $S$  is called a **permutation** of  $S$ .

The definition of  **$r$ -permutation** of  $S$  given in Definition 7.3.1 is the commonly used definition of  **$r$ -permutation**. However, one can raise the following questions:

What is meant by an arrangement? What is meant by an ordered arrangement? In the example given at the beginning of this section, we showed some arrangements of colored blocks. For example, **RG**, **RB**, **RP**, and **GR** are some of the ordered arrangements. Moreover, we remarked that **RG** and **GR** are two different arrangements. Here we have arranged the blocks side by side. However, one may place the block **G** on top of the block **R** and claim that this arrangement is different from the other arrangements. In order to overcome all these ambiguities, we give the formal definition of  **$r$ -permutation** as follows.

---

**DEFINITION 7.3.2** ▶ Let  $S$  be a set of  $n$  distinct elements,  $n > 0$ . Let  $0 < r \leq n$ . A one-one function from the set  $I_r = \{1, 2, 3, \dots, r\}$  into  $S$  is called an  **$r$ -permutation** of  $S$ .

Using Definition 7.3.2, let us explain the arrangement of two colored blocks out of the given four colored blocks.

Here by an ordered arrangement or a permutation we mean a one-one function  $f : \{1, 2\} \rightarrow \{\text{R, G, B, P}\}$ . If  $f(1) = \text{R}$  and  $f(2) = \text{G}$ , then  $f$  is a one-one function and therefore a 2-permutation of  $S$ . This gives us an ordered arrangement where we place **R** first and **G** in second place. Again  $g : \{1, 2\} \rightarrow \{\text{R, G, B, P}\}$  given by  $g(1) = \text{G}$  and  $g(2) = \text{R}$  is another one-one function.

The function  $g$  is different from the function  $f$ , so these two functions result in two different 2-permutations. That is, they result in two different arrangements, **RG** and **GR**.

Now to find all 2-permutations of  $\{\text{R, G, B, P}\}$  we have to find all one-one functions  $f : \{1, 2\} \rightarrow \{\text{R, G, B, P}\}$ . The process consists of two steps.

**Step 1.** Determine the image of 1, i.e.,  $f(1)$ .

**Step 2.** Determine the image of 2, i.e.,  $f(2)$ .

Now the image of 1 can be any one of the elements **R**, **G**, **B**, **P**. Thus, step 1 can be completed in 4 different ways.

In step 2, we have to choose the image of 2. Because we are constructing a one-one function, the image of 2 must be different than the image of 1. In step 1, we have chosen one of the four elements **R**, **G**, **B**, or **P**, and so in step 2 we choose the image of 2 from the remaining three elements. Therefore, step 2 can be done in 3 different ways. Hence, by the multiplication principle, the whole process can be completed in one of the  $4 \cdot 3 = 12$  ways. Consequently, there are exactly 12 one-one functions  $f : \{1, 2\} \rightarrow \{\mathbf{R}, \mathbf{G}, \mathbf{B}, \mathbf{P}\}$ . In other words, the number of 2-permutations of  $\{\mathbf{R}, \mathbf{G}, \mathbf{B}, \mathbf{P}\}$  is 12.

Let  $S = \{x_1, x_2, \dots, x_n\}$  be a set with  $n > 0$  distinct elements and let  $0 < r \leq n$ . Suppose  $f$  is an  $r$ -permutation of  $S$ . Then  $f$  is a one-one function from  $\{1, 2, 3, \dots, r\}$  into  $S$ . Let  $f(i) = a_i$  for  $i = 1, 2, \dots, r$ . We call the expression  $a_1 a_2 a_3 \cdots a_r$  an *ordered arrangement* of  $r$  distinct elements of  $S = \{x_1, x_2, \dots, x_n\}$ .

It follows that corresponding to every  $r$ -permutation of  $S$  there is an ordered arrangement of  $r$  distinct elements of  $S$ .

Conversely, suppose that  $a_1 a_2 a_3 \cdots a_r$  is an ordered arrangement of  $r$  distinct elements of  $S = \{x_1, x_2, \dots, x_n\}$ . Then the function  $f : \{1, 2, \dots, r\} \rightarrow S$  defined by  $f(i) = a_i$  for  $i = 1, 2, \dots, r$  is a one-one function and therefore an  $r$ -permutation of  $S$ .

We thus see that there is a one-to-one correspondence between the set of  $r$ -permutations of  $S$  and the set of ordered arrangements of  $r$  distinct elements of  $S$ . Hence, from now on, we will not distinguish between ordered arrangements and one-one functions.

Notice that the ordered arrangement  $x_1 x_2 \cdots x_r$  is different than the ordered arrangement  $x_2 x_1 \cdots x_r$ .

Next we determine a formula for determining the number of  $r$ -permutations of a set with  $n > 0$  distinct elements. However, first let us make the following definition.

---

**DEFINITION 7.3.3** ▶ Let  $S$  be a set with  $n$  distinct elements,  $n > 0$ . Let  $0 < r \leq n$ . Then  $P(n, r)$  denotes the number of  $r$ -permutations of  $S$ .

**Theorem 7.3.4:** Let  $S$  be a set with  $n$  distinct elements,  $n > 0$ .

Let  $0 < r \leq n$ . Then  $P(n, r)$  is given by the following formula:

$$P(n, r) = n(n - 1)(n - 2) \cdots (n - r + 1).$$

**Proof:** An  $r$ -permutation of  $S$  is an ordered arrangement of  $r$  elements of  $S$ . Therefore, an  $r$ -permutation of  $S$  can be constructed in a sequence of  $r$ -successive steps:

$$\overbrace{\quad \quad \quad \cdots \quad}^{s_1 \ s_2 \ s_3 \ \cdots \ s_r},$$

where  $s_i$  denotes the  $i$ th step. In the first step, we choose the first element of the  $r$ -permutation; in the second step, we choose the second element, and so on. Next we determine the number of ways each of these  $r$  steps can be completed.

Initially, all  $n$  elements of  $S$  are available. Any of these  $n$  elements can be selected as the first element of the  $r$ -permutation. Therefore, the first step can be completed in  $n$  different ways. After completing the first step, only  $n - 1$  elements are left. Therefore, step 2 can be completed in  $n - 1$  different ways. In general, after selecting the first  $i - 1$  elements of the  $r$ -permutation, for the  $i$ th element  $n - (i - 1) = n - i + 1$  elements are left, where  $1 \leq i \leq r$ . Any of these  $n - i + 1$  elements can be selected as the  $i$ th element of the  $r$ -permutation. Therefore, the  $i$ th step can be completed in  $n - i + 1$  different ways.

It now follows by the multiplication principle that the number of ways an  $r$ -permutation can be constructed is:

$$n(n - 1)(n - 2) \cdots (n - r + 1)$$

ways. Hence,

$$P(n, r) = n(n - 1)(n - 2) \cdots (n - r + 1). \blacksquare$$

---

**REMARK 7.3.5** ▶ Let  $n$  and  $r$  be two integers such that  $1 \leq r \leq n$ . Recall that for any positive integer  $n$ , the product  $n(n - 1)(n - 2) \cdots 3 \cdot 2 \cdot 1$  is denoted by  $n!$ . Moreover,  $0! = 1$ . Now notice that

$$\begin{aligned} \frac{n!}{(n - r)!} &= \frac{n(n - 1)(n - 2) \cdots (n - r + 1)(n - r)!}{(n - r)!} \\ &= n(n - 1)(n - 2) \cdots (n - r + 1). \end{aligned}$$

Hence,

$$P(n, r) = n(n - 1)(n - 2) \cdots (n - r + 1) = \frac{n!}{(n - r)!}.$$

**Corollary 7.3.6:** Let  $S$  be a set with  $n$  distinct elements,  $n > 0$ . Then  $P(n, n)$ —the number of  $n$ -permutations of  $S$ —is given by:

$$P(n, n) = n!.$$

**Proof:** By Remark 7.3.5,

$$P(n, n) = \frac{n!}{(n - n)!} = \frac{n!}{0!} = \frac{n!}{1} = n!. \blacksquare$$

## WORKED-OUT EXERCISES

**Exercise 1:** Let  $S$  be a set with 50 elements. Find the number of 3-permutations of  $S$  and find the number of permutations of  $S$ .

**Solution:** The number of 3-permutations of  $S$  is:

$$P(50, 3) = 50 \cdot 49 \cdot 48 = 117,600.$$

The number of permutations of  $S$  is:

$$P(50, 50) = 50!.$$

**Exercise 2:** How many dance pairs, (dance pairs means a pair  $(W, M)$ , where  $W$  stands for a woman and  $M$  for man), can be formed from a group of 6 women and 10 men?

**Solution:** The problem is equivalent to finding all one-one functions from the set of 6 women to the set of 10 men, which is the same as finding all 6-permutations of a set of 10 elements. Now the number of 6-permutations of a set of 10 elements is  $P(10, 6) = \frac{10!}{(10-6)!} = \frac{10!}{4!} = 10 \cdot 9 \cdot 8 \cdot 7 \cdot 6 \cdot 5 = 151,200$ .

**Exercise 3:** How many four-letter words can be formed from the letters  $G, R, O, U, P, S$  if no letter is to be used more than once in any word?

**Solution:** We will arrange four distinct letters in a row to form a four-letter word. If we change the ordering of letters we will get a different word.

Therefore, the number of four-letter words where no letter is to be used more than once in any word

$$\begin{aligned} &= \text{the number of ordered arrangements of the letters } G, R, O, U, P, S \text{ taken four at a time} \\ &= \text{the number of 4-permutations of } \{G, R, O, U, P, S\} \\ &= \text{the number of permutations of six distinct elements taken four at a time} \\ &= P(6, 4) = \frac{6!}{(6-4)!} = \frac{6!}{2!} = 6 \cdot 5 \cdot 4 \cdot 3 = 360. \end{aligned}$$

**Exercise 4:** How many numbers between 20000 and 50000 can be formed with the digits 1, 2, 3, 4, 5, 6 such that no digits are repeated in any of the numbers so formed?

**Solution:** Any number between 20000 and 50000 is a five-digit number of the form  $a_5 a_4 a_3 a_2 a_1$  such that  $a_i \in \{1, 2, 3, 4, 5, 6\}$  and  $a_5 \neq 1, 5, 6$ . If all the five digits are distinct and there are no restrictions for  $a_5$ , then the number of five-digit numbers is the same as the number of permutations of five elements of the set  $\{1, 2, 3, 4, 5, 6\}$ , which equals  $P(6, 5) = \frac{6!}{(6-5)!} = \frac{6!}{1!} = 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 = 720$ .

Let us now find in how many of the numbers  $a_5 a_4 a_3 a_2 a_1$ ,  $a_5$  is 1, 5, or 6.

If  $a_5$  is 1, then we will find the numbers  $1 a_4 a_3 a_2 a_1$ , where  $a_i \in \{2, 3, 4, 5, 6\}$  for  $i = 1, 2, 3, 4$  and the number of such numbers is the same as the number of permutations of four elements of the set  $\{2, 3, 4, 5, 6\}$ , which equals  $P(5, 4) = \frac{5!}{(5-4)!} = \frac{5!}{1!} = 5 \cdot 4 \cdot 3 \cdot 2 = 120$ .

Similarly, there are 120 numbers of the form  $5 a_4 a_3 a_2 a_1$  and also there are 120 numbers of the form  $6 a_4 a_3 a_2 a_1$ . Because we will not count the numbers  $1 a_4 a_3 a_2 a_1$ ,  $5 a_4 a_3 a_2 a_1$ , and  $6 a_4 a_3 a_2 a_1$ , we find that there are  $720 - 120 - 120 - 120 = 360$  numbers lying between 20000 and 50000 that can be formed with the digits 1, 2, 3, 4, 5, 6 such that digits are not repeated in any of the numbers so formed.

**Exercise 5:** In how many ways can six boys and five girls stand in a line so that no two girls are next to each other?

**Solution:** According to the given conditions, between two girls there must be a boy. Suppose the six boys are  $B_1, B_2$ ,

$B_3, B_4, B_5$ , and  $B_6$  and they stand in a line

$$G \quad B_1 \quad G \quad B_2 \quad G \quad B_3 \quad G \quad B_4 \quad G \quad B_5 \quad G \quad B_6 \quad G$$

where  $G$ 's denote the positions for girls. For girls there are seven positions. In these seven positions five different girls can stand in  $P(7, 5)$  different ways (because it is a 5-permutation of seven elements). After each arrangement of the girls the six boys can stand in six different places in  $P(6, 6)$  different ways (because it is a 6-permutation of six elements). Hence, by the multiplication principle, the number of ways six boys and five girls may stand in a line so that no two girls are next to each other is

$$\begin{aligned} P(7, 5) \cdot P(6, 6) &= \frac{7!}{(7-5)!} \cdot 6! = \frac{7!}{2!} \cdot 6! \\ &= \frac{7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2!}{2!} \cdot 6! = 1814400. \end{aligned}$$

**Exercise 6:** In how many ways can six boys and five girls stand in a line so that all the boys stand side by side and all the girls stand side by side?

**Solution:** An arrangement of the required type looks as follows:

$$G \quad G \quad G \quad G \quad G \quad B \quad B \quad B \quad B \quad B \quad B$$

or

$$B \quad B \quad B \quad B \quad B \quad G \quad G \quad G \quad G \quad G$$

Here  $G$  denotes the position of a girl and  $B$  denotes the position of a boy. Now five girls can be arranged in a row in  $P(5, 5)$  ways (because it is a 5-permutation of five elements). After each arrangement of the girls the six boys can stand in six different places in  $P(6, 6)$  different ways (because it is a 6-permutation of six elements). Hence, by the multiplication principle, the number of ways five girls and six boys may stand in a line so that all the girls stand first and then all the boys is  $P(5, 5) \cdot P(6, 6) = 5! \cdot 6!$ . Similarly, the number of ways five girls and six boys can stand in a line so that all the boys stand first and then all the girls is

$$P(6, 6) \cdot P(5, 5) = 6! \cdot 5!$$

Hence the number of ways six boys and five girls stand in a line so that all the boys stand side by side and all the girls stand side by side is

$$5! \cdot 6! + 6! \cdot 5! = 2 \cdot 5! \cdot 6!$$

## SECTION REVIEW

### Key Terms

permutation

$n$ -permutation

$P(n, r)$

$r$ -permutation

## Some Key Results

- Let  $S$  be a set with  $n$  distinct elements,  $n > 0$ . Let  $0 < r \leq n$ . Then  $P(n, r)$  is given by the following formula:  $P(n, r) = n(n - 1)(n - 2) \cdots (n - r + 1)$ .
- Let  $S$  be a set with  $n$  distinct elements,  $n > 0$ . Then  $P(n, n)$ -the number of  $n$ -permutations of  $S$  is given by  $P(n, n) = n!$ .

## EXERCISES

---

- Find  $P(10, 3), P(15, 10), P(6, 0), P(6, 6)$ .
- Find the positive integer  $n$  such that  $P(n + 1, 3) = 10 \cdot P(n - 1, 2)$ .
- Show that for any positive integer  $n$ ,  $P(n, n - 1) = n!$
- Find the number of different one-one functions from the set  $\{1, 2, 3, 4\}$  into  $\{E, F, G, H, I, J\}$ .
- Find the number of different one-to-one correspondences from a set of four distinct elements into itself.
- Find the number of different ways a grade of A, B, C, or D can be assigned to three students of a class so that no two students receive the same grade.
- Three friends go to a movie where they find seven vacant seats in a row. In how many different ways they can seat themselves?
- There are 12 hospitals in a town. How many different ways can seven patients be sent to the hospitals so that no two patients may be in the same hospital?
- How many three-letter words can be formed from the letters  $A, N, G, R, Y$  if no letter is to be used more than once in a word?
- How many distinct five-letter words can be formed from the letters  $A, N, G, R, Y$  if no letter is to be used more than once in a word?
- How many different ways can the letters of the word *MONDAY* be arranged? How many of them start with *M* and do not end with *Y*?
- Find the different numbers of three digits that can be formed with the digits 1, 2, 3, 4, 5, 6 such that no digit is repeated in any of the numbers so formed.
- How many numbers between 3000 and 4000, with distinct digits, can be formed using the digits 1, 3, 4, 5, 6?
- How many numbers greater than 3000, with distinct digits, can be formed using the digits 1, 3, 4, 5, 6?
- How many even numbers greater than 500, with distinct digits, can be formed using the digits 3, 4, 5, 6, 7?
- Determine the four-digit numbers, with distinct digits, that can be formed using the digits 0, 1, 2, 3, 4, 5, 6 such that none of the numbers have a leading 0.
- In how many ways can six boys and eight girls be arranged in a line so that no two boys may occupy consecutive positions?
- How many ways can six boys and six girls be seated in a row if the boys and girls are to have alternate seats?
- Seven rooms are available in a motel. Four visitors come to the motel and ask for separate rooms. In how many ways can the manager assign the rooms?
- Find the number of seating arrangements in a row of eight students so that two particular students will not sit side by side.
- Find the number of six-letter words that can be formed from the letters of the word *HISTORY* if no letter is used more than once in any word subject to the conditions given below.
  - The first letter of each word is *H*.
  - The first letter of each word is either *H* or *Y*.
  - The word starts with *HIS*.
  - The word contains *HIS* as a substring.
- How many dance pairs (*G, B*), where *G* stands for girl and *B* for boy, can be formed from a group of 10 girls and 15 boys?
- How many different four-digit numbers with distinct digits can be formed using the digits 1, 2, 3, 4, 5, 6?

## 7.4 COMBINATIONS

---

In the previous section, we were interested not only in selecting certain elements, but also in arranging them in a row. However, there are many situations in which we are only interested in selecting certain elements. For example, consider the following problem.

Suppose that there are 4 students,  $s_1, s_2, s_3$ , and  $s_4$ , interested in serving on a committee that handles dorm-room assignments. How many ways can such a committee of 2 students be formed? Notice that here we are only interested in selecting 2 students out of 4. Let  $S = \{s_1, s_2, s_3, s_4\}$  be the set of these students. A committee of 2 students out of these 4 students is a 2-element subset of  $S$ . Thus, the number of ways to form a committee of 2 students reduces to finding the number

of 2-element subsets of  $S$ . Notice that the subsets of  $S$  consisting of 2 elements or the 2-element subsets of  $S$  are:  $\{s_1, s_2\}$ ,  $\{s_1, s_3\}$ ,  $\{s_1, s_4\}$ ,  $\{s_2, s_3\}$ ,  $\{s_2, s_4\}$ , and  $\{s_3, s_4\}$ . Thus, there are 6 ways a committee of 2 students out of 4 students can be formed.

We find that the above problem is the same as counting the number of subsets of a set that contains a specified number of elements. The following theorem tells how to find the  $r$ -element subsets of a set with  $n$  elements, where  $0 \leq r \leq n$ .

**Theorem 7.4.1:** Let  $S$  be a set with  $n > 0$  elements. Let  $r$  be an integer such that  $0 \leq r \leq n$ . The number of subsets that contains  $r$  elements of  $S$  is  $\frac{n!}{r!(n-r)!}$ .

**Proof:** See the Worked-Out Exercise 3, page 445. ■

The counting problems described at the beginning of this section are referred to as the *combination problems*. In the remainder of this section, in addition to deriving a formula to answer such problems, we also look at other similar problems.

Notice that the sets

$$\{s_1, s_2\} = \{s_2, s_1\},$$

but the arrangement

$$s_1s_2 \neq s_2s_1.$$

---

**DEFINITION 7.4.2** ▶ Let  $S$  be a set with  $n > 0$  elements. Let  $r$  be an integer such that  $0 \leq r \leq n$ . An  **$r$ -combination** of  $S$  or a **combination** of the elements of  $S$  taken  $r$  at a time is a subset  $A$  of  $S$  such that  $A$  contains  $r$  elements of  $S$ .

---

**DEFINITION 7.4.3** ▶ Let  $S$  be a set with  $n > 0$  elements. Let  $r$  be an integer such that  $0 \leq r \leq n$ . Then  $C(n, r)$  denotes the number of  $r$ -combinations or the number of combinations of elements of  $S$  taken  $r$  at a time, i.e., the number of  $r$ -element subsets of  $S$ .

---

**REMARK 7.4.4** ▶ Another common notation for  $C(n, r)$  is  $\binom{n}{r}$ . Moreover,  $C(n, r)$  and  $\binom{n}{r}$  are read as the number of  $r$ -combinations of a set with  $n$  distinct elements.

The following theorem gives an explicit formula for  $C(n, r)$ .

**Theorem 7.4.5:** Let  $S$  be a set with  $n > 0$  elements. Let  $r$  be an integer such that  $0 \leq r \leq n$ . Then

$$C(n, r) = \frac{n!}{r!(n-r)!}.$$

**Proof:** Because  $C(n, r)$  denotes the number of subsets that contain  $r$  elements of  $S$ , the theorem follows from Theorem 7.4.1. However, let us give another proof of this theorem.

We prove this result by deriving a formula for  $P(n, r)$  in terms of  $C(n, r)$  and then use Theorem 7.3.4, where  $P(n, r)$  is the number of  $r$ -permutations of  $S$ .

An  $r$ -permutation of  $S$  is an ordered arrangement of  $r$  elements of  $S$ . Now an  $r$ -permutation can be constructed in two successive steps.

Step 1: Select an  $r$ -element subset of  $S$ , say  $S_r$ .

Step 2: Construct a permutation of  $S_r$ .

Because  $C(n, r)$  denotes the number of  $r$ -element subsets of  $S$ , step 1 can be completed in  $C(n, r)$  ways. Now  $S_r$  is a set with  $r$  elements. Therefore, the number of permutations of  $S_r$  is  $P(r, r)$ .

Now step 1 can be completed in  $C(n, r)$  ways, and step 2 can be completed in  $P(r, r)$  ways. Hence, by the multiplication principle, the number of ways an  $r$ -permutation can be formed is

$$C(n, r)P(r, r).$$

It now follows that

$$P(n, r) = C(n, r)P(r, r).$$

This implies that

$$C(n, r) = \frac{P(n, r)}{P(r, r)}.$$

By Remark 7.3.5,

$$P(n, r) = \frac{n!}{(n - r)!}$$

and by Corollary 7.3.6,

$$P(r, r) = r!.$$

Hence,

$$C(n, r) = \frac{P(n, r)}{P(r, r)} = \frac{n!}{r!(n - r)!}. \blacksquare$$

**Corollary 7.4.6:** Let  $S$  be a set with  $n > 0$  elements. Let  $r$  be an integer such that  $0 \leq r \leq n$ . Then

$$C(n, r) = C(n, n - r).$$

**Proof:** By Theorem 7.4.5,

$$\begin{aligned} C(n, n - r) &= \frac{n!}{(n - r)!(n - (n - r))!} \\ &= \frac{n!}{(n - r)!(n - n + r)!} \\ &= \frac{n!}{(n - r)!(r)!} \\ &= \frac{n!}{r!(n - r)!} \\ &= C(n, r). \blacksquare \end{aligned}$$

**REMARK 7.4.7** ▶ The proof of Corollary 7.4.6 can also be given as follows: Let  $0 \leq r \leq n$ . In a selection of  $r$  elements, we do not select the other  $n - r$  elements. Therefore, we have two subsets of  $S$ , one with  $r$  elements and the other with  $n - r$  elements. For each selection of  $r$  elements, we have a selection of  $n - r$  elements. And conversely, for each selection of  $n - r$  elements, we have a selection of  $r$  elements. Hence,  $C(n, r) = C(n, n - r)$ .

**Corollary 7.4.8:** Let  $n$  be a nonnegative integer. Then

$$C(n, n) = 1.$$

## WORKED-OUT EXERCISES

**Exercise 1:** In how many ways can a soccer team of 11 players be selected from a group of 20 players?

**Solution:** We are to select 11 players out of 20 players. For this we find all 11-combinations of a set of 20 elements. The number of such combinations is

$$\begin{aligned} C(20, 11) &= \frac{20!}{11!(20-11)!} = \frac{20!}{11!9!} \\ &= \frac{20 \cdot 19 \cdot 18 \cdot 17 \cdot 16 \cdot 15 \cdot 14 \cdot 13 \cdot 12}{9 \cdot 8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2} = 167960. \end{aligned}$$

**Exercise 2:** If  $C(16, r) = C(16, r + 2)$ , then find  $r$ .

**Solution:**  $C(16, r) = C(16, r + 2)$  implies either  $r = r + 2$  or  $r + (r + 2) = 16$ . Now  $r \neq r + 2$ . Therefore,  $r + (r + 2) = 16$ , which implies  $r = 7$ .

We can also determine the value of  $r$  using a direct computation as follows:

$$\begin{aligned} C(16, r) &= C(16, r + 2) \\ \Rightarrow \frac{16!}{r!(16-r)!} &= \frac{16!}{(r+2)!(16-r-2)!} \\ \Rightarrow (r+2)!(16-r-2)! &= r!(16-r)! \\ \Rightarrow (r+2)(r+1)r!(16-r-2)! &= r!(16-r) \\ &\quad (16-r-1)(16-r-2)! \\ \Rightarrow (r+2)(r+1) &= (16-r)(16-r-1) \\ \Rightarrow r^2 + 3r + 2 &= (16-r)^2 - (16-r) \\ \Rightarrow r^2 + 3r + 2 &= 256 - 32r + r^2 - 16 + r \\ \Rightarrow 3r + 2 &= 240 - 31r \\ \Rightarrow 34r &= 238 \\ \Rightarrow r &= 7. \end{aligned}$$

**Exercise 3:** Let  $S$  be a set containing  $n$  elements, where  $n$  is a positive integer. If  $r$  is an integer such that  $0 \leq r \leq n$ ,

then show that the number of subsets of  $S$  containing exactly  $r$  elements is

$$\frac{n!}{r!(n-r)!}.$$

**Solution:** We prove this result by induction on  $n$ .

Let  $P(n)$  be the statement: If  $S$  is a set containing  $n$  ( $> 0$ ) elements, then the number of subsets of  $S$  containing exactly  $r$  elements ( $0 \leq r \leq n$ ) is  $\frac{n!}{r!(n-r)!}$ .

*Basis step:* Suppose  $n = 1$ . Then  $S$  has only one element, say  $a$ . Then  $\emptyset$  and  $\{a\}$  are the only subsets of  $S$ . Thus, if  $r = 0$ , then  $\emptyset$  is the only subset with 0 element. Hence,

$$1 = \frac{1!}{0!(1-0)!}.$$

Again for  $r = 1$ ,  $\{a\}$  is the only subset with 1 element. So,

$$1 = \frac{1!}{1!(1-1)!}.$$

Hence, we see that the statement is true for  $n = 1$ .

*Inductive hypothesis:* Suppose  $k$  is a positive integer. Assume that  $P(k)$  holds for any set with  $k$  elements.

*Inductive step:* Let  $S$  be a set with  $k + 1$ ,  $k \geq 1$ , elements. Let us write  $S = \{a_1, a_2, \dots, a_k, a_{k+1}\}$ . We now determine the number of subsets of  $S$  containing exactly  $r$  elements where  $0 \leq r \leq k + 1$ .

If  $r = 0$ , then the empty set,  $\emptyset$ , is the only subset with zero elements. If  $r = k + 1$ , then the set  $S$  is the only subset with  $k + 1$  elements. In both of these cases,  $P(k + 1)$  holds because

$$1 = \frac{(k+1)!}{0!((k+1)-0)!} \quad \text{and} \quad 1 = \frac{(k+1)!}{(k+1)!(k+1-k-1)!}.$$

Now, let  $A$  be any subset with exactly  $r$  elements where  $0 < r < k + 1$ . There are two cases to be considered.

**Case 1:**  $a_{k+1} \notin A$ . In this case,  $A$  is a subset of the set  $\{a_1, a_2, \dots, a_k\}$ . By the inductive hypothesis, the number of such subsets is

$$\frac{k!}{r!(k-r)!}.$$

**Case 2:**  $a_{k+1} \in A$ . In this case, if we remove  $a_{k+1}$  from  $A$ , then  $A \setminus \{a_{k+1}\}$  is a subset of  $\{a_1, a_2, \dots, a_k\}$  and the number of elements in  $A \setminus \{a_{k+1}\}$  is  $r-1$ . By the inductive hypothesis, the number of such subsets is

$$\frac{k!}{(r-1)!(k-(r-1))!}.$$

Now from Case 1 and Case 2, we find that the total number of subsets  $A$  of  $S$  with  $r$  elements is

$$\begin{aligned} \frac{k!}{r!(k-r)!} + \frac{k!}{(r-1)!(k-r+1)!} &= \frac{k!(k-r+1) + k!r}{r!(k-r+1)!} \\ &= \frac{(k+1)!}{r!(k+1-r)!}. \end{aligned}$$

Hence,  $P(k+1)$  is true. The result now follows by induction.

**Exercise 4:** Let  $X = \{0, 1, 2, 3, 4\}$ . Find the number of subsets of  $X$  that contain three elements of  $X$ .

**Solution:** The number of subsets of  $X$  that contain three elements of  $X$  is the number of 3-combinations of  $X$ . This number is  $C(5, 3) = \frac{5!}{3!(5-3)!} = \frac{5!}{3!2!} = \frac{5 \cdot 4}{2} = 10$ .

**Exercise 5:** A committee of six is to be made from four students and eight teachers. In how many ways can this be done?

- If the committee contains exactly three students?
- If the committee contains at least three students?

**Solution:**

- We are to select a committee of six from four students and eight teachers such that the committee contains exactly three students. Hence, the other three members are the teachers.

Now, we can select three students out of four in  $C(4, 3)$  ways. For each of these selections, the remaining three members are selected from eight teachers and this can be done in  $C(8, 3)$  ways. Hence, the number of ways a committee of six from four students and eight teachers such that the committee contains exactly three students is

$$\begin{aligned} C(4, 3) \cdot C(8, 3) &= \frac{4!}{3!(4-3)!} \cdot \frac{8!}{3!(8-3)!} \\ &= \frac{4!}{3!} \cdot \frac{8!}{3!(5)!} = 4 \cdot \frac{8 \cdot 7 \cdot 6}{3 \cdot 2} = 224. \end{aligned}$$

- We have to consider two cases.

**Case 1:** 3 students and 3 teachers.

**Case 2:** 4 students and 2 teachers.

In Case 1, from part (a) we find that the number of ways the committee can be formed is 224.

In Case 2, proceeding as in part (a), the number of ways the committee can be formed is

$$\begin{aligned} C(4, 4) \cdot C(8, 2) &= \frac{4!}{4!(4-4)!} \cdot \frac{8!}{2!(8-2)!} \\ &= \frac{4!}{4!} \cdot \frac{8!}{2!(6)!} = \frac{8 \cdot 7}{2} = 28 \end{aligned}$$

Now combining parts (a) and (b) (using the addition principle), a committee of six from four students and eight teachers such that the committee contains at least three students is  $224 + 28 = 252$ .

**Exercise 6:** A student is required to answer 7 out of 12 questions, which are divided into two groups, each containing 6 questions. The student is not permitted to answer more than 5 questions from either group. In how many different ways can the student choose the 7 questions?

**Solution:** The student can choose 7 questions satisfying the given restrictions in the following ways:

| Number of questions from group A | Number of questions from group B |
|----------------------------------|----------------------------------|
| 5                                | 2                                |
| 4                                | 3                                |
| 3                                | 4                                |
| 2                                | 5                                |

Hence, the total number of ways in which the student can choose the 7 questions is  $C(6, 5) \cdot C(6, 2) + C(6, 4) \cdot C(6, 3) + C(6, 3) \cdot C(6, 4) + C(6, 2) \cdot C(6, 5) = 90 + 300 + 300 + 90 = 780$ .

**Exercise 7:** Seema has six friends. In how many ways can she invite one or more friends to a dinner party?

**Solution:** Seema may invite one friend out of six, or two out of six, or three out of six, or four out of six, or five out of six, or six out of six. Hence, the number of ways Seema can invite her friends is

$$\begin{aligned} &C(6, 1) + C(6, 2) + C(6, 3) + C(6, 4) + C(6, 5) + C(6, 6) \\ &= \frac{6!}{1!(6-1)!} + \frac{6!}{2!(6-2)!} + \frac{6!}{3!(6-3)!} + \frac{6!}{4!(6-4)!} \\ &\quad + \frac{6!}{5!(6-5)!} + \frac{6!}{6!(6-6)!} \\ &= \frac{6!}{1!5!} + \frac{6!}{2!4!} + \frac{6!}{3!3!} + \frac{6!}{4!2!} + \frac{6!}{5!1!} + \frac{6!}{6!0!} \\ &= 6 + 15 + 20 + 15 + 6 + 1 = 63 \end{aligned}$$

**Another Solution:** Seema may invite one friend out of six, or two out of six, or three out of six, or four out of six, or five out of six, or six out of six. Thus, we need to find the number of nonempty subsets of a set of six elements. Now the number of subsets of a set of six elements is  $2^6 = 64$ . One of these subsets is empty. Therefore, the number of nonempty subsets of a set of six elements is  $64 - 1 = 63$ . Hence, there are 63 ways Seema can invite one or more friends to a dinner party.

## SECTION REVIEW

### Key Terms

combination

 $C(n, r)$  $\binom{n}{r}$  $r$ -combination

### Some Key Results

- Let  $S$  be a set with  $n > 0$  elements. Let  $r$  be an integer such that  $0 \leq r \leq n$ . The number of subsets that contain  $r$  elements of  $S$  is  $\frac{n!}{r!(n-r)!}$ .
- Let  $S$  be a set with  $n > 0$  elements. Let  $r$  be an integer such that  $0 \leq r \leq n$ . Then  $C(n, r) = \frac{n!}{r!(n-r)!}$ .
- Let  $n$  and  $r$  be nonnegative integers such that  $0 \leq r \leq n$ . Then  $C(n, r) = C(n, n - r)$ .

## EXERCISES

- Find  $C(10, 3)$ ,  $C(15, 10)$ ,  $C(6, 0)$ ,  $C(6, 6)$ .
- Find the positive integer  $n$  such that  $C(20, 2n) = C(20, 2n + 4)$ .
- Let  $X = \{2, 3, 4, 5, 6, 7\}$ . Find the number of subsets of  $X$  that contain four elements.
- Let  $X = \{1, 2, 3, 4, 5, 6, 7, 8\}$ . Find the number of subsets of  $X$  that contain four odd integers.
- Find the number of subsets of  $\{A, B, C, D, E\}$  that contain the letters  $A$  and  $B$ .
- How many four-element subsets of  $\{a, b, c, d, e, i, o, u, x\}$  contain no vowels?
- Find the number of committees consisting of four different members from a group of 16 people.
- A box contains 15 apples. How many different selections of 3 apples can be made so as to include a particular apple?
- How many ways can a soccer team of 11 players be selected from 18 players?
- How many different triangles can be formed by joining the vertices of a square?
- In an athletic club, there are 16 male members and 10 female members. How many ways can we form a committee of 7 members subject to the following conditions:
  - there must be 3 males and 4 females.
  - the committee must contain at least 2 females.
  - the committee must contain at least 2 males.
- Find the number of ways we can form a committee of four Republicans and three Democrats from a group of ten Republicans and eight Democrats.
- From a group of seven Democrats and four Republicans a committee of five with at least one Republican is to be formed. In how many ways can this be done?
- A test consists of 12 questions that are divided into three sections. There are 5 questions in the first section, 4 in the second section, and 3 in the third section. A student is required to answer 6 out of these 12 questions. In how many ways can the student answer 6 questions if the student selects 3 from the first section, 2 from the second section, and 1 from the third section?
- An exam consists of 10 questions that are divided into two sections. Each section contains 5 questions. A student is required to answer 6 out of 10 questions. The student is not permitted to answer more than 4 questions from any group. In how many ways can the student select the questions?
- How many different words consisting of four consonants and three vowels can be formed from an alphabet of ten consonants and four vowels? (Repetition of a letter in a word is not allowed.)
- In a group of 20 students 8 students are girls. In how many ways can 12 students be selected so as to include
  - exactly 7 girls
  - at least 7 girls
- A club has 12 members. A president, a vice president, a secretary, and a treasurer are to be chosen from these members. A member cannot serve in more than one position. In how many ways can these officers be chosen?
- A club has two groups of players labeled group A and group B. The number of players in group A is 6 and the number of players in group B is 8. Find the number of ways a football team of 11 players can be formed from the two groups so that the team contains at least 4 players from group A.
- Hamid has seven friends. In how many ways can he invite one or more of them to a dinner?

## 7.5 GENERALIZED PERMUTATIONS AND COMBINATIONS<sup>1</sup>

Consider the word  $w = \text{SUCCESS}$ . This is a word on the alphabet  $\{S, U, C, E\}$ . The length of  $w$  is the number of occurrences of the letters  $S, U, C, E$ . Thus, the length of  $\text{SUCCESS}$  is 7. Notice that in  $w$ ,  $S$  occurs three times,  $C$  occurs two times,  $U$  occurs one time, and  $E$  occurs one time.

Suppose we want to know the number of words of length 7 on the alphabet  $\{S, U, C, E\}$  such that  $S$  occurs three times,  $C$  occurs two times,  $U$  occurs one time, and  $E$  occurs one time.

Now an arbitrary word of length 7 is a sequence  $\{a_i\}, i = 1, 2, \dots, 6, 7$ , where  $a_i \in \{S, U, C, E\}$ . In our case, we are looking for those sequences in which three  $a_i$ 's are  $S$ , two  $a_i$ 's are  $C$ , one  $a_i$  is  $U$ , and one  $a_i$  is  $E$ .

Recall that a sequence of length 7 is a function  $f : \{1, 2, 3, \dots, 6, 7\} \rightarrow \{S, U, C, E\}$  such that  $f(i) = a_i$ . Hence, for three occurrences of  $S$  in a word we choose three distinct integers from the set  $\{1, 2, 3, \dots, 6, 7\}$ . This can be done in  $C(7, 3)$  ways because we are choosing 3-element subsets of a set of seven elements. After these three choices, the number of remaining integers in the set  $\{1, 2, 3, \dots, 6, 7\}$  is four. Hence, for two occurrences of  $C$ , we choose 2-element subsets of a set of four integers. This can be done in  $C(4, 2)$  ways. After these two choices the number of remaining integers is two. For one occurrence of  $U$ , we choose 1-element subsets from a set of two elements. This can be done in  $C(2, 1)$  ways. Finally, for one occurrence of  $E$ , we choose a 1-element subset from a set of one element. This can be done in  $C(1, 1)$ . Hence, by the multiplication principle, the number of words length 7 satisfying the above conditions is:

$$C(7, 3) \cdot C(7 - 3, 2) \cdot C(7 - 3 - 2, 1) \cdot C(7 - 3 - 2 - 1, 1),$$

which equals

$$C(7, 3) \cdot C(4, 2) \cdot C(2, 1) \cdot C(1, 1) = \frac{7!}{3!4!} \cdot \frac{4!}{2!2!} \cdot \frac{2!}{1!1!} \cdot \frac{1!}{1!0!} = \frac{7!}{3!2!1!1!} = \frac{7!}{3!2!}.$$

Notice that to construct words of length 7 with the above requirement, first we chose  $S$ , then  $U$ , and so on. However, one can verify that the number of such words does not depend on the order in which the letters are chosen.

Using the preceding method of selecting letters, we can prove the following theorem.

**Theorem 7.5.1:** Suppose there is a collection of  $n$  objects of  $k$  different types. Assume that objects of the same type cannot be distinguished from each other. Suppose each type contains  $n_i$  objects,  $i = 1, 2, \dots, k$ , ( $n = n_1 + n_2 + \dots + n_k$ ). Then the total number of different arrangements of these  $n$  objects of  $k$  different types taken all at a time is

$$C(n, n_1) \cdot C(n - n_1, n_2) \cdot C(n - n_1 - n_2, n_3) \cdots C(n - n_1 - n_2 - \cdots - n_{k-1}, n_k),$$

which equals

$$\frac{n!}{n_1!n_2!\cdots n_{k-1}!n_k!}.$$

<sup>1</sup>The material covered in this section may be considered optional.

Each of the  $\frac{n!}{n_1!n_2!\dots n_{k-1}!n_k!}$  arrangements of Theorem 7.5.1 is called a **generalized arrangement**, or **generalized permutation**, of the  $n$  objects of  $k$  different types taken all at a time.

**Corollary 7.5.2:** Let  $n$  and  $k$  be integers such that  $0 < n$  and  $0 \leq k \leq n$ . The number of bit strings of length  $n$  that contain exactly  $k$  number of 1's is  $C(n, k)$ .

**Proof:** A bit string of length  $n$  is of the form  $a_1a_2a_3\dots a_n$ , where  $a_i \in \{0, 1\}$ . We want to determine the number of bit strings of length  $n$  that have exactly  $k$  1's. This implies that  $k$  of the  $a_i$ 's are 1 and the  $n - k$  of the  $a_i$ 's are 0. Hence, by Theorem 7.5.1, the number of bit strings of length  $n$  that contain exactly  $k$  number of 1's is

$$C(n, k) \cdot C(n - k, n - k) = C(n, k) \cdot 1 = C(n, k). \blacksquare$$

In the following counting problem, we will use Corollary 7.5.2.

Suppose there are five identical marbles of the same color and four different boxes. We want to put the marbles into the boxes without any conditions. We may put as many marbles out of five as we like in a box; we may even not put any marbles in some box. Suppose for some distribution of the marbles, Box 1 receives  $x_1$  marbles, Box 2 receives  $x_2$  marbles, Box 3 receives  $x_3$  marbles, and Box 4 receives  $x_4$  marbles. Then  $x_1 + x_2 + x_3 + x_4 = 5$ . Hence, to find the number of ways five marbles can be distributed is the same as finding the number of integer solutions of the equation  $x_1 + x_2 + x_3 + x_4 = 5$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0$ .

We describe some of these distributions in the following diagrams. Suppose  $x_1 = 3, x_2 = 1, x_3 = 1, x_4 = 0$ . This implies that Box 1 receives three marbles, Box 2 receives one marble, Box 3 receives one marble, and Box 4 does not receive any marbles. Pictorially, we can represent this solution as follows:

| Box 1 | Box 2 | Box 3 | Box 4 |
|-------|-------|-------|-------|
| 000   | 0     | 0     | x     |

Here, each 0 represents a marble. Because Box 1 received three marbles, three 0's are shown in this box. Similar conventions apply for Box 2 and Box 3. An x in Box 4 means that this box did not receive any marbles.

Next, corresponding to this distribution of marbles, we can construct a bit string with exactly three 1's as follows: First we write a 0 for each marble of Box 1, and then write a 1. Next we write a 0 for each marble (if any), of Box 2, and then write 1. After this we write a 0 for each marble (if any), of Box 3, and then write a 1. Finally, we write a 0 for each marble of Box 4. For the preceding distribution, we obtain the bit string 00010101. In other words, the number of 0's before the first 1 specifies the number of marbles in Box 1; the number of 0's between the first 1 and the second 1 specifies the number of marbles in Box 2; the number of 0's between the second 1 and the third 1 specifies the number of marbles in Box 3; and the number of 0's after the third 1 specifies the number of marbles in Box 4.

Thus, pictorially, the integer solution  $x_1 = 3, x_2 = 1, x_3 = 1, x_4 = 0$  of the equation  $x_1 + x_2 + x_3 + x_4 = 5$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0$ , the distribution of marbles, and the bit string corresponding to the distribution can be shown

as follows:

|                                                      | Box 1 | Box 2 | Box 3 | Box 4 | $\leftrightarrow$ | 00010101 |
|------------------------------------------------------|-------|-------|-------|-------|-------------------|----------|
| $x_1 = 3, x_2 = 1, x_3 = 1, x_4 = 0 \leftrightarrow$ | 000   | 0     | 0     | x     |                   | (7.3)    |

The following diagrams shows various solutions of the given equation, the distribution of marbles, and the corresponding bit string.

|                                                      | Box 1 | Box 2 | Box 3 | Box 4 | $\leftrightarrow$ | 10100001 |
|------------------------------------------------------|-------|-------|-------|-------|-------------------|----------|
| $x_1 = 0, x_2 = 1, x_3 = 4, x_4 = 0 \leftrightarrow$ | x     | 0     | 0000  | x     |                   | (7.4)    |

|                                                      | Box 1 | Box 2 | Box 3 | Box 4 | $\leftrightarrow$ | 10001010 |
|------------------------------------------------------|-------|-------|-------|-------|-------------------|----------|
| $x_1 = 0, x_2 = 3, x_3 = 1, x_4 = 1 \leftrightarrow$ | x     | 000   | 0     | 0     |                   | (7.5)    |

|                                                      | Box 1 | Box 2 | Box 3 | Box 4 | $\leftrightarrow$ | 00000111 |
|------------------------------------------------------|-------|-------|-------|-------|-------------------|----------|
| $x_1 = 5, x_2 = 0, x_3 = 0, x_4 = 0 \leftrightarrow$ | 00000 | x     | x     | x     |                   | (7.6)    |

|                                                      | Box 1 | Box 2 | Box 3 | Box 4 | $\leftrightarrow$ | 00011100 |
|------------------------------------------------------|-------|-------|-------|-------|-------------------|----------|
| $x_1 = 3, x_2 = 0, x_3 = 0, x_4 = 2 \leftrightarrow$ | 000   | x     | x     | 00    |                   | (7.7)    |

|                                                      | Box 1 | Box 2 | Box 3 | Box 4 | $\leftrightarrow$ | 00101010 |
|------------------------------------------------------|-------|-------|-------|-------|-------------------|----------|
| $x_1 = 2, x_2 = 1, x_3 = 1, x_4 = 1 \leftrightarrow$ | 00    | 0     | 0     | 0     |                   | (7.8)    |

In the first case, (7.4), we put no marble in Box 1, one marble in Box 2, four marbles in Box 3, and no marble in Box 4 ( $\leftrightarrow x_1 = 0, x_2 = 1, x_3 = 4, x_4 = 0$ ). In the second case, (7.5), we put no marble in Box 1, three marbles in Box 2, one marble in Box 3, and one marble in Box 4 ( $\leftrightarrow x_1 = 0, x_2 = 3, x_3 = 1, x_4 = 1$ ), and so on.

We find that the first arrangement, (7.3), is the same as the bit string 00010101 of length 8. There are three 0's before the first position of 1, which implies that Box 1 contains three marbles; there is only one 0 between the first position and the second position of 1, which indicates that there is one marble in Box 2; there is only one 0 between the second position and the third position of 1, which indicates that there is one marble in Box 3; and finally, there is no zero after the third 1, which indicates that there are no marbles in Box 4.

For case 2, the bit string of length 8 is 10100001.

For case 3, the bit string of length 8 is 10001010.

For case 4, the bit string of length 8 is 00000111.

For case 5, the bit string of length 8 is 00011100.

For case 6, the bit string of length 8 is 00101010.

Now, corresponding to a bit string of length 8 with three 1's, say 00010101, we put the marbles in the following way:

|                            | Box 1 | Box 2 | Box 3 | Box 4 | $\leftrightarrow$ | $x_1 = 3, x_2 = 1, x_3 = 1, x_4 = 0$ |
|----------------------------|-------|-------|-------|-------|-------------------|--------------------------------------|
| $00010101 \leftrightarrow$ | 000   | 0     | 0     | x     |                   |                                      |

We can establish a one-to-one correspondence between the set of all possible distributions of five marbles into four boxes and the set of bit strings of length 8 that contain exactly three 1's. Also we find that the number of integer solutions of the equation  $x_1 + x_2 + x_3 + x_4 = 5$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0$  is the same as the number of ways five marbles can be distributed into four different boxes without any prior conditions.

Now any bit string of length 8 that contains exactly three 1's also contains five 0's. Thus, by Corollary 7.5.2, the number of bit strings of length 8 that contain three 1's and five 0's is

$$\frac{8!}{3!5!}.$$

Hence, the number of ways to put five identical, same-color marbles into four boxes is  $\frac{8!}{3!5!}$ . This implies that the number of integer solutions of the equation  $x_1 + x_2 + x_3 + x_4 = 5$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0$  is  $\frac{8!}{3!5!} = C(8, 3)$ .

From the above discussions we derive the following theorem.

**Theorem 7.5.3:** The number of integer solutions of the equation

$x_1 + x_2 + x_3 + \dots + x_k = n$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, \dots, x_k \geq 0$ , where  $n > 0$  is  $C(n + k - 1, k - 1)$ .

Suppose now there are four boxes, say Box 1, Box 2, Box 3, and Box 4. Box 1 contains identical red marbles, Box 2 contains identical blue marbles, Box 3 contains identical green marbles, and Box 4 contains identical yellow marbles. We want to select three marbles from these boxes without any restrictions.

For example, we can select three marbles from Box 1; or we can select one from Box 1 and two from Box 4; or one from Box 1, one from Box 2, and one from Box 3; or three from Box 4. Corresponding to these selections, we have

|              | Box 1 | Box 2 | Box 3 | Box 4 |
|--------------|-------|-------|-------|-------|
| Selection 1: | 000   | x     | x     | x     |
| Selection 2: | 0     | x     | x     | 00    |
| Selection 3: | 0     | 0     | 0     | x     |
| Selection 4: | x     | x     | x     | 000   |

Now, corresponding to each of these selections we can construct a bit string of length 6 that contains three 1's. For example, the bit string of length 6 with three 1's corresponding to selection 1 is 000111; the bit string of length 6 with three 1's corresponding to selection 2 is 011100; the bit string of length 6 with three 1's corresponding to selection 3 is 010101; and the bit string of length 6 with three 1's corresponding to selection 4 is 111000.

Hence, the number of selections of three marbles with repetitions allowed from four categories is the same as the number of bit strings of length 6 that

contain exactly three 1's, which equals

$$C(6, 3) = C(4 - 1 + 3, 3).$$

The preceding problem is the same as the distribution of three marbles into four boxes. Therefore, this problem can be considered as the problem of finding the number of integer solutions of the equation  $x_1 + x_2 + x_3 + x_4 = 3$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0$ . Here this number is  $C(6, 3)$ .

A selection of three marbles from four categories is called a *3-combination of 4 objects with repetitions allowed*.

The following theorem gives the number of  $r$ -combinations of  $n$  objects with repetitions allowed.

**Theorem 7.5.4:** Let  $n$  and  $r$  be two positive integers such that  $r \leq n$ . Then the number of  $r$ -combinations of  $n$  objects with repetitions allowed is  $C(n - 1 + r, r)$ .

We now consider the following example.

### EXAMPLE 7.5.5

Suppose a candy shop has six different varieties of candy. Chelsea wants to buy four candies. We want to know the number of ways Chelsea can do this.

Because there are six different varieties, we consider six boxes.

| B1 | B2 | B3 | B4 | B5 | B6 |
|----|----|----|----|----|----|
|    |    |    |    |    |    |

Suppose Chelsea chooses two candies from the first variety and one each from the fourth and sixth varieties. Corresponding to this selection, we can make the following distribution of candies into the six boxes.

| B1 | B2 | B3 | B4 | B5 | B6 |
|----|----|----|----|----|----|
| 00 | ×  | ×  | 0  | ×  | 0  |

Corresponding to this distribution, we construct the bit string 001110110, of length 9, which contains exactly five (6 - 1) 1's.

Suppose she chooses the candies as follows: two from the second variety and one each from the fifth and sixth varieties. Corresponding to this selection, we can make the following distribution of candies into the six boxes.

| B1 | B2 | B3 | B4 | B5 | B6 |
|----|----|----|----|----|----|
| ×  | 00 | ×  | ×  | 0  | 0  |

The bit string of length 9 corresponding to this distribution with exactly five 1's is 100111010.

It follows that the number of ways to select four candies from six varieties of candies is the number of bit strings of length 9 that contains exactly five 1's, which is  $C(9, 5) = \frac{9!}{4!5!} = 126$ .

Notice that this problem is the same as the problem of finding the number of integer solutions of the equation  $x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 4$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0, x_6 \geq 0$ .


**WORKED-OUT EXERCISES**

**Exercise 1:** Find the number of words of length 11 on the alphabet  $\{E, N, G, I, R\}$  such that  $E$  occurs three times,  $N$  occurs three times,  $I$  occurs two times,  $G$  occurs two times and  $R$  occurs one time.

**Solution:** Now an arbitrary word of length 11 is a sequence  $\{a_i\}$ ,  $i = 1, 2, \dots, 10, 11$ , where  $a_i \in \{E, N, G, I, R\}$ . In our case, we are looking for those sequences in which three  $a_i$ 's are  $E$ , two  $a_i$ 's are  $N$ , two  $a_i$ 's are  $G$ , two  $a_i$ 's are  $I$ , and one  $a_i$  is  $R$ . Recall that a sequence of length 11 is a function  $f : \{1, 2, 3, \dots, 10, 11\} \rightarrow \{E, N, G, I, R\}$  such that  $f(i) = a_i$ . Hence, for three occurrences of  $E$  in the word we choose three distinct integers from the set  $\{1, 2, 3, \dots, 10, 11\}$ . This can be done in  $C(11, 3)$  ways because we are choosing 3-element subsets of a set of 11 elements. After making these choices, the number of remaining integers in the set  $\{1, 2, 3, \dots, 10, 11\}$  is eight. Hence, for three occurrences of  $N$  we choose 3-element subsets from the set of remaining eight integers, and this can be done in  $C(8, 3)$  ways. After the preceding two types of choices, the number of remaining integers is five. For two occurrences of  $I$ , we choose 2-element subsets from the set of remaining five integers, which can be done in  $C(5, 2)$  ways. After these choices, the number of remaining integers is three.

For two occurrences of  $G$ , we choose 2-element subsets from the set of remaining three integers, which can be done in  $C(3, 2)$  ways.

Finally one integer is left, and for one occurrence of  $R$ , we choose 1-element subsets of a set with one element. This can be done in  $C(1, 1)$  ways. Hence, by the multiplication principle, the number of sequences of length 11 satisfying the above conditions is  $C(11, 3) \cdot C(8, 3) \cdot C(5, 2) \cdot C(3, 2) \cdot C(1, 1)$ , which equals

$$\frac{11!}{8!3!} \cdot \frac{8!}{5!3!} \cdot \frac{5!}{3!2!} \cdot \frac{3!}{2!1!} \cdot \frac{1!}{1!} = \frac{11!}{3!3!2!2!1!}$$

**Exercise 2:** A sailing club has 10 white flags, 7 red flags, and 3 green flags. In how many ways can these flags be displayed in a row?

**Solution:** The total number of flags is 20. Out of these 20 flags 10 are white, 7 are red, and 3 are green. Hence, the number of ordered arrangements of these flags in a row is  $\frac{20!}{10!7!3!}$ .

**Exercise 3:** Find the number of bit strings of length 10 that have exactly three 1's.

**Solution:** A bit string of length 10 is of the form  $a_1 a_2 a_3 \dots a_9 a_{10}$ , where  $a_i \in \{0, 1\}$ . We want to determine the number of bit strings of length 10 that have exactly three 1's. This implies that three of the  $a_i$ 's are 1 and the seven of the  $a_i$ 's are 0. Hence, the number of bit strings of length 10 that have exactly three 1's is

$$\frac{10!}{7!3!} = \frac{10 \cdot 9 \cdot 8}{3 \cdot 2} = 120.$$

**Exercise 4:** Find the number of integer solutions of the equation  $x_1 + x_2 + x_3 + x_4 = 12$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0$ .

**Solution:** The problem is the same as finding the number of ways 12 marbles can be distributed in four boxes  $x_1, x_2, x_3, x_4$ .

| Box $x_1$ | Box $x_2$ | Box $x_3$ | Box $x_4$ |
|-----------|-----------|-----------|-----------|
|           |           |           |           |

For example, corresponding to the solution  $x_1 = 1, x_2 = 1, x_3 = 5, x_4 = 5$ , the distribution of marbles is the following:

| Box $x_1$ | Box $x_2$ | Box $x_3$ | Box $x_4$ |
|-----------|-----------|-----------|-----------|
| 0         | 0         | 00000     | 00000     |

which is equivalent to the string 010100000100000.

Let us consider another solution:  $x_1 = 1, x_2 = 0, x_3 = 11, x_4 = 0$ . The distribution of marbles corresponding to this solution is the following:

| Box $x_1$ | Box $x_2$ | Box $x_3$    | Box $x_4$ |
|-----------|-----------|--------------|-----------|
| 0         | x         | 000000000000 | x         |

which is equivalent to the string 011000000000000.

Hence, the given problem is equivalent to finding the number of bit strings of length 15 that have exactly three 1's. The required number is:

$$C(15, 3) = \frac{15!}{12!3!} = \frac{15 \cdot 14 \cdot 13}{3 \cdot 2} = 13 \cdot 7 \cdot 5 = 495.$$

**Exercise 5:** Find the number of integer solutions of the equation  $x_1 + x_2 + x_3 = 12$  such that  $x_1 \geq 1, x_2 \geq 1, x_3 \geq 1$ .

**Solution:** The problem is the same as the distribution of 12 marbles in three boxes  $x_1, x_2, x_3$  such that each box contains at least one marble.

| Box $x_1$ | Box $x_2$ | Box $x_3$ |
|-----------|-----------|-----------|
| 0         | 0         | 0         |

For example, corresponding to the solution  $x_1 = 2, x_2 = 3, x_3 = 7$ , the distribution of marbles is the following:

| Box $x_1$ | Box $x_2$ | Box $x_3$ |
|-----------|-----------|-----------|
| 0         | 00        | 000000    |
| 0         | 0         | 0         |

which is equivalent to the string 0010001000000.

Let us consider another solution:  $x_1 = 1, x_2 = 5, x_3 = 6$ . The distribution of marbles is the following:

| Box $x_1$ | Box $x_2$ | Box $x_3$ |
|-----------|-----------|-----------|
| x         | 0000      | 00000     |
| 0         | 0         | 0         |

which is equivalent to the string 01000001000000.

Hence, under the given conditions we will distribute only 9 marbles in three boxes. So we have to find the number of bit strings of length 11 that have exactly two 1's. The required number is

$$C(11, 2) = \frac{11!}{9!2!} = \frac{11 \cdot 10}{2} = 55.$$

**Exercise 6:** Find the number of integer solutions of the equation  $x_1 + x_2 + x_3 = 12$  such that  $x_1 \geq 1, x_2 \geq 1, x_3 \geq 3$ .

**Solution:** In this problem, each of boxes  $x_1$  and  $x_2$  contains at least one marble and box  $x_3$  contains at least three marbles.

| Box $x_1$ | Box $x_2$ | Box $x_3$ |
|-----------|-----------|-----------|
| 0         | 0         | 000       |

For example, corresponding to the solution  $x_1 = 2, x_2 = 3, x_3 = 7$ , the distribution of marbles is the following:

| Box $x_1$ | Box $x_2$ | Box $x_3$ |
|-----------|-----------|-----------|
| 0         | 00        | 0000      |
| 0         | 0         | 000       |

Hence, we will distribute only seven marbles. So we have to find the number of bit strings of length 9 that have exactly

two 1's. The required number is

$$C(9, 2) = \frac{9!}{7!2!} = \frac{9 \cdot 8}{2} = 36.$$

**Exercise 7:** Suppose there are three boxes containing identical white, red, and blue balls. Each box contains at least 12 balls.

- a. In how many ways can we select 12 balls?
- b. In how many ways can we select 12 balls such that we must have at least one ball of each color?

**Solution:** Let  $x_1$  denote the number of white balls,  $x_2$  denote the number of red balls, and  $x_3$  denote the number of blue balls.

- a. The number of ways we can select 12 balls is the same as the number of integral solutions of  $x_1 + x_2 + x_3 = 12$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$ .

As in Worked-Out Exercise 4 (of this section), we find that the number of solutions is  $C(14, 2)$ .

- b. The number of ways we can select 12 balls is the same as the number of integral solutions of  $x_1 + x_2 + x_3 = 12$  such that  $x_1 \geq 1, x_2 \geq 1, x_3 \geq 1$ .

As in Worked-Out Exercise 4, we find that the number of solutions is  $C(11, 2) = \frac{11!}{9!2!} = \frac{11 \cdot 10}{2} = 55$ .

**Exercise 8:** A candy shop carries 10 types of chocolates. Shelly wants to buy 4 chocolates. In how many different ways can she do this?

**Solution:** There are 10 types of chocolates. Suppose Shelly chooses  $x_i \geq 0, i = 1, \dots, 10$  chocolates from the  $i$ th variety. Hence, this problem is the same as the problem of finding the number of integer solutions of the equation  $x_1 + x_2 + x_3 + x_4 + \dots + x_9 + x_{10} = 4$  such that  $x_i \geq 0, i = 1, \dots, 10$ . This number is  $C(13, 9) = \frac{13!}{9!4!} = \frac{13 \cdot 12 \cdot 11 \cdot 10}{4 \cdot 3 \cdot 2} = 715$ .

## SECTION REVIEW

### Key Terms

generalized arrangement

generalized permutation

### Some Key Results

1. Suppose there is a collection of  $n$  objects of  $k$  different types. Assume that objects of the same type cannot be distinguished from each other. Suppose each type contains  $n_i$  objects,  $i = 1, 2, \dots, k$ , ( $n = n_1 + n_2 + \dots + n_k$ ). Then the total number of different arrangements of these  $n$  objects of  $k$  different types taken all at a time is:

$$C(n, n_1) \cdot C(n - n_1, n_2) \cdot C(n - n_1 - n_2, n_3) \cdots C(n - n_1 - n_2 - \dots - n_{k-1}, n_k),$$

which equals

$$\frac{n!}{n_1!n_2!\cdots n_{k-1}!n_k!}.$$

2. Let  $n$  and  $k$  be integers such that  $0 < n$  and  $0 \leq k \leq n$ . The number of bit strings of length  $n$  that contain exactly  $k$  number of 1's is  $C(n, k)$ .
3. The number of integer solutions of the equation  $x_1 + x_2 + x_3 + \cdots + x_k = n$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, \dots, x_k \geq 0$ , where  $n > 0$  is  $C(n+k-1, k-1)$ .
4. Let  $n$  and  $r$  be two positive integers such that the number of  $r$ -combinations of  $n$  objects with repetitions allowed is  $C(n-1+r, r)$ .

## EXERCISES

---

1. Find the number of different ways the letters of the word *SUCCESSIVE* can be arranged.
2. Find the number of different ways the letters of the word *BALLOON* can be arranged.
3. Find the number of different ways the letters of the word *MATHEMATICS* can be arranged.
4. A library has seven copies of one book, four copies of another book, and single copies of four other books. In how many ways can all these books be arranged in a row?
5. In how many ways can 15 houses be painted so that 5 houses are red, 3 houses are green, and 7 houses are blue?
6. Find the number of bit strings of length 8 that have exactly three 0's.
7. Find the number of bit strings of length 10 that have exactly four 1's.
8. How many bit strings of length 15 contain exactly five 1's?
9. Find the number of integer solutions of the equation  $x_1 + x_2 + x_3 + x_4 = 20$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0$ .
10. Find the number of integer solutions of the equation  $x_1 + x_2 + x_3 = 19$  such that  $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$ .
11. Find the number of integer solutions of the equation  $x_1 + x_2 + x_3 = 15$  such that  $x_1 \geq 1, x_2 \geq 1, x_3 \geq 1$ .
12. Find the number of integer solutions of the equation  $x_1 + x_2 + x_3 = 16$  such that  $x_1 \geq 2, x_2 \geq 1, x_3 \geq 1$ .
13. Find the number of integer solutions of the equation  $x_1 + x_2 + x_3 = 16$  such that  $x_1 \geq 2, x_2 \geq 1, x_3 = 1$ .
14. Find the number of integer solutions of the equation  $x_1 + x_2 + x_3 = 16$  such that  $0 \leq x_1 \leq 3, 0 \leq x_2 \leq 3, 0 \leq x_3 \leq 4$ .
15. A bakery makes five different varieties of donuts. Carl wants to buy ten donuts. How many different ways can he do it?
16. A bakery makes ten different varieties of donuts. Shawn wants to buy five donuts. How many different ways can he do it?
17. Find the number of different collections of eight coins that can be made from identical pennies, identical nickels, identical dimes, and identical quarters.
18. Suppose there are three boxes of identical white, red, and blue balls. Each box contains at least 15 balls.
  - a. In how many ways can we select 15 balls?
  - b. In how many ways can we select 15 balls such that we must have at least one ball from each color?
  - c. In how many ways can we select 15 balls such that we must have at least one red ball?
  - d. In how many ways can we select 15 balls such that we must have exactly one white ball?

## 7.6 BINOMIAL COEFFICIENTS

---

In Section 7.4, we came across the terms  $C(n, r)$ —the number of ways to select  $r$  elements from a set of  $n$  elements. We derived the following formula for  $C(n, r)$  and used it to answer certain counting problems.

$$C(n, r) = \frac{n!}{r!(n-r)!},$$

where  $0 \leq r \leq n$ .

In this section, we further discuss the basic properties of  $C(n, r)$  as well as describe computer algorithms to compute it.

Interestingly, the term  $C(n, r)$  also appears in the expansion of the expression  $(x + y)^n$ . Below we will obtain a formula for  $(x + y)^n$  in terms of  $C(n, r)$ .

We call the expression  $x + y$  a **binomial expression** as it is the sum of two terms. We call the expression  $(x + y)^n$  a *binomial expression of order n*. Before we derive a formula for  $(x + y)^n$ , let us evaluate the binomial expressions  $(x + y)^2$  and  $(x + y)^3$ .

$$\begin{aligned}(x + y)^2 &= (x + y)(x + y) \\&= x(x + y) + y(x + y) \\&= xx + xy + yx + yy \\&= xx + xy + xy + yy \\&= xx + 2xy + yy \\&= C(2, 0)x^2 + C(2, 1)xy + C(2, 2)y^2,\end{aligned}$$

because  $C(2, 0) = 1$ ,  $C(2, 1) = 2$ , and  $C(2, 2) = 1$ .

$$\begin{aligned}(x + y)^3 &= (x + y)(x + y)(x + y) \\&= (x + y)(xx + xy + xy + yy) \\&= x(xx + xy + xy + yy) + y(xx + xy + xy + yy) \\&= xxx + xxy + xxy + xyy + yxx + yxy + yxy + yyy \\&= xxx + xxy + xxy + xyy + xxy + xyy + xyy + yyy \\&= x^3 + x^2y + x^2y + xy^2 + x^2y + xy^2 + xy^2 + y^3 \\&= x^3 + x^2y + x^2y + x^2y + xy^2 + xy^2 + xy^2 + y^3 \\&= x^3 + 3x^2y + 3xy^2 + y^3.\end{aligned}$$

Let us observe the following: To compute  $(x + y)^3$ , we multiply the factors  $(x + y)$ ,  $(x + y)$ , and  $(x + y)$ . The expansion of  $(x + y)^3$  consists of the terms  $x^3$ ,  $x^2y$ ,  $xy^2$ , and  $y^3$ . This is due to the fact that each term in the expansion of  $(x + y)^3$  is a product of terms, one term from each of the factors  $(x + y)$ ,  $(x + y)$ , and  $(x + y)$ . For example, to obtain the term  $x^2y$ , we can choose  $x$  from the first and second factors and  $y$  from the third factor. Next we determine how many times each of the terms  $x^3$ ,  $x^2y$ ,  $xy^2$ , and  $y^3$  appear in the expansion.

Now to form the term  $x^3$  we must take  $x$  from each of the three factors, and this can be done in only one way. To form the term  $x^2y$  we must choose  $y$  from only one factor. Now there are three factors, and we need to choose  $y$  from only one of the factors. This can be done in  $C(3, 1)$  ways; i.e., number of ways to select one item from a set of three items. After choosing  $y$  we can select  $x$  from the remaining two factors, which can be done in one way as we are selecting two items from a set of two items. Thus, the number of ways the term  $x^2y$  can be formed is  $C(3, 1)$ . Similarly, the number of ways the term  $xy^2$  can be formed is  $C(3, 2)$ , and the number of ways the term can be formed is  $1 = C(3, 3)$ . Also note that  $C(3, 0) = 1$ . Using these combinatorial arguments,

$$\begin{aligned}(x + y)^3 &= C(3, 0)x^3 + C(3, 1)x^2y + C(3, 2)xy^2 + C(3, 3)y^3. \\&= x^3 + 3x^2y + 3xy^2 + y^3.\end{aligned}$$

In general, we have the following binomial theorem.

**Theorem 7.6.1: Binomial Theorem.** Let  $n$  be an integer such that  $n \geq 1$ . If  $x$  and  $y$  are variables, then

$$\begin{aligned}(x+y)^n &= \sum_{i=0}^n C(n, i)x^{n-i}y^i \\ &= C(n, 0)x^n + C(n, 1)x^{n-1}y + C(n, 2)x^{n-2}y^2 \\ &\quad + \cdots + C(n, n-1)xy^{n-1} + C(n, n)y^n.\end{aligned}$$

**Proof:** We have

$$(x+y)^n = \underbrace{(x+y)(x+y) \cdots (x+y)}_{n \text{ times}}.$$

That is,  $(x+y)^n$  is a product of the  $n$  factors  $x+y$ . Each term in the expansion of  $(x+y)^n$  is a product of terms, one term from each of these factors. Now each factor  $x+y$  has two terms. Therefore, from a factor we must choose  $x$  or  $y$ . Let  $0 \leq i \leq n$ . Suppose we choose  $y$ 's from  $i$  factors, which can be done in  $C(n, i)$  ways because we are choosing  $y$  from  $i$  factors from a set of  $n$  factors. Then we must choose  $x$  from the remaining  $n-i$  factors, which can be done in one way because we are choosing  $x$  from the set of remaining  $n-i$  factors. Moreover, if we choose  $y$  from  $i$  factors and  $x$  from  $n-i$  factors and then multiply them, we obtain the term  $x^{n-i}y^i$ . It now follows that the number of ways the term  $x^{n-i}y^i$  can be formed is  $C(n, i) \cdot 1 = C(n, i)$  ways. In other words, in the expansion of  $(x+y)^n$  the term  $x^{n-i}y^i$  would appear  $C(n, i)$  times. Thus, the coefficient of  $x^{n-i}y^i$  in the expansion of  $(x+y)^n$  is  $C(n, i)$ . It now follows that

$$\begin{aligned}(x+y)^n &= C(n, 0)x^n + C(n, 1)x^{n-1}y + C(n, 2)x^{n-2}y^2 \\ &\quad + \cdots + C(n, n-1)xy^{n-1} + C(n, n)y^n \\ &= \sum_{i=0}^n C(n, i)x^{n-i}y^i. \blacksquare\end{aligned}$$

**REMARK 7.6.2** ▶ The preceding proof of the binomial theorem uses combinatorial arguments. We can also prove the binomial theorem using induction (see Exercise 20, page 469).

**DEFINITION 7.6.3** ▶ The term  $C(n, i)$ , is also, called the **binomial coefficient**.

Using the binomial theorem, Theorem 7.6.1, we can obtain many interesting results, some of which are given next as corollaries.

**Corollary 7.6.4:** Let  $n$  be an integer such that  $n \geq 0$ . Then

$$2^n = \sum_{i=0}^n C(n, i).$$

**Proof:** Let  $x = 1$  and  $y = 1$ . Then by Theorem 7.6.1,

$$\begin{aligned} 2^n &= (1+1)^n \\ &= \sum_{i=0}^n C(n, i) 1^{n-i} 1^i \\ &= \sum_{i=0}^n C(n, i). \quad \blacksquare \end{aligned}$$

**Corollary 7.6.5:** Let  $n$  be an integer such that  $n \geq 0$ . Then

$$\sum_{i=0}^n (-1)^i C(n, i) = 0.$$

**Proof:** Let  $x = 1$  and  $y = -1$ . Then by Theorem 7.6.1,

$$\begin{aligned} 0^n &= (1-1)^n \\ &= \sum_{i=0}^n C(n, i) 1^{n-i} (-1)^i \\ &= \sum_{i=0}^n (-1)^i C(n, i). \quad \blacksquare \end{aligned}$$

**Corollary 7.6.6:** Let  $n$  be an integer such that  $n \geq 0$ . Then

$$3^n = \sum_{i=0}^n C(n, i) 2^i.$$

**Proof:** Let  $x = 1$  and  $y = 2$ . Then by Theorem 7.6.1,

$$\begin{aligned} 3^n &= (1+2)^n \\ &= \sum_{i=0}^n C(n, i) 1^{n-i} 2^i \\ &= \sum_{i=0}^n C(n, i) 2^i. \quad \blacksquare \end{aligned}$$

In the next theorem, **Pascal's identity**, we describe an interesting and important property of  $C(n, r)$ , which will lead to the construction of algorithms to compute it.

**Theorem 7.6.7: Pascal's Identity.** Let  $n$  and  $r$  be integers such that  $1 \leq r \leq n$ . Then

$$C(n+1, r) = C(n, r-1) + C(n, r).$$

**Proof:** Now  $C(n + 1, r)$  is the number of ways of selecting  $r$  elements from a set of  $n + 1$  elements. Let  $S$  be a set of  $n + 1$  elements. Then the number of  $r$  element subsets of  $S$  is  $C(n + 1, r)$ .

Next we determine the number of  $r$  element subsets of  $S$  using a different technique.

Let  $x \in S$ . Let  $T = S \setminus \{x\}$ . Then  $T$  is set with  $n$  elements and  $T \subseteq S$ . Consider the following sets:

$$P = \{C \mid C \text{ is a subset of } T, \text{ and number of elements in } C \text{ is } r\}$$

and

$$Q = \{D \mid D \text{ is a subset of } T, \text{ and number of elements in } D \text{ is } r - 1\}.$$

That is,  $P$  is the set of  $r$  element subsets of  $T$ , and  $Q$  is the set of  $r - 1$  element subsets of  $T$ . Because  $T$  has  $n$  elements, the number of  $r$  element subsets of  $T$  is  $C(n, r)$ , and the number of  $r - 1$  element subsets of  $T$  is  $C(n, r - 1)$ . Hence,

$$|P| = C(n, r) \quad \text{and} \quad |Q| = C(n, r - 1).$$

Moreover, observe that  $P \cap Q = \emptyset$ . Hence, by the addition principle,

$$|P \cup Q| = |P| + |Q| = C(n, r) + C(n, r - 1).$$

Let  $A$  be a subset of  $S$  such that  $A$  has  $r$  elements. We have two cases as follows:

**Case 1:** Suppose that  $x \notin A$ . Then  $A$  is an  $r$  element subset of  $T$ , so  $A \in P$ . Because every subset of  $T$  is a subset of  $S$ , it follows that the number of such  $A$  is  $C(n, r)$ .

**Case 2:** Suppose that  $x \in A$ . Let  $B = A \setminus \{x\}$ . Then  $B$  is an  $r - 1$  element subset of  $T$ , so  $B \in Q$ . On the other hand, let  $D$  be an  $r - 1$  element subset of  $T$ , that is,  $D \in Q$ . Then  $D \cup \{x\}$  is an  $r$  element subset of  $S$ . It now follows that if  $x \in A$ , then  $A$  can be constructed from  $T$  by taking an  $r - 1$  element subset of  $T$  and then including  $x$  in that subset. Therefore, the number of such  $A$  is  $C(n, r - 1)$ .

It now follows from Cases 1 and 2 that the number of  $r$  element subsets of  $S$  is:

$$C(n, r) + C(n, r - 1).$$

Hence,

$$C(n + 1, r) = C(n, r - 1) + C(n, r). \blacksquare$$

#### EXAMPLE 7.6.8

- (i)  $C(3, 2) = C(2, 1) + C(2, 2)$ .
- (ii)  $C(7, 4) = C(6, 3) + C(6, 4)$ .
- (iii)  $C(10, 6) = C(9, 5) + C(9, 6)$ .

We know that  $C(n + 1, r)$  is a binomial coefficient appearing in the expansion of  $(x + y)^{n+1}$ , and  $C(n, r - 1)$  and  $C(n, r)$  are binomial coefficients appearing in the expansion of  $(x + y)^n$ .

Blaise Pascal, a French philosopher and mathematician, discovered he could obtain the number  $C(n, r)$  by constructing a triangular array. This array is usually called **Pascal's triangle**. The construction of the array is as follows.

The row 0, i.e., the first row of the triangle, contains the single entry 1. The row 1, i.e., the second row, contains a pair of entries each equal to 1. The following shows the first two rows of Pascal's triangle.

$$\begin{array}{c} 1 \\ 1 \quad 1 \end{array}$$

Next we calculate the  $n$ th row of the triangle from the preceding row by the following rules.

1. The first ( $k = 0$ ) and the last ( $k = n$ ) entries are both equal to 1.
2. For  $1 \leq k \leq n - 1$ , the  $k$ th entry in the  $n$ th row is the sum of the  $(k - 1)$ th and  $k$ th entries of the  $(n - 1)$ th row.

The first five rows of Pascal's triangle are shown in the following diagram.

$$\begin{array}{ccccccc} & & & 1 & & & \\ & & & \swarrow + \searrow & & & 1 \\ & & 1 & & 1 & & \\ & & & \swarrow + \searrow & & \swarrow + \searrow & \\ 1 & & & 1+1=2 & & 1+1=2 & 1 \\ & & & \swarrow + \searrow & & \swarrow + \searrow & \\ & 1 & & 1+2=3 & & 2+1=3 & \\ & & & \swarrow + \searrow & & \swarrow + \searrow & \\ & & 1 & & 1+3=4 & & 3+3=6 & \\ & & & & & & & 3+1=4 & \\ & & & & & & & & 1 \end{array}$$

Now we construct a triangular array for the numbers  $C(n, r)$  such that  $n \geq 0$  and  $0 \leq r \leq n$  according to the following rules.

1. Rows are labeled  $n = 0, n = 1, n = 2, \dots$  and positions within the  $n$ th row are labeled  $k = 0, k = 1, k = 2, \dots, k = n$ .
2. The number  $C(n, k)$  is placed in the  $k$ th position of the  $n$ th row.

We obtain the following triangular array of binomial coefficients. (In the following table we write  $C(n, r)$  as  $\binom{n}{r}$ .)



**Blaise Pascal**  
(1623–1662)

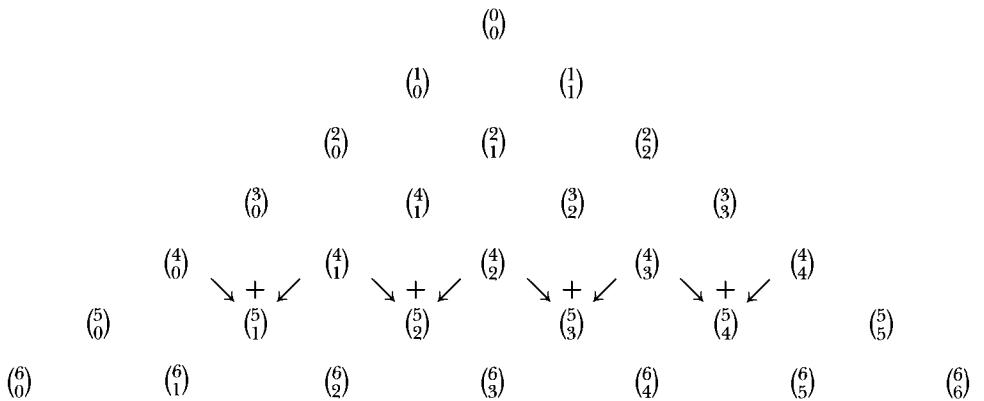
Pascal's mother died when he was just three. Etienne Pascal, his father, raised Pascal as well as his three sisters. Pascal grew up in Paris and was educated by his father, who believed mathematics should not be taught until the age of 15. Pascal, like other children who are denied something, began studying geometry at the

### Historical Notes

age of 12, on his own. When his father learned of this, he acquiesced, and Pascal began a more formal study of mathematics. This study included attending Mersenne's meetings with his father. Pascal even offered a paper at one meeting, which included his mystic hexagon.

Pascal would go on to publish papers on conic sections, projective geometry, and binomial coefficients. The publication of the latter led to Newton's binomial theorem. In addition to his

work in mathematics, Pascal created a calculator, which he hoped would help his father in his work as a tax collector. Pascal also theorized about physical principles, including the laws of pressure. Additionally, Pascal was deeply religious, and he wrote theological and philosophical works about the human condition and faith in God. His most notable work is called *Pensées*.



If we evaluate the numbers  $\binom{n}{r}$ , then we find that we obtain Pascal's triangle:

|   |   |    |   |    |   |    |   |
|---|---|----|---|----|---|----|---|
|   |   |    | 1 |    |   |    |   |
|   |   |    | 1 |    | 1 |    |   |
|   |   | 1  |   | 2  |   | 1  |   |
|   |   | 1  |   | 3  |   | 3  |   |
|   | 1 |    | 4 |    | 6 |    | 4 |
| 1 |   | 5  |   | 10 |   | 10 |   |
| 1 |   | 6  |   | 15 |   | 20 |   |
| 1 |   | 15 |   | 20 |   | 15 |   |
| 1 |   | 6  |   | 4  |   | 5  |   |
| 1 |   | 1  |   | 1  |   | 1  |   |

Pascal's triangle has many interesting properties. For example, consider the 3rd row, i.e., row 2.

$$1 \quad 2 \quad 1$$

We find that this row gives the number  $121 = 11^2$ . Now consider the 4th row, i.e., row 3.

$$1 \quad 3 \quad 3 \quad 1$$

This gives the number  $1331 = 11^3$ . Consider the 5th row, i.e., row 4.

$$1 \quad 4 \quad 6 \quad 4 \quad 1$$

This gives the number  $14641 = 11^4$ . Similarly, the 6th row gives the number  $11^5$ , and so on.

Our main interest in Pascal's triangle is its relation with the expansion of  $(x + y)^n$ . We find that the members of the  $n$ th row,  $n = 0, 1, 2, \dots$ , give the coefficients of the expansion of  $(x + y)^n$ . For example, the elements in row 1 are the binomial coefficients of the expansion  $(x + y)^1 = 1x + 1y$ , the elements in row 2 are the binomial coefficients of the expansion  $(x + y)^2 = 1x^2 + 2xy + 1y^2$ , the elements in row 3 are the binomial coefficients of the expansion  $(x + y)^3 = 1x^3 + 3x^2y + 3xy^2 + 1y^3$ , and so on.

Now suppose we want to expand  $(x + y)^{10}$ . We can create a Pascal's triangle of 10 rows similar to the one given above. Then members of row 10 are

$$\binom{10}{0}, \binom{10}{1}, \binom{10}{2}, \binom{10}{3}, \binom{10}{4}, \binom{10}{5}, \binom{10}{6}, \binom{10}{7}, \binom{10}{8}, \binom{10}{9}, \binom{10}{10}$$

and these members are the coefficients of  $x^{10}y^0, x^9y^1, x^8y^2, x^7y^3, x^6y^4, x^5y^5, x^4y^6, x^3y^7, x^2y^8, x^1y^9, x^0y^{10}$ , respectively. Then  $(x + y)^{10} = \sum_{i=0}^{10} \binom{10}{i} x^{10-i} y^i$ .

Later in this section, using Pascal's triangle, we will design an efficient algorithm to compute  $C(n, r)$ .

## Algorithms to Compute the Factorial and $C(n, r)$

Let  $X$  be a set with  $n$  elements, where  $n \geq 1$ . In this and the preceding sections, while counting the elements of  $X$ , satisfying certain conditions, we came across various formulas. For example, the number of permutations of  $X$  is  $n!$ . The number of  $r$ -permutations of  $X$  is  $P(n, r) = \frac{n!}{(n-r)!}$ , where  $1 \leq r \leq n$ . Similarly, the number of  $r$ -combinations of  $X$  (i.e., the number of  $r$ -element subsets of  $X$ ) is  $C(n, r) = \frac{n!}{r!(n-r)!}$ ,  $1 \leq r \leq n$ . We also encountered various problems that required us to compute these terms.

In this section, we describe the following algorithms:

1. Compute  $n!$ , where  $n$  is a nonnegative integer.
2. Compute  $C(n, r)$ .

### Computing the Factorial

Let  $n$  be a nonnegative integer. Then

$$0! = 1,$$

and if  $n \geq 1$ , then

$$n! = 1 \cdot 2 \cdot 3 \cdots (n-1)n.$$

Using this formula, we can write the following algorithm to compute  $n!$ .

#### ALGORITHM 7.1: Determine the factorial of a nonnegative integer.

*Input:*  $n$ —a positive integer

*Output:*  $n!$

1. **function** factorial( $n$ )
2. **begin**
3.     fact = 1;
4.     **for**  $i := 2$  **to**  $n$  **do**
5.         fact := fact \*  $i$ ;
6.     **return** fact;
7. **end**

### Computing $C(n, r)$

Let  $n$  be a positive integer. Recall that

$$C(n, r) = \frac{n!}{r!(n-r)!}$$

where  $1 \leq r \leq n$ .

Earlier in this section, we described an algorithm to find the factorial of a number. We can, of course, use that algorithm to compute  $C(n, r)$ . However, the algorithms given in the preceding section are only good for moderately sized

integers. If a computer uses 32 bits to store a positive integer, then the largest integer that can be stored is  $2^{31} - 1$ . Therefore, determining  $C(n, r)$  using the definition of a factorial is not efficient.

First we describe the technique known as **divide and conquer** to compute  $C(n, r)$ . In the divide-and-conquer technique, a problem is divided into a fixed number, say  $k$ , of smaller problems of the same kind. Typically,  $k = 2$ . Each of the smaller problems is then divided into  $k$  smaller problems of the same kind, and so on, until the smaller problem is reduced to a case in which the solution is easily obtained. The solutions of the smaller problems are then put together to obtain the solution of the original problem. Typically,  $k = 2$ .

Let us now design an algorithm to compute  $C(n, r)$ . We know that

$$C(n, 0) = 1$$

$$C(n, n) = 1$$

$$C(n, r) = C(n - 1, r - 1) + C(n - 1, r), \quad 0 < r < n.$$

The following recursive algorithm determines  $C(n, r)$ .

**ALGORITHM 7.2:** Compute  $C(n, r)$ .

```

Input:  n and r
Output: C(n, r)
1. function combination(n, r)
2. begin
3.   if (r = 0) or (r = n) then
4.     return 1;
5.   else
6.     return combination(n-1, r-1) + combination(n-1, r);
7. end
```

**EXAMPLE 7.6.9**

In this example, we use the preceding algorithm to determine

$$C(4, 3),$$

i.e., `combination(4, 3)`.

Here  $n = 4$  and  $r = 3$ .

Because  $r \neq 0$  and  $r \neq 4$ , the statement in Line 6 executes, that is, it computes

$$\text{combination}(3, 2) + \text{combination}(3, 3).$$

Each of the expressions `combination(3, 2)` and `combination(3, 3)` is a function call.

Let us first determine `combination(3, 2)`. Here  $n = 3$  and  $r = 2$ . Thus, again the statement in Line 6 executes, that is, it computes

$$\text{combination}(2, 1) + \text{combination}(2, 2).$$

Again each of the expressions `combination(2, 1)` and `combination(2, 2)` is a function call.

Let us first determine  $\text{combination}(2, 1)$ . Here  $n = 2$  and  $r = 1$ . Thus, again the statement in Line 6 executes, that is, it computes

$$\text{combination}(1, 0) + \text{combination}(1, 1).$$

As before, each of the expressions  $\text{combination}(1, 0)$  and  $\text{combination}(1, 1)$  is a function call.

To determine  $\text{combination}(1, 0)$ , the statement in Line 4 executes, which returns 1. Similarly,  $\text{combination}(1, 1)$  returns 1. Hence,  $\text{combination}(2, 1) = 1 + 1 = 2$ .

To compute  $\text{combination}(3, 2)$  we need to compute  $\text{combination}(2, 1) + \text{combination}(2, 2)$ . We have already computed  $\text{combination}(2, 1)$ , which is 2. To compute  $\text{combination}(2, 2)$ , which is a function call, notice that here  $n = 2$  and  $r = 2$ , so the statement in Line 4 executes, which returns 1. Thus,  $\text{combination}(2, 2) = 1$ . Hence,  $\text{combination}(3, 2) = 2 + 1 = 3$ .

To compute  $\text{combination}(4, 3)$  we need to compute  $\text{combination}(3, 2) + \text{combination}(3, 3)$ . We have already computed  $\text{combination}(3, 2)$ , which is 3. To compute  $\text{combination}(3, 3)$ , which is a function call, notice that here  $n = 3$  and  $r = 3$ , so the statement in Line 4 executes, which returns 1. Thus,  $\text{combination}(3, 3) = 1$ . Hence,  $\text{combination}(4, 3) = 3 + 1 = 4$ .

Let us take a closer look at the function  $\text{combination}$ . To do this we compute  $C(10, 6)$ .

Now

$$C(10, 6) = C(9, 5) + C(9, 6).$$

Moreover,

$$C(9, 5) = C(8, 4) + C(8, 5),$$

$$C(9, 6) = C(8, 5) + C(8, 6).$$

Computing  $C(9, 5)$  and  $C(9, 6)$  requires us to compute  $C(8, 5)$ . Therefore,  $C(8, 5)$  will be computed twice. In cases such as this, the divide-and-conquer technique is inefficient because it might solve the same instance several times.

Earlier in this section, we described Pascal's triangle. Pascal's triangular array leads to an efficient algorithm to compute  $C(n, r)$ . This algorithm, which we describe next, falls into the category of a programming technique known as dynamic programming.

In **dynamic programming**, we use a sequence of arrays to construct the solution in a bottom-up approach. Let  $C[0 \dots n, 0 \dots n]$  be a two-dimensional array such that

$$C[i, j] = \binom{i}{j}.$$
<sup>2</sup>

Notice that the rows and columns of the array  $C$  start at 0.

To determine  $\binom{n}{r}$ , we start by determining the elements of the first row of the array  $C[0 \dots n, 0 \dots n]$ , then the second row, and so on.

---

<sup>2</sup>Here we write  $\binom{i}{j}$  for  $C(i, j)$ .

Let us determine  $\binom{5}{3}$ . Here  $n = 5$  and  $r = 3$ . Let  $C[0 \dots 5, 0 \dots 5]$  be the two-dimensional array. Notice that  $C[0 \dots 5, 0 \dots 5]$  is an array of 6 rows and 6 columns and the index of the first row is 0.

Row 0:  $C[0, 0] = \binom{0}{0} = 1$ .

$$C = \begin{bmatrix} 1 \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \end{bmatrix}.$$

Row 1: Determine  $C[1, 0] = \binom{1}{0}$  and  $C[1, 1] = \binom{1}{1}$ .

$$C[1, 0] = 1,$$

$$C[1, 1] = 1.$$

$$C = \begin{bmatrix} 1 \\ 1 & 1 \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \end{bmatrix}.$$

Row 2: Determine  $C[2, 0] = \binom{2}{0}$ ,  $C[2, 1] = \binom{2}{1}$ , and  $C[2, 2] = \binom{2}{2}$ .

$$C[2, 0] = 1,$$

$$C[2, 1] = C[1, 0] + C[1, 1] = 1 + 1 = 2,$$

$$C[2, 2] = 1.$$

$$C = \begin{bmatrix} 1 \\ 1 & 1 \\ 1 & 2 & 1 \\ & & & & & \\ & & & & & \\ & & & & & \end{bmatrix}.$$

Row 3: Determine  $C[3, 0]$ ,  $C[3, 1]$ ,  $C[3, 2]$ , and  $C[3, 3]$ .

$$C[3, 0] = 1,$$

$$C[3, 1] = C[2, 0] + C[2, 1] = 1 + 2 = 3,$$

$$C[3, 2] = C[2, 1] + C[2, 2] = 2 + 1 = 3,$$

$$C[3, 3] = 1.$$

$$C = \begin{bmatrix} 1 \\ 1 & 1 \\ 1 & 2 & 1 \\ 1 & 3 & 3 & 1 \\ & & & & & \\ & & & & & \end{bmatrix}.$$

Row 4: Determine  $C[4, 0]$ ,  $C[4, 1]$ ,  $C[4, 2]$ ,  $C[4, 3]$ , and  $C[4, 4]$ .

$$C[4, 0] = 1,$$

$$C[4, 1] = C[3, 0] + C[3, 1] = 1 + 3 = 4,$$

$$C[4, 2] = C[3, 1] + C[3, 2] = 3 + 3 = 6,$$

$$C[4, 3] = C[3, 2] + C[3, 3] = 3 + 1 = 4,$$

$$C[4, 4] = 1.$$

$$C := \begin{bmatrix} 1 \\ 1 & 1 \\ 1 & 2 & 1 \\ 1 & 3 & 3 & 1 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}.$$

Row 5: Determine  $C[5, 0]$ ,  $C[5, 1]$ ,  $C[5, 2]$ ,  $C[5, 3]$ ,  $C[5, 4]$ , and  $C[5, 5]$ .

$$C[5, 0] = 1,$$

$$C[5, 1] = C[4, 0] + C[4, 1] = 1 + 4 = 5,$$

$$C[5, 2] = C[4, 1] + C[4, 2] = 4 + 6 = 10,$$

$$C[5, 3] = C[4, 2] + C[4, 3] = 6 + 4 = 10,$$

$$C[5, 4] = C[4, 3] + C[4, 4] = 4 + 1 = 5,$$

$$C[5, 5] = 1.$$

$$C := \begin{bmatrix} 1 \\ 1 & 1 \\ 1 & 2 & 1 \\ 1 & 3 & 3 & 1 \\ 1 & 4 & 6 & 4 & 1 \\ 1 & 5 & 10 & 10 & 5 & 1 \end{bmatrix}.$$

This implies that  $\binom{5}{3} = C[5, 3] = 10$ .

The preceding example illustrates that when we compute  $\binom{n}{r}$ , its smaller versions are already computed.

**ALGORITHM 7.3:** Determine  $C(n, r)$  using dynamic programming.

*Input:*  $n, r, n > 0, r > 0, r \leq n$

*Output:*  $C(n, r)$

```

1. function combDynamicProg(n, r)
2. begin
3.   for i = 0 to n do
4.     for j = 0 to min(i, r) do
5.       if j = 0 or j = i then

```

```

6.           C[i,j] := 1;
7.       else
8.           C[i,j] := C[i-1, j-1] + C[i-1, j];
9.       return C[n, r];
10.      end

```

## WORKED-OUT EXERCISES

**Exercise 1:** Expand  $(3a + 5b)^4$ .

**Solution:** Let  $n = 4$ ,  $x = 3a$ , and  $y = 5b$ . Then  $(3a + 5b)^4 = (x + y)^4$ . Hence, by the binomial theorem:

$$\begin{aligned} & (3a + 5b)^4 \\ &= C(4,0)(3a)^4 + C(4,1)(3a)^3(5b) + C(4,2)(3a)^2(5b)^2 \\ &\quad + C(4,3)(3a)(5b)^3 + C(4,4)(5b)^4 \\ &= 81a^4 + 4 \cdot 27 \cdot 5a^3b + 6 \cdot 9 \cdot 25a^2b^2 + 4 \cdot 3 \cdot 125ab^3 + 625b^4 \\ &= 81a^4 + 540a^3b + 1350a^2b^2 + 1500ab^3 + 625b^4. \end{aligned}$$

Notice that  $C(4,0) = 1$ ,  $C(4,1) = 4$ ,  $C(4,2) = 6$ ,  $C(4,3) = 4$ , and  $C(4,4) = 1$ .

**Exercise 2:** Expand  $(2a - 3b)^5$ .

**Solution:** Let  $n = 5$ ,  $x = 2a$ , and  $y = -3b$ . Then  $(2a - 3b)^5 = (x + y)^5$ . Hence, by the binomial theorem:

$$\begin{aligned} & (2a - 3b)^5 \\ &= C(5,0)(2a)^5 + C(5,1)(2a)^4(-3b) + C(5,2)(2a)^3(-3b)^2 \\ &\quad + C(5,3)(2a)^2(-3b)^3 + C(5,4)(2a)(-3b)^4 + C(5,5)(-3b)^5 \\ &= 32a^5 + 5 \cdot 16 \cdot (-3)a^4b + 10 \cdot 8 \cdot 9a^3b^2 + 10 \cdot 4 \cdot (-27)a^2b^3 \\ &\quad + 5 \cdot 2 \cdot 81ab^4 + (-243)b^5. \\ &= 32a^5 - 240a^4b + 720a^3b^2 - 1080a^2b^3 + 810ab^4 - 243b^5. \end{aligned}$$

Notice that  $C(5,0) = 1$ ,  $C(5,1) = 5$ ,  $C(5,2) = 10$ ,  $C(5,3) = 10$ ,  $C(5,4) = 5$ , and  $C(5,5) = 1$ .

**Exercise 3:** Determine the coefficient of  $x^8y^5$  in the expansion of  $(x + y)^{13}$ .

**Solution:** Let  $n = 13$  and  $i = 5$ . Thus,  $x^8y^5 = x^{n-i}y^i$ . Hence, by Theorem 7.6.1, the coefficient of  $x^8y^5$  is

$$\begin{aligned} C(n, i) &= C(13, 5) \\ &= \frac{13!}{5!(13-5)!} \\ &= \frac{13!}{5!8!} \\ &= \frac{13 \cdot 12 \cdot 11 \cdot 10 \cdot 9 \cdot 8!}{5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 \cdot 8!} \\ &= 1287. \end{aligned}$$

**Exercise 4:** Find the coefficient of  $x^{14}y^{12}$  in the expansion of  $(x + y)^{26}$ .

**Solution:** Let  $n = 26$  and  $i = 12$ . Then  $x^{14}y^{12} = x^{n-i}y^i$  and  $(x + y)^{26} = (x + y)^n$ . Hence, by the binomial theorem, the coefficient of  $x^{14}y^{12}$  is:

$$C(26, 12) = \frac{26!}{12!(26-12)!} = \frac{26!}{12!14!}.$$

**Exercise 5:** Find the coefficient of  $x^3y^2$  in the expansion of  $(3x - 7y)^5$ .

**Solution:** By the binomial theorem,

$$\begin{aligned} (3x - 7y)^5 &= \sum_{i=0}^5 C(5, i)(3x)^{5-i}(-7y)^i \\ &= \sum_{i=0}^5 C(5, i)(3^{5-i})x^{5-i}(-7)^iy^i \\ &= \sum_{i=0}^5 C(5, i)(3^{5-i})(-7)^ix^{5-i}y^i. \end{aligned}$$

Let  $i = 2$ . Then the term

$$\begin{aligned} C(5, i)3^{5-i}(-7)^i x^{5-i}y^i &= C(5, 2)3^{5-2}(-7)^2 x^{5-2}y^2 \\ &= C(5, 2)3^3(-7)^2 x^3y^2. \end{aligned}$$

Hence, the coefficient of  $x^3y^2$  is:

$$C(5, 2)3^3(-7)^2 = 10 \cdot 27 \cdot 49 = 13230.$$

**Exercise 6:** Find the coefficient of  $x^3$  in the expansion of  $(x + \frac{1}{x})^7$ .

**Solution:** Let  $x^3$  be in the  $(n+1)$ th term. Now the  $(n+1)$ th term of  $(x + \frac{1}{x})^7$  is

$$C(7, n)x^{7-n}\left(\frac{1}{x}\right)^n = C(7, n)x^{7-2n}.$$

Hence,  $7 - 2n = 3$ . This implies that  $n = 2$ . Therefore, the coefficient of  $x^3$  in the expansion of  $(x + \frac{1}{x})^7$  is  $C(7, 2) = 21$ .

## SECTION REVIEW

---

### Key Terms

binomial expression

Pascal's identity

divide and conquer

binomial coefficient

Pascal's triangle

dynamic programming

### Some Key Results

1. Let  $n$  be an integer such that  $n \geq 1$ . If  $x$  and  $y$  are variables, then

$$\begin{aligned}(x+y)^n &= \sum_{i=0}^n C(n, i)x^{n-i}y^i \\ &= C(n, 0)x^n + C(n, 1)x^{n-1}y + C(n, 2)x^{n-2}y^2 \\ &\quad + \cdots + C(n, n-1)xy^{n-1} + C(n, n)y^n.\end{aligned}$$

2. Let  $n$  and  $r$  be integers such that  $1 \leq r \leq n$ . Then

$$C(n+1, r) = C(n, r-1) + C(n, r).$$

## EXERCISES

---

1. Using the binomial theorem, expand and then simplify.

a.  $(3a+b)^5$       b.  $(x-2y)^6$   
c.  $\left(x+\frac{2}{x}\right)^4$       d.  $(1-x^2)^6$

2. Find the coefficient of  $x^3y^4$  in the expansion of  $(x+2y)^7$ .

3. Determine the coefficient of  $x^8y^3$  in the expansion of  $(x-y)^{11}$ .

4. Find the coefficient of  $x^{15}$  in the expansion of  $(x-x^2)^{10}$ .

5. Consider the binomial expansion of  $(x+y)^{15}$ .

- a. Find the first two terms of the expansion.  
b. Find the last two terms of the expansion.  
c. Find the seventh and eighth terms of the expansion.

6. Consider the binomial expansion of  $(2x-y)^9$ .

- a. Find the first two terms of the expansion.  
b. Find the last two terms of the expansion.  
c. Find the fourth and fifth terms of the expansion.  
d. Find the coefficient of  $x^3y^6$  in the expansion.

7. Using the binomial theorem, expand the following and then simplify.

a.  $\left(x+\frac{1}{x^2}\right)^5$       b.  $\left(x^2-\frac{1}{x}\right)^5$

8. Find the term in the expansion of  $(2x+\frac{1}{x^2})^9$  that does not contain  $x$ .

9. Find the coefficient of  $x^3$  in the expansion of  $(x-\frac{1}{x})^7$ .

10. Find the term that does not contain  $x$  in the expansion of  $(x-\frac{1}{x})^{10}$ .

11. Find the coefficient of  $x^{32}$  in the expansion of  $(x^4+\frac{1}{x^3})^{15}$ .

12. Find the fourth term in the binomial expansion of  $(x^2-x)^{10}$ .

13. Find the fifth term in the binomial expansion of  $(x^2+2y)^8$ .

14. Use Pascal's identity to express  $C(n+2, r)$  in terms of  $C(n, r)$ ,  $C(n, r-1)$ , and  $C(n, r-2)$ .

15. Find the row of the Pascal's triangle that corresponds to  $n = 7$ .

16. Find the row of the Pascal's triangle that corresponds to  $n = 9$ .

17. Prove that

$$C(n, 0) + 2C(n, 1) + 3C(n, 2) + \cdots + (n+1)C(n, n) = (n+2)2^{n-1}$$

for all nonnegative integers  $n$ .

18. Prove that

$$C(n, 0)^2 + C(n, 1)^2 + C(n, 2)^2 + \cdots + C(n, n)^2 = \frac{(2n)!}{(n!)^2}$$

for all nonnegative integers  $n$ .

19. Prove that

$$C(2n+1, 0) + C(2n+1, 1) + C(2n+1, 2) + \cdots + C(2n+1, n) = 2^{2n}$$

for all nonnegative integers  $n$ .

20. Prove the binomial theorem, Theorem 7.6.1, using the principle of induction.

21. Apply the algorithm combination as given in this section to compute the following.

a.  $C(6, 4)$       b.  $C(10, 3)$

22. Apply the algorithm combDynamicProg as given in this section to compute the following.

a.  $C(5, 2)$       b.  $C(6, 4)$       c.  $C(8, 5)$

## 7.7 GENERATING PERMUTATIONS AND COMBINATIONS

A thief breaks into a jewelry store carrying a knapsack. The thief sees various items on display. Associated with each item,  $x$ , is its weight,  $w$ , and a profit,  $p$ , if the thief decides to steal the item. Moreover, if the thief decides to steal an item, the thief must steal all of that particular item and put it in the knapsack. The knapsack the thief is carrying can hold only a certain amount of weight, which we call the *capacity* of the knapsack and denote it by  $c$ . Therefore, if the total weight of the items in the knapsack exceeds  $c$ , the knapsack will break. The thief's objective is to maximize the total profit of the items in the knapsack without breaking the knapsack. This is known as the 0 – 1 knapsack problem because either the thief steals all of a particular item or the thief does not steal that item at all.

More formally, we can state the 0 – 1 knapsack problem as follows. Let  $S$  denote the set of  $n$  items  $x_1, x_2, \dots, x_n$ ;  $p_i$  denote the profit of stealing item  $x_i$ ;  $w_i$  denote the weight of  $x_i$ ; and  $c$  denote the capacity of the knapsack.

$$S = \{x_1, x_2, \dots, x_n\},$$

$$W = \{w_1, w_2, \dots, w_n\},$$

$$P = \{p_1, p_2, \dots, p_n\},$$

$c$  = the capacity of the knapsack, that is, the maximum weight the knapsack can hold, where  $w_i, p_i, i = 1, 2, \dots, n$ , and  $c$  are positive integers.

Determine a subset  $A$  of  $S$  such that

$$\sum_{x_i \in A} p_i \text{ is maximum subject to the condition that } \sum_{x_i \in A} w_i \leq c.$$

One way to solve this problem is to generate all subsets of  $S$  and choose the one that yields the largest profit. Now  $S$  has  $n$  elements, so  $S$  has  $2^n$  subsets. For large values of  $n$ , it may take a considerable amount of time to find the solution. However, our objective is to show that the solution can be obtained as proposed.

Let us consider another famous problem known as the traveling salesperson tour. Suppose that a salesperson wants to visit, say 7, cities starting at one city and visiting each city only once. Moreover suppose that there is a direct route from one city to any other city. Before starting on the tour the salesperson wants to know the order in which the cities should be visited so that the total traveling time is minimal. One way to solve this problem is to generate all  $7!$  permutations of the cities and then choose the one with the shortest traveling time.

There are many other such problems for which one would like to generate all permutations or combinations of a given set to find the solution. The objective of this section is to design algorithms to generate those permutations and combinations.

Let  $A = \{1, 2, \dots, n\}$  be a set with  $n$  elements,  $n \geq 1$ .

Let  $P_1 : a_1 a_2 \cdots a_n$  and  $P_2 : b_1 b_2 \cdots b_n$  be different permutations of  $A$ . Because these are different permutations, there must exist some  $i$  such that  $a_i \neq b_i$ . We say that  $P_1$  precedes  $P_2$  in the **lexicographic order**, written  $P_1 \prec P_2$ , if there exists a positive integer  $i$ ,  $1 \leq i \leq n$ , such that  $a_1 = b_1, a_2 = b_2, \dots, a_{i-1} = b_{i-1}$  and  $a_i < b_i$ . In other words, the first  $i - 1$  elements of  $P_1$  and  $P_2$  are the same and the  $i$ th element of  $P_1$  is less than the  $i$ th element of  $P_2$ .

**EXAMPLE 7.7.1**

Let  $A = \{1, 2, 3, 4, 5\}$ . Then  $12354 \prec 12435$ .

Let  $P_1$  be a permutation of the elements of  $A$ . A permutation  $P_2$  of the elements of  $A$  is called the **next-largest permutation** after  $P_1$  if there is no permutation  $P$  of  $A$  such that  $P_1 \prec P \prec P_2$ .

**EXAMPLE 7.7.2**

The permutation  $56128743$  is the next-largest permutation after  $56128734$ .

**EXAMPLE 7.7.3**

Let  $A = \{1, 2, \dots, n\}$ , where  $n \geq 1$ . The permutation  $1234 \cdots n$  is the smallest permutation (in the lexicographic order) and  $n(n-1)(n-2) \cdots 321$  is the largest permutation (in the lexicographic order) of the set  $A$ .

Our objective is to generate all the permutations of set  $A$ . Because  $A$  has  $n$  elements, there are  $n!$  permutations of set  $A$ . To generate all permutations of set  $A$ , we start with the smallest permutation and continue to generate the next-largest permutation (from the previous permutation) in the lexicographic order until all permutations are generated. Next, we illustrate how to generate the next-largest permutation from a given permutation.

**EXAMPLE 7.7.4**

Let  $A = \{1, 2, 3, 4, 5, 6, 7\}$ . Consider the permutation  $P : 1732546$ . Let  $a_1 = 1, a_2 = 7, a_3 = 3, a_4 = 2, a_5 = 5, a_6 = 4$ , and  $a_7 = 6$ . To generate the next-largest permutation, first we look at the last two elements,  $a_6$  and  $a_7$ , of  $P$ . Here we see that  $a_6 < a_7$ . In this case, to get the next-largest permutation, we simply interchange  $a_6$  and  $a_7$ . It can be shown that the next-largest permutation after  $P$  is  $1732564$ .

**EXAMPLE 7.7.5**

Let  $A = \{1, 2, 3, 4, 5, 6, 7\}$ . Consider the permutation  $P : 1264753$ . Let  $a_1 = 1, a_2 = 2, a_3 = 6, a_4 = 4, a_5 = 7, a_6 = 5$ , and  $a_7 = 3$ .

1. As in the preceding example, to generate the next-largest permutation, first we look at the last two elements,  $a_6$  and  $a_7$ , of  $P$ . Here we see that  $a_6 > a_7$ .
2. Next, we look at the elements  $a_5$  and  $a_6$  and check whether  $a_5 < a_6$ , which is not the case here.
3. Next we look at the elements  $a_4$  and  $a_5$  and check whether  $a_4 < a_5$ . Here we find that  $a_4 < a_5$ .
  - a. Next we look at the elements  $a_5, a_6$ , and  $a_7$  and choose the smallest of these that is also larger than  $a_4$ . In this case,  $a_6 = 5$  is the smallest of  $a_5, a_6$ , and  $a_7$  such that  $a_6 > a_4$ .
  - b. We interchange  $a_4$  and  $a_6$ . So now we have  $a_4 = 5, a_5 = 7, a_6 = 4$ , and  $a_7 = 3$ .

- c. When listing the elements  $a_5$ ,  $a_6$ , and  $a_7$  in the permutation, we list them in increasing order, in this case as 347.

The next-largest permutation after  $P$  is therefore 1265347.

Following these examples, we describe the algorithm to generate the next-largest permutation (in the lexicographic order) after a given permutation as follows: (We assume that the elements of the given permutation are stored in an array  $P$  of length  $n$ . Moreover, we also assume that the given permutation is not the largest permutation.)

**ALGORITHM 7.4:** Generate the next-largest permutation.

*Input:*  $P$ —array containing a permutation  
 $n$ —the number of elements in the permutation

*Output:*  $L$ —array containing the next-largest permutation after  $P$

```

1. procedure nextLargestPermutation( $P, L, n$ )
2. begin
3.   for  $i := 1$  to  $n$  do
4.      $L[i] := P[i];$ 
5.    $i := n - 1;$ 
6.   //Find the largest index  $i$  such
7.   //that  $L[i] < L[i+1]$ 
8.   while  $L[i] > L[i+1]$  do
9.      $i := i - 1;$ 
10.    //Find the index  $j$  of the smallest element
11.    //in  $L[i+1 \dots n]$  such that  $L[i] < L[j]$ 
12.     $j := n;$ 
13.    while  $L[i] > L[j]$  do
14.       $j := j - 1;$ 
15.      swap( $L[i], L[j]$ );
16.      //Arrange the element in  $L[i+1 \dots n]$  in
17.      //increasing order
18.       $s := i + 1;$ 
19.       $t := n;$ 
20.      while  $t > s$  do
21.        begin
22.          swap( $L[s], L[t]$ );
23.           $s := s + 1;$ 
24.           $t := t - 1;$ 
25.        end
26.      end

```

**EXAMPLE 7.7.6**

In this example, we apply the preceding algorithm to generate the next-largest permutation after 27148653. Here we write  $L_i$  to denote  $L[i]$ , the  $i$ th element of the permutation. Thus,  $L_1 = 2$ ,  $L_2 = 7$ ,  $L_3 = 1$ ,  $L_4 = 4$ ,  $L_5 = 8$ ,  $L_6 = 6$ ,  $L_7 = 5$ , and  $L_8 = 3$ .

At Line 5,  $i$  is set to 7. Now  $L_5 > L_6 > L_7 > L_8$ , so the while loop at Line 8 sets  $i$  to 4.

Next, at Line 12,  $j$  is set to 8. The while loop at Line 13 finds the smallest element among  $L_5$ ,  $L_6$ ,  $L_7$ , and  $L_8$  that is larger than  $L_4$ . This loop sets  $j$  to 7.

At Line 15,  $L_i$  is swapped with  $L_j$ ; i.e.,  $L_4$  is swapped with  $L_7$ . At this stage, the permutation looks like 27158643. Next we list the elements starting at  $i + 1 = 5$  in increasing order. At Line 18,  $s$  is set to 5 and at Line 19,  $t$  is set to 8. The while loop at Line 20 rearranges the new elements,  $L_5$ ,  $L_6$ ,  $L_7$ , and  $L_8$ , in increasing order to get the permutation 27153468.

Thus, the largest permutation after 27148653 is 27153468.

Next we discuss how to generate the  $r$ -combinations of a set with  $n$  elements, where  $1 \leq r \leq n$ .

Let  $X = \{1, 2, 3, \dots, n\}$  be a set with  $n$  elements, where  $n \geq 1$ . Let  $1 \leq r \leq n$ . Then, recall that an  $r$ -combination of  $X$  is an  $r$ -element subset of  $X$ . If  $r = n$ , then an  $r$ -combination of  $X$  is an  $n$ -element subset of  $X$ . We know that there is only one  $n$ -element subset of  $X$ , which is  $X$  itself. It follows that generating  $n$ -combinations of  $X$  is trivial.

Let us consider the  $r$ -combinations of  $X$ , where  $1 \leq r < n$ . Let  $A = \{a_1, a_2, \dots, a_r\}$  be an  $r$ -combination of  $X$ , where  $a_i \in \{1, 2, \dots, n\}$  for all  $i = 1, 2, \dots, r$ . When listing the elements of an  $r$ -combination  $\{a_1, a_2, \dots, a_r\}$ , we assume that  $a_1 < a_2 < \dots < a_r$ .

**EXAMPLE 7.7.7**

Let  $X = \{1, 2, 3, 4, 5, 6, 7\}$ . Then  $\{1, 2, 3, 4\}$ ,  $\{2, 4, 5, 6\}$ ,  $\{3, 4, 5, 6\}$ , and  $\{4, 5, 6, 7\}$  are some 4-combinations of  $X$ .

Let  $a = \{a_1, a_2, \dots, a_r\}$  and  $b = b_1 b_2 \dots b_r$  be two different  $r$ -combinations of  $X$ . Let  $a = a_1 a_2 \dots a_r$  be the string corresponding to the  $r$ -combination  $\{a_1, a_2, \dots, a_r\}$  and  $b = b_1 b_2 \dots b_r$  be the string corresponding to the  $r$ -combination  $\{b_1, b_2, \dots, b_r\}$  of  $X$ .

We say that  $a$  precedes  $b$  in the lexicographic order, written  $a \prec b$  if there exists a positive integer  $i$ ,  $1 \leq i \leq n$ , such that  $a_1 = b_1, a_2 = b_2, \dots, a_{i-1} = b_{i-1}$  and  $a_i < b_i$ . In other words, the first  $i - 1$  elements of  $a$  and  $b$  are same, and the  $i$ th element of  $a$  is less than the  $i$ th element of  $b$ . Moreover, we say that  $b$  is the **next-largest  $r$ -combination** after  $a$  if there is no string  $s$ , corresponding to an  $r$ -combination of  $X$ , such that  $a \prec s \prec b$ .

**EXAMPLE 7.7.8**

Let  $X = \{1, 2, 3, 4, 5, 6, 7\}$ . Then 1234 is the string corresponding to the 4-combination  $\{1, 2, 3, 4\}$ . Similarly, 24567 is the string corresponding to the 5-combination  $\{2, 4, 5, 6, 7\}$  of  $X$ .

**EXAMPLE 7.7.9**

Let  $X = \{1, 2, 3, 4, 5, 6, 7\}$ .

- (i) Let  $a = 2356$ ,  $b = 2357$ , and  $c = 2367$ . Then  $a \prec b$  and  $a \prec c$ . It can be shown that  $b$  is the next-largest string after  $a$ .
- (ii) The string 1234 is the smallest string among the strings corresponding to the 4-combinations of  $X$ . Also, the string 4567 is the largest string among the strings corresponding to the 4-combinations of  $X$ .

Notice that in the largest string, 4567, the last number, which is 7, is the largest element of  $X$ , the second-to-last element, which is 6, is the second-largest element of  $X$ , and so on.

Let  $X = \{1, 2, 3, \dots, n\}$ . It can be shown that the string corresponding to the  $r$ -combinations  $\{1, 2, 3, \dots, r\}$  is the smallest among the strings corresponding to  $r$ -combinations of  $X$ . Also, the string corresponding to the  $r$ -combination  $\{n - r + 1, \dots, n - 1, n\}$  is the largest string among the strings corresponding to the  $r$ -combinations of  $X$ .

As in the case of generating permutations, to generate all  $r$ -combinations of  $X$ , we start with the string corresponding to the  $r$ -combinations  $\{1, 2, 3, \dots, r\}$  and continue to generate the next-largest string after the current string until we have generated all such strings.

Let  $X = \{1, 2, 3, \dots, 7\}$ . Consider the string  $a = a_1 a_2 a_3 a_4$  corresponding to a 4-combinations of  $X$ . Now  $a_1 < a_2 < a_3 < a_4$ . It can be shown that the largest value of  $a_4$  is 7, the largest value of  $a_3$  is 6, the largest value of  $a_2$  is 5, and the largest value of  $a_1$  is 4.

In general, suppose  $X = \{1, 2, 3, \dots, n\}$  and  $a = a_1 a_2 \dots a_{r-1} a_r$  is a string corresponding to an  $r$ -combination of  $X$ . Then

1.  $a_1 < a_2 < \dots < a_{r-1} < a_r$ .
2. The largest value of  $a_r$  is  $n$ , the largest value of  $a_{r-1}$  is  $n - 1, \dots$ , the largest value of  $a_1$  is  $n - r + 1$ . In general, the largest value of  $a_i$ , where  $1 \leq i \leq r$ , is  $n - r + i$ , i.e.,  $\max\_value(a_i) = n - r + i$ .

#### EXAMPLE 7.7.10

Let  $X = \{1, 2, 3, \dots, 7, 8\}$  and let  $a = a_1 a_2 a_3 a_4 a_5 a_6$  be a string corresponding to a 6-combination of  $X$ . Here  $n = 8$  and  $r = 6$ . The largest value of  $a_6 = n - r + 6 = 8 - 6 + 6 = 8$  and the largest value of  $a_2 = n - r + 2 = 8 - 6 + 2 = 4$ .

We use these facts to generate all  $r$ -combinations of  $X$ .

#### EXAMPLE 7.7.11

Let  $X = \{1, 2, 3, 4, 5, 6, 7\}$ . Suppose we want to generate all 4-combinations of  $X$ . We start with the string 1234 and generate the next-largest string after this string. Notice that the next-largest string after 1234 is 1235. The next-largest string after 1235 is 1236, and the next-largest string after 1236 is 1237. To find the next-largest string after 1237, we cannot increase the last number any more because  $X$  has only seven elements. That is, the last element of string has its maximum value. It follows that to find the next-largest string, we increase the last number until it reaches its maximum value.

So let us see how we find the next-largest string after 1237. Because the last number has its maximum value, we look at the second-to-last number, which is 3. We know that the maximum value the second-to-last element can have is 6. To generate the next-largest string after 1237, we first increase 3 to 4 and then set the last element to  $4 + 1 = 5$ . Thus, the next-largest string after 1237 is 1245.

#### EXAMPLE 7.7.12

Let  $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ . Consider the string  $a = 134789$  corresponding to the 6-combination  $\{1, 3, 4, 7, 8, 9\}$ . Let us generate the next-largest string after  $a$ . Let  $a_1 = 1$ ,  $a_2 = 3$ ,  $a_3 = 4$ ,  $a_4 = 7$ ,  $a_5 = 8$ , and  $a_6 = 9$ . We see that  $a_4$ ,  $a_5$ , and  $a_6$  have their maximum values. So we look at  $a_3 = 4$ . The largest value  $a_3$  can have is 6. Therefore,  $a_3$  does not have its maximum value. So we increment  $a_3$  by 1, so  $a_3 = 5$ . We then set  $a_4 = a_3 + 1 = 5 + 1 = 6$ ,  $a_5 = a_3 + 2 = 5 + 2 = 7$ , and  $a_6 = a_3 + 3 = 5 + 3 = 8$ . Thus, the next-largest string is 135678.

Notice that in the string 135678,  $a_4 = a_3 + 1$ ,  $a_5 = a_4 + 1$ , and so on. That is,  $a_4$  is the immediate successor of  $a_3$ ,  $a_5$  is the immediate successor of  $a_4$ , and so on.

Following this discussion, given a string  $a$  corresponding to an  $r$ -combination of  $X$ , we can describe the algorithm to generate the next largest string after  $a$  corresponding to an  $r$ -combination of  $X$ .

(We assume that the elements of the given string are stored in an array  $A$  of the length  $n$ . Moreover, we also assume that the given string is not the largest string.)

**ALGORITHM 7.5:** Generate the next-largest  $r$ -combination.

*Input:*  $A$ —array containing  $r$ -combination of a set of  $n$  elements  
 $n$ —the number of elements in the set

*Output:*  $L$ —array containing the next-largest string after  $A$

```

1. procedure nextLargestRCombination(A, L, n, r)
2. begin
3.   for i := 1 to r do
4.     L[i] := A[i];
5.     //Find the largest index i such L[i]
6.     //does not have its maximum value.
7.   i := r;
8.   max := n;
9.   while L[i] = max do
10.    begin
11.      i := i - 1;
12.      max := max - 1;
13.    end
14.    //Increment L[i] by 1
15.    L[i] := L[i] + 1;
16.    //Set the elements to the right of L[i] as the
17.    //successor of the previous element.
18.    for j := i + 1 to r do
19.      L[j] := L[j - 1] + 1;
20.  end

```

**EXAMPLE 7.7.13**

In this example, we illustrate how the preceding algorithm works.

Let  $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$  and let  $a = 1236789$  be the string corresponding to the 7-combination  $\{1, 2, 3, 6, 7, 8, 9\}$ . We generate the next-largest string after  $a$ . We write  $L_i$  for  $L[i]$ .

The **for** loop at Line 3 sets  $L_1 = 1$ ,  $L_2 = 2$ ,  $L_3 = 3$ ,  $L_4 = 6$ ,  $L_5 = 7$ ,  $L_6 = 8$ , and  $L_7 = 9$ .

The statement at Line 7 sets  $i$  to 7, and the statement at Line 8 sets  $max$  to 9.

The **while** loop at Line 9 finds the largest  $i$  such that  $L_i$  does not have its maximum value. Because  $L_7 = 9 = max$ , the statements in Lines 11 and 12 execute

setting  $i$  to 6 and  $\max$  to 8. Next the expression in the `while` evaluates to true because  $L_i = L_6 = 8 = \max$ . So the statements in Lines 11 and 12 execute setting  $i$  to 5 and  $\max$  to 7. Next the expression in the `while` evaluates to true because  $L_i = L_5 = 7 = \max$ . So the statements in Lines 11 and 12 execute setting  $i$  to 4 and  $\max$  to 6. Once again the expression in the `while` evaluates to true because  $L_i = L_4 = 6 = \max$ . So the statements in Lines 11 and 12 execute setting  $i$  to 3 and  $\max$  to 5. The next time the expression in the `while` evaluates to false because  $L_i = L_3 = 3 \neq 5 = \max$ . Therefore, the largest  $i$ , such that  $L_i$  does not have its maximum value, is 3.

Next the statement in Line 15 executes, which sets  $L_3 := L_3 + 1 = 3 + 1 = 4$ .

The `for` loop at Line 18 sets  $L_4 = 5$ ,  $L_5 = 6$ ,  $L_6 = 7$ , and  $L_7 = 8$ .

Thus, the largest string after 1236789 is 1245678. It follows that the 7-combination generated after  $\{1, 2, 3, 6, 7, 8, 9\}$  is  $\{1, 2, 4, 5, 6, 7, 8\}$ .

We leave it as an exercise to write the algorithm that uses the algorithm `nextLargestRCombination` to generate all  $r$ -combinations of  $X$  starting with the string corresponding to the  $r$ -combination  $\{1, 2, \dots, r\}$ .

## WORKED-OUT EXERCISES

**Exercise 1:** List the following permutations of a set with seven elements in the lexicographic order: 3654127, 3145276, 5476312, 2315647, 2346517.

**Solution:** Consider the permutations 2315647 and 2346517. We see that the first two elements of these permutations are the same. The third element of 2315647, which is 1, is smaller than the third element, 4, of 2346517. Therefore,  $2315647 < 2346517$ . In a similar manner, we can show that  $2346517 < 3145276 < 3654127 < 5476312$ . Hence, the permutations in the lexicographic order are 2315647, 2346517, 3145276, 3654127, and 5476312.

**Exercise 2:** Let  $A$  be a set with seven elements. Find the next-largest permutations, in the lexicographic order, after the following permutations of set  $A$ .

(a)  $P : 5412367$

(b)  $P : 1543762$ .

**Solution:**

- (a) Let  $a_1 = 5$ ,  $a_2 = 4$ ,  $a_3 = 1$ ,  $a_4 = 2$ ,  $a_5 = 3$ ,  $a_6 = 6$ , and  $a_7 = 7$ . Here we see that  $a_6 < a_7$ . Therefore, to get the next-largest permutation after  $P$ , we simply interchange  $a_6$  and  $a_7$ . Hence, the next-largest permutation after  $P$  is 5412376.
- (b) Let  $a_1 = 1$ ,  $a_2 = 5$ ,  $a_3 = 4$ ,  $a_4 = 3$ ,  $a_5 = 7$ ,  $a_6 = 6$ , and  $a_7 = 2$ . Here we see that  $a_5 > a_6 > a_7$  and  $a_4 < a_5$ . Therefore, to find the next-largest permutation after  $P$ , first we find the smallest of  $a_5$ ,  $a_6$ , and  $a_7$  that is also larger than  $a_4$ . We see that  $a_6$  is the smallest of  $a_5$ ,  $a_6$ , and  $a_7$  such that  $a_6 > a_4$ . Next we interchange  $a_4$  and  $a_6$  to get  $a_4 = 6$ , and  $a_6 = 3$ . After this while listing the elements of  $P$ , we list  $a_5$ ,  $a_6$ , and  $a_7$  in increasing order. It now follows that the next-largest permutation, in the lexicographic order, after  $P$  is 1546237.

**Exercise 3:** Let  $X = \{1, 2, 3, 4, 5, 6, 7\}$ .

- (a) Let  $a = 135$  be the string corresponding to the 3-combination  $\{1, 3, 5\}$ . Find the largest value of each digit in  $a$ . Which of the digits are at their maximum value?
- (b) Let  $a = 2467$  be the string corresponding to the 4-combination  $\{2, 4, 6, 7\}$ . Find the largest value of each digit in  $a$ . Which of the digits are at their maximum value?

**Solution:**

- (a) Let  $a_1 = 1$ ,  $a_2 = 3$ , and  $a_3 = 5$ . Here  $n = 7$ , the number of elements in  $X$ , and  $r = 3$ . The maximum value of  $a_3 = n - r + 3 = 7 - 3 + 3 = 7$ ; the maximum value of  $a_2 = n - r + 2 = 7 - 3 + 2 = 6$ ; the maximum value of  $a_1 = n - r + 1 = 7 - 3 + 1 = 5$ .

None of the digits is at its maximum value.

- (b) Let  $a_1 = 2$ ,  $a_2 = 4$ ,  $a_3 = 6$ , and  $a_4 = 7$ . Here  $n = 7$ , the number of elements in  $X$ , and  $r = 4$ . The maximum value of  $a_4 = n - r + 4 = 7 - 4 + 4 = 7$ ; the maximum value of  $a_3 = n - r + 3 = 7 - 4 + 3 = 6$ ; the maximum value of  $a_2 = n - r + 2 = 7 - 4 + 2 = 5$ ; the maximum value of  $a_1 = n - r + 1 = 7 - 4 + 1 = 4$ .

The digits  $a_3$  and  $a_4$  are at their maximum value.

**Exercise 4:** Let  $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ .

- (a) Let  $a = 24678$  be the string corresponding to the 5-combination  $\{2, 4, 6, 7, 8\}$ . Find the next-largest string after  $a$ , in the lexicographic order, corresponding to a 5-combination of  $A$ .
- (b) Let  $a = 14589$  be the string corresponding to the 5-combination  $\{1, 4, 5, 8, 9\}$ . Find the next-largest string after  $a$ , in the lexicographic order, corresponding to a 5-combination of  $A$ .

**Solution:**

- (a) Let  $a_1 = 2$ ,  $a_2 = 4$ ,  $a_3 = 6$ ,  $a_4 = 7$ , and  $a_5 = 8$ . We see that  $a_5$  does not have its maximum values. So to get the next-largest string after  $a$ , we simply increment  $a_5$  by 1. Hence, the next-largest string after  $a$ , in the lexicographic order, corresponding to a 5-combination of  $A$  is 24679.

- (b) Let  $a_1 = 1$ ,  $a_2 = 4$ ,  $a_3 = 5$ ,  $a_4 = 8$ , and  $a_5 = 9$ . We see that  $a_4$  and  $a_5$  have their maximum values. So we look at  $a_3 = 5$ . The maximum value  $a_3$  can have is 7. So we increment  $a_3$  by 1 to get  $a_3 = 6$ . We then set  $a_4 = a_3 + 1 = 6 + 1 = 7$  and  $a_5 = a_3 + 2 = 6 + 2 = 8$ . Thus, the next-largest string after  $a$  is 14678.

**SECTION REVIEW****Key Terms**

lexicographic order

next-largest permutation

next-largest  $r$ -combination**Some Key Results**

1. Suppose  $X = \{1, 2, 3, \dots, n\}$  and  $a = a_1 a_2 \cdots a_{r-1} a_r$  is a string corresponding to an  $r$ -combination of  $X$ . Then
  - a.  $a_1 < a_2 < \cdots < a_{r-1} < a_r$ .
  - b. The largest value of  $a_r$  is  $n$ , the largest value of  $a_{r-1}$  is  $n - 1, \dots$ , the largest value of  $a_1$  is  $n - r + 1$ . In general, the largest value of  $a_i$ , where  $1 \leq i \leq r$ , is  $n - r + i$ , i.e.,  $\text{max\_value}(a_i) = n - r + i$ .

**EXERCISES**

1. Let  $A = \{1, 2, 3, 4, 5, 6, 7, 8\}$ . List the following permutations of  $A$  in the lexicographic order.

$$\begin{array}{lll} 13267548, & 26754381, & 37284165, \\ 13587462, & 53728164, & 26753184. \end{array}$$

2. Let  $A = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ . Find the next-largest permutation, in the lexicographic order, after the following permutations.

a. 168957432      b. 198652734

3. Let  $A = \{1, 2, 3, 4, 5\}$ . Starting with 51234, list all the permutations that are generated by repeatedly using the algorithm **nextLargestPermutation**.

4. Let  $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ .

- a. Let  $a = 256$  be the string corresponding to the 3-combination  $\{2, 5, 6\}$ . Find the largest value of each digit in  $a$ . Which of the digits are at their maximum value?
- b. Let  $a = 15679$  be the string corresponding to the 5-combination  $\{1, 5, 6, 7, 9\}$ . Find the largest value of each digit in  $a$ . Which of the digits are at their maximum value?

- c. Let  $a = 3489$  be the string corresponding to the 4-combination  $\{3, 4, 8, 9\}$ . Find the largest value of each digit in  $a$ . Which of the digits are at their maximum value?

5. Let  $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ . Find the next-largest  $r$ -combination, where  $1 \leq r \leq 9$ , in the lexicographic order, after the following  $r$ -combinations.

- a.  $r = 3$ ,  $r$ -combination  $\{3, 5, 7\}$
- b.  $r = 5$ ,  $r$ -combination  $\{2, 3, 6, 7, 8\}$
- c.  $r = 6$ ,  $r$ -combination  $\{1, 3, 4, 7, 8, 9\}$

6. Show that the algorithm **nextLargestPermutation** correctly generates the next-largest permutation.

7. Show that the algorithm **nextLargestRCombination** correctly generates the next-largest  $r$ -combination.

8. Write an algorithm to generate all  $n$ -permutations of a set with  $n$  elements.

9. Write an algorithm to generate all  $r$ -combinations of a set with  $n$  elements.

10. List all 3-permutations of  $\{1, 2, 3, 4, 5\}$ .

11. Let  $A$  be a set with  $n$  elements and  $1 \leq r \leq n$ . Write an algorithm to generate all  $r$ -permutations of  $A$ .

## 7.8 DISCRETE PROBABILITY

The theory of probability plays a crucial role in making inferences. Intuitively, probability measures how likely something is to occur. According to Pierre Simon de Laplace, probability is the ratio of the number of favorable cases to the total number of cases, assuming that all of the various cases are equally possible. To apply this definition, one needs to know the number of favorable cases and the total number of cases, so one needs to know various counting techniques.

In this section, we present a preview to motivate an elementary treatment of probability theory using examples and an appeal to the intuition.

The origin of the theory of probability is associated with two French mathematicians, Pierre de Fermat and Blaise Pascal. We remarked in Chapter 5 that although Fermat did not publish his works, he did correspond with other mathematicians. In 1654, Pascal and Fermat discussed the following problem through letters. (An interesting series of their letters concerning this problem is available in the *Source Book in Mathematics*, by D. E. Smith, published by McGraw-Hill.)

Some gamblers in France wanted to know how to determine the result of an interrupted game, knowing the position of the scores of the two equally skilled players at the time of interruption.

Consider the simple action of throwing a single balanced die. There are only six possible outcomes, each representing the number appearing on the upper face. We will use the term *experiment* to denote this action. The set of all possible outcomes associated with this experiment is given by  $\{1, 2, 3, 4, 5, 6\}$ , which is in general referred to as the *sample space*. The event of observing an odd number can also be expressed as the set  $\{1, 3, 5\}$ .

To define a mathematical structure for the concept of probability, we need the following five terminologies.

1. Experiment
2. Sample space
3. Sample points
4. Events
5. Simple events

### HISTORICAL NOTES



**Pierre Simon de Laplace**  
(1749–1827)  
Laplace was born in Normandy to a farming family. There has been speculation regarding the affluence of his family, and whether it was his family's influence or that of his wealthy neighbors that propelled Laplace's early education. In either case, Laplace distinguished himself while a student at a Benedictine priory school. Owing to family pressure to make a life in the church, Laplace then entered Caen University to study the-

ology. However, while at Caen he became fascinated with mathematics and instead of finishing his degree headed to Paris to pursue an alternative career. He secured a position teaching mathematics at École Militaire and began publishing papers on a variety of mathematical subjects, including integral calculus, differential equations, the potential function, and the Laplace coefficients.

Laplace's academic success continued along with his work in mathematics. For the next several years, Laplace offered papers and presented

theories on a wide array of mathematical topics, including the least squares rule, which gave the theory of probability a solid foundation. He also wrote on thermodynamics, proved the stability of the solar system, and helped usher in the use of the metric system in France. Laplace's mercurial political beliefs permitted him to survive both the turmoil of the Reign of Terror and Napoleonic rule. Laplace was ultimately awarded the position of marquis when the Bourbons were returned to power.

**DEFINITION 7.8.1** ► A **probabilistic experiment**, or **random experiment**, or simply an **experiment**, is the process by which an observation is made.

In probability theory, any action or process that leads to an observation is referred to as an experiment. Some typical probabilistic experiments are given in the following example.

**EXAMPLE 7.8.2**

Some examples of probabilistic experiments are:

1. Tossing a pair of fair coins. (Fair means that both heads and tails have an equal chance of appearing when the coin is tossed.)
2. Throwing a balanced die.
3. Observing the number of incoming phone calls to a switchboard during a given hour.
4. Determining the amount of dosage that must be given to a patient until the patient reacts positively.

Once an experiment has been performed, exactly one of many possible outcomes is observed. It is impossible to predict which one will occur definitely. For example, if a single balanced die is thrown, one of the six possible faces will appear and the number on the upper face will be observed. It is difficult to predict which one of 1, 2, 3, 4, 5, or 6 will appear on the upper face. The set of all possible outcomes of a probabilistic (or random) experiment is known as the *sample space* of the experiment.

**DEFINITION 7.8.3** ►

The **sample space** associated with a probabilistic experiment is the set consisting of all possible outcomes of the experiment and is denoted by  $S$ . The elements of the sample space are referred to as **sample points**. A **discrete sample space** is one that contains either a finite or a countable number of distinct sample points.

With an experiment, we can often associate more than one sample space, depending on what we want to record as an outcome.

**EXAMPLE 7.8.4**

Refer to Example 7.8.2. The respective sample spaces are:

- (i)  $S = \{\text{HH}, \text{HT}, \text{TH}, \text{TT}\}$ , where H stands for heads and T stands for tails.
- (ii)  $S = \{1, 2, 3, 4, 5, 6\}$ .
- (iii)  $S = \{0, 1, 2, 3, \dots\}$ .
- (iv)  $S = (0, \infty)$ , the set of all numbers greater than 0.

In Example 7.8.4, the sample spaces in the first two cases have four and six sample points, respectively. The sample space in the third case has countably infinitely many sample points, and the sample space in the last case is an uncountable set of infinitely many sample points. Thus, the first three are examples of discrete sample spaces.

**EXAMPLE 7.8.5**

Let us consider Experiment (1) of Example 7.8.2. Suppose we want to record an outcome of the number of heads observed. For this the sample space is  $\{2, 1, 0\}$ .

Any event associated with a probabilistic experiment can be characterized as a subset of the sample space, as illustrated in Example 7.8.9.

**DEFINITION 7.8.6** ► An **event** in a discrete sample space  $S$  is a collection of sample points, i.e., any subset of  $S$ . In other words, an event is a set consisting of possible outcomes of the experiment.

**DEFINITION 7.8.7** ► A **simple event** is an event that cannot be decomposed. Each simple event corresponds to one and only one sample point. Any event that can be decomposed into more than one simple event is called a **compound event**.

**DEFINITION 7.8.8** ► Let  $A$  be an event connected with a probabilistic experiment  $E$  and let  $S$  be the sample space of  $E$ . The event  $B$  of nonoccurrence of  $A$  is called the **complementary event** of  $A$ . This means that the subset  $B$  is the complement  $A'$  of  $A$  in  $S$ .

The conversion of an event into a set notation is demonstrated in the following example.

### EXAMPLE 7.8.9

Consider the experiment of throwing a single balanced die and recording the number on the top face. Because the number on the top face may be one of 1, 2, 3, 4, 5, or 6, the sample space is

$$S = \{1, 2, 3, 4, 5, 6\}.$$

Let  $A$  be the event that the number on the top face is even. Then  $A = \{2, 4, 6\}$ . Let  $B$  be the event that the number on the top face is less than 4. Then  $B = \{1, 2, 3\}$ . If  $C$  is the event that the number on the top face is prime, then  $C = \{2, 3, 5\}$ . Let us consider the following events.

$A$  : The number on the top face is an even number.

$B$  : The number on the top face is less than 4.

$C$  : The number on the top face is prime.

$D$  : The number on the top face is 2 or 5.

$E_1$  : The number on the top face is 1.

$E_2$  : The number on the top face is 2.

$E_3$  : The number on the top face is 3.

$E_4$  : Observe a 4.

$E_5$  : Observe a 5.

$E_6$  : Observe a 6.

$F$  : The number on the top face is an odd number.

These can be expressed as

$$A = \{2, 4, 6\},$$

$$B = \{1, 2, 3\},$$

$$C = \{2, 3, 5\},$$

$$D = \{2, 5\},$$

$$E_1 = \{1\},$$

$$E_2 = \{2\},$$

$$E_3 = \{3\},$$

$$\begin{aligned}E_4 &= \{4\}, \\E_5 &= \{5\}, \\E_6 &= \{6\}, \\F &= \{1, 3, 5\}.\end{aligned}$$

Here  $C = \{2, 3, 5\} = \{2, 5\} \cup \{3\} = D \cup E_3$ . Hence,  $C$  is a compound event. Similarly,  $A$ ,  $B$ , and  $D$  are also compound events. Here  $E_1, E_2, \dots, E_6$  are all simple events. The event  $F$  is the complementary event of  $A$ .

In an experiment, two or more events are said to be **equally likely** if, after taking into consideration all relevant evidences, none can be expected in preference to another. The simple events  $\{H\}$  and  $\{T\}$  connected with the experiment “tossing a fair coin” are equally likely events. Similarly, the simple events  $\{1\}, \{2\}, \{3\}, \{4\}, \{5\}$ , and  $\{6\}$  connected with the experiment “throwing a single balanced die and recording the number on the top face” are all equally likely events.

The likelihood of any outcome in a sample space is given by a probability function that assigns to each outcome a real number called the probability of the outcome. Probabilities are real numbers  $x$  such that  $0 \leq x \leq 1$ . The following is the classical definition of probability.

---

**DEFINITION 7.8.10** ▶ Let  $S$  be the sample space of a probabilistic experiment  $E$ . Suppose each outcome of the experiment is equally likely and the number of outcomes is finite. If  $A$  is an event connected with  $E$ , then the **probability** of the occurrence of  $A$ ,  $P(A)$ , is given by

$$P(A) = \frac{|A|}{|S|}.$$

**EXAMPLE 7.8.11**

Consider the experiment of throwing a single balanced die and recording the number on the top face. Because we are considering a balanced die, we can assume that all outcomes of the throws are equally likely. In Example 7.8.9, we have seen there are six possible outcomes. Consider the event:

$A$  : The number on the top face is an even number. Then  $A = \{2, 4, 6\}$ . The probability of the occurrence of  $A$ ,  $P(A)$ , is given by

$$P(A) = \frac{|A|}{|S|} = \frac{3}{6} = \frac{1}{2}.$$

Next we consider the event:

$B$  : The number on the top face is an odd number. Then  $B = \{1, 3, 5\}$ . The probability of the occurrence of  $B$ ,  $P(B)$ , is given by

$$P(B) = \frac{|B|}{|S|} = \frac{3}{6} = \frac{1}{2}.$$

Note that events  $A$  and  $B$  of Example 7.8.11 cannot occur at the same time. These two events are called **mutually exclusive** events.

---

**DEFINITION 7.8.12** ▶ Two events  $A$  and  $B$  connected with a probabilistic experiment  $E$  are said to be **mutually exclusive** if they cannot occur simultaneously.

Considering events  $A$  and  $B$  as subsets of the sample space connected with  $E$ , we say that events  $A$  and  $B$  are mutually exclusive if and only if  $A \cap B = \emptyset$ , i.e., if and only if  $P(A \cap B) = 0$ .

The simple events  $\{1\}$ ,  $\{2\}$ ,  $\{3\}$ ,  $\{4\}$ ,  $\{5\}$ , and  $\{6\}$  connected with the experiment “throwing a single balanced die and recording the number on the top face” are all mutually exclusive events. Also we find that

$$P(\{1\}) = P(\{2\}) = P(\{3\}) = P(\{4\}) = P(\{5\}) = P(\{6\}) = \frac{1}{6}.$$

This implies that in the experiment “throwing a single balanced die and recording the number on the top face,” all of the outcomes of the throw are equally likely, because each possible outcome has the same probability,  $\frac{1}{6}$ .

## Axiomatic Approach

Analyzing the concept of equally likely probability, we see that three conditions must hold.

1. The probability of occurrence of any event must be greater than or equal to 0.
2. The probability of the whole sample space must be 1.
3. If two events are mutually exclusive, the probability of their union is the sum of their respective probabilities.

These three fundamental concepts form the basis of our definition of probability.

## Axioms of Probability

Let  $S$  denote the sample space associated with a given experiment. We assign to every event  $A$  in  $S$  (that is,  $A$  is a subset of  $S$ ) a number,  $P(A)$ , called the probability of  $A$ , so that the following axioms hold.

*Axiom 1:*  $P(A) \geq 0$ .

*Axiom 2:*  $P(S) = 1$ .

*Axiom 3:* For any sequence of mutually exclusive events  $A_1, A_2, \dots$  in  $S$ ,

$$P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i).$$

In particular, Axiom 3, which is stated in terms of an infinite sequence of mutually exclusive events, implies a similar property for a finite sequence. That is, if  $A_1, A_2, \dots, A_n$  in  $S$  are pairwise mutually exclusive events, then

$$P(A_1 \cup A_2 \cup A_3 \cup \dots \cup A_n) = \sum_{i=1}^n P(A_i).$$

### EXAMPLE 7.8.13

Suppose that three items are selected at random from a manufacturing process. Each item is inspected and classified defective, D, or nondefective, N. Then we see that the sample space is

$$S = \{\text{DDD}, \text{DDN}, \text{DND}, \text{DNN}, \text{NDD}, \text{NDN}, \text{NND}, \text{NNN}\}.$$

Accordingly, the simple events are denoted by

$$\begin{aligned} E_1 &= \{\text{DDD}\}, & E_2 &= \{\text{DDN}\}, & E_3 &= \{\text{DND}\}, & E_4 &= \{\text{DNN}\}, \\ E_5 &= \{\text{NDD}\}, & E_6 &= \{\text{NDN}\}, & E_7 &= \{\text{NND}\}, & E_8 &= \{\text{NNN}\}. \end{aligned}$$

Then the sample space is the union of the eight distinct sample events, i.e.,

$$S = E_1 \cup E_2 \cup E_3 \cup E_4 \cup E_5 \cup E_6 \cup E_7 \cup E_8.$$

If we assign

$$P(E_i) = \frac{1}{8},$$

for  $i = 1, 2, \dots, 8$ , then

$$P(S) = \sum_{i=1}^8 P(E_i) = \frac{1}{8} + \frac{1}{8} = 1$$

and

$$P(E_2 \cup E_3 \cup E_5) = P(E_2) + P(E_3) + P(E_5) = \frac{1}{8} + \frac{1}{8} + \frac{1}{8} = \frac{3}{8}.$$

From the axiomatic definition of probability let us show that the classical definition follows.

Let  $S$  be a sample space of a probabilistic experiment  $E$ . Suppose each outcome of the experiment is equally likely and the number of outcomes is finite. Suppose  $S$  consists of a finite number of equally likely outcomes  $a_1, a_2, a_3, \dots, a_n$ . Then  $\{a_1\}, \{a_2\}, \{a_3\}, \dots, \{a_n\}$  are pairwise mutually exclusive events and

$$\{a_1\} \cup \{a_2\} \cup \{a_3\} \cup \dots \cup \{a_n\} = S.$$

Hence, by Axiom 3,

$$P(S) = P(\{a_1\} \cup \{a_2\} \cup \{a_3\} \cup \dots \cup \{a_n\}) = \sum_{i=1}^n P(\{a_i\}).$$

Because  $a_1, a_2, a_3, \dots, a_n$  are equally likely outcomes,

$$P(\{a_i\}) = P(\{a_2\}) = P(\{a_3\}) = \dots = P(\{a_n\}).$$

Hence,

$$1 = P(S) = \sum_{i=1}^n P(\{a_i\}) = nP(\{a_i\}),$$

for  $i = 1, 2, \dots, n$ . Then  $P(\{a_i\}) = \frac{1}{n}$ , for  $i = 1, 2, \dots, n$ .

Let  $A$  be an event connected with  $E$  and let

$$A = \{a_{i_1}, a_{i_2}, a_{i_3}, \dots, a_{i_m}\} \subseteq \{a_1, a_2, a_3, \dots, a_n\}.$$

Then

$$\begin{aligned} P(A) &= P(\{a_{i_1}\} \cup \{a_{i_2}\} \cup \{a_{i_3}\} \cup \dots \cup \{a_{i_m}\}) \\ &= P(\{a_{i_1}\}) + P(\{a_{i_2}\}) + P(\{a_{i_3}\}) + \dots + P(\{a_{i_m}\}) \end{aligned}$$

because  $\{a_{i_1}\}, \{a_{i_2}\}, \{a_{i_3}\}, \dots, \{a_{i_m}\}$  are mutually exclusive events. Hence,

$$P(A) = m \cdot \frac{1}{n} = \frac{|A|}{|S|}.$$

The following theorem lists various properties of probabilities.

**Theorem 7.8.14:** Let  $A$ ,  $B$ , and  $C$  be events in the sample space  $S$ . Then

- (i)  $P(A') = 1 - P(A)$ .
- (ii)  $0 \leq P(A) \leq 1$ .
- (iii)  $P(A) = P(A \cap B) + P(A \cap B')$ .
- (iv)  $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ .

**Proof:**

- (i)  $A$  and  $A'$  are mutually exclusive events. Now  $S = A \cup A'$ . Hence,  $1 = P(S) = P(A \cup A') = P(A) + P(A')$  implies that  $P(A') = 1 - P(A)$ .
- (ii) From part (i),  $1 = P(A) + P(A')$  implies that  $P(A) \leq 1$ . Hence,  $0 \leq P(A) \leq 1$ .
- (iii) For events  $A$  and  $B$ , we find that events  $A \cap B$  and  $A \cap B'$  are mutually exclusive and  $A = (A \cap B) \cup (A \cap B')$ . Hence,

$$P(A) = P((A \cap B) \cup (A \cap B')) = P(A \cap B) + P(A \cap B').$$

- (iv) We have  $A \cup B = (A \cap B) \cup (A \cap B') \cup (A \cap B) \cup (A' \cap B) = (A \cap B) \cup (A \cap B') \cup (A' \cap B)$ . Now events  $(A \cap B)$ ,  $(A \cap B')$ , and  $(A' \cap B)$  are pairwise disjoint. Hence,

$$\begin{aligned} P(A \cup B) &= P(A \cap B) + P(A \cap B') + P(A' \cap B) \\ &= P(A \cap B) + P(A) - P(A \cap B) + P(B) - P(A \cap B) \quad \text{from part (iii)} \\ &= P(A) + P(B) - P(A \cap B). \blacksquare \end{aligned}$$

## Conditional Probability

Consider the throw of two distinct balanced dice. We want to find the probability of getting a sum of 7, when it is given that the digit in the first die is greater than that in the second. In the probabilistic experiment of throwing two dice the sample space  $S$  consists of  $6 \times 6 = 36$  outcomes. We can assume that each of these outcomes is equally likely. Let  $A$  be the event: The sum of the digits of the two dice is 7, and let  $B$  be the event: The digit in the first die is greater than the second. Then

$$\begin{aligned} A : &\{(6, 1), (5, 2), (4, 3), (3, 4), (2, 5), (1, 6)\} \\ B : &\{(6, 1), (6, 2), (6, 3), (6, 4), (6, 5), (5, 1), (5, 2), (5, 3), \\ &(5, 4), (4, 1), (4, 2), (4, 3), (3, 1), (3, 2), (2, 1)\}. \end{aligned}$$

Let  $C$  be the event: The sum of the digits in the two dice is 7 but the digit in the first die is greater than the second. Then

$$C : \{(6, 1), (5, 2), (4, 3)\} = A \cap B.$$

It is given that event  $B$  has occurred. Therefore, we reduce the original sample space of 36 outcomes to the sample space  $B$ . In this sample space, we consider the event  $C$  and the probability of  $C$  in  $B$  is

$$\frac{|C|}{|B|} = \frac{3}{15} = \frac{1}{5}.$$

This is a conditional probability because it is given that the digit in the first die is greater than that in the second. We denote it by the symbol  $P(A | B)$ .

Now  $B$  and  $C$  are events in  $S$ . Thus,

$$P(B) = \frac{|B|}{|S|} = \frac{15}{36} = \frac{5}{12},$$

$$P(C) = \frac{|C|}{|S|} = \frac{3}{36} = \frac{1}{12}$$

and

$$\frac{P(C)}{P(B)} = \frac{P(A \cap B)}{P(B)} = \frac{\frac{1}{12}}{\frac{5}{12}} = \frac{1}{5}.$$

So we find that

$$P(A | B) = \frac{P(A \cap B)}{P(B)}.$$

Formally, we give the following definition.

---

**DEFINITION 7.8.15** ▶ Let  $A$  and  $B$  be two events connected with a probabilistic experiment. Then the **conditional probability** of  $A$ , when it is given that event  $B$  has occurred, i.e.,  $P(B) \neq 0$ , is denoted by the symbol  $P(A | B)$  and is given by

$$P(A | B) = \frac{P(A \cap B)}{P(B)}.$$

## WORKED-OUT EXERCISES

---

**Exercise 1:** Suppose a properly balanced coin is tossed four times and the result is recorded after each toss. List all possible outcomes in the sample space. Find the event  $E$  that contains only the outcomes in which one tail appears. What is the probability of obtaining exactly one tails? What is the probability of obtaining exactly two tails?

**Solution:** For each toss of the coin there are two possible outcomes, heads (H) or tails (T). Hence, the list of all possible outcomes for four tosses is the following:

|      |      |      |      |
|------|------|------|------|
| HHHH | HHHT | HHTH | HTHH |
| THHH | HHTT | HTTH | HTHT |
| THHT | TTHH | THTH | HTTT |
| TTHT | TTTH | THTT | TTTT |

Hence, the sample space  $S$  consists of these equally likely possible outcomes. Now

$$S = \{HHHT, HHTH, HTHH, THHH\}$$

The probability of obtaining exactly one tails is

$$P(E) = \frac{|E|}{|S|} = \frac{4}{16} = \frac{1}{4}.$$

Let  $E$  be the event that contains only the outcomes in which two tails appear. Then

$$E = \{HHTT, HTTH, HTHT, TTHH, THTH, THHT\}.$$

Hence, the probability of obtaining exactly two tails is  $P(E) = \frac{|E|}{|S|} = \frac{6}{16} = \frac{3}{8}$ .

**Exercise 2:** Two coins are tossed. List all possible outcomes in the sample space. Find the number of outcomes and list the outcomes in the event  $E$  that both coins show heads only.

**Solution:** We are tossing two coins. So we consider two places

1st coin      2nd coin

For each toss of the coin there are two possible outcomes, heads (H) or tails (T), for each coin. After the toss, the first coin may show H and the second coin may show also H or the first coin may show H and the second coin may show T. Here we use the multiplicative principle to count the number of elements of  $S$ . For the first coin the number of possible outcomes is 2, and for the second coin also the number of possible outcomes is 2. Hence, the total number of possible outcomes is  $2 \times 2 = 4$ . Now the list of all possible outcomes is

$$S = \{HH, TH, HT, TT\}.$$

Then  $E = \{HH\}$ .

**Exercise 3:** What is the probability that if a fair coin is tossed six times you will get (a) exactly two heads? (b) at least two heads?

**Solution:**

(a) To keep the records, we may consider the six places

|          |          |          |
|----------|----------|----------|
| 1st toss | 2nd toss | 3rd toss |
| 4th toss | 5th toss | 6th toss |

For each toss of the coin there are two possible outcomes, heads (H) or tails (T). So for each toss the sample space consists of two members H and T. Now there are six tosses. With each outcome of the first toss we can associate two possible outcomes of the 2nd toss and after the 2nd toss we see that there are  $2 \times 2 = 4$  possible outcomes HH, TH, HT, TT. With each of these possible outcomes we can associate two possible outcomes, H and T, of the 3rd toss. For example, corresponding to the outcome HH, the possible outcomes after the 3rd toss are HHH, HHT. So after the 3rd toss the number of possible outcomes is  $2 \cdot 2 \cdot 2 = 8$ . So, we use the multiplicative principle and see that the number of possible outcomes after the 6th toss is  $2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 = 64$ . Hence, the number of elements in the sample space S is 64. Now out of these possible outcomes we now find the number of possible outcomes that contain exactly two H's. Some typical examples of these possible outcomes are HTHTTT, THTHTT, and TTHHTT. Therefore, it follows that the number of possible outcomes that contain exactly two H's is the same as the number of permutations of the letters of the word HTTHHTT, and this number is  $\frac{6!}{2!4!} = 15$ . If E denotes the event of possible outcomes that contain exactly two H's, then  $P(E) = \frac{|E|}{|S|} = \frac{15}{64}$ .

- (b) As in part (a), the sample space S consists of 64 elements. Let A be the event that at least two heads are observed and let B be the event that less than two heads are observed. Then  $B = \{\text{TTTTTT, HTHTTT, THTHTT, TTTHTT, TTTTHT, TTTITH}\}$ . Thus,  $P(B) = \frac{7}{64}$ . Now  $S = A \cup B$  and  $A \cap B = \emptyset$ . Hence,  $P(A) = 1 - P(B) = 1 - \frac{7}{64} = \frac{57}{64}$ .

**Exercise 4:** Consider the experiment of throwing two distinct balanced dice and recording the number on the top faces. What is the probability that the sum of the numbers showing on the top faces of the two dice is exactly 7?

**Solution:** The number on the top face of each die is one of 1, 2, 3, 4, 5, or 6. Let  $A = \{1, 2, 3, 4, 5, 6\}$ . Hence, an outcome of the experiment can be considered as an ordered pair  $(a, b)$ , where  $a$  is from the first die and  $b$  is from the second die. Hence, the sample space is

$$S = \{(a, b) \in A \times A \mid a, b \in A\}.$$

Then  $|S| = 36$ . Let E be the event that the sum of the numbers on the top faces is 7. Then

$$E = \{(1, 6), (6, 1), (2, 5), (5, 2), (3, 4), (4, 3)\}.$$

The probability of event E is  $P(E) = \frac{|E|}{|S|} = \frac{6}{36} = \frac{1}{6}$ .

**Exercise 5:** Suppose there are exactly 3 red balls in a bucket of 15 balls. If we choose 4 balls at random, what is the probability that we do not choose a red ball?

**Solution:** We have to choose 4 balls at random from among the 15 balls in a bucket. Hence, the set of all selections of 4 balls from among the 15 balls is the sample space S. Hence,  $|S| = C(15, 4)$ . If we do not choose a red ball, then the selection of 4 balls will be done from the remaining 12 balls. Hence, the event E is the set of all selections of 4 balls from among the 12 balls. Then  $|E| = C(12, 4)$ . The probability of event E is

$$P(E) = \frac{|E|}{|S|} = \frac{C(12, 4)}{C(15, 4)} = \frac{495}{1365} = \frac{33}{91}.$$

**Exercise 6:** There are five red balls and four white balls in a box. Four balls are selected at random from these balls. Find the probability that two of the selected balls will be red and two will be white.

**Solution:** Here the sample space S is the set of all selections of four balls out of nine balls ignoring the order of selections. Hence,  $|S| = C(9, 4)$ . Event E is the set of all selections of four balls such that two of the selected balls are red and two are white.

Because there are five red balls and four white balls, we are going to choose two red balls from five red balls and simultaneously two white balls from four white balls. Then  $|E| = C(5, 2) \cdot C(4, 2)$ . Therefore, the probability of the event E is

$$P(E) = \frac{|E|}{|S|} = \frac{C(5, 2) \cdot C(4, 2)}{C(9, 4)} = \frac{10 \cdot 6}{126} = \frac{10}{21}.$$

**Exercise 7:** A balanced die is thrown three times and the resulting sequence of digits on the upper face is recorded. What is the probability of the event A that either all three digits are equal or none of them is 3?

**Solution:** Because the die is balanced, we can assume that all the outcomes are equally likely. Also, because the die is thrown three times, the outcomes may be one of the 3-tuples  $(a, b, c)$ , where  $a, b, c \in \{1, 2, 3, 4, 5, 6\}$ . Hence, the number of outcomes in the sample space S is  $6 \times 6 \times 6 = 216$ . Let B be the event: All three digits are equal, and C be the event: None of the digits in the upper face is 3. Hence,  $A = B \cup C$ . Then

$$P(A) = P(B \cup C) = P(B) + P(C) - P(B \cap C).$$

Now

$$B = \{(x, x, x) \in S \mid x \in \{1, 2, 3, 4, 5, 6\}\}.$$

Hence,  $|B| = 6$ ,

$$C = \{(a, b, c) \in S \mid a, b, c \in \{1, 2, 4, 5, 6\}\}.$$

Therefore,  $|C| = 5 \cdot 5 \cdot 5 = 125$  and  $|B \cap C| = 5$  (because every member of B is also a member of C except (3, 3, 3)).

Now  $P(B) = \frac{|B|}{|S|} = \frac{6}{216}$ ,  $P(C) = \frac{|C|}{|S|} = \frac{125}{216}$ ,  $P(B \cap C) = \frac{|B \cap C|}{|S|} = \frac{5}{216}$ . Hence,

$$P(A) = \frac{6}{216} + \frac{125}{216} - \frac{5}{216} = \frac{126}{216} = \frac{7}{12}.$$

**Exercise 8:** Find the probability that a 6 is obtained on one of the dice in a throw of two dice given that the sum of the digits on the upper faces is 7.

**Solution:** In the probabilistic experiment of throwing two dice, the sample space  $S$  consists of  $6 \cdot 6 = 36$  outcomes. We can assume that each of these outcomes is equally likely. Let  $B$  be the event: The sum of digits in the two dice is 7, and  $A$

be the event: 6 is obtained on one of the dice. Then

$$B : \{(6, 1), (5, 2), (4, 3), (3, 4), (2, 5), (1, 6)\},$$

$$A : \{(6, 1), (6, 2), (6, 3), (6, 4), (6, 5), (1, 6),$$

$$(2, 6), (3, 6), (4, 6), (5, 6), (6, 6)\},$$

$$A \cap B = \{(6, 1), (1, 6)\},$$

$$P(A \cap B) = \frac{|A \cap B|}{|S|} = \frac{2}{36} = \frac{1}{18},$$

and

$$P(B) = \frac{|B|}{|S|} = \frac{6}{36} = \frac{1}{6}.$$

It is a conditional probability. Hence,

$$P(A | B) = \frac{P(A \cap B)}{P(B)} = \frac{\frac{1}{18}}{\frac{1}{6}} = \frac{1}{3}.$$

## SECTION REVIEW

### Key Terms

probabilistic experiment  
random experiment  
experiment  
sample space

sample points  
discrete sample space  
simple event  
compound event

complementary event  
equally likely  
probability  
conditional probability

### Some Key Definitions

- The sample space associated with a probabilistic experiment is the set consisting of all possible outcomes of the experiment and is denoted by  $S$ . The elements of the sample space are referred to as sample points. A discrete sample space is one that contains either a finite or a countable number of distinct sample points.
- Let  $S$  be the sample space of a probabilistic experiment  $E$ . Suppose each outcome of the experiment is equally likely and the number of outcomes is finite. If  $A$  is an event connected with  $E$ , then the probability of the occurrence of  $A$ ,  $P(A)$ , is given by  $P(A) = \frac{|A|}{|S|}$ .
- Two events  $A$  and  $B$  connected with a probabilistic experiment  $E$  are said to be mutually exclusive if they cannot occur simultaneously.

### Key Result

- Let  $A$ ,  $B$ , and  $C$  be events in the sample space  $S$ . Then
  - $P(A') = 1 - P(A)$ .
  - $0 \leq P(A) \leq 1$ .
  - $P(A) = P(A \cap B) + P(A \cap B')$ .
  - $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ .

## EXERCISES

---

1. If two distinct dice are cast, what is the probability that the sum of the number showing on the top faces of the dice is less than
  - a. 7?
  - b. 6?
  - c. 12?
2. Consider an experiment in which two marbles are drawn at random from an urn containing eight red, six blue, four green, and two white marbles. Determine the probability when both marbles are
  - a. white.
  - b. green.
  - c. red.
3. If a properly balanced coin is tossed three times in sequence, then list all possible outcomes in the sample space. Find the event  $E$  that contains only the outcomes in which one tail appears. What is the probability of obtaining exactly one tail?
4. Three distinct coins are tossed. List all possible outcomes in the sample space. List the outcomes in the event  $E$  that two coins show heads only. What is the probability of obtaining exactly two heads?
5. Find the associated sample space when three letters are chosen simultaneously at random from the letters  $a, b, c, d$ , and  $e$ .
6. Consider the experiment of throwing a balanced die twice in sequence and recording the number on the top face. Let  $E, F$ , and  $G$  be the following events:  $E =$  one of the numbers is at least 4,  $F =$  one of the numbers is prime,  $G =$  one of the numbers is a multiple of 2. Describe the events as subsets of the sample space.
7. If four distinct coins are tossed, find the probability of observing three heads at a time.
8. A coin is tossed six times in sequence. What is the probability of obtaining four heads and two tails?
9. In the experiment of rolling a balanced die, find the probability of obtaining a number greater than 3.
10. Consider the experiment of throwing two distinct balanced dice and recording the number on the top faces. What is the probability that the sum of the number showing on the top faces of the two dice is exactly 10?
11. Consider the experiment of throwing two distinct balanced dice and recording the number on the top faces. What is the probability that the sum of the number showing on the top faces of the two dice is exactly 5?
12. Consider an experiment of throwing two distinct balanced dice and recording the number on the top faces. What is the probability that the sum of the number showing on the top faces of the two dice is less than 5?
13. Suppose there are eight applicants, five men and three women, for some job. These men and women will be interviewed in a random order. What is the probability that the three women will all be interviewed before the men?
14. Suppose there are exactly 4 blue balls in a bucket of 15 balls. If we choose 5 balls at random, what is the probability that we do not choose a blue ball?
15. Five balls are chosen at random from eight red balls and five yellow balls. Find the probability that
  - a. all five balls are red,
  - b. at least two balls are red,
  - c. three balls are red and two balls are yellow.
16. A bag contains seven red and five white balls. Four balls are drawn at random. What is the probability that
  - a. all of them are red?
  - b. two of them are red and two are white?
17. Four students are selected at random from a class consisting of eight boys and six girls. Find the probability that only four girl students are selected.
18. Find the probability of a randomly chosen permutation of the letters of the word *combination* so that no two vowels will be adjacent.
19. We would like to form at random a number of five digits using the digits 1, 2, 3, 4, 5 and 6. Find the probability that it contains three 2's and two 3's.
20. A five-member committee is to be formed at random from four boys and six girls. Find the probability that the committee consists of exactly two boys and three girls.
21. What is the probability that a randomly chosen subset of  $\{a, b, c, d, e, f, g\}$  contains both  $c$  and  $f$ ?
22. A balanced die is thrown three times and the resulting sequence of digits on the upper face is recorded. What is the probability of the event  $A$  that either all three digits are equal or none of them is 6?
23. A certain defective die is tossed. The digits 1, 2, 3, 4, 5, and 6 will be shown on the upper face with the following probabilities:  $P(\{1\}) = \frac{3}{18}$ ,  $P(\{2\}) = \frac{2}{18}$ ,  $P(\{3\}) = \frac{4}{18}$ ,  $P(\{4\}) = \frac{2}{18}$ , and  $P(\{6\}) = \frac{6}{18}$ . Find
  - a. The probability  $P(\{5\})$ .
  - b. The probability of the events
 

*A* : The number on the top face is an even number.  
*B* : The number on the top face is less than 4.
24. Find the probability that a 5 is obtained on one of the dice in a throw of two dice given that the sum of the digits on the upper faces is 7.

## PROGRAMMING EXERCISES

1. Write a program to find the number of integer solutions of an equation of the form  $x_1 + x_2 + \dots + x_{10} = a$  such that  $a > 0$ ,  $x_1 \geq a_1 \geq 0$ ,  $x_2 \geq a_2 \geq 0, \dots, x_{10} \geq a_{10} \geq 0$ , where  $a$  and  $a_i$ ,  $i = 1, 2, \dots, 10$ , are integers.
2. Write a program that uses the divide-and-conquer technique to implement the algorithm to compute  $C(n, r)$ .
3. Write a program that uses the dynamic programming technique to implement the algorithm to compute  $C(n, r)$ .
4. Write a program to implement the algorithm `nextLargestPermutation`.
5. Write a program to generate all permutations of a set.
6. Write a program to implement the algorithm `nextLargestRCombination`.
7. Write a program to generate all  $r$ -combinations of a set.

## Recurrence Relations

The objectives of this chapter are to:

- Learn about recurrence relations
- Learn the relationship between sequences and recurrence relations
- Explore how to solve recurrence relations by iteration
- Learn about linear homogeneous recurrence relations and how to solve them
- Become familiar with linear nonhomogeneous recurrence relations

In Chapter 7, we described various counting techniques, such as the multiplication principle, the addition principle, the pigeonhole principle, permutations, combinations, and probability. However, there are other, more advanced counting methods available when these methods are not sufficient to solve the problem. Some advanced counting techniques, such as recurrence relations, can be adapted to suit the needs of particular counting problems.

In this chapter, we introduce recurrence relations. Roughly speaking, a recurrence relation relates the  $n$ th term of a sequence to some of its preceding terms. We begin by describing the relationship between sequences and recurrence relations.

## 8.1 SEQUENCES AND RECURRENCE RELATIONS

Let us begin our discussion of the relationship between sequences and recurrence relations by considering the following problems.

1. Sam received a yearly bonus and deposited \$10,000 in a local bank yielding 7% interest compounded annually. He wants to know the total amount to be accumulated after, say 10 years, or 30 years, or, in general,  $n$  years. Let  $A_n$  denote the total amount accumulated after  $n$  years. Then  $A_0 = 10000$ , the initial amount;  $A_{10}$  = the amount after 10 years; and  $A_{30}$  = the amount after 30 years. Determining  $A_{10}$  requires us to find  $A_9$ , the amount after 9 years, which is also the amount at the beginning of the 10th year. We thus see that we can construct the following sequence:  $A_0, A_1, A_2, \dots, A_9, A_{10}, \dots, A_{30}, \dots, A_{n-1}, A_n, \dots$ . As we shall see in Example 8.1.10, we can define  $A_n = (1.07)A_{n-1}$ , for all  $n \geq 1$ ; that is,  $A_n$  can be defined in terms of  $A_{n-1}$ . Such an equation is called a recurrence relation, i.e., the  $n$ th term of the sequence can be defined by using some of the previous terms.
2. Suppose we are interested in determining the number of  $n$ -bit binary strings, i.e., binary strings of length  $n$ , that do not contain 00 as a substring for  $n = 1, 2, 3, 4, \dots$ . Let  $B_n$  = the number of  $n$ -bit binary strings that do not contain 00 as a substring. Then we can construct the sequence  $B_1, B_2, \dots, B_n, \dots$ . In Example 8.1.7, we show that  $B_n = B_{n-1} + B_{n-2}$ ,  $n \geq 3$ .

---

**REMARK 8.1.1** ▶ Throughout this chapter when we speak of an  $n$ -bit binary string we mean a binary string of length  $n$ . Moreover, a binary string is a string consisting of 0's and 1's.

As these examples show, there is a relationship between sequences and recurrence relations. In the remainder of this section, we provide detailed examples illustrating this relationship and discuss a method for solving recurrence relations. By “solving” a recurrence relation, we mean finding an explicit formula, a term we will define shortly, for the  $n$ th term. We begin with the following example.

### EXAMPLE 8.1.2

Consider the following two sequences:

$$S_1 : 3, 5, 7, 9, \dots$$

$$S_2 : 3, 9, 27, 81, \dots$$

We can find a formula for the  $n$ th term of sequences  $S_1$  and  $S_2$  by observing the pattern of the sequences.

$$S_1 : 2 \cdot 1 + 1, 2 \cdot 2 + 1, 2 \cdot 3 + 1, 2 \cdot 4 + 1, \dots$$

$$S_2 : 3^1, 3^2, 3^3, 3^4, \dots$$

For  $S_1$ ,  $a_n = 2n + 1$  for  $n \geq 1$ , and for  $S_2$ ,  $a_n = 3^n$  for  $n \geq 1$ . This type of formula is called an **explicit formula** for the sequence, because using this formula we can directly find any term of the sequence without using other terms of the sequence. For example,  $a_3 = 2 \cdot 3 + 1 = 7$ .

In the preceding example, it was easy to find an explicit formula for the  $n$ th term of the sequence. However, there are sequences for which finding an explicit formula is not obvious. Let us consider the following example.

**EXAMPLE 8.1.3**

Let  $S$  denote the sequence

$$1, 1, 2, 3, 5, 8, 13, 21, \dots$$

For this sequence, the explicit formula is not obvious. If we observe closely, however, we find that the pattern of the sequence is such that any term after the second term is the sum of the preceding two terms. Now

$$\begin{aligned} \text{3rd term} &= 2 = 1 + 1 = \text{1st term} + \text{2nd term} \\ \text{4th term} &= 3 = 1 + 2 = \text{2nd term} + \text{3rd term} \\ \text{5th term} &= 5 = 2 + 3 = \text{3rd term} + \text{4th term} \\ \text{6th term} &= 8 = 3 + 5 = \text{4th term} + \text{5th term} \\ \text{7th term} &= 13 = 5 + 8 = \text{5th term} + \text{6th term} \end{aligned}$$

Hence, the sequence  $S$  can be defined by the equation

$$f_n = f_{n-1} + f_{n-2} \quad (8.1)$$

for all  $n \geq 3$  and

$$\begin{aligned} f_1 &= 1, \\ f_2 &= 1. \end{aligned} \quad (8.2)$$

This sequence is called the Fibonacci sequence in honor of the Italian mathematician Leonardo Fibonacci, and the terms of the sequence are called Fibonacci numbers. We see that we can find the  $n$ th term,  $n \geq 3$ , of the sequence from the preceding two terms using Equation 8.1. Notice that the values of  $f_1$  and  $f_2$  are given explicitly. Now because

$$f_3 = f_1 + f_2,$$

using  $f_1$  and  $f_2$ , we can determine  $f_3$ . Similarly, we have

$$f_4 = f_2 + f_3.$$

Therefore, using  $f_2$  and  $f_3$ , we can determine  $f_4$ , and so on.

Because  $f_n$  is defined in terms of previous terms of the sequence, equations of the form (8.1) are called recurrence equations of recurrence relations.

Given a sequence  $a_0, a_1, a_2, \dots, a_n, \dots$  on a set  $S$ , we can define a function  $f : \mathbb{N}^0 \rightarrow S$  such that  $f(n) = a_n$  for all  $n \geq 0$ , where  $\mathbb{N}^0 = \{0, 1, 2, 3, \dots\}$  is the set of all nonnegative integers. On the other hand, if  $g : \mathbb{N}^0 \rightarrow S$  is a function, then we can define a sequence  $\{s_n\}$  on  $S$  by defining  $s_n = g(n)$ . Thus, there is a one-to-one correspondence between the set of functions from  $\mathbb{N}^0$  into  $S$  and the set of all sequences on  $S$ .

Next, we define the notion of recursive definition. To do this, let us consider the function  $f : \mathbb{N}^0 \rightarrow \mathbb{Z}^+$  defined by

$$f(n) = 2f(n-1) + f(n-2) \quad \text{for all } n \geq 2. \quad (8.3)$$

and

$$\begin{aligned} f(0) &= 5, \\ f(1) &= 7. \end{aligned} \quad (8.4)$$

We see that if we know  $f(n-1)$  and  $f(n-2)$ , then we can compute  $f(n)$  for all  $n \geq 2$ . For example, suppose we want to know the value of  $f(4)$ . Now  $f(4) =$

$2f(3) + f(2)$ . Hence, to find  $f(4)$  we need to know  $f(3)$  and  $f(2)$ . We know that  $f(0) = 5$  and  $f(1) = 7$ . Thus,

$$\begin{aligned}f(2) &= 2f(1) + f(0) = 2 \cdot 7 + 5 = 19, \\f(3) &= 2f(2) + f(1) = 2 \cdot 19 + 7 = 45, \\f(4) &= 2f(3) + f(2) = 2 \cdot 45 + 19 = 109.\end{aligned}$$

The definition of the function  $f$  as given in (8.3) is called a **recursive definition**. In this case, the equation

$$f(n) = 2f(n-1) + f(n-2) \quad \text{for all } n \geq 2$$

is called a *recurrence relation*, and  $f(0) = 5$  and  $f(1) = 7$  are called the *initial conditions* for the function  $f$ .

Let us consider another recursive definition.

### EXAMPLE 8.1.4

Consider the function  $f : \mathbb{N}^+ \rightarrow \mathbb{Z}^+$  defined by

$$\begin{aligned}f(0) &= 1, \\f(n) &= nf(n-1) \quad \text{for all } n \geq 1.\end{aligned}$$

Then

$$\begin{aligned}f(0) &= 1 = 0!, \\f(1) &= 1 \cdot f(0) = 1 = 1!, \\f(2) &= 2 \cdot f(1) = 2 \cdot 1 = 2 = 2!, \\f(3) &= 3 \cdot f(2) = 3 \cdot 2 \cdot 1 = 6 = 3!,\end{aligned}$$

and so on. Here  $f(n) = nf(n-1)$  for all  $n \geq 1$  is the recurrence relation, and  $f(0) = 1$  is the initial condition for the function  $f$ . Notice that the function  $f$  is nothing but the factorial function, i.e.,  $f(n) = n!$  for all  $n \geq 0$ .

Let us consider the function  $f$  as given in (8.3). If we write  $a_n = f(n)$ , then (8.3) translates into the following equation:

$$a_n = 2a_{n-1} + a_{n-2} \quad \text{for all } n \geq 2.$$

That is,  $a_n$  is defined in terms of  $a_{n-1}$  and  $a_{n-2}$ . As remarked previously, such an equation is called a recurrence relation. Moreover, (8.4) translates into  $a_0 = 5$  and  $a_1 = 7$ . These are called the initial conditions for the recurrence relation.

Let us give a formal definition of recurrence relations and initial conditions.

---

**DEFINITION 8.1.5** ► A **recurrence relation** for a sequence  $a_0, a_1, a_2, \dots, a_n, \dots$  is an equation that relates  $a_n$  to some of the terms  $a_0, a_1, a_2, \dots, a_{n-2}, a_{n-1}$  for all integers  $n$  with  $n \geq k$ , where  $k$  is a nonnegative integer. The **initial conditions** for the recurrence relation are a set of values that explicitly define some of the members of  $a_0, a_1, a_2, \dots, a_{k-1}$ .

The equation

$$a_n = 2a_{n-1} + a_{n-2} \quad \text{for all } n \geq 2,$$

as defined above, relates  $a_n$  to  $a_{n-1}$  and  $a_{n-2}$ . Here  $k = 2$ . So this is a recurrence relation with initial conditions  $a_0 = 5$  and  $a_1 = 7$ .

### EXAMPLE 8.1.6

For the sequence of Example 8.1.3, equation (8.2) gives the initial conditions.

We now give some examples of interesting sequences defined by a recurrence relation and initial conditions.

**EXAMPLE 8.1.7**

**Number of  $n$ -bit binary strings that do not contain 00 and contain at least 1 bit.** Let  $a_i \in \{0, 1\}$  and  $a_0 a_1 a_2 \dots a_{n-1} a_n$  be any  $n$ -bit binary string. Suppose we are interested to know the number of  $n$ -bit binary strings that do not contain 00 as a substring for  $n = 1, 2, 3, 4, \dots$ .

Let  $B_n$  = the number of  $n$ -bit binary strings that do not contain 00 as a substring.

Now 0 and 1 are the only 1-bit binary strings and these strings do not contain 00, so  $B_1 = 2$ .

Next, 00, 01, 10, and 11 are the only 2-bit binary strings. However, among these, 01, 10, and 11 are the only 2-bit binary strings that do not contain 00, so  $B_2 = 3$ .

Next we find that 000, 001, 010, 100, 101, 011, 110, and 111 are the only 3-bit strings. Among these, 010, 101, 011, 110, and 111 are the only 3-bit binary strings that do not contain 00. Thus,  $B_3 = 5$ .

Hence,  $B_1 = 2$ ,  $B_2 = 3$ , and  $B_3 = 5$ .

Let us now find a recurrence relation and initial conditions for the sequence  $B_1, B_2, B_3, B_4, \dots$

We assume  $n \geq 3$ . To count the number of  $n$ -bit binary strings that do not contain 00 as a substring we consider the following two types of strings:

- (i)  $n$ -bit binary strings that begin with 1 and that do not contain 00 as a substring,
- (ii)  $n$ -bit binary strings that begin with 0 and that do not contain 00 as a substring.

Now the set of strings of type (i) and the set of strings of type (ii) are disjoint. Hence, by the addition principle, the total number of strings of the required type is the sum of the strings of type (i) and type (ii).

Consider an  $n$ -bit binary string of type (i), i.e.,

$$1 a_2 a_3 a_4 \cdots a_{n-1} a_n.$$

It follows that the  $(n-1)$ -bit binary string  $a_2 a_3 a_4 \cdots a_{n-1} a_n$  does not contain 00 as a substring. On the other hand, if we consider an  $(n-1)$ -bit binary string  $a_2 a_3 a_4 \cdots a_{n-1} a_n$  that does not contain 00 as a substring, then the  $n$ -bit binary string  $1 a_2 a_3 a_4 \cdots a_{n-1} a_n$  does not contain 00 as a substring. Hence, the number of  $n$ -bit binary strings of type (i) is  $B_{n-1}$ .

Next consider an  $n$ -bit binary string of type (ii), i.e.,

$$0 a_2 a_3 a_4 \cdots a_{n-1} a_n.$$

Here it follows that  $a_2 = 1$  and the  $(n-2)$ -bit binary string  $a_3 a_4 \cdots a_{n-1} a_n$  does not contain 00 as a substring. On the other hand, if we consider an  $(n-2)$ -bit binary string  $a_3 a_4 \cdots a_{n-1} a_n$  that does not contain 00 as a substring, then the  $n$ -bit binary string  $01 a_3 a_4 \cdots a_{n-1} a_n$  does not contain 00 as a substring. Hence, the number of  $n$ -bit binary strings of type (ii) is the same as the number of  $(n-2)$ -bit binary strings  $a_3 a_4 \cdots a_{n-1} a_n$  that do not contain 00 as a part of the string, and this equals  $B_{n-2}$ . Hence,  $B_n = B_{n-1} + B_{n-2}$ ,  $n \geq 3$ . Therefore, a recurrence relation for the sequence  $B_1, B_2, B_3, B_4, \dots$  is

$$B_n = B_{n-1} + B_{n-2}, \quad n \geq 3$$

and the initial conditions are  $B_1 = 2$  and  $B_2 = 3$ .

**EXAMPLE 8.1.8****Number of  $n$ -bit binary strings that do not contain 111 and contain at least 1 bit.**

Let  $a_i \in \{0, 1\}$  and  $a_0 a_1 a_2 \cdots a_{n-1} a_n$  be an  $n$ -bit binary string. Suppose we are interested in finding the number of  $n$ -bit binary strings that do not contain 111 as a substring for  $n = 1, 2, 3, 4, \dots$ .

Let  $B_n$  = the number of  $n$ -bit binary strings that do not contain 111 as a substring.

Now 0 and 1 are the only 1-bit binary strings and these strings do not contain 111. Thus,  $B_1 = 2$ .

Next, 00, 01, 10, and 11 are the only 2-bit binary strings and these strings do not contain 111. Thus,  $B_2 = 4$ .

Notice that 000, 001, 010, 100, 101, 011, 110, and 111 are the only 3-bit binary strings. However, 000, 001, 010, 100, 101, 011, and 110 are the only 3-bit binary strings that do not contain 111. Thus,  $B_3 = 7$ .

Let us now consider 4-bit binary strings. There are sixteen 4-bit binary strings. Out of these, 0000, 0001, 0010, 0100, 0110, 0011, 0101, 1000, 1001, 1010, 1011, 1101, and 1100 are the only 3-bit strings that do not contain 111. Thus,  $B_4 = 13$ .

Hence,  $B_1 = 2$ ,  $B_2 = 4$ ,  $B_3 = 7$ , and  $B_4 = 13$ .

Let us now find a recurrence relation and initial conditions for the sequence  $B_1, B_2, B_3, B_4, B_5, \dots, B_n, \dots$ .

We assume  $n \geq 4$ . To count the number of  $n$ -bit binary strings that do not contain 111 as a substring we consider the following two types of strings:

- (i)  $n$ -bit strings that begin with 0 and that do not contain 111 as a substring,
- (ii)  $n$ -bit strings that begin with 1 and that do not contain 111 as a substring.

Now the set of strings of type (i) and the set of strings of type (ii) are disjoint. Hence, by the addition principle, the total number of strings of the required type is the sum of the strings of type (i) and type (ii).

Let us determine the number of type (i) strings. Consider an  $n$ -bit binary string of type (i), i.e.,

$$0 a_2 a_3 a_4 \cdots a_{n-1} a_n.$$

It follows that the  $(n - 1)$ -bit binary string  $a_2 a_3 a_4 \cdots a_{n-1} a_n$  does not contain 111 as a substring. Again, if we consider an  $(n - 1)$ -bit binary string  $a_2 a_3 a_4 \cdots a_{n-1} a_n$  that does not contain 111 as a substring, then the  $n$ -bit binary string  $0 a_2 a_3 a_4 \cdots a_{n-1} a_n$  does not contain 111 as a substring. Hence, the number of  $n$ -bit binary strings of type (i) is  $B_{n-1}$ .

Let us now determine the number of type (ii) strings. Consider an  $n$ -bit binary string of type (ii), i.e.,

$$1 a_2 a_3 a_4 \cdots a_{n-1} a_n$$

Here we consider two different cases:

$$\text{Case (ii)}_1 : a_2 = 0.$$

$$\text{Case (ii)}_2 : a_2 = 1.$$

**Case (ii)<sub>1</sub>:**  $a_2 = 0$ . Consider an  $n$ -bit binary string of type (ii)<sub>1</sub>, i.e.,

$$10 a_3 a_4 \cdots a_{n-1} a_n$$

It follows that the  $(n - 2)$ -bit binary string  $a_3 a_4 \cdots a_{n-1} a_n$  does not contain 111 as a substring. Again if we consider an  $(n - 2)$ -bit binary string  $a_3 a_4 \cdots a_{n-1} a_n$  that does not contain 111 as a substring, then the  $n$ -bit string  $10 a_3 a_4 \cdots a_{n-1} a_n$  does not

contain 111 as a substring. Hence, the number of  $n$ -bit binary strings in Case (ii)<sub>1</sub> is the same as the number of  $(n - 2)$ -bit binary strings  $a_3a_4 \cdots a_{n-1}a_n$  that do not contain 111 as a part of the string, and this is  $B_{n-2}$ .

**Case (ii)<sub>2</sub>:**  $a_2 = 1$ . Consider an  $n$ -bit binary string of type (ii)<sub>2</sub>, i.e.,

$$11a_3a_4 \cdots a_{n-1}a_n.$$

It follows that  $a_3$  must be 0 and the  $(n - 3)$ -bit binary string  $a_4 \cdots a_{n-1}a_n$  does not contain 111 as a substring. Again if we consider an  $(n - 3)$ -bit binary string  $a_4 \cdots a_{n-1}a_n$  that does not contain 111 as a substring, then the  $n$ -bit string 110a<sub>4</sub>  $\cdots$  a<sub>n-1</sub>a<sub>n</sub> does not contain 111 as a substring. Hence, the number of  $n$ -bit binary strings in Case (ii)<sub>2</sub> is the same as the number of  $(n - 3)$ -bit binary strings  $a_4 \cdots a_{n-1}a_n$  that do not contain 111 as a substring, and this is  $B_{n-3}$ .

It follows that the number of strings of type (ii) is  $B_{n-2} + B_{n-3}$ .

Consequently, the number of  $n$ -bit binary strings that do not contain 111 as a substring is  $B_n = B_{n-1} + B_{n-2} + B_{n-3}$ ,  $n \geq 4$ . Therefore, a recurrence relation for the sequence  $B_1, B_2, B_3, B_4, \dots$  is

$$B_n = B_{n-1} + B_{n-2} + B_{n-3}, \quad n \geq 4$$

and the initial conditions are  $B_1 = 2$ ,  $B_2 = 4$ ,  $B_3 = 7$ , and  $B_4 = 13$ .

### EXAMPLE 8.1.9

**Number of subsets of a finite set.** Let  $s_n$  denote the number of subsets of a set  $A$  with  $n$  elements,  $n \geq 0$ . In Worked-Out Exercise 9 (Chapter 2, page 144), we proved that

$$s_0 = 1,$$

$$s_n = 2s_{n-1}, \quad \text{if } n > 0$$

Hence, a recurrence relation for the sequence  $s_0, s_1, s_2, s_3, s_4, \dots$  is

$$s_n = 2s_{n-1}, \quad n \geq 1$$

and an initial condition is  $s_0 = 1$ .

### EXAMPLE 8.1.10

**Compound Interest.** Sam received a yearly bonus and deposited \$10,000 in a local bank yielding 7% interest compounded annually. Sam wants to know the total amount accumulated after  $n$  years. Let  $A_n$  denote the total amount accumulated after  $n$  years. Let us determine a recurrence relation and initial conditions for the sequence  $A_0, A_1, A_2, A_3, \dots$ .

The amount accumulated after one year is the initial amount plus the interest on the initial amount. Now  $A_{n-1}$  is the amount accumulated after  $n - 1$  years. This implies that the amount at the beginning of  $n$ th year is  $A_{n-1}$ . It follows that the total amount accumulated after  $n$  years is the amount at the beginning of the  $n$ th year plus the interest on this amount. Because the interest rate is 7%, the interest earned during the  $n$ th year is  $(0.07)A_{n-1}$ . Hence,

$$A_n = A_{n-1} + (0.07)A_{n-1}$$

$$= 1.07A_{n-1}, \quad n \geq 1,$$

$$A_0 = 10000.$$

Thus, we find that a recurrence relation and an initial condition for the sequence  $\{A_n\}_{n=0}^{\infty}$  are

$$A_n = 1.07A_{n-1}, \quad n \geq 1,$$

$$A_0 = 10000.$$

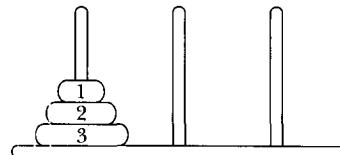
**EXAMPLE 8.1.11**

**Tower of Hanoi.** In the nineteenth century, a game called the Tower of Hanoi became popular in Europe. This game represents work that is under way in the temple of Brahma. At the creation of the universe, priests in the temple of Brahma were supposedly given three diamond pegs, with one peg containing 64 golden disks. Each golden disk is slightly smaller than the disk below it. The priests' task is to move all 64 disks from the first peg to the third peg. The rules for moving the disks are as follows:

1. Only one disk can be moved at a time.
2. The removed disk must be placed on one of the pegs.
3. A larger disk cannot be placed on top of a smaller disk.

The priests were told that once they had moved all the disks from the first peg to the third peg, the universe would come to an end.

Our objective is to determine the minimum number of moves required to transfer the disks from the first peg to the third peg. Figure 8.1 shows the Tower of Hanoi problem with three disks.



**FIGURE 8.1** Tower of Hanoi problem with three disks

Let us first consider the case in which the first peg contains only one disk. In this case, the disk can be moved directly from peg 1 to peg 3. So let us consider the case in which the first peg contains two disks. In this case, first we move the first disk from peg 1 to peg 2, and then we move the second disk from peg 1 to peg 3. Finally, we move the first disk from peg 2 to peg 3. Next, we consider the case in which the first peg contains three disks and then generalize this to the case of 64 disks (in fact, to an arbitrary number of disks).

Suppose that peg 1 contains three disks. To move disk number 3 to peg 3, the top two disks must first be moved to peg 2. Disk number 3 can then be moved from peg 1 to peg 3. To move the top two disks from peg 2 to peg 3, we use the same strategy as before. This time we use peg 1 as the intermediate peg. Figure 8.2 shows a solution to the Tower of Hanoi problem with three disks.

Let us now generalize this problem to the case of 64 disks. To begin, the first peg contains all 64 disks. Disk number 64 cannot be moved from peg 1 to peg 3 unless the top 63 disks are on the second peg. So first we move the top 63 disks from peg 1 to peg 2, and then we move disk number 64 from peg 1 to peg 3. Now the top 63 disks are all on peg 2. To move disk number 63 from peg 2

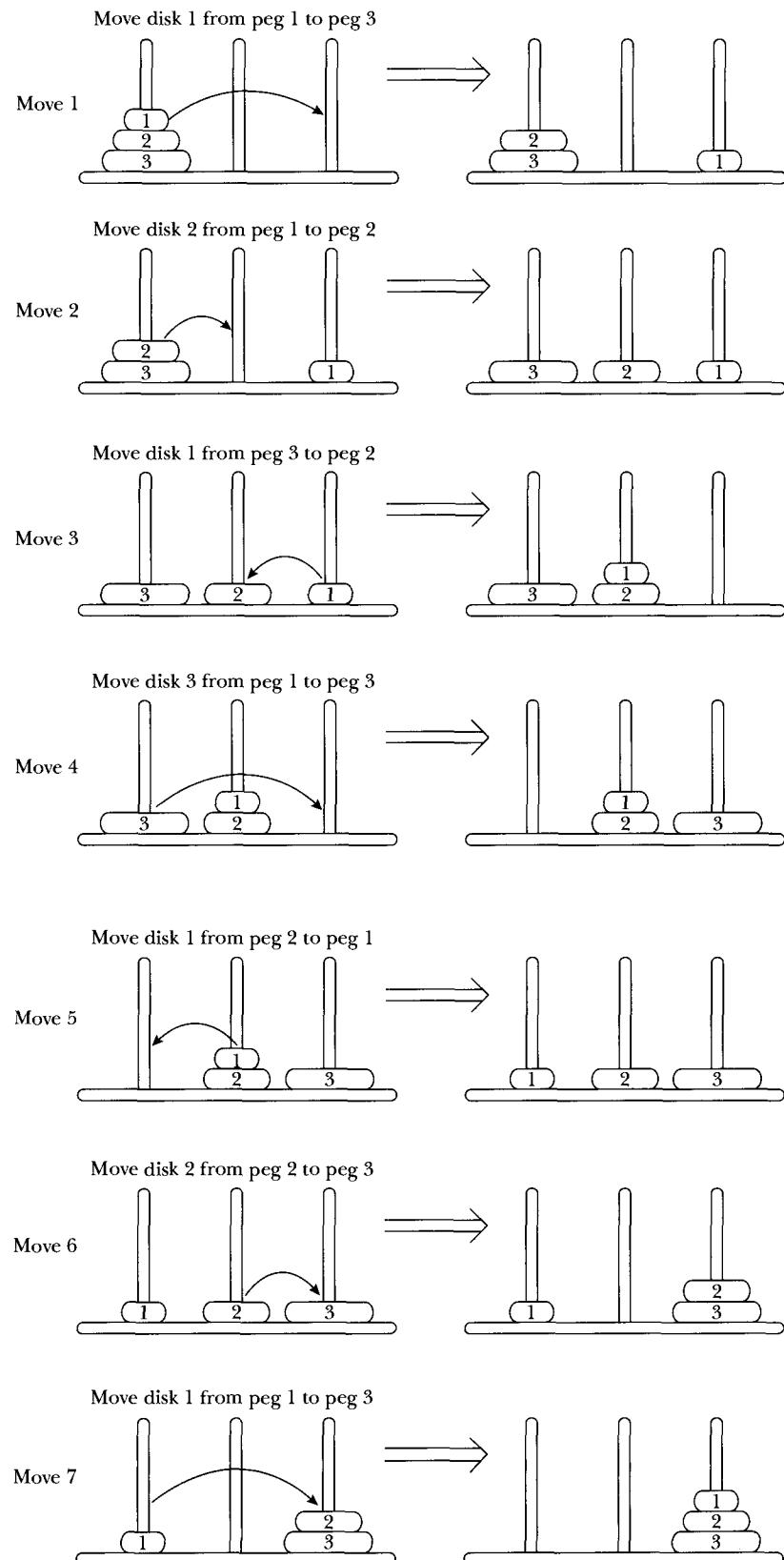


FIGURE 8.2 A solution to the Tower of Hanoi problem with three disks

to peg 3, we first move the top 62 disks from peg 2 to peg 1, and then we move disk number 63 from peg 2 to peg 3. To move the remaining 62 disks, we follow a similar procedure.

In general, let peg 1 contain  $n \geq 1$  disks.

1. Move the top  $n - 1$  disks from peg 1 to peg 2 using peg 3 as the intermediate peg.
2. Move disk number  $n$  from peg 1 to peg 3.
3. Move the top  $n - 1$  disks from peg 2 to peg 3 using peg 1 as the intermediate peg.

Let  $c_n$  denote the number of moves required to move  $n$  disks,  $n \geq 0$ , from peg 1 to peg 3. Step (1) requires us to move the top  $n - 1$  disks from peg 1 to peg 2, which requires  $c_{n-1}$  moves. Step (2) requires us to move the  $n$ th disk from peg 1 to peg 3, which requires 1 move. Step (3) requires us to move  $n - 1$  disks from peg 2 to peg 3, which requires  $c_{n-1}$  moves. Thus, it follows that

$$c_n = 2c_{n-1} + 1, \quad \text{if } n > 1, \quad (8.5)$$

and

$$c_1 = 1. \quad (8.6)$$

Now (8.5) is a recurrence relation for the sequence  $\{c_n\}_{n=1}^{\infty}$  with the initial condition given by (8.6).

### EXAMPLE 8.1.12

**Rabbits On An Island.** The following problem was posed by Leonardo Fibonacci in the thirteenth century in his book *Liber abaci*.

A pair of newborn rabbits (one male and one female) is kept on an island where there are no other rabbits. A pair of rabbits does not breed until they are two months old. After a pair becomes two months old, each pair of rabbits (of opposite sexes) produces another pair (of opposite sexes) each month. Assuming that no rabbits ever die, find a recurrence relation for the number of pairs of rabbits on the island just after  $n$  months.

Let  $a_n$  denote the number of pairs of rabbits on the island just after  $n$  months. At the end of the first month, the number of pairs of rabbits on the island is

$$a_1 = 1.$$

This pair of rabbits does not breed during the second month. Thus, the number of pairs of rabbits just after the second month is

$$a_2 = 1.$$

Now the number of pairs of rabbits,  $a_n$ , just after  $n$  months is the number of pairs after  $n - 1$  months plus the number of newborn pairs in the  $n$ th month. The number of newborn pairs in the  $n$ th month is the number of pairs just after the  $(n - 2)$ th month because each newborn pair is produced by a pair of rabbits at least two months old. Hence,

$$a_n = a_{n-1} + a_{n-2}, \quad n \geq 3, \quad (8.7)$$

which is a recurrence relation. The initial conditions are  $a_1 = 1$  and  $a_2 = 1$ . Now,

$$\begin{aligned}a_3 &= a_2 + a_1 = 1 + 1 = 2, \\a_4 &= a_3 + a_2 = 2 + 1 = 3, \\a_5 &= a_4 + a_3 = 3 + 2 = 5,\end{aligned}$$

and so on. We see that the sequence defined by (8.7) is: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ... .

## Solving Recurrence Relations by Iteration (Substitution)

In the preceding section, we saw various examples of recurrence relations. Our goal in solving a recurrence relation is to find an explicit formula for the general term  $a_n$  of the recurrence relation. In this section, we describe how to find an explicit formula by iteration, or substitution.

---

**DEFINITION 8.1.13** ► Suppose a recurrence relation for a sequence  $a_0, a_1, a_2, \dots, a_n, \dots$ , is given. By a *solution of the recurrence relation* we mean to obtain an explicit formula for  $a_n$ , i.e., to find an expression for  $a_n$  that does not involve any other  $a_i$ .

The following example clarifies Definition 8.1.13.

### EXAMPLE 8.1.14

Let  $S$  be the sequence  $\{a_n\}_{n=0}^{\infty}$ , where

$$a_n = 7a_{n-1} - 6a_{n-2} \quad \text{for all } n \geq 2. \quad (8.8)$$

Because  $a_n$  is defined in terms of the preceding terms  $a_{n-1}$  and  $a_{n-2}$ , Equation (8.8) is a recurrence relation.

Let us show that  $a_n = 5 + 0 \cdot n$  is a solution of Equation (8.8). Here  $a_0 = 5$ ,  $a_1 = 5$ ,  $a_2 = 5, \dots, a_n = 5$ , and so on. Let us evaluate the right side of Equation (8.8), i.e.,

$$7a_{n-1} - 6a_{n-2} = 7 \cdot 5 - 6 \cdot 5 = 35 - 30 = 5 = a_n.$$

Hence,  $a_n = 5, n \geq 0$  is a solution of the recurrence relation (8.8).

Now let  $a_n = 6^n$ . Here  $a_0 = 6^0 = 1$ ,  $a_1 = 6^1 = 6$ ,  $a_2 = 6^2 = 36, \dots, a_{n-2} = 6^{n-2}$ ,  $a_{n-1} = 6^{n-1}$ ,  $a_n = 6^n$ , and so on. Let us evaluate the right side of Equation (8.8), using the terms of this sequence. We have

$$\begin{aligned}7a_{n-1} - 6a_{n-2} &= 7 \cdot 6^{n-1} - 6 \cdot 6^{n-2} \\&= 7 \cdot 6^{n-1} - 6^{n-1} \\&= (7 - 1) \cdot 6^{n-1} \\&= 6 \cdot 6^{n-1} \\&= 6^n \\&= a_n.\end{aligned}$$

Therefore,  $a_n = 6^n, n \geq 0$  is also a solution of the recurrence relation (8.8).

Note that the expression  $a_n = 2^n, n \geq 0$  is not a solution of Equation (8.8).

**REMARK 8.1.15** ▶ Notice that in Example 8.1.14, we determined two solutions of a recurrence relation. This is due to the fact that we did not specify any initial conditions in Example 8.1.14.

**EXAMPLE 8.1.16**

Let  $S : a_1, a_2, \dots, a_n, \dots$  be a sequence defined by

$$a_1 = 4, \quad (8.9)$$

$$a_n = a_{n-1} + 7, \quad \text{if } n > 1. \quad (8.10)$$

Equation (8.9) specifies the initial condition and Equation (8.10) specifies the recurrence relation.

Let us determine  $a_5$ . Here  $n = 5 > 1$ , so we use Equation (8.10). Thus,

$$a_5 = a_4 + 7.$$

This requires us to determine  $a_4$ . Here  $n = 4 > 1$ , so we use Equation (8.10). Thus,

$$a_4 = a_3 + 7,$$

which in turn requires us to determine  $a_3$ . Here  $n = 3 > 1$ , so we again use Equation (8.10). Hence,

$$a_3 = a_2 + 7,$$

which in turn requires us to determine  $a_2$ . Here  $n = 2 > 1$ , so we again use Equation (8.10). Hence,

$$a_2 = a_1 + 7.$$

We can now substitute the value of  $a_1$  and obtain  $a_2 = 4 + 7 = 11$ . Next we use the value of  $a_2$  and obtain  $a_3 = a_2 + 7 = 11 + 7 = 18$ . This gives us  $a_4 = a_3 + 7 = 18 + 7 = 25$ . Finally,  $a_5 = a_4 + 7 = 25 + 7 = 32$ .

This shows that by using Equations (8.9) and (8.10), we can determine any term of the sequence. For example, suppose that we want to determine  $a_8$ . We can determine  $a_8$  as follows: By repeatedly using Equation (8.10), we have

$$\begin{aligned} a_8 &= a_7 + 7 \\ &= a_6 + 7 + 7 \\ &= a_5 + 7 + 7 + 7 \\ &= a_4 + 7 + 7 + 7 + 7 \\ &= a_3 + 7 + 7 + 7 + 7 + 7 \\ &= a_2 + 7 + 7 + 7 + 7 + 7 + 7 \\ &= a_1 + 7 + 7 + 7 + 7 + 7 + 7 + 7 \\ &= 4 + 7 + 7 + 7 + 7 + 7 + 7 + 7 \\ &= 53. \end{aligned}$$

Given a recurrence relation with initial conditions, we can determine any term of the sequence by using techniques similar to those shown in Example 8.1.16. The problem with this is that it requires us to apply the recurrence relations repeatedly until we arrive at one of the initial conditions. If we could determine a formula for  $a_n$ , the  $n$ th term of the sequence, that does not involve any other  $a_i$ , then we could determine any term of the sequence without knowing the values of other

terms of the sequence. In this section and the next, we present various ways of determining a formula for  $a_n$  that does not involve any other  $a_i$ .

The technique used to determine  $a_8$  and  $a_5$  in Example 8.1.16 is called the **iteration, or substitution, method**. In the next few examples, we determine an explicit formula for  $a_n$  using the iteration technique.

**EXAMPLE 8.1.17**

Consider the recurrence relation of Example 8.1.16, i.e.,

$$a_n = a_{n-1} + 7, \quad \text{if } n > 1, \quad (8.11)$$

with the initial condition,

$$a_1 = 4. \quad (8.12)$$

In (8.11), replace  $n$  by  $n - 1$  to obtain

$$a_{n-1} = a_{n-2} + 7.$$

Substitute  $a_{n-1}$  into  $a_n$  to obtain

$$\begin{aligned} a_n &= a_{n-1} + 7 \\ &= (a_{n-2} + 7) + 7 \\ &= a_{n-2} + 2 \cdot 7. \end{aligned}$$

Next, in (8.11), replace  $n$  by  $n - 2$  to obtain

$$a_{n-2} = a_{n-3} + 7.$$

Substitute  $a_{n-2}$  into  $a_n$  to obtain

$$\begin{aligned} a_n &= a_{n-2} + 2 \cdot 7 \\ &= (a_{n-3} + 7) + 2 \cdot 7 \\ &= a_{n-3} + 3 \cdot 7. \end{aligned}$$

In general, we have

$$a_n = a_{n-k} + k \cdot 7, \quad k = 1, 2, 3, \dots, n-1. \quad (8.13)$$

In (8.13), substitute  $k = n - 1$  to obtain

$$\begin{aligned} a_n &= a_{n-(n-1)} + (n-1) \cdot 7 \\ &= a_1 + (n-1) \cdot 7. \end{aligned}$$

Next, substitute  $a_1 = 4$  to obtain

$$\begin{aligned} a_n &= a_1 + (n-1) \cdot 7 \\ &= 4 + (n-1) \cdot 7 \\ &= 7(n-1) + 4 \\ &= 7n - 3 \end{aligned} \quad (8.14)$$

for all  $n \geq 1$ , which gives the explicit formula for  $a_n$ .

It seems that the explicit formula given by (8.14) is correct. However, to be certain, we must verify its correctness, which we can do by induction.

So using induction we verify that

$$a_n = 7n - 3 \quad \text{for all } n \geq 1. \quad (8.15)$$

*Basis step:* Let  $n = 1$ . Then  $a_n = a_1 = 7 \cdot 1 - 3 = 7 - 3 = 4$ . Thus, the result is true for  $n = 1$ .

*Inductive hypothesis:* Assume that (8.15) is true for  $n = k \geq 1$ , i.e.,

$$a_k = 7k - 3.$$

*Inductive step:* Let  $n = k + 1$ . We have

$$\begin{aligned} a_{k+1} &= a_k + 7 && \text{by (8.11)} \\ &= (7k - 3) + 7 && \text{by the inductive hypothesis, } a_k = 7k - 3 \\ &= 7k + 7 - 3 \\ &= 7(k + 1) - 3. \end{aligned}$$

Thus, (8.15) is true for  $n = k + 1$ . Hence, (8.15) is true for all  $n \geq 1$ .

### EXAMPLE 8.1.18

In this example, we find an explicit formula for the sequence  $S$  that begins with the following terms:  $1, 3, 6, 10, 15, \dots$

In this sequence, notice that  $a_1 = 1$ ,  $a_2 = 3 = a_1 + 2$ ,  $a_3 = 6 = a_2 + 3$ ,  $a_4 = 10 = a_3 + 4$ , and so on. In general  $a_n = a_{n-1} + n$ . Thus, the sequence  $S$  can be defined by the recurrence relation

$$a_n = a_{n-1} + n, \quad \text{if } n > 1 \tag{8.16}$$

with the initial condition

$$a_1 = 1. \tag{8.17}$$

In (8.16), replace  $n$  by  $n - 1$  to obtain

$$a_{n-1} = a_{n-2} + (n - 1).$$

Substitute  $a_{n-1}$  into  $a_n$  to obtain

$$\begin{aligned} a_n &= a_{n-1} + n \\ &= (a_{n-2} + (n - 1)) + n \\ &= a_{n-2} + (n - 1) + n. \end{aligned}$$

Next, in (8.16), replace  $n$  by  $n - 2$  to obtain

$$a_{n-2} = a_{n-3} + (n - 2).$$

Substitute  $a_{n-2}$  into  $a_n$  to obtain

$$\begin{aligned} a_n &= a_{n-2} + (n - 1) + n \\ &= (a_{n-3} + (n - 2)) + (n - 1) + n \\ &= a_{n-3} + (n - 2) + (n - 1) + n. \end{aligned}$$

In general, we have

$$a_n = a_{n-k} + (n - k + 1) + \cdots + (n - 2) + (n - 1) + n, \quad k = 1, 2, 3, \dots, n - 1. \tag{8.18}$$

In (8.18), substitute  $k = n - 1$  to obtain

$$\begin{aligned} a_n &= a_{n-(n-1)} + (n - (n - 1) + 1) + \cdots + (n - 2) + (n - 1) + n \\ &= a_1 + 2 + \cdots + (n - 2) + (n - 1) + n. \end{aligned}$$

Next, substitute  $a_1 = 1$  to obtain

$$\begin{aligned} a_n &= 1 + 2 + \cdots + (n-2) + (n-1) + n \\ &= \frac{n(n+1)}{2} \end{aligned} \quad (8.19)$$

for all  $n \geq 1$ , which gives the explicit formula for  $a_n$ . In Section 2.3, we verified that

$$1 + 2 + \cdots + (n-2) + (n-1) + n = \frac{n(n+1)}{2}.$$

We leave it as an exercise to verify that (8.19) gives the explicit formula for the recurrence relation (8.16) with the initial condition given by (8.17).

### EXAMPLE 8.1.19

In this example, we show how to solve the recurrence relation given in Example 8.1.11, i.e.,

$$c_n = 2c_{n-1} + 1, \quad \text{if } n > 1, \quad (8.20)$$

and

$$c_1 = 1 \quad (8.21)$$

using iteration. Now (8.20) is a recurrence relation for the sequence  $\{c_n\}_{n=1}^{\infty}$ .

In (8.20), replace  $n$  by  $n-1$  to obtain

$$c_{n-1} = 2c_{n-2} + 1.$$

Substitute  $c_{n-1}$  into  $c_n$  to obtain

$$c_n = 2(2c_{n-2} + 1) + 1 = 2^2 c_{n-2} + 2 + 1.$$

In (8.20), replace  $n$  by  $n-2$  to obtain

$$c_{n-2} = 2c_{n-3} + 1.$$

Substitute  $c_{n-2}$  into  $c_n$  to obtain

$$c_n = 2^2(2c_{n-3} + 1) + 2 + 1 = 2^3 c_{n-3} + 2^2 + 2 + 1.$$

Apply the substitution, i.e., iterative, method to obtain

$$\begin{aligned} c_n &= 2c_{n-1} + 1 \\ &= 2^2 c_{n-2} + 2 + 1 \\ &= 2^3 c_{n-3} + 2^2 + 2 + 1 \\ &\vdots \\ &= 2^{n-k} c_{n-k} + 2^{n-k-1} + \cdots + 2^2 + 2 + 1 \\ &\vdots \\ &= 2^{n-1} c_1 + 2^{n-2} + 2^{n-3} + \cdots + 2^2 + 2 + 1 \\ &= 2^{n-1} + 2^{n-2} + 2^{n-3} + \cdots + 2^2 + 2 + 1 \quad \text{because } c_1 = 1 \\ &= 2^n - 1 \quad \text{by Worked-Out Exercise 2, page 143, Section 2.3} \end{aligned}$$

Hence, the explicit formula for  $c_n$  is

$$c_n = 2^n - 1, \quad n \geq 1. \quad (8.22)$$

We leave it as an exercise to show by induction that  $c_n$ , given by (8.22), is a solution of the recurrence relation (8.20) with the initial condition given by (8.21).

Next we show that the minimum number of moves required, to transfer  $n$  disks from peg 1 to peg 3 is given by (8.22), i.e.,  $2^n - 1$ .

Let  $d_n$  be the minimum number of moves needed to solve the puzzle. By induction, we prove that

$$c_n = d_n \quad \text{for all } n \geq 1. \quad (8.23)$$

*Basis step:* Let  $n = 1$ . Then there is only one disk on peg 1. It follows that the minimum number of moves required to move this disk is 1. Thus,  $d_1 = 1$ . Also  $c_1 = 2^1 - 1 = 2 - 1 = 1$ . Hence,  $c_1 = d_1$ .

*Inductive hypothesis:* Suppose that the result is true for  $n = k \geq 1$ , i.e.,  $c_k = d_k$ .

*Inductive step:* Let  $n = k + 1$ . Now  $d_{k+1}$  is the minimum number of moves required to solve the  $(k + 1)$ -disk puzzle. In an optimal solution, to move disk number  $k + 1$  from peg 1 to peg 3, the top  $k$  disks must be on peg 2 and peg 3 must be empty. The minimum number of moves required to move  $k$  disks from one peg to another peg is  $d_k$ . One move is required to move the disk number  $k + 1$  from peg 1 to peg 3. After moving the disk number  $k + 1$  from peg 1 to peg 3, we move the  $k$  disks from peg 2 to peg 3. Hence, we have

$$\begin{aligned} d_{k+1} &\geq 2d_k + 1 \\ &= 2c_k + 1 \quad \text{by the inductive hypothesis, } c_k = d_k \\ &= c_{k+1}. \end{aligned} \quad (8.24)$$

By definition, the minimum number of moves required to solve the  $(k + 1)$ -disk puzzle is  $d_{k+1}$ , so

$$c_{k+1} \geq d_{k+1}. \quad (8.25)$$

Hence, by (8.24) and (8.25), we have

$$c_{k+1} = d_{k+1}.$$

Thus, the result is true for  $n = k + 1$ . Consequently, by induction,  $c_n = d_n$  for all  $n \geq 1$ .

Next let us determine how long it would take to move 64 disks from peg 1 to peg 3.

If peg 1 contains 64 disks, then the number of moves required to move all 64 disks from peg 1 to peg 3 is  $2^{64} - 1$ . Because

$$2^{10} = 1024 \approx 1000 = 10^3,$$

we have

$$2^{64} = 2^4 \cdot 2^{60} \approx 2^4 \cdot 10^{18} = 1.6 \cdot 10^{19}.$$

The number of seconds in one year is approximately  $3.2 \cdot 10^7$ . Suppose that the priests move one disk per second and they do not rest. Now

$$1.6 \cdot 10^{19} = 5 \cdot 3.2 \cdot 10^{18} = 5 \cdot (3.2 \cdot 10^7) \cdot 10^{11} = (3.2 \cdot 10^7) \cdot (5 \cdot 10^{11}).$$

The time required to move all 64 disks from peg 1 to peg 3 is roughly  $5 \cdot 10^{11}$  years. It is estimated that our universe is about 15 billion ( $= 1.5 \cdot 10^{10}$ ) years old. Also,

$$5 \cdot 10^{11} = 50 \cdot 10^{10} \approx 33 \cdot (1.5 \cdot 10^{10}).$$

This calculation shows that our universe will last about 33 times as long as it already has.

Assume that a computer can generate 1 billion =  $10^9$  moves per second. Then the number of moves that the computer can generate in one year is

$$(3.2 \cdot 10^7) \cdot 10^9 = 3.2 \cdot 10^{16}.$$

So the computer time required to generate  $2^{64}$  moves is

$$2^{64} \approx 1.6 \cdot 10^{19} = 1.6 \cdot 10^{16} \cdot 10^3 = (3.2 \cdot 10^{16}) \cdot 500.$$

Thus, it would take about 500 years for the computer to generate  $2^{64}$  moves at the rate of 1 billion moves per second.

### EXAMPLE 8.1.20

Suppose there are  $2n$  ( $n > 0$ ) students  $a_1, a_2, \dots, a_{2n}$  in a class. Let  $P_n$  denote the number of ways to group these students into pairs. For example, if  $n = 1$ , then there are 2 students  $a_1, a_2$  and the only pair is  $\{a_1, a_2\}$ . Hence,  $P_1 = 1$ . If  $n = 2$ , then there are 4 students  $a_1, a_2, a_3, a_4$ . The possible groupings of pairs are

- Grouping 1 :  $\{a_1, a_2\}, \{a_3, a_4\}$ ;
- Grouping 2 :  $\{a_1, a_3\}, \{a_2, a_4\}$ ;
- Grouping 3 :  $\{a_1, a_4\}, \{a_2, a_3\}$ .

Thus,  $P_2 = 3$ .

If  $n = 3$ , then there are 6 students  $a_1, a_2, a_3, a_4, a_5, a_6$ . The possible groupings of pairs are

- Grouping 1 :  $\{a_1, a_2\}, \{a_3, a_4\}, \{a_5, a_6\}$ ;
- Grouping 2 :  $\{a_1, a_3\}, \{a_2, a_4\}, \{a_5, a_6\}$ ;
- Grouping 3 :  $\{a_1, a_2\}, \{a_3, a_5\}, \{a_4, a_6\}$ ;
- Grouping 4 :  $\{a_1, a_3\}, \{a_2, a_5\}, \{a_4, a_6\}$ ;
- Grouping 5 :  $\{a_1, a_2\}, \{a_3, a_6\}, \{a_4, a_5\}$ ;
- Grouping 6 :  $\{a_1, a_3\}, \{a_2, a_6\}, \{a_4, a_5\}$ ;
- Grouping 7 :  $\{a_1, a_4\}, \{a_3, a_2\}, \{a_5, a_6\}$ ;
- Grouping 8 :  $\{a_1, a_5\}, \{a_2, a_3\}, \{a_4, a_6\}$ ;
- Grouping 9 :  $\{a_1, a_4\}, \{a_3, a_5\}, \{a_2, a_6\}$ ;
- Grouping 10 :  $\{a_1, a_5\}, \{a_2, a_4\}, \{a_3, a_6\}$ ;
- Grouping 11 :  $\{a_1, a_4\}, \{a_3, a_6\}, \{a_2, a_5\}$ ;
- Grouping 12 :  $\{a_1, a_5\}, \{a_2, a_6\}, \{a_3, a_4\}$ ;
- Grouping 13 :  $\{a_1, a_6\}, \{a_3, a_4\}, \{a_2, a_5\}$ ;
- Grouping 14 :  $\{a_1, a_6\}, \{a_3, a_5\}, \{a_2, a_4\}$ ;
- Grouping 15 :  $\{a_1, a_6\}, \{a_3, a_2\}, \{a_4, a_5\}$ .

Thus,  $P_3 = 15$ .

We now obtain a recurrence relation and the initial conditions for the sequence  $P_1, P_2, \dots, P_n, \dots$

Consider student  $a_1$ . Then choose another student,  $b$ , from the remaining students to make the pair  $\{a_1, b\}$ . Because the number of remaining students is

$2n - 1$ , the first pair  $\{a_1, b\}$  can be constructed in  $2n - 1$  different ways. After making one of these pairs with  $a_1$ , the number of remaining students is  $2(n - 1)$ . Now the number of ways to group these  $2(n - 1)$  students into pairs is  $P_{n-1}$ . Hence, by the multiplication principle, the number of ways to group the  $2n$  students into pairs is  $(2n - 1)P_{n-1}$ . Therefore, the recurrence relation and initial condition for the sequence are

$$\begin{aligned} P_n &= (2n - 1)P_{n-1}, \quad n > 1, \\ P_1 &= 1. \end{aligned} \tag{8.26}$$

Let us determine  $P_2$  and  $P_3$  using this recurrence relation. Now

$$P_2 = (2 \cdot 2 - 1)P_1 = 3 \cdot 1 = 3$$

and

$$P_3 = (2 \cdot 3 - 1)P_2 = 5 \cdot 3 = 15.$$

We now solve the recurrence relation given in (8.26). Change  $n$  to  $n - 1$  in (8.26) to get

$$\begin{aligned} P_{n-1} &= (2(n - 1) - 1)P_{(n-1)-1} \\ &= (2n - 3)P_{(n-2)}. \end{aligned}$$

Thus,

$$P_n = (2n - 1)(2n - 3)P_{(n-2)}.$$

Change  $n$  to  $n - 2$  in (8.26) to get

$$P_{(n-2)} = (2(n - 2) - 1)P_{(n-2)-1} = (2n - 5)P_{(n-3)}.$$

Thus,

$$\begin{aligned} P_n &= (2n - 1)(2n - 3)P_{(n-2)} \\ &= (2n - 1)(2n - 3)(2n - 5)P_{(n-3)} \\ &\quad \vdots \\ &= (2n - 1)(2n - 3)(2n - 5) \cdots 3P_1 \\ &= (2n - 1)(2n - 3)(2n - 5) \cdots 3 \cdot 1 \\ &= \frac{2n(2n - 1)(2n - 2)(2n - 3)(2n - 4)(2n - 5) \cdots 3 \cdot 2 \cdot 1}{2n(2n - 2)(2n - 4)(2n - 6) \cdots 2} \\ &= \frac{(2n)!}{2^n n!}, \quad n \geq 1. \end{aligned}$$

We can verify by induction that  $P_n = \frac{(2n)!}{2^n n!}$ ,  $n \geq 1$ .



## WORKED-OUT EXERCISES

**Exercise 1:** Find the first five terms of a sequence  $a_0, a_1, a_2, \dots, a_n, \dots$  satisfying the given recurrence relation and initial conditions.

- (a)  $a_n = a_{n-1} + 5$  if  $n \geq 1$ ,  $a_0 = 5$
- (b)  $a_n = a_{n-1} + n$  if  $n \geq 1$ ,  $a_0 = 5$
- (c)  $a_n = 7a_{n-1} + 3a_{n-2} + 5$  if  $n \geq 2$ ,  $a_0 = 5$  and  $a_1 = 2$

**Solution:**

(a) The first five terms of the sequence are

$$\begin{aligned}a_0 &= 5, \\a_1 &= a_0 + 5 = 5 + 5 = 10, \\a_2 &= a_1 + 5 = 10 + 5 = 15, \\a_3 &= a_2 + 5 = 15 + 5 = 20, \\a_4 &= a_3 + 5 = 20 + 5 = 25.\end{aligned}$$

(b) The first five terms of the sequence are

$$\begin{aligned}a_0 &= 5, \\a_1 &= a_0 + 1 = 5 + 1 = 6, \\a_2 &= a_1 + 2 = 6 + 2 = 8, \\a_3 &= a_2 + 3 = 8 + 3 = 11, \\a_4 &= a_3 + 4 = 11 + 4 = 15.\end{aligned}$$

(c) The first five terms of the sequence are

$$\begin{aligned}a_0 &= 5 \\a_1 &= 2, \\a_2 &= 7a_1 + 3a_0 + 5 = 7 \cdot 2 + 3 \cdot 5 + 5 = 34, \\a_3 &= 7a_2 + 3a_1 + 5 = 7 \cdot 34 + 3 \cdot 2 + 5 = 249, \\a_4 &= 7a_3 + 3a_2 + 5 = 7 \cdot 249 + 3 \cdot 34 + 5 = 1850.\end{aligned}$$

**Exercise 2:** Find recurrence relation and initial conditions for the sequence  $S : 1, 5, 13, 29, 61, \dots$ **Solution:** In the sequence, notice that  $a_1 = 1$ ,  $a_2 = 5 = 2a_1 + 3$ ,  $a_3 = 13 = 2a_2 + 3$ ,  $a_4 = 29 = 2a_3 + 3$ , and so on. In general,  $a_n = a_{n-1} + 3$ . Thus, we find that recurrence relation and initial conditions for the sequence  $S$  are given by

$$\begin{aligned}a_n &= 2a_{n-1} + 3, \quad \text{if } n > 1 \\a_1 &= 1.\end{aligned}$$

**Exercise 3:** Find the recurrence relation and initial conditions for the sequence  $S : 0, 2, 8, 26, 80, \dots, 3^n - 1, \dots$ **Solution:** In the sequence  $S$ , notice that  $a_0 = 0$ ,  $a_1 = 2 = 3 \cdot a_0 + 2$ ,  $a_2 = 8 = 3 \cdot a_1 + 2$ ,  $a_3 = 26 = 3 \cdot a_2 + 2$ ,  $a_4 = 80 = 3 \cdot a_3 + 2$ , and so on. In general,

$$a_n = 3^n - 1 = 3 \cdot (3^{n-1} - 1) + 2 = 3 \cdot a_{n-1} + 2.$$

Thus we find that recurrence relation and initial conditions for the sequence  $S$  are given by

$$\begin{aligned}a_n &= 3 \cdot a_{n-1} + 2, \quad \text{if } n > 1, \\a_0 &= 0.\end{aligned}$$

**Exercise 4:** Carmen and Andrew are purchasing a new vacation house costing \$300,000 with a down payment of \$60,000 and a long-term mortgage. The unpaid balance is being financed at a monthly rate of 1.2% on the unpaid balance and a payment of \$2000 per month. Find a recurrence relation and an initial condition to determine the unpaid balance after  $n$  monthly payments.**Solution:** Let  $a_n$  denote the unpaid balance after  $n$  payments. Note that the unpaid balance after  $n$  payments is the unpaid balance after  $n - 1$  payments plus the monthly interest on it minus one payment.

$$a_n = a_{n-1} + \frac{1.2}{100}a_{n-1} - 2000.$$

That is,

$$\begin{aligned}a_n &= a_{n-1} + 0.012a_{n-1} - 2000 \\&= 1.012a_{n-1} - 2000, \quad n \geq 1.\end{aligned}$$

The initial unpaid balance is the cost of the house minus the down payment, i.e., \$240,000. Thus, the initial condition is

$$a_0 = 240,000.$$

**Exercise 5:** During allergy season Kevin needs to take an allergy tablet containing 25 mg of a drug each morning. During the day 20% of the amount of the drug is eliminated.

- (a) Write a recurrence relation, with initial condition(s), describing the amount of the drug in Kevin's body immediately after he takes the  $n$ th tablet.
- (b) How much of the drug is in Kevin's body immediately after he takes the 9th tablet?
- (c) What quantity of the drug will eventually accumulate in Kevin's body?

**Solution:**

- (a) Let  $a_n$  be the amount of the drug in Kevin's body immediately after he takes the  $n$ th tablet. Observe that just before he takes the  $n$ th tablet, the remaining amount of the drug in the body is 80% of  $a_{n-1}$ , where  $a_{n-1}$  is the amount of the drug in the body immediately after taking the  $(n - 1)$ th tablet. Thus, it follows that

$$a_n = 0.8a_{n-1} + 25, \quad n \geq 1. \quad (8.27)$$

Before taking any tablet, the amount of drug in the body,  $a_0$ , is 0; i.e., the initial condition is  $a_0 = 0$ . We claim that the general solution of this recurrence relation is:

$$a_n = 125(1 - (0.8)^n), \quad n \geq 0.$$

We verify this using induction.

*Basis step:* Let  $n = 0$ , then  $a_0 = 125(1 - (0.8)^0) = 125(1 - 1) = 0$ . Hence, the result is true for  $n = 0$ .*Inductive hypothesis:* Suppose the result is true for  $n = k$ , i.e.,

$$a_k = 125(1 - (0.8)^k),$$

for some  $k \geq 0$ .

*Inductive step:* Let  $n = k + 1$ . Now

$$\begin{aligned} a_{k+1} &= 0.8a_k + 25, \\ &= 0.8(125(1 - (0.8)^k)) + 25 \\ &= 0.8 \cdot 125 - 0.8 \cdot 125(0.8)^k + 25 \\ &= 100 - 125(0.8)^{k+1} + 25 \\ &= 125 - 125(0.8)^{k+1} \\ &= 125(1 - (0.8)^{k+1}). \end{aligned}$$

Thus, the result is true for  $n = k + 1$ . Hence, by induction, we have

$$a_n = 125(1 - (0.8)^n), \quad n \geq 0.$$

(In Worked-Out Exercise 1, page 541, we will show another way to obtain the general solution of (8.27).)

- (b) The amount of the drug in Kevin's body just after he takes the 9th tablet is:

$$a_9 = 125(1 - (0.8)^9) = 125 \cdot 0.865782272 = 108.22.$$

Hence, the amount of the drug just after taking the 9th tablet is 108.22 mg.

- (c) Now  $a_n = 125(1 - (0.8)^n)$ . As  $n$  increases, the quantity  $(0.8)^n$  decreases to 0. This implies that when  $n$  is very large,  $a_n$  approaches 125. Thus 125 mg of the drug will be eventually accumulated in Kevin's body.

**Exercise 6:** The  $n$ th term  $a_n$  of the sequence  $a_1, a_2, \dots, a_n, \dots$  satisfies the recurrence relation

$$a_n = 7a_{n-1} - 12a_{n-2} + 6, \quad n \geq 3,$$

with initial conditions  $a_1 = 2$  and  $a_2 = 8$ . Prove that

$$a_n = 4^n - 3^n + 1, \quad n \geq 1.$$

**Solution:** We prove the result by induction on  $n$ .

*Basis Step:* If  $n = 1, 2$ , then using the initial conditions, we have

$$\begin{aligned} a_1 &= 2 = 4^1 - 3^1 + 1, \\ a_2 &= 8 = 4^2 - 3^2 + 1. \end{aligned}$$

This implies that the result is true for  $n = 1$  and 2.

*Inductive hypothesis:* Suppose that the result is true for all positive integers  $n = 1, 2, \dots, k, k \geq 2$ .

*Inductive step:* Let  $n = k + 1$ . Because  $k + 1 \geq 3$ ,

$$a_{k+1} = 7a_k - 12a_{k-1} + 6.$$

By the inductive hypothesis,  $a_k = 4^k - 3^k + 1$  and  $a_{k-1} = 4^{k-1} - 3^{k-1} + 1$ . This implies that

$$\begin{aligned} a_{k+1} &= 7a_k - 12a_{k-1} + 6. \\ &= 7(4^k - 3^k + 1) - 12(4^{k-1} - 3^{k-1} + 1) + 6 \\ &= 7 \cdot 4^k - 7 \cdot 3^k + 7 - 12 \cdot 4^{k-1} + 12 \cdot 3^{k-1} - 12 + 6 \\ &= 7 \cdot 4^k - 12 \cdot 4^{k-1} - 7 \cdot 3^k + 12 \cdot 3^{k-1} + 7 - 12 + 6 \\ &= 7 \cdot 4^k - 3 \cdot 4 \cdot 4^{k-1} - 7 \cdot 3^k + 4 \cdot 3 \cdot 3^{k-1} + 1 \\ &= 7 \cdot 4^k - 3 \cdot 4^k - 7 \cdot 3^k + 4 \cdot 3^k + 1 \\ &= (7 - 3)4^k - (7 - 4)3^k + 1 \\ &= 4 \cdot 4^k - 3 \cdot 3^k + 1 \\ &= 4^{k+1} - 3^{k+1} + 1. \end{aligned}$$

Thus, the result is true for  $n = k + 1$ . The result now follows by induction, i.e.,

$$a_n = 4^n - 3^n + 1 \quad \text{for all } n \geq 1.$$

**Exercise 7:** Use iterations to find the explicit formula for the sequence  $\{b_n\}_{n=1}^{\infty}$ , where  $b_n$  is defined by

$$b_n = 5b_{n-1} + 3, \quad n \geq 2,$$

with the initial condition  $b_1 = 2$ .

**Solution:** We have

$$\begin{aligned} b_n &= 5b_{n-1} + 3 \\ &= 5(5b_{n-2} + 3) + 3 && \text{because } b_{n-1} = 5b_{n-2} + 3 \\ &= 5 \cdot 5b_{n-2} + 5 \cdot 3 + 3 \\ &= 5^2 b_{n-2} + 5 \cdot 3 + 3 \\ &= 5^2(5b_{n-3} + 3) + 5 \cdot 3 + 3 && \text{because } b_{n-2} = 5b_{n-3} + 3 \\ &= 5^2 \cdot 5b_{n-3} + 5^2 \cdot 3 + 5 \cdot 3 + 3 \\ &= 5^3 b_{n-3} + 5^2 \cdot 3 + 5 \cdot 3 + 3. \end{aligned}$$

In general,

$$b_n = 5^k b_{n-k} + 5^{k-1} \cdot 3 + 5^{k-2} \cdot 3 + \cdots + 5 \cdot 3 + 3.$$

Substitute  $k = n - 1$ . We have

$$\begin{aligned} b_n &= 5^{n-1} b_{n-(n-1)} + 5^{n-2} \cdot 3 + 5^{n-3} \cdot 3 + \cdots + 5 \cdot 3 + 3 \\ &= 5^{n-1} b_1 + 5^{n-2} \cdot 3 + 5^{n-3} \cdot 3 + \cdots + 5 \cdot 3 + 3 \\ &= 5^{n-1} b_1 + 3(5^{n-2} + 5^{n-3} + \cdots + 5 + 1) \\ &= 2 \cdot 5^{n-1} + 3(5^{n-2} + 5^{n-3} + \cdots + 5 + 1). \quad \text{because } b_1 = 2 \end{aligned}$$

By induction, we can show that  $5^{n-2} + 5^{n-3} + \cdots + 5 + 1 = \frac{5^{n-1}-1}{4}$ . Hence,

$$\begin{aligned} b_n &= 2 \cdot 5^{n-1} + 3 \left( \frac{5^{n-1}-1}{4} \right) \\ &= 2 \cdot 5^{n-1} + \frac{3}{4} 5^{n-1} - \frac{3}{4} \\ &= \frac{11}{4} 5^{n-1} - \frac{3}{4} \\ &= \frac{1}{4}(11 \cdot 5^{n-1} - 3), \quad n \geq 1. \end{aligned}$$

Next, using induction we verify that

$$b_n = \frac{1}{4}(11 \cdot 5^{n-1} - 3), \quad n \geq 1.$$

*Basic step:* Let  $n = 1$ . Then

$$\begin{aligned} b_1 &= \frac{1}{4}(11 \cdot 5^{1-1} - 3) \\ &= \frac{1}{4}(11 \cdot 5^0 - 3) = \frac{1}{4}(11 - 3) = \frac{8}{4} = 2. \end{aligned}$$

Thus, the result is true for  $n = 1$ .

*Inductive hypothesis:* Suppose that the result is true for  $n = k \geq 1$ , that is,

$$b_k = \frac{1}{4}(11 \cdot 5^{k-1} - 3).$$

*Inductive step:* Let  $n = k + 1$ . Now

$$\begin{aligned} b_{k+1} &= 5b_k + 3 \\ &= 5 \left[ \frac{1}{4}(11 \cdot 5^{k-1} - 3) \right] + 3 \quad \text{substitute } b_k \\ &= \frac{5}{4}(11 \cdot 5^{k-1} - 3) + 3 \\ &= \frac{1}{4}(11 \cdot 5 \cdot 5^{k-1} - 3 \cdot 5) + 3 \\ &= \frac{1}{4}(11 \cdot 5^k - 15) + 3 \\ &= \frac{1}{4} \cdot 11 \cdot 5^k - \frac{15}{4} + 3 \\ &= \frac{1}{4} \cdot 11 \cdot 5^k - \frac{3}{4} \\ &= \frac{1}{4}(11 \cdot 5^k - 3). \end{aligned}$$

This implies that the result is true for  $n = k + 1$ . Hence, by induction  $b_n = \frac{1}{4}(11 \cdot 5^{n-1} - 3)$  for all  $n \geq 1$ .

**Exercise 8:** Sam received his yearly bonus and deposited \$10,000 in a local bank yielding 7% interest compounded annually. He wants to determine the total amount accumulated after  $n$  years. More specifically, he wants to determine the total amount accumulated after 3 years. What will be the total amount after 10 years?

**Solution:** Let  $A_n$  denote the total amount accumulated after  $n$  years. Let us determine a recurrence relation for the sequence  $\{A_n\}_{n=0}^{\infty}$ , where  $A_0$  denotes the initial amount.

The amount accumulated after 1 year is the initial amount plus the interest on the initial amount. Now  $A_{n-1}$  is the amount accumulated after  $n - 1$  years. This implies that the amount at the beginning of  $n$ th year is  $A_{n-1}$ . It follows that the total amount accumulated after  $n$  years is the amount at the beginning of the  $n$ th year plus the interest on this amount. Because the interest rate is 7%, the interest earned during the  $n$ th year is  $(0.07)A_{n-1}$ . Hence,

$$\begin{aligned} A_n &= A_{n-1} + (0.07)A_{n-1} \\ &= 1.07A_{n-1}, \quad n \geq 1. \end{aligned}$$

Also

$$A_0 = 10000.$$

Now

$$\begin{aligned} A_1 &= 1.07A_0 = 1.07 \cdot 10000 = 10700, \\ A_2 &= 1.07 \cdot A_1 = 1.07 \cdot 10700 = 11449, \\ A_3 &= 1.07 \cdot A_2 = 1.07 \cdot 11449 = 12250.43. \end{aligned}$$

The total amount accumulated after 3 years is \$12,250.43.

Next, using iteration, we determine the explicit formula for  $A_n$ . Let  $n \geq 1$ . Then

$$\begin{aligned} A_n &= 1.07A_{n-1} \\ &= 1.07(1.07A_{n-2}) \\ &= 1.07^2A_{n-2} \\ &= 1.07^2(1.07A_{n-3}) \\ &= 1.07^3A_{n-3} \\ &\vdots \\ &= 1.07^kA_{n-k}. \end{aligned}$$

Substituting  $k = n$ , we get

$$A_n = 1.07^nA_{n-n} = 1.07^nA_0 = (10000)1.07^n, \quad n \geq 0. \quad (8.28)$$

We leave it as an exercise to verify, by induction, that this is the correct formula for  $A_n$ .

The total amount accumulated after 10 years is  $A_{10}$ . Substitute  $n = 10$  in (8.28) to get

$$A_{10} = (10000)1.07^{10} = 19671.51.$$

Hence, the total amount accumulated after 10 years is \$19,671.51.

**Exercise 9:** A chain of fast-food restaurants had 30 stores in 1990. Since then they have opened 4 new stores every year. Assume that this trend of opening 4 new stores continues indefinitely.

- Write a recurrence relation with the initial condition for  $R_n$ , the number of restaurants  $n$  years after 1990.
- Solve the recurrence relation by iteration.
- How many stores will be opened in 2010?

**Solution:**

- Let  $R_n$  be the number of stores  $n$  years after 1990. The number of stores 1 year after 1990, i.e., after 1991, is the number of stores after 1990 plus 4. Similarly, the number of stores  $n$  years after 1990 is the number of stores  $n - 1$  years after 1990 plus 4. Now the number of stores  $n - 1$  years after 1990 is  $R_{n-1}$ . Thus, the recurrence relation is:

$$R_n = R_{n-1} + 4, \quad \text{if } n \geq 1. \quad (8.29)$$

Because there are 30 stores in 1990,  $R_0$ , the number of stores 0 years after 1990, is 30. Thus, the initial

condition is

$$R_0 = 30.$$

- (b) We now solve the recurrence relation (8.29) by iteration as follows:

$$\begin{aligned} R_n &= R_{n-1} + 4 \\ &= (R_{n-2} + 4) + 4 \quad \text{because } R_{n-1} = R_{n-2} + 4 \\ &= R_{n-2} + 2 \cdot 4 \\ &= R_{n-3} + 4 + 2 \cdot 4, \quad R_{n-2} = R_{n-3} + 4 \\ &= R_{n-3} + 3 \cdot 4 \\ &\vdots \\ &= R_{n-k} + k \cdot 4. \end{aligned}$$

We substitute  $k = n$  to get  $R_n = R_{n-n} + n \cdot 4 = R_0 + 4n = 30 + 4n$ . Hence,

$$R_n = 4n + 30 \quad \text{for all } n \geq 1. \quad (8.30)$$

We leave it as an exercise to verify, using induction, that the explicit formula in (8.30) for  $R_n$  is correct.

- (c) In 2010, i.e., 20 years after 1990, the number of stores is  $R_{20}$ , where

$$R_{20} = 20 \cdot 4 + 30 = 110.$$

### Exercise 10:

Consider the recurrence relation

$$a_n = a_{n-2} + a_{n-4}.$$

Explain why fewer than four initial conditions are not enough to uniquely define  $a_n$  for all  $n \geq 5$ . If  $a_1 = a_2 = a_3 = a_4 = 1$ , find the first 10 terms of the sequence  $\{a_n\}_{n=1}^{\infty}$ .

**Solution:** If  $n = 5$ , then  $a_5 = a_3 + a_1$ . This implies that if either  $a_1$  or  $a_3$  is undefined, then  $a_5$  is undefined. If  $a_5$  is undefined, then the odd-numbered terms starting with  $a_5$  are undefined. Next, let  $n = 6$ . Then  $a_6 = a_4 + a_2$ . This implies that if either  $a_2$  or  $a_4$  is undefined, then  $a_6$  is undefined. If  $a_6$  is undefined, then the even-numbered terms starting with  $a_6$  are undefined. It now follows that to completely define the sequence  $\{a_n\}_{n=1}^{\infty}$  we must specify the terms  $a_1, a_2, a_3$ , and  $a_4$ .

The first 10 terms of the sequence are:  $a_1 = 1, a_2 = 1, a_3 = 1, a_4 = 1$ ,

$$\begin{aligned} a_5 &= a_3 + a_1 = 1 + 1 = 2, \\ a_6 &= a_4 + a_2 = 1 + 1 = 2, \\ a_7 &= a_5 + a_3 = 2 + 1 = 3, \\ a_8 &= a_6 + a_4 = 2 + 1 = 3, \\ a_9 &= a_7 + a_5 = 3 + 2 = 5, \\ a_{10} &= a_8 + a_6 = 3 + 2 = 5. \end{aligned}$$

## SECTION REVIEW

### Key Terms

explicit formula

recurrence relation

iteration method

recursive definition

initial conditions

substitution method

### Some Key Definitions

1. A recurrence relation for a sequence  $a_0, a_1, a_2, \dots, a_n, \dots$  is an equation that relates  $a_n$  to some of the terms  $a_0, a_1, a_2, \dots, a_{n-2}, a_{n-1}$  for all integers  $n$  with  $n \geq k$ , where  $k$  is a nonnegative integer. The initial conditions for the recurrence relation is a set of values that explicitly defines some of the members of  $a_0, a_1, a_2, \dots, a_{k-1}$ .
2. Suppose a recurrence relation for a sequence  $a_0, a_1, a_2, \dots, a_n, \dots$ , which is an equation that relates  $a_n$  to some of the terms  $a_0, a_1, a_2, \dots, a_{n-2}, a_{n-1}$ , for all integers  $n$  with  $n \geq k$ , where  $k$  is a nonnegative integer, is given. By a solution of the recurrence relation we mean to obtain an explicit formula for  $a_n$ , i.e., to find an expression for  $a_n$  that does not involve any other  $a_i$ .

## EXERCISES

---

- Find the first five terms of a sequence  $a_0, a_1, a_2, \dots, a_n, \dots$  satisfying the given recurrence relation and initial conditions.
  - $a_n = a_{n-1} + 4$ , if  $n \geq 1$ ,  $a_0 = 1$
  - $a_n = a_{n-1} + 2n$ , if  $n \geq 1$ ,  $a_0 = 5$
  - $a_n = 2a_{n-1} + n^2$ , if  $n \geq 1$ ,  $a_0 = 1$
  - $a_n = a_{n-1} + 4a_{n-2}$ , if  $n \geq 2$ ,  $a_0 = 1$  and  $a_1 = 2$
  - $a_n = 7a_{n-1} - 12a_{n-2} + 6$ , if  $n \geq 2$ ,  $a_0 = 5$  and  $a_1 = 2$
  - $a_n = a_{n-1} + a_{n-2} + n$ , if  $n \geq 2$ ,  $a_0 = -1$  and  $a_1 = 1$
  - $a_n = 2a_{n-1} + a_{n-2} + a_{n-3}$ , if  $n \geq 3$ ,  $a_0 = 2$ ,  $a_1 = 1$ , and  $a_2 = 1$
- Find the first five terms of the sequence defined by the recurrence relation

$$a_n = a_{n-1}^2, \quad \text{if } n \geq 2$$

with the initial condition  $a_1 = 1$ .

- Find the first seven terms of the sequence defined by the recurrence relation

$$a_n = a_{n-1} + a_{n-3}, \quad \text{if } n \geq 3$$

with the initial conditions  $a_0 = 1$ ,  $a_1 = 2$ , and  $a_2 = 0$ .

- Find the first six terms of the sequence defined by the recurrence relation

$$x_n = nx_{n-1} + n^2x_{n-2}, \quad \text{if } n \geq 2$$

with the initial conditions  $x_0 = x_1 = 1$ .

- Find the recurrence relation and initial conditions for the sequence  $S : 0, 1, 3, 7, 15, \dots, 2^n - 1, \dots$
- Find the recurrence relation and initial conditions for the sequence  $S : 2, 4, 10, 28, \dots, 3^n + 1, \dots$
- Find the recurrence relation and initial conditions for the sequence  $S : 1, 2, 3, 4, 5, \dots, n + 1, \dots$
- Find a recurrence relation for the sequence  $S : 1, 3, 7, 15, 31, \dots$
- Michael is purchasing a hotel costing \$500,000 with a down payment of \$60,000 and a long-term mortgage. The unpaid balance is being financed at a monthly rate of 0.9% and a payment of \$4000 per month. Find a recurrence relation and initial condition to determine the unpaid balance after  $n$  monthly payments.
- Ashley invests \$5000 at 12% interest compounded annually in a bank. Suppose  $A_n$  denotes the amount at the end of  $n$  years.
  - Find a recurrence relation and initial condition for the sequence  $A_0, A_1, A_2, A_3, \dots, A_n, \dots$
  - Find the first four terms of the sequence.
- Priya got her yearly bonus and deposited \$7000 in a local bank yielding 7.5% interest compounded annually. Suppose  $A_n$  denotes the amount at the end of  $n$  years.
  - Find a recurrence relation and initial condition for the sequence  $A_0, A_1, A_2, A_3, \dots, A_n, \dots$
  - Find the total amount accumulated after five years.

- Rohan purchased a car for \$15,000 with a down payment of \$500 and a long-term loan. The unpaid balance is financed at a monthly rate of 1.5% and a payment of \$500 per month. Find a recurrence relation and initial condition to determine the unpaid balance after  $n$  monthly payments.
- Describe a recurrence relation specifying the number of  $n$ -bit strings,  $n \geq 0$ , that do not contain two consecutive 1's, that is, the string 11.
- Sandy is purchasing a new car for \$30,000 with a down payment of \$6000 and a long-term loan. The unpaid balance is financed at a monthly rate of 1.2% and a payment of \$200 per month. Find a recurrence relation and initial condition to determine the unpaid balance after  $n$  monthly payments.
- Raj invests \$8500 at 9% interest compounded annually in a bank. Suppose  $A_n$  denotes the amount at the end of  $n$  years.
  - Find a recurrence relation and initial condition for the sequence  $A_0, A_1, A_2, A_3, \dots, A_n, \dots$
  - Find the first four terms of the sequence.
- Jessica needs to take a tablet containing 2.5 mg of a drug each morning to control allergies. During the day 15% of the amount of the drug is eliminated.
  - Write a recurrence relation, with initial condition(s), describing the amount of the drug in Jessica's body immediately after she takes the  $n$ th tablet.
  - How much of the drug is in Jessica's body immediately after she takes the 8th tablet?
  - What quantity of the drug will eventually accumulate in Jessica's body?
- The  $n$ th term,  $a_n$ , of the sequence  $a_1, a_2, \dots, a_n, \dots$  satisfies the recurrence relation

$$a_n = 12a_{n-1} - 35a_{n-2} + 4, \quad n \geq 2,$$

with initial conditions  $a_0 = 0$  and  $a_1 = 2$ . Prove that

$$a_n = \frac{4}{3}7^n - \frac{3}{2}5^n + \frac{1}{6}, \quad n \geq 0.$$

- Consider the recurrence relation

$$a_n = 2a_{n-1} - a_{n-2} \quad \text{for all } n \geq 2.$$

- Determine whether the sequence  $\{a_n\}_{n=0}^{\infty}$  is a solution of this recurrence relation, where  $a_n = 3n$  for all  $n \geq 0$ .
- Determine whether the sequence  $\{a_n\}_{n=0}^{\infty}$  is a solution of the recurrence relation, where  $a_n = 2^n$  for all  $n \geq 0$ .

- Consider the recurrence relation and initial conditions:

$$a_n = 3a_{n-1} - 2a_{n-2} \quad \text{for all } n \geq 2,$$

$a_0 = 0$ ,  $a_1 = 1$ . Solve the recurrence relation, i.e., find an explicit formula for  $a_n$ .

20. Consider the recurrence relation

$$a_n = a_1 a_{n-1} + a_2 a_{n-2} + \cdots + a_{n-1} a_1, \quad \text{if } n \geq 2$$

with the initial condition  $a_1 = 1$ . Determine the first five terms of the sequence  $\{a_n\}_{n=1}^{\infty}$ . (The numbers of the sequence  $\{a_n\}_{n=1}^{\infty}$  are known as the Catalan numbers.)

21. Consider the recurrence relation and initial conditions

$$a_n = -3a_{n-1} + 4a_{n-2},$$

$$a_0 = 5, a_1 = -5$$

Show that  $a_n = 2(-4)^n + 3$  for all  $n \geq 0$ .

22. A chain of hotels had 20 hotels in 1980. Since then they have opened five new hotels every year. Assume that this trend of opening five new hotels continues indefinitely.

- a. Write a recurrence relation with the initial condition for  $H_n$ , the number of hotels  $n$  years after 1980.
- b. Solve the recurrence relation by iteration.
- c. How many hotels will there be in 2020?

23. Let  $A$  and  $B$  be sets with  $m$  and  $n$  elements, respectively. Let  $a_{m,n}$  denote the number of onto functions from  $A$  into  $B$ . Show that

$$a_{m,n} = n^m - \sum_{i=1}^{n-1} C(n, i) a_{m,k}, \quad \text{if } m \geq n \text{ and } n > 1$$

and  $a_{m,1} = 1$ , where  $C(n, i)$  is the number of combinations of choosing  $i$  items from a set of  $n$  items.

24. Let  $A$  be a set with  $n$  elements and let  $k$  be an integer such that  $1 \leq k \leq n$ . Let  $S_{n,k}$  denote the number of ways set  $A$  can be partitioned into  $k$  subsets. For example, if  $A = \{x_1, x_2, x_3\}$  and  $k = 2$ , then the partitions of two subsets of  $A$  are  $\{\{x_1\}, \{x_2, x_3\}\}$ ,  $\{\{x_2\}, \{x_1, x_3\}\}$ ,  $\{\{x_3\}, \{x_1, x_2\}\}$ , so  $S_{3,2} = 3$ . (The number  $S_{n,k}$  is known as the Stirling number of the second kind.)

- a. Find  $S_{4,1}, S_{4,2}, S_{4,3}, S_{4,4}$ .
- b. Show that  $S_{n,k} = S_{n-1,k-1} + kS_{n-1,k}$  for all integers  $n$  and  $k$  such that  $1 \leq k \leq n$  with the initial conditions  $S_{n,1} = 1 = S_{n,n}$  for all  $n \geq 1$ .
- c. Use Part (b) to find  $S_{5,1}, S_{5,2}, S_{5,3}, S_{5,4}, S_{5,5}$ .

25. Suppose for the Tower of Hanoi problem, the three pegs are in a row and we impose the following additional restriction: A disk from a peg can be moved only to the adjacent peg. That is, from peg 1 to peg 2, peg 2 to peg 1, peg 2 to peg 3, and peg 3 to peg 2. Let  $c_n$  denote the number of moves required to move  $n$  disks,  $n \geq 0$ , from peg 1 to peg 3.

- a. Find  $c_1, c_2, c_3$ , and  $c_4$ .
- b. Show that  $c_n = 3c_{n-1} + 2$ .
- c. Use part (b) to find  $c_5$  and  $c_6$ .

## 8.2 LINEAR HOMOGENEOUS RECURRENCE RELATIONS

In Section 8.1, we showed how to solve recurrence relations using iterations, or substitutions. In this section, we consider special types of recurrence relations and discuss how to find an explicit formula for the recurrence relations. We begin with the following definition.

**DEFINITION 8.2.1** ▶ Let  $a_0, a_1, a_2, \dots, a_n, \dots$  be a sequence of numbers. A **linear homogeneous recurrence relation** of order  $k$  with constant coefficients is a recurrence relation of the form

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + c_3 a_{n-3} + \cdots + c_k a_{n-k}, \quad (8.31)$$

where  $c_k \neq 0$  and  $c_1, c_2, c_3, \dots$ , and  $c_k$  are constants.

### EXAMPLE 8.2.2

Consider the following recurrence relations.

- (i)  $a_n = 3a_{n-1} + a_{n-2}$
- (ii)  $a_n = 3a_{n-1} + 5$
- (iii)  $a_n = 3a_{n-1} + a_{n-2} \cdot a_{n-3}$
- (iv)  $a_n = 3a_{n-1} + a_{n-2} + \sqrt{2}a_{n-3}$
- (v)  $a_n = 3a_{n-1} + na_{n-2}$

Recurrence relations (i), (ii), (iii), and (iv) are recurrence relations with constant coefficients. Recurrence relation (v),  $a_n = 3a_{n-1} + na_{n-2}$ , is not a relation with constant coefficients. Notice that (i) is a linear homogeneous recurrence

relation of order 2, (ii) is not a homogeneous recurrence relation because of the constant term 5, (iii) is not a linear recurrence relation because it contains  $a_{n-2} \cdot a_{n-3}$ , the product of terms  $a_{n-2}$  and  $a_{n-3}$ , and (iv) is a linear homogeneous recurrence relation of order 3.

**DEFINITION 8.2.3** ► A sequence  $s_0, s_1, s_2, \dots, s_n, \dots$  is said to **satisfy** a linear homogeneous recurrence relation

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + c_3 a_{n-3} + \cdots + c_k a_{n-k}, \quad c_k \neq 0 \quad (8.32)$$

of order  $k$  with constant coefficients if  $s_n = c_1 s_{n-1} + c_2 s_{n-2} + c_3 s_{n-3} + \cdots + c_k s_{n-k}$ .

**DEFINITION 8.2.4** ► If a sequence  $s_0, s_1, s_2, \dots, s_n, \dots$  satisfies a linear homogeneous recurrence relation, then the sequence  $s_0, s_1, s_2, \dots, s_n, \dots$  is also called a **solution** of that recurrence relation.

### EXAMPLE 8.2.5

Consider the recurrence relation  $a_n = 3a_{n-1}$ . This is a linear homogeneous recurrence relation of order 1. Let  $t$  be a nonzero number and suppose  $a_n = t^n$  for all  $n \geq 0$ . Then  $a_n = 3a_{n-1}$  implies that  $t^n = 3t^{n-1}$ . Therefore,  $t = 3$ . Thus, we find that  $a_n = 3^n$ . Hence, the sequence  $1, 3, 3^2, 3^3, \dots, 3^n, \dots$  is a solution of the recurrence relation  $a_n = 3a_{n-1}$ .

### EXAMPLE 8.2.6

Consider the following recurrence relation of a sequence  $a_0, a_1, a_2, \dots, a_n, \dots$  of numbers.

$$a_n = 7a_{n-1} - 12a_{n-2}. \quad (8.33)$$

This is a linear homogeneous recurrence relation of order 2 with constant coefficients.

We rewrite (8.33) as follows:

$$a_n - 7a_{n-1} + 12a_{n-2} = 0$$

and substitute  $a_n = t^n$ , where  $t$  is a nonzero number, to obtain

$$t^n - 7t^{n-1} + 12t^{n-2} = 0.$$

This implies that

$$t^{n-2}(t^2 - 7t + 12) = 0.$$

Here the equation

$$t^2 - 7t + 12 = 0$$

is called the characteristic equation of the recurrence relation. We determine the roots of this equation. Now,

$$t^2 - 7t + 12 = (t - 4)(t - 3)$$

and so

$$(t - 4)(t - 3) = 0.$$

This implies that the roots of the characteristic equation are  $t = 4$ , and  $t = 3$ .

We show that the sequences  $\{4^n\}_{n=0}^{\infty}$  and  $\{3^n\}_{n=0}^{\infty}$  are solutions of the above recurrence relation. We have

$$\begin{aligned}
 & 7a_{n-1} - 12a_{n-2} \quad \text{and} \quad 7a_{n-1} - 12a_{n-2} \\
 & = 7 \cdot 4^{n-1} - 12 \cdot 4^{n-2} \quad = 7 \cdot 3^{n-1} - 12 \cdot 3^{n-2} \\
 & = 7 \cdot 4^{n-1} - 3 \cdot 4^{n-1} \quad = 7 \cdot 3^{n-1} - 4 \cdot 3^{n-1} \\
 & = 4^{n-1}(7 - 3) \quad = 3^{n-1}(7 - 4) \\
 & = 4^{n-1} \cdot 4 \quad = 3^{n-1} \cdot 3 \\
 & = 4^n \quad = 3^n \\
 & = a_n \quad = a_n
 \end{aligned}$$

Next we take  $a_n = c_1 4^n$  and  $a_n = c_2 3^n$ , where  $c_1$  and  $c_2$  are constants. Then, proceeding as above, we can show that the sequences  $\{c_1 4^n\}$ ,  $\{c_2 3^n\}$ , and  $\{c_1 4^n + c_2 3^n\}$  satisfy the above recurrence relation.

Next we consider the recurrence relation in (8.33), i.e.,  $a_n = 7a_{n-1} - 12a_{n-2}$ , with initial conditions

$$\begin{aligned}
 a_0 &= 3 \\
 a_1 &= 11.
 \end{aligned}$$

Suppose there exist constants  $c_1$  and  $c_2$  such that

$$a_n = c_1 4^n + c_2 3^n, \quad n \geq 0.$$

We substitute  $n = 0$  and  $n = 1$  to obtain

$$\begin{aligned}
 a_0 &= c_1 + c_2, \\
 a_1 &= 4c_1 + 3c_2.
 \end{aligned}$$

Using the initial conditions, we get

$$\begin{aligned}
 c_1 + c_2 &= 3, \\
 4c_1 + 3c_2 &= 11.
 \end{aligned}$$

Solving these equations for  $c_1$  and  $c_2$ , we get  $c_1 = 2$  and  $c_2 = 1$ . We now show that if  $\{a_n\}$  is a solution of the given recurrence relation with initial conditions

$$\begin{aligned}
 a_0 &= 3 \\
 a_1 &= 11,
 \end{aligned}$$

then

$$a_n = 2 \cdot 4^n + 3^n, \quad n \geq 0.$$

We verify this by induction.

*Basis step:* For  $n = 0$ ,  $a_0 = 3 = 2 \cdot 4^0 + 3^0$ . Also we find that for  $n = 1$ ,  $a_1 = 11 = 2 \cdot 4^1 + 3^1$ .

*Inductive hypothesis:* Suppose that for any integer  $k$ ,  $0 < k < n$ ,  $a_k = 2 \cdot 4^k + 3^k$ .

*Inductive step:* By the inductive hypothesis,  $a_{n-1} = 2 \cdot 4^{n-1} + 3^{n-1}$  and  $a_{n-2} = 2 \cdot 4^{n-2} + 3^{n-2}$ . Hence,

$$\begin{aligned}
 a_n &= 7a_{n-1} - 12a_{n-2} \\
 &= 7(2 \cdot 4^{n-1} + 3^{n-1}) - 12(2 \cdot 4^{n-2} + 3^{n-2}) \\
 &= 14 \cdot 4^{n-1} + 7 \cdot 3^{n-1} - 6 \cdot 4^{n-1} - 4 \cdot 3^{n-1}
 \end{aligned}$$

$$\begin{aligned}
 &= 8 \cdot 4^{n-1} + 3 \cdot 3^{n-1} \\
 &= 2 \cdot 4^n + 3^n.
 \end{aligned}$$

The result now follows by induction, i.e.,  $a_n = 2 \cdot 4^n + 3^n$  for all  $n \geq 0$ .

Hence, the solution of the given recurrence relation, (8.33), with initial conditions  $a_0 = 3$  and  $a_1 = 11$  is the sequence  $\{2 \cdot 4^n + 3^n\}$ .

From Example 8.2.6, we find that:

- (i) A linear homogeneous recurrence relation of order  $k$  with constant coefficients may have more than one solution.
- (ii) If the sequences  $\{s_n\}$  and  $\{p_n\}$  satisfy a recurrence relation, then the sequence  $\{bs_n + dp_n\}$  is also a solution, where  $b$  and  $d$  are constants.

Let us now consider a linear homogeneous recurrence relation with constant coefficients of order 2.

**Theorem 8.2.7:** Let

$$a_n = c_1 a_{n-1} + c_2 a_{n-2}, \quad c_2 \neq 0, \quad n > 1 \quad (8.34)$$

be a linear homogeneous recurrence relation with constant coefficients.

Let  $t$  be a nonzero real number. Then the sequence  $\{t^n\}$  satisfies the above recurrence relation if and only if

$$t^2 - c_1 t - c_2 = 0. \quad (8.35)$$

**Proof:** Consider the recurrence relation (8.34). Let us rewrite it as follows:

$$a_n - c_1 a_{n-1} - c_2 a_{n-2} = 0. \quad (8.36)$$

Suppose the sequence  $\{t^n\}$  satisfies the above recurrence relation. Then we obtain

$$t^n - c_1 t^{n-1} - c_2 t^{n-2} = 0.$$

This implies that

$$t^{n-2}(t^2 - c_1 t - c_2) = 0.$$

Now  $t \neq 0$  implies that  $t^{n-2} \neq 0$ . Hence, we divide both sides by  $t^{n-2}$  to obtain

$$t^2 - c_1 t - c_2 = 0.$$

Conversely, suppose

$$t^2 - c_1 t - c_2 = 0.$$

We multiply both sides by  $t^{n-2}$  to get

$$t^n - c_1 t^{n-1} - c_2 t^{n-2} = 0,$$

i.e.,

$$t^n = c_1 t^{n-1} + c_2 t^{n-2}.$$

Hence, the sequence  $\{t^n\}$  satisfies the given recurrence relation. ■

**DEFINITION 8.2.8** ▶ Let  $a_n = c_1 a_{n-1} + c_2 a_{n-2}$ ,  $c_2 \neq 0$ ,  $n > 1$  be a linear homogeneous recurrence relation with constant coefficients. The equation

$$t^2 - c_1 t - c_2 = 0$$

is called the **characteristic equation** of the recurrence relation.

**Theorem 8.2.9:** Let

$$a_n = c_1 a_{n-1} + c_2 a_{n-2}, \quad n > 1 \quad (8.37)$$

be a linear homogeneous recurrence relation of order 2, where  $c_1$  and  $c_2$  are constants and  $c_2 \neq 0$

- (i) If the sequences  $\{s_n\}$  and  $\{p_n\}$  satisfy (8.37), then for any constants  $b$  and  $d$ , the sequence  $\{bs_n + dp_n\}$  satisfies (8.37).
- (ii) Let  $r$  be a root of the characteristic equation

$$t^2 - c_1 t - c_2 = 0 \quad (8.38)$$

of (8.37). Then the sequence  $\{r^n\}$  is a solution of (8.37).

**Proof:**

- (i) Because the sequences  $\{s_n\}$  and  $\{p_n\}$  are solutions of (8.37), we have

$$\begin{aligned} s_n &= c_1 s_{n-1} + c_2 s_{n-2}, \\ p_n &= c_1 p_{n-1} + c_2 p_{n-2}. \end{aligned}$$

Let  $u_n = bs_n + dp_n$ ,  $n \geq 0$ . We show that the sequence  $\{u_n\}$  is a solution of (8.37). Now

$$\begin{aligned} u_n &= bs_n + dp_n \\ &= b(c_1 s_{n-1} + c_2 s_{n-2}) + d(c_1 p_{n-1} + c_2 p_{n-2}) \\ &= c_1(bs_{n-1} + dp_{n-1}) + c_2(bs_{n-2} + dp_{n-2}) \\ &= c_1 u_{n-1} + c_2 u_{n-2}. \end{aligned}$$

Hence, the sequence  $\{u_n\}$  is a solution of (8.37).

- (ii) Let  $r$  be a root of the quadratic equation

$$t^2 - c_1 t - c_2 = 0.$$

Then

$$r^2 - c_1 r - c_2 = 0$$

and so by Theorem 8.2.7, it follows that the sequence  $1, r, r^2, \dots, r^n, \dots$  is a solution of (8.37). ■

**Theorem 8.2.10:** Suppose that a sequence  $\{d_n\}$  is a solution of the recurrence relation (8.37). If  $\eta_1$  and  $\eta_2$  are the distinct roots of the characteristic equation (8.38), then there exist constants  $b$  and  $d$ , which

are to be determined, such that the solution of the recurrence relation (8.37) is

$$d_n = br_1^n + dr_2^n, \quad n = 0, 1, \dots$$

**Proof:** By Theorem 8.2.9, for any constants  $b$  and  $d$ , the sequence  $\{br_1^n + dr_2^n\}$  is a solution of the given recurrence relation. Suppose  $b$  and  $d$  are constants such that

$$\begin{aligned} d_0 &= br_1^0 + dr_2^0, \\ d_1 &= br_1^1 + dr_2^1 \end{aligned}$$

Then

$$b + d = d_0, \quad (8.39)$$

$$br_1 + dr_2 = d_1. \quad (8.40)$$

Multiply (8.39) by  $r_1$  and subtract (8.40) from it to obtain

$$d(r_1 - r_2) = r_1 d_0 - d_1.$$

Because  $r_1 \neq r_2$ , divide by  $r_1 - r_2$  to get

$$d = \frac{r_1 d_0 - d_1}{r_1 - r_2}.$$

Now multiply (8.39) by  $r_2$  and subtract (8.40) from it to obtain

$$b(r_2 - r_1) = r_2 d_0 - d_1.$$

As before, because  $r_1 \neq r_2$ , divide by  $r_2 - r_1$  to get

$$b = \frac{r_2 d_0 - d_1}{r_2 - r_1}$$

We prove by induction on  $n$  that

$$d_n = br_1^n + dr_2^n, \quad n = 0, 1, \dots,$$

where,  $b = \frac{r_2 d_0 - d_1}{r_2 - r_1}$ ,  $d = \frac{r_1 d_0 - d_1}{r_1 - r_2}$ .

*Basis step:* For  $n = 0$ ,  $br_1^0 + dr_2^0 = b + d = d_0$ . Also

$$br_1^1 + dr_2^1 = \left( \frac{r_2 d_0 - d_1}{r_2 - r_1} \right) r_1 + \left( \frac{r_1 d_0 - d_1}{r_1 - r_2} \right) r_2 = d_1.$$

Hence, the result is true for  $n = 0, 1$ .

*Inductive hypothesis:* Suppose  $d_k = br_1^k + dr_2^k$ , for  $k = 0, 1, \dots, n-1, n > 1$ .

*Inductive step:* By the inductive hypothesis,  $d_{n-1} = br_1^{n-1} + dr_2^{n-1}$  and  $d_{n-2} = br_1^{n-2} + dr_2^{n-2}$ . Hence,

$$\begin{aligned} d_n &= c_1 d_{n-1} + c_2 d_{n-2} && \text{because } \{d_n\} \text{ is a solution of} \\ &= c_1(br_1^{n-1} + dr_2^{n-1}) + c_2(br_1^{n-2} + dr_2^{n-2}) && a_n = c_1 a_{n-1} + c_2 a_{n-2} \\ &= (c_1 r_1 + c_2)br_1^{n-2} + (c_2 + c_1 r_2)dr_2^{n-2} \end{aligned}$$

$$\begin{aligned}
 &= r_1^2 b r_1^{n-2} + r_2^2 d r_2^{n-2} \quad \text{because } r_1, r_2 \text{ are roots of} \\
 &\quad \text{the characteristic equation} \\
 &= br^n + dr_2^n.
 \end{aligned}$$

This completes the induction and the proof of the theorem. ■

**Corollary 8.2.11:** Suppose that

$$a_0 = d_0, \quad a_1 = d_1$$

are the initial conditions for the recurrence relation (8.37), where  $d_0$  and  $d_1$  are constants. Further suppose that  $r_1$  and  $r_2$  are the roots of (8.38). If  $r_1 \neq r_2$ , then there exist constants  $b$  and  $d$ , which are to be determined by initial conditions, such that the solution of the recurrence relation (8.37) is

$$a_n = br_1^n + dr_2^n, \quad n = 0, 1, \dots$$

**Proof:** It follows from Theorem 8.2.10. ■

### EXAMPLE 8.2.12

In this example, we solve the following linear homogeneous recurrence relation:

$$a_n = 7a_{n-1} - 10a_{n-2} \quad (8.41)$$

with initial conditions

$$\begin{aligned}
 a_0 &= 1 \\
 a_1 &= 8.
 \end{aligned}$$

The characteristic equation of the given recurrence relation is:

$$t^2 - 7t + 10 = 0.$$

Next, we find the roots of this equation. Now,

$$t^2 - 7t + 10 = (t - 5)(t - 2)$$

and so

$$(t - 5)(t - 2) = 0.$$

This implies that the roots of the characteristic equation are  $t = 5$ , and  $t = 2$ . The roots are distinct. By Theorem 8.2.10, there exist constants  $c_1$  and  $c_2$ , which are to be determined from initial conditions, such that

$$a_n = c_1 5^n + c_2 2^n, \quad n \geq 0.$$

We substitute  $n = 0$  and  $n = 1$ , respectively, to obtain

$$\begin{aligned}
 a_0 &= c_1 + c_2, \\
 a_1 &= 5c_1 + 2c_2.
 \end{aligned}$$

Using the initial conditions, we get

$$\begin{aligned}
 c_1 + c_2 &= 1, \\
 5c_1 + 2c_2 &= 8.
 \end{aligned}$$

Solving these equations for  $c_1$  and  $c_2$ , we get  $c_1 = 2$  and  $c_2 = -1$ . Hence,

$$a_n = 2 \cdot 5^n - 2^n, \quad n \geq 0.$$

Hence, the sequence  $\{2 \cdot 5^n - 2^n\}$  is the solution.

**Theorem 8.2.13:** Suppose that a sequence  $\{s_n\}$  is a solution of the recurrence relation (8.37). If  $r_1$  and  $r_2$  are the roots of the characteristic equation (8.38) such that  $r_1 = r_2 = r$ , then there exist constants  $b$  and  $d$ , which are to be determined, such that the solution of the recurrence relation (8.37) is

$$s_n = br^n + dnr^n, \quad n = 0, 1, \dots$$

**Proof:** Suppose that  $r_1 = r_2 = r$ . Because  $r$  is a root of (8.38), the sequence  $\{r^n\}_{n=0}^{\infty}$  is a solution of (8.37).

Next we show that the sequence  $\{nr^n\}_{n=0}^{\infty}$  is a solution of (8.37).

Now  $c_2 \neq 0$ , so  $r \neq 0$ . Because  $r$  is a root, of multiplicity 2, of (8.38), we get

$$\begin{aligned} t^2 - c_1 t - c_2 &= (t - r)^2 \\ &= t^2 - 2rt + r^2. \end{aligned}$$

Comparing the coefficients, we get  $c_1 = 2r$  and  $c_2 = -r^2$ . Let  $a_n = nr^n$ . Then  $a_{n-1} = (n-1)r^{n-1}$  and  $a_{n-2} = (n-2)r^{n-2}$ . Now

$$\begin{aligned} c_1 a_{n-1} + c_2 a_{n-2} &= c_1(n-1)r^{n-1} + c_2(n-2)r^{n-2} \\ &= 2r(n-1)r^{n-1} + (-r^2)(n-2)r^{n-2} \quad \text{because } c_1 = 2r \text{ and } c_2 = -r^2 \\ &= 2(n-1)r^n - (n-2)r^n \\ &= (2(n-1) - (n-2))r^n \\ &= (2n-2-n+2)r^n \\ &= nr^n \\ &= a_n. \end{aligned}$$

Hence, the sequence  $\{nr^n\}_{n=0}^{\infty}$  is a solution of (8.37).

By Theorem 8.2.9, the sequence  $\{br^n + dnr^n\}_{n=0}^{\infty}$ , for any constants  $b$  and  $d$ , is a solution of (8.37).

Suppose  $b$  and  $d$  are constants such that

$$s_0 = br^0 + d0r^0,$$

$$s_1 = br^1 + d1r^1.$$

Then

$$b = s_0 \quad \text{and} \quad d = \frac{s_1 - s_0 r}{r}.$$

We prove by induction on  $n$  that

$$s_n = br^n + dnr^n, \quad n = 0, 1, \dots,$$

where  $b = s_0$ ,  $d = \frac{s_1 - s_0 r}{r}$ .

*Basis step:* For  $n = 0$ ,  $br^0 + d0r^0 = b = s_0$ .

Also we find that

$$br^1 + d1r^1 = s_0r + \frac{s_1 - s_0r}{r} \cdot r = s_1.$$

*Inductive hypothesis:* Suppose  $s_k = br^k + dkr^k$ , for  $k = 0, 1, \dots, n-1, n > 1$ .

*Inductive step:* By the inductive hypothesis,  $s_{n-1} = br^{n-1} + d(n-1)r^{n-1}$  and  $s_{n-2} = br^{n-2} + d(n-2)r^{n-2}$ . Hence,

$$\begin{aligned} s_n &= c_1s_{n-1} + c_2s_{n-2} \quad \text{because } \{s_n\} \text{ is a solution of } a_n = c_1a_{n-1} + c_2a_{n-2} \\ &= c_1(br^{n-1} + d(n-1)r^{n-1}) + c_2(br^{n-2} + d(n-2)r^{n-2}) \\ &= 2r(br^{n-1} + d(n-1)r^{n-1}) - r^2(br^{n-2} + d(n-2)r^{n-2}) \\ &\qquad\qquad\qquad \text{because } r \text{ is a repeated root} \\ &\qquad\qquad\qquad \text{of the characteristic equation} \\ &= 2br^n + 2d(n-1)r^n - br^n - d(n-2)r^n \\ &= br^n + dnr^n. \end{aligned}$$

This completes the induction, and the proof of the theorem is complete. ■

The following corollary is immediately from Theorem 8.2.13.

**Corollary 8.2.14:** Suppose that

$$a_0 = d_0, \quad a_1 = d_1$$

are the initial conditions for the recurrence relation (8.37), where  $d_0$  and  $d_1$  are constants. Further suppose that  $\eta_1$  and  $\eta_2$  are the roots of (8.38) such that  $\eta_1 = \eta_2 = r$ . Then there exist constants  $b$  and  $d$ , which are to be determined from initial conditions, such that the solution of the recurrence relation (8.37) is

$$a_n = br^n + dnr^n, \quad n = 0, 1, \dots$$

### EXAMPLE 8.2.15

In this example, we solve the following linear homogeneous recurrence relation:

$$a_n = 4a_{n-1} - 4a_{n-2}$$

with initial conditions

$$\begin{aligned} a_0 &= 4 \\ a_1 &= 12. \end{aligned}$$

The characteristic equation of this recurrence relation is the quadratic equation

$$t^2 - 4t + 4 = 0.$$

We find the roots of this equation. Now,

$$t^2 - 4t + 4 = (t - 2)(t - 2)$$

and so

$$(t - 2)(t - 2) = 0.$$

This implies that the roots of the characteristic equation are  $t = 2$ , and  $t = 2$ . The roots are not distinct. Therefore, by Theorem 8.2.13, there exist constants  $c_1$  and  $c_2$ , which are to be determined from initial conditions, such that

$$a_n = c_1 2^n + c_2 n 2^n, \quad n = 0, 1, \dots$$

We substitute  $n = 0$  and  $n = 1$ , respectively, to obtain

$$\begin{aligned} a_0 &= c_1 \\ a_1 &= 2c_1 + 2c_2. \end{aligned}$$

Using the initial conditions, we get

$$\begin{aligned} c_1 &= 4, \\ 2c_1 + 2c_2 &= 12. \end{aligned}$$

Solving these equations for  $c_1$  and  $c_2$ , we get  $c_1 = 4$  and  $c_2 = 2$ . Hence,

$$a_n = 4 \cdot 2^n + 2 \cdot n \cdot 2^n = 2 \cdot 2^{n+1} + n 2^{n+1} = (2+n)2^{n+1} = (n+2)2^{n+1}, \quad n \geq 0.$$

Thus, we find that the sequence  $\{(n+2)2^{n+1}\}$  is the solution.

We now consider linear homogeneous recurrence relations of order  $k \geq 2$ .

**Theorem 8.2.16:** Let

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + c_3 a_{n-3} + \dots + c_k a_{n-k}, \quad c_k \neq 0 \quad (8.42)$$

be a linear homogeneous recurrence relation with constant coefficients. Let  $t$  be a nonzero real number. Then the sequence  $\{t^n\}$  is a solution of the above recurrence relation if and only if

$$t^n - c_1 t^{n-1} - c_2 t^{n-2} - c_3 t^{n-3} - \dots - c_k t^{n-k} = 0.$$

**Proof:** Consider the recurrence relation (8.42). Let us rewrite it as follows:

$$a_n - c_1 a_{n-1} - c_2 a_{n-2} - c_3 a_{n-3} - \dots - c_k a_{n-k} = 0. \quad (8.43)$$

Suppose  $\{t^n\}$  is a solution of the above recurrence relation. Then we obtain

$$t^n - c_1 t^{n-1} - c_2 t^{n-2} - c_3 t^{n-3} - \dots - c_k t^{n-k} = 0.$$

This implies that

$$t^{n-k}(t^k - c_1 t^{k-1} - c_2 t^{k-2} - c_3 t^{k-3} - \dots - c_k) = 0.$$

Now  $t \neq 0$  implies that  $t^{n-k} \neq 0$ . Hence, dividing both sides by  $t^{n-k}$  we obtain the equation

$$t^k - c_1 t^{k-1} - c_2 t^{k-2} - c_3 t^{k-3} - \dots - c_k = 0.$$

Conversely, suppose

$$t^k - c_1 t^{k-1} - c_2 t^{k-2} - c_3 t^{k-3} - \dots - c_k = 0.$$

Multiplying both sides by  $t^{n-k}$ , we get

$$t^n - c_1 t^{n-1} - c_2 t^{n-2} - c_3 t^{n-3} - \dots - c_k t^{n-k} = 0,$$

i.e.,

$$t^n = c_1 t^{n-1} + c_2 t^{n-2} + c_3 t^{n-3} + \cdots + c_k t^{n-k}.$$

Hence,  $\{t^n\}$  is a solution of the given recurrence relation. ■

**DEFINITION 8.2.17** ▶ Let  $a_n = c_1 a_{n-1} + c_2 a_{n-2} + c_3 a_{n-3} + \cdots + c_k a_{n-k}$ ,  $c_k \neq 0$  be a linear homogeneous recurrence relation with constant coefficients. The equation

$$t^k - c_1 t^{k-1} - c_2 t^{k-2} - c_3 t^{k-3} - \cdots - c_k = 0$$

is called the **characteristic equation** of this linear homogeneous recurrence relation.

**REMARK 8.2.18** ▶ To obtain the characteristic equation of the recurrence relation  $a_n = c_1 a_{n-1} + c_2 a_{n-2} + c_3 a_{n-3} + \cdots + c_k a_{n-k}$ ,  $c_k \neq 0$ , substitute  $a_n = t^n$ ,  $t \neq 0$ , to get

$$t^n = c_1 t^{n-1} + c_2 t^{n-2} + c_3 t^{n-3} + \cdots + c_k t^{n-k}.$$

Thus,

$$\begin{aligned} t^n &= c_1 t^{n-1} + c_2 t^{n-2} + c_3 t^{n-3} + \cdots + c_k t^{n-k} \\ \Rightarrow t^n - c_1 t^{n-1} - c_2 t^{n-2} - c_3 t^{n-3} - \cdots - c_k t^{n-k} &= 0 \\ \Rightarrow t^{n-k}(t^k - c_1 t^{k-1} - c_2 t^{k-2} - c_3 t^{k-3} - \cdots - c_k) &= 0. \end{aligned}$$

Because  $t \neq 0$ , we have,  $t^k - c_1 t^{k-1} - c_2 t^{k-2} - c_3 t^{k-3} - \cdots - c_k = 0$ , which is the characteristic equation.

### Theorem 8.2.19: Let

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + c_3 a_{n-3} + \cdots + c_k a_{n-k} \quad (8.44)$$

be a linear homogeneous recurrence relation of order  $k$ , where  $c_k \neq 0$  and  $c_1, c_2, c_3, \dots$ , and  $c_k$  are constants. Let

$$t^k - c_1 t^{k-1} - c_2 t^{k-2} - c_3 t^{k-3} - \cdots - c_k = 0$$

be the characteristic equation of (8.44).

- (i) If the sequences  $\{s_n\}_{n=0}^{\infty}$  and  $\{p_n\}_{n=0}^{\infty}$  are solutions of (8.44), then for any constants  $b$  and  $d$ , the sequence  $\{bs_n + dp_n\}_{n=0}^{\infty}$  is a solution of (8.44).
- (ii) If  $r$  is a root of the characteristic equation, then the sequence  $1, r, r^2, \dots, r^n, \dots$  is a solution of (8.44).
- (iii) If  $r_1, r_2, \dots, r_k$  are distinct roots of the characteristic equations, then there exist constants  $b_1, b_2, b_3, \dots, b_k$ , which are to be determined from initial conditions, such that a solution of (8.44) is given by

$$a_n = b_1 r_1^n + b_2 r_2^n + b_3 r_3^n + \cdots + b_k r_k^n,$$

- (iv) If  $r$  is a root, of multiplicity  $m$ , of the characteristic equation, then  $a_n = r^n$ ,  $a_n = nr^n$ ,  $a_n = n^2r^n$ , ..., and  $a_n = n^{m-1}r^n$  are solutions of (8.44).
- (v) Suppose that

$$a_0 = d_0, a_1 = d_1, \dots, a_{n-1} = d_{n-1}$$

are the initial conditions for the recurrence relation (8.44), where  $d_0, d_1, \dots$ , and  $d_{n-1}$  are constants. If  $r_1, r_2, \dots$ , and  $r_t$  are  $t$  distinct roots of the characteristic equation with multiplicities  $m_1, m_2, \dots, m_t$  and  $m_1 + m_2 + \dots + m_t = k$ , then there exist constants  $c_{ij}$ , which are to be determined from the initial conditions, such that the solution of the recurrence relation (8.44) is

$$\begin{aligned} a_n = & (c_{00} + c_{01}n + \dots + c_{0m_1}n^{m_1-1})r_1^n \\ & + (c_{10} + c_{11}n + \dots + c_{1m_2}n^{m_2-1})r_2^n \\ & + \dots + (c_{t0} + c_{t1}n + \dots + c_{tm_t}n^{m_t-1})r_t^n, \quad n = 0, 1, \dots \end{aligned}$$

The proof of Theorem 8.2.19 is beyond the scope of this book. However, in the next example, we apply this theorem to solve a third-order linear homogeneous recurrence relation.

### EXAMPLE 8.2.20

In this example, we solve the following recurrence relation:

$$a_n = 5a_{n-1} - 8a_{n-2} + 4a_{n-3} \quad (8.45)$$

with initial conditions

$$a_0 = 0$$

$$a_1 = 2$$

$$a_2 = 10.$$

First we find the characteristic equation for (8.45). Rewrite (8.45) as follows:

$$a_n - 5a_{n-1} + 8a_{n-2} - 4a_{n-3} = 0,$$

and substitute  $a_n = t^n$  to obtain

$$t^n - 5t^{n-1} + 8t^{n-2} - 4t^{n-3} = 0.$$

This implies that

$$t^{n-3}(t^3 - 5t^2 + 8t - 4) = 0.$$

Thus, the characteristic equation is:

$$t^3 - 5t^2 + 8t - 4 = 0.$$

Next, we find the roots of this equation. Now,

$$t^3 - 5t^2 + 8t - 4 = (t - 1)(t - 2)^2.$$

Therefore,

$$(t - 1)(t - 2)^2 = 0$$

This implies that the roots of the characteristic equation are  $t = 1$  and  $t = 2$ , and 2 is a root of multiplicity 2. Thus, there exist constants  $c_1$ ,  $c_2$ , and  $c_3$  which are to be determined, such that the solution of (8.45) is

$$a_n = c_1 1^n + c_2 2^n + c_3 n 2^n, \quad n \geq 0. \quad (8.46)$$

Substitute  $n = 0$ ,  $n = 1$ , and  $n = 2$  to obtain

$$\begin{aligned} a_0 &= c_1 + c_2 \\ a_1 &= c_1 + 2c_2 + 2c_3 \\ a_2 &= c_1 + 4c_2 + 8c_3. \end{aligned}$$

Using the initial conditions, we get

$$\begin{aligned} c_1 + c_2 &= 0 \\ c_1 + 2c_2 + 2c_3 &= 2 \\ c_1 + 4c_2 + 8c_3 &= 10. \end{aligned}$$

Solving these equations for  $c_1$ ,  $c_2$ , and  $c_3$ , we get  $c_1 = 2$ ,  $c_2 = -2$ , and  $c_3 = 2$ . Hence,

$$a_n = 2 \cdot 1^n - 2 \cdot 2^n + 2n2^n, \quad n \geq 0,$$

i.e.,

$$a_n = 2 - 2^{n+1} + n2^{n+1}, \quad n \geq 0.$$

## WORKED-OUT EXERCISES

**Exercise 1:** Find an explicit formula for the following linear homogeneous recurrence relation:

$$a_n = -4a_{n-1} - 3a_{n-2}, \quad \text{if } n \geq 2, \quad (8.47)$$

with the initial conditions  $a_0 = 4$  and  $a_1 = 8$ .

**Solution:** Let us rewrite the recurrence relation as

$$a_n + 4a_{n-1} + 3a_{n-2} = 0.$$

The corresponding characteristic equation is

$$t^2 + 4t + 3 = 0,$$

which can be factored as

$$(t + 1)(t + 3) = 0.$$

The roots of this characteristic equation are  $t = -1$  and  $t = -3$ . Then there exist constants  $c_1$  and  $c_2$ , which are to be determined from the initial conditions, such that the solution of the given recurrence relation is

$$a_n = c_1(-1)^n + c_2(-3)^n \quad \text{for all } n \geq 0. \quad (8.48)$$

Next we determine  $c_1$  and  $c_2$  using the initial conditions. We substitute  $n = 0$  and  $n = 1$  in (8.48) to get

$$\begin{aligned} a_0 &= c_1 + c_2, \\ a_1 &= -c_1 - 3c_2. \end{aligned}$$

Thus, using initial conditions, we have

$$\begin{aligned} c_1 + c_2 &= 4, \\ -c_1 - 3c_2 &= 8. \end{aligned}$$

Solving these equations for  $c_1$  and  $c_2$ , we get  $c_1 = 10$  and  $c_2 = -6$ . Hence, the explicit formula is:

$$\begin{aligned} a_n &= 10(-1)^n + (-6)(-3)^n \\ &= 10(-1)^n - 6(-3)^n \quad \text{for all } n \geq 0. \end{aligned}$$

**Exercise 2:** Find an explicit formula for the following linear homogeneous recurrence relation:

$$a_n = 6a_{n-1} - 9a_{n-2}, \quad \text{if } n \geq 2, \quad (8.49)$$

with the initial conditions  $a_0 = 4$  and  $a_1 = 9$ .

**Solution:** The characteristic equation for the recurrence relation (8.49) is

$$t^2 - 6t + 9 = 0,$$

which can be factored as

$$(t - 3)(t - 3) = 0.$$

The root of this characteristic equation is  $t = 3$ , which is a root of multiplicity 2. Then there exist constants  $c_1$  and  $c_2$ , which are to be determined from the initial conditions, such

that the solution of the given recurrence relation is

$$a_n = c_1 3^n + c_2 n 3^n \quad \text{for all } n \geq 0. \quad (8.50)$$

Next we determine  $c_1$  and  $c_2$  using the initial conditions. We substitute  $n = 0$  and  $n = 1$  in (8.50) to get

$$\begin{aligned} a_0 &= c_1, \\ a_1 &= 3c_1 + 3c_2. \end{aligned}$$

Thus, using initial conditions, we have

$$\begin{aligned} c_1 &= 4, \\ 3c_1 + 3c_2 &= 9. \end{aligned}$$

Solving these equations for  $c_1$  and  $c_2$ , we get  $c_1 = 4$  and  $c_2 = -1$ . Hence, the explicit formula is:

$$\begin{aligned} a_n &= 4 \cdot 3^n - 1n 3^n \\ &= 4 \cdot 3^n - n 3^n \\ &= (4 - n) 3^n \quad \text{for all } n \geq 0. \end{aligned}$$

**Exercise 3:** Find an explicit formula for the following linear homogeneous recurrence relation:

$$3a_n = 7a_{n-1} - 2a_{n-2}, \quad \text{if } n > 1, \quad (8.51)$$

with the initial conditions  $a_0 = -2$  and  $a_1 = 1$ .

**Solution:** We can rewrite the recurrence relation as

$$a_n - \frac{7}{3}a_{n-1} + \frac{2}{3}a_{n-2} = 0.$$

Thus, the characteristic equation for the recurrence relation is:

$$t^2 - \frac{7}{3}t + \frac{2}{3} = 0,$$

or

$$3t^2 - 7t + 2 = 0,$$

which can be factored as

$$(3t - 1)(t - 2) = 0.$$

The roots of this characteristic equation are  $t = 2$  and  $t = \frac{1}{3}$ . The roots are distinct. Then there exist constants  $c_1$  and  $c_2$ , which are to be determined from the initial conditions such that the solution of the given recurrence relation is

$$a_n = c_1 2^n + c_2 \left(\frac{1}{3}\right)^n \quad \text{for all } n \geq 0. \quad (8.52)$$

Next we determine  $c_1$  and  $c_2$  using the initial conditions. We substitute  $n = 0$  and  $n = 1$  in (8.51) to get

$$\begin{aligned} a_0 &= c_1 + c_2, \\ a_1 &= 2c_1 + \frac{1}{3}c_2. \end{aligned}$$

Thus, using initial conditions, we have

$$\begin{aligned} c_1 + c_2 &= -2, \\ 2c_1 + \frac{1}{3}c_2 &= 1. \end{aligned}$$

or

$$\begin{aligned} c_1 + c_2 &= -2, \\ 6c_1 + c_2 &= 3. \end{aligned}$$

Solving these equations for  $c_1$  and  $c_2$ , we get  $c_1 = 1$  and  $c_2 = -3$ . Hence, the explicit formula is:

$$a_n = 2^n - 3\left(\frac{1}{3}\right)^n \quad \text{for all } n \geq 0.$$

**Exercise 4:** Solve the recurrence relation

$$f_n = f_{n-1} + f_{n-2}, \quad \text{if } n \geq 3 \quad (8.53)$$

with the initial conditions  $f_1 = 1$  and  $f_2 = 2$ .

**Solution:** First we obtain the characteristic equation of (8.53), which is:

$$t^2 - t - 1 = 0.$$

The roots of this equation are:

$$t = \frac{1 \pm \sqrt{5}}{2}.$$

The roots are distinct. Hence,

$$f_n = c_1 \left(\frac{1 + \sqrt{5}}{2}\right)^n + c_2 \left(\frac{1 - \sqrt{5}}{2}\right)^n, \quad n \geq 1.$$

Using the initial conditions, we have

$$\begin{aligned} c_1 \left(\frac{1 + \sqrt{5}}{2}\right) + c_2 \left(\frac{1 - \sqrt{5}}{2}\right) &= 1, \\ c_1 \left(\frac{1 + \sqrt{5}}{2}\right)^2 + c_2 \left(\frac{1 - \sqrt{5}}{2}\right)^2 &= 2, \end{aligned}$$

or

$$\begin{aligned} c_1(1 + \sqrt{5}) + c_2(1 - \sqrt{5}) &= 2, \\ c_1(1 + \sqrt{5})^2 + c_2(1 - \sqrt{5})^2 &= 8. \end{aligned}$$

We leave it as an exercise to show that

$$\begin{aligned} c_1 &= \frac{1 + \sqrt{5}}{2\sqrt{5}}, \\ c_2 &= -\frac{1 - \sqrt{5}}{2\sqrt{5}}. \end{aligned}$$

Then the general solution for the given recurrence relation is:

$$\begin{aligned} f_n &= \frac{1 + \sqrt{5}}{2\sqrt{5}} \left(\frac{1 + \sqrt{5}}{2}\right)^n + \left(-\frac{1 - \sqrt{5}}{2\sqrt{5}}\right) \left(\frac{1 - \sqrt{5}}{2}\right)^n \\ &= \frac{1}{2^{n+1}\sqrt{5}} \{(1 + \sqrt{5})^{n+1} - (1 - \sqrt{5})^{n+1}\}, \quad n \geq 1. \end{aligned}$$

**Exercise 5:** Solve the recurrence relation

$$a_n = 3a_{n-1} + 10a_{n-2} - 24a_{n-3}, \quad n \geq 3,$$

with initial conditions  $a_0 = 0$ ,  $a_1 = 2$ , and  $a_2 = 4$ .

**Solution:** First let us rewrite the recurrence relation as

$$a_n - 3a_{n-1} - 10a_{n-2} + 24a_{n-3} = 0.$$

The characteristic equation of the recurrence relation is:

$$t^3 - 3t^2 - 10t + 24 = 0.$$

This equation can be factored as:

$$(t - 2)(t + 3)(t - 4) = 0.$$

This implies that the roots of the characteristic equation are  $t = 2$ ,  $t = -3$ , and  $t = 4$ . Then there exist constants  $c_1$ ,  $c_2$ , and  $c_3$ , which are to be determined, such that the solution of the given recurrence relation is

$$a_n = c_1 2^n + c_2 (-3)^n + c_3 4^n, \quad n \geq 0, \quad (8.54)$$

for some constants  $c_1$ ,  $c_2$ , and  $c_3$ . Substitute  $n = 0, 1$ , and  $2$  in

(8.54) to get

$$a_0 = c_1 + c_2 + c_3,$$

$$a_1 = 2c_1 - 3c_2 + 4c_3,$$

$$a_2 = 4c_1 + 9c_2 + 16c_3.$$

Using the initial conditions, we have

$$c_1 + c_2 + c_3 = 0,$$

$$2c_1 - 3c_2 + 4c_3 = 2,$$

$$4c_1 + 9c_2 + 16c_3 = 4.$$

Next we solve these equations to get  $c_1 = -\frac{7}{35}$ ,  $c_2 = -\frac{8}{35}$ , and  $c_3 = \frac{3}{7}$ . Substituting these values into (8.54), we get the following general solution of the given recurrence relation:

$$a_n = -\frac{7}{35} 2^n - \frac{8}{35} (-3)^n + \frac{3}{7} 4^n, \quad n \geq 0.$$

## SECTION REVIEW

### Key Terms

linear homogeneous recurrence relation  
satisfy

solution  
characteristic equation

### Some Key Definitions

- Let  $a_0, a_1, a_2, \dots, a_n, \dots$  be a sequence of numbers. A linear homogeneous recurrence relation of order  $k$  with constant coefficients is a recurrence relation of the form  $a_n = c_1 a_{n-1} + c_2 a_{n-2} + c_3 a_{n-3} + \dots + c_k a_{n-k}$ , where  $c_k \neq 0$  and  $c_1, c_2, c_3, \dots$ , and  $c_k$  are constants.
- A sequence  $s_0, s_1, s_2, \dots, s_n, \dots$  is said to satisfy a linear homogeneous recurrence relation  $a_n = c_1 a_{n-1} + c_2 a_{n-2} + c_3 a_{n-3} + \dots + c_k a_{n-k}$ ,  $c_k \neq 0$  of order  $k$  with constant coefficients if  $s_n = c_1 s_{n-1} + c_2 s_{n-2} + c_3 s_{n-3} + \dots + c_k s_{n-k}$ .

### Some Key Results

- Let  $a_n = c_1 a_{n-1} + c_2 a_{n-2}$ ,  $n > 1$ , be a linear homogeneous recurrence relation of order 2, where  $c_1$  and  $c_2$  are constants and  $c_2 \neq 0$ .
  - If the sequences  $\{s_n\}$  and  $\{p_n\}$  satisfy  $a_n = c_1 a_{n-1} + c_2 a_{n-2}$ , then for any constants  $b$  and  $d$ , the sequence  $\{bs_n + dp_n\}$  satisfies  $a_n = c_1 a_{n-1} + c_2 a_{n-2}$ .
  - Let  $r$  be a root of the characteristic equation  $t^2 - c_1 t - c_2 = 0$  of  $a_n = c_1 a_{n-1} + c_2 a_{n-2}$ . Then the sequence  $\{r^n\}$  is a solution of  $a_n = c_1 a_{n-1} + c_2 a_{n-2}$ .

2. Suppose that a sequence  $\{d_n\}$  is a solution of the recurrence relation  $a_n = c_1 a_{n-1} + c_2 a_{n-2}$ . If  $r_1$  and  $r_2$  are the distinct roots of the characteristic equation  $t^2 - c_1 t - c_2 = 0$ , then there exist constants  $b$  and  $d$ , which are to be determined, such that the solution of the recurrence relation  $a_n = c_1 a_{n-1} + c_2 a_{n-2}$  is

$$d_n = br_1^n + dr_2^n, \quad n = 0, 1, \dots$$

3. Suppose that a sequence  $\{s_n\}$  is a solution of the recurrence relation  $a_n = c_1 a_{n-1} + c_2 a_{n-2}$ . If  $r_1$  and  $r_2$  are the roots of the characteristic equation  $t^2 - c_1 t - c_2 = 0$  such that  $r_1 = r_2 = r$ , then there exist constants  $b$  and  $d$ , which are to be determined, such that the solution of the recurrence relation  $a_n = c_1 a_{n-1} + c_2 a_{n-2}$  is

$$s_n = br^n + dnr^n, \quad n = 0, 1, \dots$$

## EXERCISES

---

1. Which of the following are linear homogeneous recurrence relations with constant coefficients?
- $a_n = -3a_{n-1} + 4$
  - $a_n = 3a_{n-1} + 6a_{n-2}$
  - $a_n = 4a_{n-1} + n^2$
  - $a_n = a_{n-1} + 6a_{n-2} \cdot a_{n-3}$
  - $a_n = \sqrt{3}a_{n-1} + 2a_{n-2} + \sqrt{2}a_{n-3}$
  - $a_n = 3a_{n-1} + na_{n-2}$
2. Consider the recurrence relation  $a_n = -3a_{n-1} + 4a_{n-2}$ . Show that the sequence  $\{s_n\}_{n=0}^{\infty}$  is a solution of this recurrence relation, where  $s_n = 2(-4)^n + 3$  for all  $n \geq 0$ .
3. Find an explicit formula for the linear homogeneous recurrence relation

$$a_n = 4a_{n-1} - 4a_{n-2}, \quad \text{if } n \geq 2,$$

with the initial conditions  $a_0 = 4$  and  $a_1 = 9$ .

4. Find an explicit formula for the linear homogeneous recurrence relation

$$2a_n = 7a_{n-1} - 3a_{n-2}, \quad \text{if } n > 1,$$

with the initial conditions  $a_0 = 1$  and  $a_1 = 1$ .

5. Solve the recurrence relation

$$a_n = a_{n-1} + 12a_{n-2}, \quad \text{if } n > 1,$$

with the initial conditions  $a_0 = 3$  and  $a_1 = 5$ .

6. Solve the recurrence relation

$$a_n = 3a_{n-1} - 3a_{n-2} + a_{n-3}, \quad \text{if } n > 2,$$

with the initial conditions  $a_0 = 2$ ,  $a_1 = 1$ , and  $a_2 = 1$ .

7. Solve the recurrence relation

$$\sqrt{a_n} = 2\sqrt{a_{n-1}} + 3\sqrt{a_{n-2}}, \quad \text{if } n \geq 2,$$

with the initial conditions  $a_0 = 1$  and  $a_1 = 2$ . (Here we consider only the positive square root.)

8. Solve the recurrence relation

$$a_n = \sqrt{a_{n-1}a_{n-2}}, \quad \text{if } n \geq 2,$$

with the initial conditions  $a_0 = 1$  and  $a_1 = 4$ . (Here we consider only the positive square root.)

## 8.3 LINEAR NONHOMOGENEOUS RECURRENCE RELATIONS

---

A common technique for solving a problem is to divide the problem into smaller subproblems and solve each subproblem. This is known as the divide-and-conquer technique. While analyzing problems that use divide-and-conquer techniques, we often come across recurrence relations of the following form:

$$T(n) = a_n T\left(\frac{n}{k}\right) + f(n),$$

where  $k$  is an integer,  $1 \leq k \leq n$ ,  $f$  is a nonzero nonnegative real-valued function, and  $a_n$  is a constant. Such recurrence relations are called nonhomogeneous recurrence relations. More formally, we have the following definition.

**DEFINITION 8.3.1** ► A **linear nonhomogeneous recurrence relation** with constants coefficients is a recurrence relation of the form

$$a_n + c_1 a_{n-1} + \cdots + c_k a_{n-k} = f(n), \quad (8.55)$$

where  $c_i, i = 1, 2, \dots, k$ , are constants,  $c_k \neq 0$ , and  $f(n)$  is a nonzero real-valued function.

If  $f(n) = 0$ , then (8.55) is a linear homogeneous equation (which we discussed in the previous section). There is no known general method for solving nonhomogeneous linear recurrence equations. However, we can develop a method for solving the special case

$$c_0 a_n + c_1 a_{n-1} + \cdots + c_k a_{n-k} = b^n p(n), \quad (8.56)$$

where  $b$  is a constant and  $p(n)$  is a polynomial in  $n$ .

### EXAMPLE 8.3.2

Consider the recurrence

$$a_n + 5a_{n-1} + 6a_{n-2} = 3^n.$$

This is a nonhomogeneous recurrence relation of the form (8.56). Here  $k = 2$ ,  $b = 3$ , and  $p(n) = 1$ .

### EXAMPLE 8.3.3

Consider the recurrence

$$a_n + 5a_{n-1} + 6a_{n-2} = 3^n(n^2 + 6n + 5).$$

This is a nonhomogeneous recurrence relation of the form (8.56). Here  $k = 2$ ,  $b = 3$ , and  $p(n) = n^2 + 6n + 5$ .

### EXAMPLE 8.3.4

In this example, we solve the nonhomogeneous recurrence relation

$$a_n - 5a_{n-1} = 3^n, \quad n \geq 1 \quad (8.57)$$

with the initial conditions

$$a_0 = 1. \quad (8.58)$$

Let  $\{p_n\}$  be a particular solution of the given recurrence relation. Then

$$p_n - 5p_{n-1} = 3^n. \quad (8.59)$$

Suppose  $\{t_n\}$  is another solution of the given recurrence relation. Then

$$t_n - 5t_{n-1} = 3^n. \quad (8.60)$$

We subtract (8.59) from (8.60) to get

$$(t_n - p_n) - 5(t_{n-1} - p_{n-1}) = 0.$$

This shows that  $\{u_n\} = \{t_n - p_n\}$  is a solution of the linear homogeneous equation

$$a_n - 5a_{n-1} = 0.$$

Now  $u_n = t_n - p_n$ , and this implies that  $t_n = p_n + u_n$ . Hence, we find that any arbitrary solution of (8.57) is a sum of a particular solution and a solution of

the homogeneous part. Conversely, it can be shown that a sum of a particular solution and a solution of the homogeneous part is a solution of the given recurrence relation. Now the next step is to find a particular solution.

Because  $f(n) = 3^n$ , to find a particular solution we try  $p_n = d \cdot 3^n$ . Then  $p_n - 5p_{n-1} = 3^n$  implies that

$$\begin{aligned} d \cdot 3^n - 5d \cdot 3^{n-1} &= 3^n \\ \Rightarrow d \cdot 3 - 5d &= 3 \\ \Rightarrow d &= -\frac{3}{2} \\ \Rightarrow p_n &= -\frac{3}{2} \cdot 3^n = -\frac{3^{n+1}}{2}. \end{aligned}$$

Notice that

$$\begin{aligned} p_n - 5p_{n-1} &= -\frac{3^{n+1}}{2} - 5\left(-\frac{3^n}{2}\right) \\ &= -\frac{3^{n+1}}{2} + 5\frac{3^n}{2} \\ &= \frac{3^n}{2}(-3 + 5) \\ &= \frac{3^n}{2}2 \\ &= 3^n. \end{aligned}$$

Thus,  $p_n = -\frac{3^{n+1}}{2}$  is a particular solution of the given recurrence relation.

Next we consider the homogeneous part,

$$a_n - 5a_{n-1} = 0.$$

Its characteristic equation is  $t - 5 = 0$ , and the root of this characteristic equation is 5. Thus, there exists a constant  $c_1$ , which is to be determined, such that  $u_n = c_1 \cdot 5^n$ .

Thus,

$$t_n = p_n + u_n = -\frac{3^{n+1}}{2} + c_1 \cdot 5^n. \quad (8.61)$$

Now  $t_0 = 1$ . Next, we put  $n = 0$  in (8.61) to get

$$1 = t_0 = -\frac{3^{0+1}}{2} + c_1 \cdot 5^0.$$

This implies that

$$c_1 = 1 + \frac{3}{2} = \frac{5}{2}.$$

Hence,

$$t_n = -\frac{3^{n+1}}{2} + \frac{5^{n+1}}{2} = -3 \cdot \frac{3^n}{2} + 5 \cdot \frac{5^n}{2}.$$

Thus, we find that the sequence  $\{-3 \cdot \frac{3^n}{2} + 5 \cdot \frac{5^n}{2}\}$  is the solution of the given recurrence relation.

Example 8.3.4 illustrates that a linear nonhomogeneous recurrence relation can be solved by first finding a particular solution of recurrence relations, then solving the homogeneous part of the recurrence relation, and finally adding the solution of the homogeneous part and the particular solution. The following theorem shows that this result is true in general.

**Theorem 8.3.5:** Let

$$a_n + c_1 a_{n-1} + \cdots + c_k a_{n-k} = f(n) \quad (8.62)$$

be a nonhomogeneous recurrence relation, where  $c_i, i = 1, 2, \dots, k$ , are constants,  $c_k \neq 0$ , and  $f(n)$  is a nonzero real-valued function. Suppose  $\{r_n\}$  is a particular solution of (8.62). Then  $\{u_n\}$  is a solution of (8.62) if and only if  $u_n = r_n + s_n$ , for all  $n$ , and  $\{s_n\}$  is a solution of the associated homogeneous part,  $a_n + c_1 a_{n-1} + \cdots + c_k a_{n-k} = 0$ .

**Proof:** Because  $\{r_n\}$  is a solution of (8.62), we have

$$r_n + c_1 r_{n-1} + \cdots + c_k r_{n-k} = f(n). \quad (8.63)$$

Suppose  $\{u_n\}$  is a solution of (8.62). Then

$$u_n + c_1 u_{n-1} + \cdots + c_k u_{n-k} = f(n). \quad (8.64)$$

Subtract the corresponding sides of (8.63) and (8.62) to obtain

$$(r_n - u_n) + c_1(r_{n-1} - u_{n-1}) + \cdots + c_k(r_{n-k} - u_{n-k}) = 0. \quad (8.65)$$

This implies that  $\{r_n - u_n\}$  is a solution of the associated homogeneous recurrence relation

$$a_n + c_1 a_{n-1} + \cdots + c_k a_{n-k} = 0.$$

Let us write  $s_n = r_n - u_n$ . Then  $\{s_n\}$  is a solution of the associated homogeneous part,  $a_n + c_1 a_{n-1} + \cdots + c_k a_{n-k} = 0$ , and  $u_n = r_n + s_n$ .

Now suppose that  $\{s_n\}$  is a solution of the associated homogeneous recurrence relation  $a_n + c_1 a_{n-1} + \cdots + c_k a_{n-k} = 0$ . Then

$$s_n + c_1 s_{n-1} + \cdots + c_k s_{n-k} = 0. \quad (8.66)$$

Add the corresponding sides of (8.63) and (8.66) to get

$$(r_n + s_n) + c_1(r_{n-1} + s_{n-1}) + \cdots + c_k(r_{n-k} + s_{n-k}) = f(n).$$

This implies that  $\{r_n + s_n\}$  is a solution of (8.62). ■

By Theorem 8.3.5, we can solve a nonhomogeneous recurrence relation by first finding a particular solution. So the key part is finding a particular solution. As we will show later in this section, finding a particular solution may not be obvious. We will also develop another method of solving a linear nonhomogeneous recurrence relation when  $f(n)$  is a polynomial.

The polynomial  $p(n)$  in Example 8.3.4 equals 1. In such cases, i.e., when  $p(n)$  is a constant polynomial, we can solve the recurrence relation without first finding a particular solution. This important result is proved in the next theorem.

**Theorem 8.3.6:** Let

$$a_n - da_{n-1} = b^n u, \quad n \geq 1 \quad (8.67)$$

be a nonhomogeneous linear recurrence relation, with the initial condition

$$a_0 = e_0, \quad (8.68)$$

where  $d$ ,  $b$ ,  $u$ , and  $e_0$  are constants, and  $b$  and  $u$  are nonzero. This nonhomogeneous linear recurrence relation can be transformed into the following linear homogeneous recurrence relation:

$$a_n - (b + d)a_{n-1} + bda_{n-2} = 0, \quad n \geq 2$$

with the initial conditions  $a_0 = e_0$  and  $a_1 = de_0 + bu$ .

Moreover,

- (i) if  $b \neq d$ , then there exists a constant  $c_0$ , which is to be determined from the initial condition, such that

$$a_n = c_0 d^n + \left( \frac{bu}{b-d} \right) b^n.$$

- (ii) if  $b = d$ , then there exists a constant  $c_0$ , which is to be determined from the initial condition, such that

$$a_n = c_0 b^n + unb^n.$$

**Proof:** Now

$$a_n - da_{n-1} = b^n u. \quad (8.69)$$

We change  $n$  to  $n - 1$  to get

$$a_{n-1} - da_{n-2} = b^{n-1} u. \quad (8.70)$$

Now a solution of (8.67) is also a solution of (8.69) and (8.70). We multiply (8.70) by  $b$  to get

$$ba_{n-1} - bda_{n-2} = b^n u. \quad (8.71)$$

We subtract (8.71) from (8.69) to get

$$a_n - (b + d)a_{n-1} + bda_{n-2} = 0, \quad n \geq 2. \quad (8.72)$$

By (8.68), we have  $a_0 = e_0$ . Next we substitute  $n = 1$  in (8.67) to get  $a_1 = da_0 + bu = de_0 + bu$ .

It follows that the solution of (8.67) with the given initial condition is also the solution of (8.72) with the initial conditions  $a_0 = e_0$  and  $a_1 = de_0 + bu$ . Next, we obtain the solution of (8.72) with the initial conditions  $a_0 = e_0$  and  $a_1 = de_0 + bu$ .

Now (8.72) is linear homogeneous equation and its characteristic equation is

$$t^2 - (b + d)t + bd = 0$$

or

$$(t - d)(t - b) = 0.$$

Thus, the roots of the characteristic equation are  $d$  and  $b$ .

**Case (i):** Suppose  $b \neq d$ . Then

$$a_n = c_0 d^n + c_1 b^n, \quad (8.73)$$

where  $c_0$  and  $c_1$  are constants, which are to be determined from the initial conditions.

We substitute  $n = 0$  in (8.73) to get

$$\begin{aligned} e_0 &= a_0 = c_0 d^0 + c_1 b^0 = c_0 + c_1 \\ c_0 + c_1 &= e_0. \end{aligned} \quad (8.74)$$

Next we substitute  $n = 1$  in (8.73) to get

$$a_1 = c_0 d + c_1 b,$$

i.e.,

$$c_0 d + c_1 b = d e_0 + b u. \quad (8.75)$$

We multiply (8.74) by  $d$  and subtract from (8.75) to get

$$c_1 b - c_1 d = b u,$$

i.e.,

$$c_1 = \frac{bu}{b-d}.$$

Hence, the sequence  $\{c_0 d^n + \frac{bu}{b-d} b^n\}$  is the solution of the recurrence relation

$$a_n - (b+d)a_{n-1} + bda_{n-2} = 0$$

with the initial conditions  $a_0 = e_0$  and  $a_1 = d e_0 + b u$ , where  $c_0$  is a constant satisfying the initial conditions.

Now any solution of

$$a_n = da_{n-1} + b^n u$$

with the initial condition  $a_0 = e_0$  is the solution of

$$a_n - (b+d)a_{n-1} + bda_{n-2} = 0$$

with the initial conditions  $a_0 = e_0$  and  $a_1 = d e_0 + b u$ . Hence,

$$a_n = c_0 d^n + \frac{bu}{b-d} b^n,$$

where  $c_0$  is a constant satisfying the initial conditions.

Similarly, we can prove part (ii). ■

### EXAMPLE 8.3.7

In this example, we use Theorem 8.3.6 to solve the recurrence relation

$$a_n - 4a_{n-1} = 8^n, \quad n \geq 1,$$

with the initial condition

$$a_0 = 1.$$

This is a recurrence relation of the form

$$a_n - da_{n-1} = b^n u,$$

where  $d = 4$ ,  $b = 8$ , and  $u = 1$ . Because  $b \neq d$ ,

$$\begin{aligned} a_n &= c_0 d^n + \frac{bu}{b-d} b^n \\ &= c_0 4^n + \frac{8}{4} 8^n \\ &= c_0 4^n + 2 \cdot 8^n \end{aligned}$$

for all  $n \geq 0$ , where  $c_0$  is a constant satisfying the initial condition.

Now

$$1 = a_0 = c_0 4^0 + 2 \cdot 8^0 = c_0 + 2.$$

Hence,  $c_0 = -1$ . This implies that  $a_n = -1 \cdot 4^n + 2 \cdot 8^n$  for all  $n \geq 0$ .

**REMARK 8.3.8** ▶ We can also solve the recurrence relation of Example 8.3.7 by finding a particular solution and a solution of the associated homogeneous part.

The polynomial  $p(n)$  in Theorem 8.3.6 is a constant polynomial. Next we consider the case when the polynomial  $p(n)$  is not constant.

### EXAMPLE 8.3.9

In this example, we solve the recurrence relation

$$a_n - a_{n-1} = n, \quad n > 1 \quad (8.76)$$

with the initial condition

$$a_0 = 0. \quad (8.77)$$

First we find a particular solution  $p_n$  of this recurrence relation. Because  $f(n) = n$  is a polynomial of degree 1, we first try  $p_n = c + d \cdot n$ .

Now  $p_n - p_{n-1} = n$  or  $p_n = p_{n-1} + n$ . This implies that

$$\begin{aligned} c + d \cdot n &= (c + d \cdot (n-1)) + n \\ \Rightarrow c + d \cdot n &= (c - d) + (d+1)n. \end{aligned}$$

Comparing the coefficients of  $n$  and the constant term, we get

$$c = c - d \quad \text{and} \quad d = d + 1,$$

i.e.,  $1 = 0$ , a contradiction. So  $p_n = c + d \cdot n$  does not work.

Next we take  $p_n = c + b \cdot n + d \cdot n^2$ . Again  $p_n - p_{n-1} = n$  or  $p_n = p_{n-1} + n$ . This implies that

$$\begin{aligned} c + b \cdot n + d \cdot n^2 &= (c + b \cdot (n-1) + d \cdot (n-1)^2) + n \\ \Rightarrow c + b \cdot n + d \cdot n^2 &= (c - b + d) + (b - 2d + 1) \cdot n + d \cdot n^2. \end{aligned}$$

Comparing the coefficients of  $n^2$ ,  $n$ , and the constant terms, we get

$$c = c - b + d \quad \text{and} \quad b = b - 2d + 1.$$

This implies that  $d = \frac{1}{2}$  and  $b = \frac{1}{2}$ . Because  $c$  is arbitrary, let us choose  $c = 0$ . Thus,  $p_n = \frac{1}{2}n + \frac{1}{2}n^2$ .

Now

$$\begin{aligned} p_n - p_{n-1} &= \frac{1}{2}n + \frac{1}{2}n^2 - \left(\frac{1}{2}(n-1) + \frac{1}{2}(n-1)^2\right) \\ &= \frac{1}{2}n + \frac{1}{2}n^2 - \frac{1}{2}n + \frac{1}{2} - \frac{1}{2}n^2 + n - \frac{1}{2} \\ &=: n. \end{aligned}$$

Hence, we find that  $p_n = \frac{1}{2}n + \frac{1}{2}n^2$  is a particular solution.

Now we consider the homogeneous part,  $a_n - a_{n-1} = 0$ . The characteristic equation is  $t - 1 = 0$  and its root is 1. Then there exists a constant  $c_1$  which is to be determined, such that  $u_n = c_1 \cdot 1^n$ . Hence,

$$t_n = p_n + u_n = \frac{1}{2} \cdot n + \frac{1}{2} \cdot n^2 + c_1 \cdot 1^n.$$

From the initial conditions,

$$\begin{aligned} 0 &= \frac{1}{2} \cdot 0 + \frac{1}{2} \cdot 0^2 + c_1 \cdot 1^0 \\ \Rightarrow c_1 &= 0. \end{aligned}$$

Hence,

$$t_n = \frac{1}{2}n + \frac{1}{2}n^2.$$

This shows that the sequence  $\{\frac{1}{2}n + \frac{1}{2}n^2\}$  is the solution of the given recurrence relation.

We also notice that finding a particular solution is not so obvious. Let us solve recurrence relation (8.76) as follows.

Now

$$a_n - a_{n-1} = n. \quad (8.78)$$

Change  $n$  to  $n - 1$  to get

$$a_{n-1} - a_{n-2} = n - 1. \quad (8.79)$$

Now subtract (8.79) from (8.78) to get

$$a_n - 2a_{n-1} + a_{n-2} = 1. \quad (8.80)$$

Change  $n$  to  $n - 1$  in (8.80) to get

$$a_{n-1} - 2a_{n-2} + a_{n-3} = 1. \quad (8.81)$$

Now subtract (8.81) from (8.80) to get

$$a_n - 3a_{n-1} + 3a_{n-2} - a_{n-3} = 0. \quad (8.82)$$

Let  $\{s_n\}$  be a solution of (8.82). Then

$$s_n - 3s_{n-1} + 3s_{n-2} - s_{n-3} = 0.$$

The characteristic equation of this recurrence relation is

$$t^3 - 3t^2 + 3t - 1 = 0,$$

i.e.,

$$(t - 1)^3 = 0.$$

Thus, 1 is a root of multiplicity 3. From this it follows that  $s_n$  is of the form

$$s_n = k_0 1^n + k_1 n 1^n + k_2 n^2 1^n = k_0 + k_1 n + k_2 n^2,$$

where  $k_0$ ,  $k_1$ , and  $k_2$  are some suitable constants. Because  $\{s_n\}$  is a solution of (8.76), we have  $s_0 = 0$ . Using (8.76) and  $s_0$ , we can find two more initial conditions,  $s_1 = s_0 + 1 = 1$  and  $s_2 = s_1 + 2 = 3$ . Using these initial conditions, we can show that  $k_0 = 0$ ,  $k_1 = \frac{1}{2}$ , and  $k_2 = \frac{1}{2}$ . Thus,

$$s_n = \frac{1}{2}n + \frac{1}{2}n^2.$$

This is the same solution as obtained earlier.

Notice that in  $s_n = k_0 + k_1 n + k_2 n^2$ ,  $k_0$  is a solution of the homogeneous part and  $k_1 n + k_2 n^2$  is a particular solution for suitable constants  $k_1$  and  $k_2$ .

In the next theorem, we consider  $p(n)$  to be a polynomial of degree 1.

**Theorem 8.3.10:** Let

$$a_n - da_{n-1} = b^n(un + v), \quad n \geq 1, \quad (8.83)$$

be a nonhomogeneous linear recurrence relation, with the initial condition

$$a_0 = e_0, \quad (8.84)$$

where  $d$ ,  $b$ ,  $u$ ,  $v$ , and  $e_0$  are constants, and  $b$  and  $u$  are nonzero. This nonhomogeneous linear recurrence relation can be transformed into the following linear homogeneous recurrence relation:

$$a_n - (2b + d)a_{n-1} + b(2d + b)a_{n-2} - b^2da_{n-3} = 0, \quad n \geq 3 \quad (8.85)$$

with the initial conditions

$$a_0 = e_0 \quad \text{and} \quad a_1 = de_0 + b(u + v).$$

Moreover, the characteristic equation of (8.85) is

$$(t - d)(t - b)^2 = 0. \quad (8.86)$$

Let  $\{r_n\}$  be a solution of (8.83).

- (i) Suppose  $b \neq d$ . Then  $r_n$  is of the form

$$r_n = c_0 d^n + c_1 b^n + c_2 n b^n,$$

where  $c_0$ ,  $c_1$ , and  $c_2$  are some constants.

- (ii) Suppose  $b = d$ . Then  $\{r_n\}$  is of the form

$$r_n = c_0 b^n + c_1 n b^n + c_2 n^2 b^n,$$

where  $c_0$ ,  $c_1$ , and  $c_2$  are some constants.

**Proof:** Now

$$a_n - da_{n-1} = b^n(un + v). \quad (8.87)$$

Change  $n$  to  $n - 1$  to get

$$a_{n-1} - da_{n-2} = b^{n-1}(u(n - 1) + v) = b^{n-1}(un + v) - b^{n-1}u. \quad (8.88)$$

Multiply (8.88) by  $b$  to get

$$ba_{n-1} - bda_{n-2} = b^n(un + v) - b^n u. \quad (8.89)$$

Subtract (8.89) from (8.87) to get

$$a_n - (b + d)a_{n-1} + bda_{n-2} = ub^n, \quad n \geq 2. \quad (8.90)$$

In (8.90), change  $n$  to  $n - 1$  to get

$$a_{n-1} - (b + d)a_{n-2} + bda_{n-3} = ub^{n-1}. \quad (8.91)$$

Multiply (8.91) by  $b$  and then subtract it from (8.90) to get

$$a_n - (2b + d)a_{n-1} + b(2d + b)a_{n-2} - b^2da_{n-3} = 0, \quad n \geq 3 \quad (8.92)$$

which is the same as (8.85).

Now  $a_0 = e_0$ . Substitute  $n = 1$  in (8.83) to get  $a_1 = da_0 + b(u + v) = de_0 + b(u + v)$ .

Suppose  $\{r_n\}$  is a solution of (8.83). Then  $\{r_n\}$  is a solution of (8.87) and (8.88). It follows that  $\{r_n\}$  is a solution of (8.89). Because (8.90) is obtained by subtracting (8.89) from (8.87), it follows that  $\{r_n\}$  is a solution of (8.90). This in turn implies that  $\{r_n\}$  is a solution of (8.91). It now follows that  $\{r_n\}$  is a solution of (??).

Hence,

$$r_n - (2b + d)r_{n-1} + b(2d + b)r_{n-2} - b^2dr_{n-3} = 0, \quad n \geq 3. \quad (8.93)$$

Moreover, it also follows that if  $\{r_n\}$  satisfies the initial condition (8.84), then it also satisfies the additional initial condition  $r_1 = de_0 + b(u + v)$ .

Now (8.93) is a linear homogeneous recurrence relation and its characteristic equation is:

$$t^3 - (2b + d)t^2 + b(2d + b)t - b^2d = 0$$

or

$$(t - d)(t - b)^2 = 0.$$

The roots of this equation are  $d$  and  $b$ , and  $b$  is a root of multiplicity 2.

- (i) Suppose  $b \neq d$ . Then  $b$  is a root of multiplicity 2. Hence, by Theorem 8.2.19,  $r_n$  is of the form

$$r_n = c_0d^n + c_1b^n + c_2nb^n,$$

where  $c_0$ ,  $c_1$ , and  $c_2$  are some constants.

- (ii) Suppose  $b = d$ . Then  $b$  is a root of multiplicity 3. By Theorem 8.2.19,  $r_n$  is of the form

$$r_n = c_0b^n + c_1nb^n + c_2n^2b^n,$$

where  $c_0$ ,  $c_1$ , and  $c_2$  are some constants. ■

### EXAMPLE 8.3.11

Consider the recurrence relation

$$a_n - 3a_{n-1} = 2^n(4n + 3), \quad n > 1 \quad (8.94)$$

with initial conditions

$$a_0 = 0,$$

$$a_1 = 14.$$

This is a recurrence relation of the form

$$a_n - da_{n-1} = b^n(un + v).$$

Here  $d = 3$ ,  $b = 2$ ,  $u = 4$ , and  $v = 3$ .

We can solve this recurrence by using the technique of Theorem 8.3.10 and obtaining

$$a_n = c_0 3^n + c_1 2^n + c_2 n 2^n,$$

where  $c_0$ ,  $c_1$ , and  $c_2$  are constants, which are to be determined from the initial conditions.

Put  $n = 2$  in (8.92) to get

$$a_2 - 3a_1 = 2^2(4 \cdot 2 + 3) = 44.$$

Because  $a_1 = 14$ , we get

$$a_2 = 3 \cdot 14 + 44 = 86.$$

Thus,

$$\begin{aligned} a_0 &= c_0 + c_1 = 0 \\ a_1 &= c_0 \cdot 3 + c_1 \cdot 2 + c_2 \cdot 2 = 14 \\ a_2 &= c_0 \cdot 3^2 + c_1 \cdot 2^2 + c_2 \cdot 2 \cdot 2^2 = 86 \end{aligned}$$

This implies that

$$\begin{aligned} c_0 + c_1 &= 0 \\ 3c_0 + 2c_1 + 2c_2 &= 14 \\ 9c_0 + 4c_1 + 8c_2 &= 86 \end{aligned}$$

We solve these equations for  $c_0$ ,  $c_1$ , and  $c_2$  to obtain  $c_0 = 30$ ,  $c_1 = -30$ , and  $c_2 = -8$ . Thus, we find that

$$a_n = 30(3^n) - 30(2^n) - n2^{n+3}, \quad n \geq 0. \quad (8.95)$$

We leave it as an exercise to verify that this  $a_n$  is the solution of the recurrence relation in (8.94) with the initial conditions  $a_0 = 0$  and  $a_1 = 14$ .

### EXAMPLE 8.3.12

Let us consider again the recurrence relation of Example 8.3.9, i.e.,

$$a_n - a_{n-1} = n, \quad n > 1$$

with the initial condition

$$a_0 = 0.$$

Let us apply Theorem 8.3.10 to find the solution of this recurrence relation. This is a recurrence relation of the form

$$a_n - da_{n-1} = b^n(un + v),$$

where  $d = 1$ ,  $b = 1$ ,  $u = 1$ , and  $v = 0$ . Here we notice that  $b = d$ .

Using the technique of Theorem 8.3.10, we obtain

$$a_n = c_0 1^n + c_1 n 1^n + c_2 n^2 1^n,$$

where  $c_0$ ,  $c_1$ , and  $c_2$  are constants, which are to be determined from the initial conditions.

Now from the given recurrence relation and initial condition we find that  $a_1 - a_0 = 1$ . Thus,  $a_1 = 1$ . Now  $a_2 - a_1 = 2$  implies that  $a_2 = 3$ . Hence,

$$\begin{aligned} a_0 &=: c_0 = 0 \\ a_1 &=: c_0 \cdot 1 + c_1 \cdot 1 + c_2 \cdot 1 = 1 \\ a_2 &=: c_0 \cdot 1 + c_1 \cdot 2 + c_2 \cdot 4 \cdot 1 = 3 \end{aligned}$$

Solving these equations we get  $c_0 = 0$ ,  $c_1 = \frac{1}{2}$ , and  $c_2 = \frac{1}{2}$ . Hence,

$$a_n = 0 + \frac{1}{2}n + \frac{1}{2}n^2 = \frac{1}{2}(n^2 + n) = \frac{n(n+1)}{2}, \quad n \geq 0.$$

We leave it as an exercise to verify that this  $a_n$  is the solution of the given recurrence relation with the given initial condition.

The characteristic polynomials in Theorems 8.3.6 and 8.3.10 are special cases of the following theorem, which can be proved by induction. We leave the proof as an exercise.

**Theorem 8.3.13:** Let

$$a_n + d_1 a_{n-1} + \cdots + d_k a_{n-k} = b^n p(n) \quad (8.96)$$

be a nonhomogeneous linear recurrence relation, where  $p(n)$  is a polynomial of degree  $m$ . Then from this nonhomogeneous linear recurrence relation we can obtain a linear homogeneous recurrence that has following characteristic equation:

$$(t^k + d_1 t^{k-1} + \cdots + d_k)(t - b)^{m+1} = 0. \quad (8.97)$$

Moreover, a solution of (8.96) is also a solution of the linear homogeneous recurrence whose characteristic equation is given by (8.97).

---

**REMARK 8.3.14** ► As we can see, there are two ways to solve a linear nonhomogeneous equation of the form

$$a_n + d_1 a_{n-1} + \cdots + d_k a_{n-k} = b^n p(n)$$

with some given initial conditions.

1. First find a particular solution and then add the particular solution to a solution of the associated linear homogeneous recurrence relation.
2. First obtain a linear homogeneous recurrence relation from the given linear nonhomogeneous recurrence relation, as shown in this section. Then a solution of the given linear nonhomogeneous recurrence relation is also a solution of the linear homogeneous equation obtained. Next find a solution of the homogeneous recurrence relation and use the initial conditions

of the nonhomogeneous recurrence solution to find the constants. Finally, verify that the solution obtained satisfies the linear nonhomogeneous recurrence relation. The following example illustrates this.

**EXAMPLE 8.3.15**

Consider the nonhomogeneous recurrence relation

$$a_n + 2a_{n-1} - 3a_{n-2} = 2^n(n^2 + n + 1) \quad (8.98)$$

with the initial conditions  $a_0 = 0$  and  $a_1 = 1$ . Here  $b^n p(n) = 2^n(n^2 + n + 1)$ . Thus,  $b = 2$  and  $p(n) = n^2 + n + 1$ . Now  $p(n)$  is a polynomial of degree  $m = 2$ . We can therefore obtain a homogeneous recurrence relation whose characteristic equation is

$$(t^2 + 2t - 3)(t - b)^{m+1} = 0,$$

i.e.,

$$(t^2 + 2t - 3)(t - 2)^3 = 0,$$

i.e.,

$$(t - 1)(t + 3)(t - 2)^3 = 0. \quad (8.99)$$

The roots of this equation are 1, 2, and  $-3$ . Moreover, 2 is a root of multiplicity 3. Hence, a solution of the homogeneous recurrence relation whose characteristic equation is (8.99) is

$$a_n = c_0 1^n + c_1 (-3)^n + c_2 2^n + c_3 n 2^n + c_4 n^2 2^n$$

or

$$a_n = c_0 + c_1 (-3)^n + c_2 2^n + c_3 n 2^n + c_4 n^2 2^n \quad (8.100)$$

for some constants  $c_0, c_1, c_2, c_3$ , and  $c_4$ .

Now  $a_0 = 0$  and  $a_1 = 1$ . Using these and (8.98), we get  $a_2 = 26$ ,  $a_3 = 55$ , and  $a_4 = 304$ . Using these initial conditions and (8.100), we can show that  $c_0 = \frac{-2750}{1000}$ ,  $c_1 = \frac{734}{1000}$ ,  $c_2 = \frac{2016}{1000}$ ,  $c_3 = \frac{160}{1000}$ , and  $c_4 = \frac{800}{1000}$ . Thus,

$$a_n = \frac{-2750}{1000} + \frac{734}{1000}(-3)^n + \frac{2016}{1000}2^n + \frac{160}{1000}n2^n + \frac{800}{1000}n^22^n.$$

A straightforward computation shows that this  $a_n$  is the solution of (8.98) with the given initial conditions.

In the preceding examples, we considered linear nonhomogeneous recurrence relations of the form

$$a_n + d_1 a_{n-1} + \cdots + d_k a_{n-k} = b^n p(n).$$

Very often we come across linear nonhomogeneous recurrence relations of the form

$$a_n + d_1 a_{n-1} + \cdots + d_k a_{n-k} = b_1^n p(n) + b_2^n q(n),$$

where  $b_1$  and  $b_2$  are constants and  $p(n)$  and  $q(n)$  are polynomials. It turns out that these types of recurrence relations can be solved using the techniques developed in this section. In other words, we can first obtain a homogeneous recurrence relation and find a solution of this homogeneous recurrence relation, and then

use this solution with the initial conditions of the nonhomogeneous recurrence relation to obtain the solution of the nonhomogeneous recurrence relation. The following example illustrates this.

**EXAMPLE 8.3.16**

In this example we consider the linear nonhomogeneous recurrence relation

$$a_n - 2a_{n-1} = n + 2^n, \quad n \geq 1 \quad (8.101)$$

with the initial condition

$$a_0 = 0. \quad (8.102)$$

This recurrence relation is not of the form

$$a_n - da_{n-1} = b^n(un + v), \quad n \geq 1,$$

so we cannot apply Theorem 8.3.10 directly. However, we can still obtain a linear homogeneous recurrence relation as follows.

In (8.101), change  $n$  to  $n - 1$  to get

$$a_{n-1} - 2a_{n-2} = n - 1 + 2^{n-1}. \quad (8.103)$$

Multiply (8.101) by 2 to get

$$2a_{n-1} - 4a_{n-2} = 2n - 2 + 2^n. \quad (8.104)$$

Subtract (8.104) from (8.103) to get

$$a_n - 4a_{n-1} + 4a_{n-2} = -n + 2, \quad n \geq 2. \quad (8.105)$$

From (8.101) and (8.102), we can obtain  $a_0 = 0$  and  $a_1 = 1 + 2 = 3$ .

In (8.105), change  $n$  to  $n - 1$  to get

$$a_{n-1} - 4a_{n-2} + 4a_{n-3} = -(n - 1) + 2. \quad (8.106)$$

Subtract (8.106) from (8.105) to get

$$a_n - 5a_{n-1} + 8a_{n-2} - 4a_{n-3} = -1, \quad n \geq 3. \quad (8.107)$$

From (8.101) and (8.102), we can obtain  $a_0 = 0$ ,  $a_1 = 3$ , and  $a_2 = 12$ .

In (8.107), change  $n$  to  $n - 1$  to get

$$a_{n-1} - 5a_{n-2} + 8a_{n-3} - 4a_{n-4} = -1. \quad (8.108)$$

Subtract (8.108) from (8.107) to get

$$a_n - 6a_{n-1} + 13a_{n-2} - 12a_{n-3} + 4a_{n-4} = 0, \quad n \geq 4. \quad (8.109)$$

From (8.101) and (8.102), we can obtain  $a_0 = 0$ ,  $a_1 = 3$ ,  $a_2 = 12$ , and  $a_3 = 35$ .

It follows that the solution of (8.101) with the initial condition  $a_0 = 0$  is also the solution of (8.109) with the initial conditions  $a_0 = 0$ ,  $a_1 = 3$ ,  $a_2 = 12$ , and  $a_3 = 35$ .

Now (8.109) is a linear homogeneous recurrence relation. Its characteristic equation is

$$t^4 - 6t^3 + 13t^2 - 12t + 4 = 0,$$

i.e.,

$$(t - 2)^2(t - 1)^2 = 0. \quad (8.110)$$

The roots of this equation are  $t = 1$  and  $t = 2$ , and both roots are of multiplicity 2. Then

$$a_n = c_1 1^n + c_2 n 1^n + c_3 2^n + c_4 n 2^n$$

or

$$a_n = c_1 + c_2 n + c_3 2^n + c_4 n 2^n, \quad (8.111)$$

where  $c_1, c_2, c_3$ , and  $c_4$  are constants, which are to be determined from the initial conditions of the recurrence relation (8.101). From  $a_0 = 0$  and using (8.101), we can show that  $a_1 = 3$ ,  $a_2 = 12$ , and  $a_3 = 35$ .

Next, using these initial conditions, we obtain  $c_1 = -2$ ,  $c_2 = -1$ ,  $c_3 = 2$ , and  $c_4 = 1$ . Thus,

$$a_n = -2 - n + 2 \cdot 2^n + n \cdot 2^n = -2 - n + 2^{n+1} + n \cdot 2^n, \quad n \geq 0.$$

It can be verified that this  $a_n$  is the solution of (8.101) with the given initial conditions.

---

**REMARK 8.3.17** ▶ Let us take another look at recurrence relation (8.101), i.e.,

$$a_n - 2a_{n-1} = n + 2^n, \quad n \geq 1.$$

There are two terms on the right side of the linear nonhomogeneous equation:  $n$  and  $2^n$ . Now

$$n = (1^n)n^1 \quad \text{and} \quad 2^n = (2^n)n^0.$$

Let us consider the term  $n = (1^n)n^1$ . Here  $b = 1$  and  $p(n) = n$  is of degree  $m = 1$ . This is why we get the factor  $(t - 1)^{m+1} = (t - 1)^2$  in (8.110).

Now consider the term  $2^n = (2^n)n^0$ . Here  $b = 2$  and  $p(n) = 1 = n^0$  is of degree 0. This is why we get the factor  $(t - 2)^{m+0} = (t - 2)$  in (8.110).

Notice that the other factor,  $(t - 2)$ , in (8.110) is from the associated linear homogeneous recurrence relation.



## WORKED-OUT EXERCISES

---

**Exercise 1:** In Worked-Out Exercise 5, page 507, we obtained the recurrence relation

$$a_n = 0.8a_{n-1} + 25, \quad n \geq 1 \quad (8.112)$$

with the initial condition  $a_0 = 0$ . Show that the general solution of this recurrence relation is  $a_n = 125(1 - (0.8)^n)$ ,  $n \geq 0$ .

**Solution:** The recurrence relation  $a_n = 0.8a_{n-1} + 25$ ,  $n \geq 1$  is a linear nonhomogeneous recurrence relation of the form (8.67). Here  $d = 0.8$ ,  $b = 1$ , and  $u = 25$ . Thus, by Theorem 8.3.6, a solution is given by

$$\begin{aligned} a_n &= a_0 d^n + \frac{bu}{b-d} b^n = a_0 (0.8)^n + \frac{1 \cdot 25}{1 - 0.8} 1^n \\ &= a_0 (0.8)^n + \frac{25}{0.2} = a_0 (0.8)^n + 125 \end{aligned}$$

i.e.,

$$a_n = c_0 (0.8)^n + 125, \quad (8.113)$$

for some constant  $c_0$ . Next we determine the value of  $c_0$ . For this we use the initial condition,  $a_0 = 0$ .

We substitute  $n = 0$  in (8.113) to get

$$a_0 = c_0 + 125.$$

Thus,

$$c_0 + 125 = 0,$$

i.e.,

$$c_0 = -125.$$

Thus, the solution is

$$a_n = 125(1 - (0.8)^n), \quad n \geq 0.$$

**Exercise 2:** Solve the recurrence

$$a_n - 3a_{n-1} = 2n, \quad n \geq 1 \quad (8.114)$$

with the initial condition  $a_0 = 0$ .

**Solution:** This is a recurrence relation of the form

$$a_n - da_{n-1} = b^n(un + v)$$

with  $d = 3$ ,  $b = 1$ ,  $u = 3$ , and  $v = 0$ . Here, notice that  $b \neq d$ .

Using the technique of Theorem 8.3.10, we can show that

$$a_n = c_0 3^n + c_1 1^n + c_2 n 1^n, \quad (8.115)$$

where  $c_0$ ,  $c_1$ , and  $c_2$  are constants, which are to be determined from the initial conditions.

Now from the given recurrence relation and the initial condition  $a_0 = 0$ , we find that  $a_1 - 3a_0 = 2$ , i.e.,  $a_1 = 2$ . Also,  $a_2 - 3a_1 = 4$  implies that  $a_2 = 10$ .

Substitute  $n = 0, 1$ , and  $2$ , respectively, in (8.115) to get

$$a_0 = c_0 + c_1 = 0$$

$$a_1 = 3c_0 + c_1 + c_2 = 2$$

$$a_2 = 9c_0 + c_1 + 2c_2 = 10.$$

Solve these equations to get  $c_0$ ,  $c_1$ , and  $c_2$  to obtain  $c_0 = \frac{3}{2}$ ,  $c_1 = -\frac{3}{2}$ , and  $c_2 = -1$ . Hence,

$$a_n = \frac{3}{2}(3^n) - \frac{3}{2} - n, \quad n \geq 0. \quad (8.116)$$

It can be verified that  $a_n$ , given by (8.116), is the solution of (8.114) with the given initial conditions.

**Exercise 3:** Solve the following recurrence relation

$$a_n - 2a_{n-1} - 3a_{n-2} = 5^n, \quad n \geq 2 \quad (8.117)$$

with the initial conditions  $a_0 = -1$  and  $a_1 = 1$ .

**Solution:** In (8.117), replace 5 by  $n - 1$  to get

$$a_{n-1} - 2a_{n-2} - 3a_{n-3} = 5^{n-1}. \quad (8.118)$$

Multiply (8.118) by 5 and subtract from (8.117) to get

$$a_n - 7a_{n-1} + 7a_{n-2} + 15a_{n-3} = 0. \quad (8.119)$$

This is a linear homogeneous recurrence relation. It follows that a solution of (8.117) is also a solution of (8.119). The characteristic equation of (8.119) is

$$t^3 - 7t^2 + 7t + 15 = 0,$$

i.e.,

$$(t+1)(t-3)(t-5) = 0. \quad (8.120)$$

(Notice that  $(t+1)(t-3) = 0$  is the characteristic equation of the associated homogeneous part.) The roots of (8.120) are  $t = -1$ ,  $t = 3$ , and  $t = 5$ . Thus,

$$a_n = c_0(-1)^n + c_1 3^n + c_2 5^n \quad (8.121)$$

where  $c_0$ ,  $c_1$ , and  $c_2$  are constants, which are to be determined by initial conditions.

Now from (8.117) and the given initial conditions  $a_0 = -1$ ,  $a_1 = 1$ , we can get  $a_2 = 26$ .

Substitute  $n = 0, 1$ , and  $2$ , respectively, in (8.121) to get

$$a_0 = c_0 + c_1 = -1$$

$$a_1 = -c_0 + 3c_1 + 5c_2 = 1$$

$$a_2 = c_0 + 9c_1 + 25c_2 = 26.$$

Solve these equations to get  $c_0 = \frac{1}{8}$ ,  $c_1 = -\frac{27}{8}$ , and  $c_2 = \frac{9}{4}$ . Hence,

$$a_n = \frac{1}{8} \cdot (-1)^n - \frac{27}{8} \cdot 3^n + \frac{9}{4} \cdot 5^n, \quad n \geq 0.$$

It can be verified that this  $a_n$  satisfies (8.117) and its initial conditions.

**Exercise 4:** Solve the recurrence relation

$$\begin{aligned} w_n &= 2w_{\frac{n}{2}} + n - 1, & n > 1, \\ &n = 2^k \text{ for some integer } k, \end{aligned} \quad (8.122)$$

with the initial condition

$$w_1 = 0. \quad (8.123)$$

**Solution:** First we transform this recurrence into a recurrence relation that can be solved using the techniques of Theorem 8.3.6. So we put  $n = 2^k$  into (8.122) to obtain the equivalent recurrence relation

$$w_{2^k} = 2w_{2^{k/2}} + 2^k - 1$$

or

$$w_{2^k} = 2w_{2^{k-1}} + 2^k - 1. \quad (8.124)$$

Let us write  $a_k = w_{2^k}$  and substitute it into (8.124) to obtain the recurrence relation

$$a_k = 2a_{k-1} + 2^k - 1$$

or

$$a_k - 2a_{k-1} = 2^k - 1, \quad k \geq 1. \quad (8.125)$$

Also,  $a_0 = w_{2^0} = w_1 = 0$ ,  $a_1 = w_{2^1} = w_2 = 1$ , and  $a_2 = w_{2^2} = w_4 := 5$ .

Now (8.125) is a linear nonhomogeneous recurrence relation. The homogeneous part is  $a_k - 2a_{k-1} = 0$ . Therefore, the characteristic equation corresponding to the homogeneous part is  $t - 2 = 0$ .

There are two nonhomogeneous terms in the right side, which are  $2^k$  and  $-1$ .

For the term  $2^k$ , we have  $b = 2$  and  $p(k) = 1$ . Thus, the term for the characteristic equation corresponding to this term is

$$(t - b)^{0+1} = t - 2.$$

For the second term,  $-1 = 1^k(-1)$ , so we can take  $b = 1$  and  $p(k) = -1$ . Thus, the term for the characteristic equation corresponding to this term is

$$(t - b)^{0+1} = t - 1.$$

Therefore, the characteristic equation corresponding to (8.125) is

$$(t - 2)(t - 2)(t - 1) = 0.$$

The roots of this equation are  $t = 1$  and  $t = 2$ , and 2 is a root of multiplicity 2. Thus,

$$a_k = c_1 + c_2 2^k + c_3 k 2^k, \quad (8.126)$$

where  $c_1$ ,  $c_2$ , and  $c_3$  are constants, which are to be determined from the initial conditions  $a_0 = 0$ ,  $a_1 = 1$ , and  $a_2 = 5$ . Using these initial conditions, we can show that  $c_1 = 1$ ,  $c_2 = -1$ , and  $c_3 = 1$ . Hence,

$$a_k = 1 - 1 \cdot 2^k + k 2^k.$$

Next, we substitute  $a_k = w_{2^k} = w_n$  and  $k = \lg n$  to obtain

$$w_n = 1 - n + n \lg n = n \lg n - (n - 1),$$

where  $\lg n = \log_2 n$ . It is easy to see that it is a solution of (8.122) when  $n$  is a power of 2.

Note that to obtain a solution we first showed that  $a_k = 1 - 1 \cdot 2^k + k 2^k$ . However, we can obtain the same solution as follows.

In (8.126), substitute  $a_k = w_{2^k} = w_n$  and  $k = \lg n$  to obtain

$$w_n = c_1 + c_2 n + c_3 n \lg n.$$

We now use the initial condition of (8.122), which is (8.123), to determine the values of  $c_1$ ,  $c_2$ , and  $c_3$ . Because we have only one initial condition,  $w_1 = 0$ , and three unknowns, we use this initial condition to obtain two more initial conditions by substituting  $n = 2$  and  $n = 4$  in (8.122):

$$\begin{aligned} w_2 &= 2w_1 + 2 - 1 = 1 \\ w_4 &= 2w_2 + 4 - 1 = 5. \end{aligned}$$

Next, using the initial conditions, we get the following equations:

$$\begin{aligned} c_1 + c_2 &= 0 \\ c_1 + 2c_2 + 2c_3 &= 1 \\ c_1 + 4c_2 + 8c_3 &= 5. \end{aligned}$$

It can be shown that  $c_1 = 1$ ,  $c_2 = -1$ , and  $c_3 = 1$ . Thus,  $w_n$  is

$$w_n = 1 - n + n \lg n = n \lg n - (n - 1),$$

where  $n = 2^k$  for some integer  $k$ . It can be verified that this  $w_n$  satisfies the given recurrence relation.

**Exercise 5:** Solve the recurrence relation

$$b_n = 3b_{n/2} + \left(\frac{n}{2}\right)^2 - 5\left(\frac{n}{2}\right) + 7, \quad (8.127)$$

where  $n = 2^k$ , for some positive integer  $k$ , with the initial condition  $b_1 = 0$ .

**Solution:** As in the previous exercise, first we transform the recurrence relation (8.127) into a recurrence relation that can be solved using the techniques of Theorem 8.3.6. So we put  $n = 2^k$  in (8.127) to obtain the equivalent recurrence

relation

$$b_{2^k} = 3b_{2^{k/2}} + \left(\frac{2^k}{2}\right)^2 - 5\left(\frac{2^k}{2}\right) + 7$$

or

$$b_{2^k} = 3b_{2^{k-1}} + (2^{k-1})^2 - 5 \cdot 2^{k-1} + 7,$$

or

$$b_{2^k} = 3b_{2^{k-1}} + \frac{1}{4}4^k - \frac{5}{2}2^k + 7. \quad (8.128)$$

Let us write  $a_k = b_{2^k}$  and substitute it in (8.128) to obtain the recurrence relation

$$a_k = 3a_{k-1} + \frac{1}{4}4^k - \frac{5}{2}2^k + 7$$

or

$$a_k - 3a_{k-1} = \frac{1}{4}4^k - \frac{5}{2}2^k + 7. \quad (8.129)$$

This is a linear nonhomogeneous recurrence relation. The homogeneous part is  $a_k - 3a_{k-1} = 0$ . Therefore, the characteristic equation corresponding to the homogeneous part is  $t - 3 = 0$ .

There are three nonhomogeneous terms in the right side, which are  $\frac{1}{4}4^k = 4^k \frac{1}{4}$ ,  $-\frac{5}{2}2^k = 2^k(-\frac{5}{2})$ , and  $7 = 1^k 7$ .

For the term  $4^k \frac{1}{4}$ , we have  $b = 4$  and  $p(k) = \frac{1}{4}$ . Thus, the term for the characteristic equation corresponding to this term is

$$(t - b)^{0+1} = t - 4.$$

For the second term,  $2^k(-\frac{5}{2})$  we can take  $b = 2$  and  $p(k) = -\frac{5}{2}$ . Thus, the term for the characteristic equation corresponding to this term is

$$(t - 2)^{0+1} = t - 2.$$

For the third term,  $1^k 7$  we can take  $b = 1$  and  $p(k) = 7$ . Thus, the term for the characteristic equation corresponding to this term is

$$(t - 1)^{0+1} = t - 1.$$

We can obtain a linear homogeneous recurrence relation (8.129) so that the characteristic equation of the linear homogeneous recurrence relation obtained is

$$(t - 1)(t - 2)(t - 3)(t - 4) = 0 = 0. \quad (8.130)$$

Moreover, a solution of (8.129) is also a solution of the linear homogeneous recurrence relation whose characteristic equation is given in (8.130). The roots of this equation are  $t = 1$ ,  $t = 2$ ,  $t = 3$ , and  $t = 4$ . Thus,  $a_n$  is of the form

$$a_k = c_1 + c_2 2^k + c_3 3^k + c_4 4^k.$$

Next, we substitute  $a_k = b_{2^k} = b_n$  and  $k = \lg n$ , to obtain

$$b_n = c_1 + c_2 n + c_3 3^{\lg n} + c_4 n^2,$$

where  $\lg n = \log_2 n$ .

We now use the initial conditions of the given nonhomogeneous recurrence relation to determine the values of  $c_1$ ,  $c_2$ ,  $c_3$ , and  $c_4$ . Because we have only one initial condition,

$b_1 = 0$ , and four unknowns, we use this initial condition to obtain three more initial conditions by substituting  $n = 2$ ,  $n = 4$ , and  $n = 8$  in (8.127):

$$\begin{aligned} b_2 &= 3b_1 + \left(\frac{2}{2}\right)^2 - 5\left(\frac{2}{2}\right) + 7 = 3b_1 + 1 - 5 + 7 = 3, \\ b_4 &= 3b_2 + \left(\frac{4}{2}\right)^2 - 5\left(\frac{4}{2}\right) + 7 = 3b_2 + 4 - 10 + 7 = 10, \\ b_8 &= 3b_4 + \left(\frac{8}{2}\right)^2 - 5\left(\frac{8}{2}\right) + 7 = 3b_4 + 16 - 20 + 7 = 33. \end{aligned}$$

Next, using the initial conditions, we get the following equations:

$$\begin{aligned} c_1 + c_2 + c_3 + c_4 &= 0 \\ c_1 + 2c_2 + 3c_3 + 4c_4 &= 3 \\ c_1 + 4c_2 + 9c_3 + 16c_4 &= 10 \\ c_1 + 8c_2 + 27c_3 + 64c_4 &= 33. \end{aligned}$$

It can be shown that  $c_1 = -\frac{7}{2}$ ,  $c_2 = 5$ ,  $c_3 = -\frac{5}{2}$ , and  $c_4 = 1$ . Thus,  $b_n$  is

$$\begin{aligned} b_n &= -\frac{7}{2} + 5n - \frac{5}{2}3^{\lg n} + n^2 \\ &\approx -3.5 + 5n - 2.5n^{1.58} + n^2, \quad \text{because } 3^{\lg n} = n^{\lg 3} \approx n^{1.58} \end{aligned}$$

where  $n = 2^k$  for some integer  $k$ . It can be shown that  $b_n = -\frac{7}{2} + 5n - \frac{5}{2}3^{\lg n} + n^2$  satisfies the given recurrence relation.

## SECTION REVIEW

### Key Term

linear nonhomogeneous recurrence relation

### Key Definition

1. A linear nonhomogeneous recurrence relation with constants coefficients is a recurrence relation of the form  $a_n + c_1a_{n-1} + \cdots + c_k a_{n-k} = f(n)$ , where  $c_i$ ,  $i = 1, 2, \dots, k$ , are constants,  $c_k \neq 0$ , and  $f(n)$  is a nonzero real-valued function.

### Some Key Results

1. Let  $a_n - da_{n-1} = b^n u$ ,  $n \geq 1$  be a nonhomogeneous linear recurrence relation, with the initial condition  $a_0 = e_0$ , where  $d$ ,  $b$ ,  $u$ , and  $e_0$  are constants and  $b$  and  $u$  are nonzero. This nonhomogeneous linear recurrence relation can be transformed into the linear homogeneous recurrence relation  $a_n - (b+d)a_{n-1} + bda_{n-2} = 0$ ,  $n \geq 2$ , with the initial conditions  $a_0 = e_0$  and  $a_1 = de_0 + bu$ .

Moreover,

- (i) if  $b \neq d$ , then there exists a constant  $c_0$ , which is to be determined from the initial condition, such that  $a_n = c_0d^n + (\frac{bu}{b-d})b^n$ .
- (ii) if  $b = d$ , then there exists a constant  $c_0$ , which is to be determined from the initial condition, such that  $a_n = c_0b^n + unb^n$ .

2. Let  $a_n - da_{n-1} = b^n(un + v)$ ,  $n \geq 1$ , be a nonhomogeneous linear recurrence relation, with the initial condition  $a_0 = e_0$ , where  $d$ ,  $b$ ,  $u$ ,  $v$ , and  $e_0$  are constants and  $b$  and  $u$  are nonzero. This nonhomogeneous linear recurrence relation can be transformed into the linear homogeneous recurrence relation  $a_n - (2b+d)a_{n-1} + b(2d+b)a_{n-2} - b^2da_{n-3} = 0$ ,  $n \geq 3$ , with the initial conditions  $a_0 = e_0$  and  $a_1 = de_0 + b(u+v)$ . Moreover, the characteristic equation of the preceding linear homogeneous recurrence relation is  $(t-d)(t-b)^2 = 0$ .

Let  $\{r_n\}$  be a solution of  $a_n = da_{n-1} + b^n(un + v)$ ,  $n \geq 1$ .

- (i) Suppose  $b \neq d$ . Then  $r_n$  is of the form  $r_n = c_0d^n + c_1b^n + c_2nb^n$ , where  $c_0$ ,  $c_1$ , and  $c_2$  are some constants.
- (ii) Suppose  $b = d$ . Then  $\{r_n\}$  is of the form  $r_n = c_0b^n + c_1nb^n + c_2n^2b^n$ , where  $c_0$ ,  $c_1$ , and  $c_2$  are some constants.

## EXERCISES

1. Let  $S_n = 1 + 2 + \dots + n$ . Construct a recurrence relation for the sequence  $S_1, S_2, S_3, S_4, \dots$ . Is your recurrence relation homogeneous or nonhomogeneous?
2. Solve the recurrence relation  $a_n - 9a_{n-1} = 8^n$ ,  $n \geq 1$ ,  $a_0 = 4$ .
3. Solve the recurrence relation  $a_n - a_{n-1} = n$ ,  $n \geq 1$ ,  $a_0 = 9$ .
4. Solve the recurrence relation  $a_n - 3a_{n-1} - 4a_{n-2} = 5$ ,  $n \geq 2$ ,  $a_0 = 4$ ,  $a_1 = 3$ .
5. Solve the recurrence relation  $a_n - 8a_{n-1} = 10^{n-1}$  for all  $n \geq 1$  and  $a_0 = 1$ .
6. Solve the recurrence relation  $a_n + a_{n-1} = 2^n$ , if  $n \geq 1$ ,  $a_0 = 3$ .
7. Solve the recurrence relation  $a_n + 3a_{n-1} - 4a_{n-2} = 5^n$ ,  $n \geq 2$ ,  $a_0 = 4$ ,  $a_1 = 3$ .
8. Solve the recurrence relation  $a_n + a_{n-1} = n2^n$ , if  $n \geq 1$  and  $a_0 = 1$ ,  $a_1 = 2$ .
9. Solve the recurrence relation  $a_n + 3a_{n-1} - 4a_{n-2} = n2^n$ ,  $n \geq 2$ ,  $a_0 = 1$ ,  $a_1 = 1$ .
10. Solve the recurrence relation  $a_n - 4a_{n-1} + 4a_{n-2} = n$ ,  $n \geq 2$ ,  $a_0 = 1$ ,  $a_1 = 1$ .
11. Solve the recurrence relation  $a_n - 6a_{n-1} + 9a_{n-2} = 2^n$ , if  $n \geq 2$ , with the initial conditions  $a_0 = 4$  and  $a_1 = 9$ .
12. Solve the recurrence relation  $a_n - a_{n-1} - 20a_{n-2} = 3^n$ , if  $n > 1$ , with the initial conditions  $a_0 = -1$  and  $a_1 = 5$ .
13. Prove Theorem 8.3.13.
14. Solve each of the following recurrence relations with the given initial conditions.
  - a.  $a_n = 8a_{n-1} + 10^{n-1}$  for all  $n \geq 1$  and  $a_0 = 1$ .
  - b.  $a_n = 8a_{n-1} + 3$  for all  $n \geq 1$  and  $a_0 = 2$ .
  - c.  $a_n = 3a_{n-1} + 4^{n-1}$  for all  $n \geq 1$  and  $a_0 = 1$ .
  - d.  $a_n = a_{n-1} + n$  for all  $n \geq 1$  and  $a_0 = 1$ .
15. In Worked-Out Exercise 4, show that  $c_1 = 1$ ,  $c_2 = -1$ , and  $c_3 = 1$ .

## ► PROGRAMMING EXERCISES

1. Write a program to solve the Tower of Hanoi problem as described in Example 8.1.11. The program should print the moves. Test run your program for  $n = 6$  and 7 disks.
2. Write a program that takes as input the first two numbers of a Fibonacci sequence and the position, say  $k$ , of the desired number, i.e.,  $f_k$ , in the sequence,  $k \geq 1$ . The program outputs  $f_k$ .
3. Write a program to solve a linear homogeneous recurrence relation of order 2. (If the characteristic equation has real roots, then output the solution. If the characteristic equation has complex roots, then only output that the roots of the characteristic equation are complex.)
4. Write a program to solve a nonlinear homogeneous recurrence relation as described in Theorem 8.3.6.

# Algorithms and Time Complexity

The objectives of this chapter are to:

- Learn about algorithm analysis
- Become familiar with big-O, omega, and theta notations
- Explore various algorithms and their time complexity

In the preceding chapters, we presented various algorithms to speed up computations. Just as in everyday life, there is usually more than one way to accomplish a task. Similarly, there is usually more than one way to design an algorithm to accomplish a task. For example, there are various algorithms to sort or search a list. There is more than one way to design an algorithm to find the factorial of a nonnegative integer or find the  $C(n, r)$ . Some algorithms take too long to yield results, while others do the computation quickly. Some algorithms may appear to be simple and easy to follow, yet may fail to yield the desired result when the problem size increases. For example, because computer memory is limited and in a programming language the size of the memory needed to store an integer is limited, it may not be possible to compute  $n!$  even for relatively small values of  $n$ , say for example,  $n = 20$ . In the preceding chapters, we presented the algorithm, but we did not provide a formal analysis. In this chapter, we first present the standard notions used in algorithm analysis, and then present various algorithms and an analysis of each algorithm.

## 9.1 ALGORITHM ANALYSIS

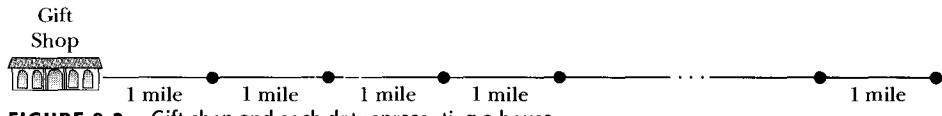
The first step in designing an algorithm to solve a problem is to analyze the problem. Just as a problem is analyzed before the algorithm and computer program are written, an algorithm should also be analyzed after it is designed. To repeat what we said before, there are various ways to design a particular algorithm.

Let us consider the following problem. The holiday season is approaching and a gift shop is expecting sales to be double or even triple the regular amount. The shop has hired extra drivers to deliver the packages on time. The company calculates the shortest distance from the shop to a particular destination and hands the route to the driver. Suppose that 50 packages are to be delivered to 50 different houses. The shop, while making the route, finds that the 50 houses are located 1 mile apart and are in the same area. The first house is also 1 mile from the shop (see Figure 9.1).



**FIGURE 9.1** Gift shop and the 50 houses

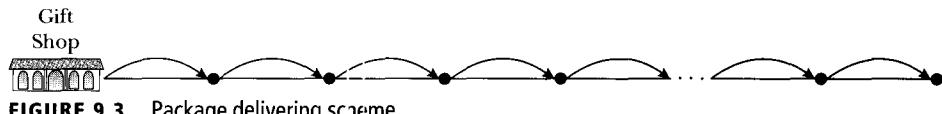
To simplify this figure, we use Figure 9.2.



**FIGURE 9.2** Gift shop and each dot representing a house

Each dot represents a house and the distance between houses is 1 mile, as shown in Figure 9.2.

To deliver 50 packages to their destinations, one of the drivers picks up all 50 packages, drives 1 mile to the first house, and delivers the first package. Then the driver drives another mile and delivers the second package, drives another mile and delivers the third package, and so on. Figure 9.3 illustrates this delivery scheme.



**FIGURE 9.3** Package delivering scheme

Using this scheme, the distance driven to deliver the packages is:

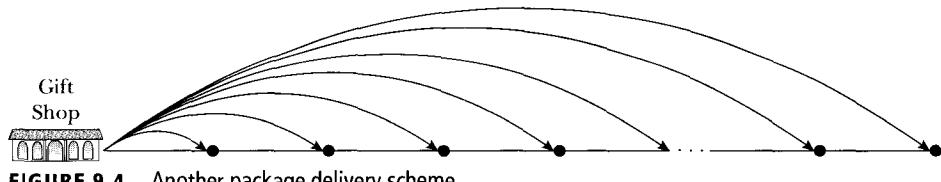
$$1 + 1 + 1 + \cdots + 1 = 50 \text{ miles}$$

Therefore, the total distance traveled to deliver the packages and then return to the shop is:

$$50 + 50 = 100 \text{ miles}$$

Another driver is given a similar route to deliver another set of 50 packages. The driver looks at the route and delivers the packages as follows: The driver picks up the first package, drives 1 mile to the first house, delivers the package, and then returns to the shop. Next, the driver picks up the second package, drives 2 miles,

delivers the package, and returns to the shop. The driver then picks up the third package, drives 3 miles, delivers the package, and returns to the shop. Figure 9.4 illustrates this delivery scheme.



**FIGURE 9.4** Another package delivery scheme

This driver delivers only one package at a time. After delivering a package, the driver comes back to the shop to pick up and deliver the second package. Using this scheme, the total distance traveled is:

$$2 \cdot (1 + 2 + 3 + \cdots + 50) = 2550 \text{ miles}$$

Now suppose that there are  $n$  packages to be delivered to  $n$  houses, and each house is 1 mile apart as shown in Figure 9.2. If the packages are delivered using the first scheme, the following equation gives the total distance traveled:

$$\underbrace{1 + 1 + \cdots + 1}_{n \text{ times}} + n = 2n \quad (9.1)$$

If the packages are delivered using the second method, the distance traveled is:

$$2 \cdot (1 + 2 + 3 + \cdots + n) = 2 \cdot \frac{n(n+1)}{2} = n^2 + n. \quad (9.2)$$

In Equation (9.1), we say that the distance traveled is a function of  $n$ . Let us consider Equation (9.2). In this equation, for large values of  $n$ , we will find that the term consisting of  $n^2$  will become the dominant term and the term containing  $n$  will become negligible. In this case, we say that the distance traveled is a function of  $n^2$ . Table 9.1 evaluates Equations (9.1) and (9.2) for certain values of  $n$ . (This table also shows the value of  $n^2$ .)

**Table 9.1** Values of  $n$ ,  $2n$ ,  $n^2$ , and  $n^2 + n$

| $n$   | $2n$  | $n^2$     | $n^2 + n$ |
|-------|-------|-----------|-----------|
| 1     | 2     | 1         | 2         |
| 10    | 20    | 100       | 110       |
| 100   | 200   | 10000     | 10100     |
| 1000  | 2000  | 1000000   | 1001000   |
| 10000 | 20000 | 100000000 | 100010000 |

When we analyze a particular algorithm, we usually count the number of operations that the algorithm executes. We focus on the number of operations, not on the actual computer time to execute the algorithm. This is because a given algorithm can be implemented on a variety of computers and the speed of the computer can affect the execution time. However, the number of operations performed by the algorithm would be the same on each computer. Let us consider the following examples.

**EXAMPLE 9.1.1**

Consider the following algorithm.

```

1. print "Enter two numbers";
2. read num1, num2;
3. if num1 >= num2 then
4.   max := num1;
5. else
6.   max := num2;
7. print max;
```

Statement 3 has one operation,  $\geq$ ; Statement 4 has one operation,  $:=$ ; Statement 6 has one operation,  $:=$ . Either Statement 4 or Statement 6 executes. Therefore, the total number of operations executed in the preceding code is

$$1 + 1 = 2.$$

In this algorithm, the number of operations executed is fixed.

**REMARK 9.1.2 ▶**

An algorithm might use statements such as **read** and **print**. However, we might not know how these statements are implemented by a particular algorithm. Therefore, when we count the number of operations in an algorithm, we usually focus on arithmetic, relational, and assignment operations.

**EXAMPLE 9.1.3**

Consider the following algorithm.

```

1. print "Enter positive integers ending with -1"
2. count := 0
3. sum := 0;
4. read num;
5. while num  $\neq$  -1 do
6.   begin
7.     sum := sum + num;
8.     count := count + 1;
9.   read num;
10.  end
11. print sum;
12. if count  $\neq$  0 then
13.   average := sum / count;
14. else
15.   average := 0;
16. print average;
```

This algorithm has two operations (Statements 1 through 4) before the **while** loop. Similarly, there are two or three operations after the **while** loop, depending on whether Statement 13 or Statement 15 executes.

Line 5 has one operation, and four operations within the **while** loop (Lines 7 through 9). Thus, Lines 5 through 9 have five operations. If the **while** loop

executes 10 times, these five operations execute 10 times. One extra operation is also executed at Line 5 to terminate the loop. Therefore, the number of operations executed is 51 from Lines 5 through 9.

If the `while` loop executes 10 times, the total number of operations executed is:

$$10 \cdot 5 + 1 + 2 + 3 \quad \text{or} \quad 10 \cdot 5 + 1 + 2 + 2,$$

i.e.,

$$10 \cdot 5 + 6 \quad \text{or} \quad 10 \cdot 5 + 5$$

We can generalize it to the case when the `while` loop executes  $n$  times. If the `while` loop executes  $n$  times, the number of operations executed is:

$$5n + 6 \quad \text{or} \quad 5n + 5$$

In these expressions, for very large values of  $n$ , the term  $5n$  becomes the dominating term and the terms 6 and 5 become negligible.

In an algorithm, certain operations are usually dominant. For example, in the algorithm in Example 9.1.3, to add numbers, the dominant operation is in Line 7. Similarly, in a search algorithm, because the search item is compared with the items in the list, the dominant operation would be comparison, that is, the relational operation. Therefore, in the case of a search algorithm, we would count the number of comparisons. As another example, suppose that we write a program to multiply matrices. The multiplication of matrices involves addition and multiplication. Because multiplication takes more computer time to execute, to analyze a matrix-multiplication algorithm, we count the number of multiplications.

In this and the subsequent chapters, not only do we develop algorithms, but we also provide a reasonable analysis of each algorithm. In fact, if there are various algorithms to accomplish a particular task, the algorithm analysis allows the programmer to choose among various options.

Suppose that an algorithm performs  $f(n)$  basic operations to accomplish a task, where  $n$  is the size of the problem. Suppose that we want to determine whether an item is in a list. Moreover, suppose that the size of the list is  $n$ . To determine whether or not the item is in the list, there are various algorithms we can use, as we will see later in this chapter. However, the basic method is to compare the item with the items in the list. Therefore, the performance of the algorithm depends on the number of comparisons.

Thus, in the case of a search,  $n$  is the size of the list and  $f(n)$  becomes the count function; that is,  $f(n)$  gives the number of comparisons done by the search algorithm. Suppose that on a particular computer it takes  $c$  units of computer time to execute one operation. Thus, the computer time it would take to execute  $f(n)$  operations is  $cf(n)$ . Clearly, the constant  $c$  depends on the speed of the computer, and it therefore varies from computer to computer. However,  $f(n)$ , the number of basic operations, is the same on each computer. If we know how the function  $f(n)$  grows as the size of the problem grows, we can determine the efficiency of the algorithm.

Table 9.2 shows how certain functions grow as the parameter  $n$ , that is, the problem size, grows. Suppose that the problem size is doubled. If we look at the table, we find that if the number of basic operations is a function of  $f(n) = n^2$ , the number of basic operations is quadrupled. If the number of basic operations is a function of  $f(n) = 2^n$ , then the number of basic operations is squared. However,

**Table 9.2** Growth of various functions

| $n$ | $\lg n$ | $n \lg n$ | $n^2$ | $2^n$      |
|-----|---------|-----------|-------|------------|
| 1   | 0       | 0         | 1     | 2          |
| 2   | 1       | 2         | 4     | 4          |
| 4   | 2       | 8         | 16    | 16         |
| 8   | 3       | 24        | 64    | 256        |
| 16  | 4       | 64        | 256   | 65536      |
| 32  | 5       | 160       | 1024  | 4294967296 |

if the number of operations is a function of  $f(n) = \lg n$ , the change in the number of basic operations is very small.

---

**REMARK 9.1.4** ► The most commonly used notation for  $\log_2 n$ , logarithm to the base 2, is  $\lg n$ ; and the most commonly used notation for  $\log_e n$ , logarithm to the base  $e$ , is  $\ln n$ . We therefore use these notations throughout the chapter.

Suppose that a computer can execute 1 billion steps per second. Table 9.3 shows the time that computer takes to execute  $f(n)$  steps.

**Table 9.3** Time for  $f(n)$  instructions on a computer that executes 1 billion instructions per second

| $n$       | $f(n) = n$   | $f(n) := \lg n$ | $f(n) = n \lg n$ | $f(n) = n^2$ | $f(n) = 2^n$            |
|-----------|--------------|-----------------|------------------|--------------|-------------------------|
| 10        | 0.01 $\mu$ s | 0.003 $\mu$ s   | 0.033 $\mu$ s    | 0.1 $\mu$ s  | 1 ms                    |
| 20        | 0.02 $\mu$ s | 0.004 $\mu$ s   | 0.086 $\mu$ s    | 0.4 $\mu$ s  | 1                       |
| 30        | 0.03 $\mu$ s | 0.005 $\mu$ s   | 0.147 $\mu$ s    | 0.9 $\mu$ s  | 1 s                     |
| 40        | 0.04 $\mu$ s | 0.005 $\mu$ s   | 0.213 $\mu$ s    | 1.6 $\mu$ s  | 18.3 min                |
| 50        | 0.05 $\mu$ s | 0.006 $\mu$ s   | 0.282 $\mu$ s    | 2.5 $\mu$ s  | 13 days                 |
| 100       | 0.10 $\mu$ s | 0.007 $\mu$ s   | 0.664 $\mu$ s    | 10 $\mu$ s   | $4 \cdot 10^{13}$ years |
| 1000      | 1.00 $\mu$ s | 0.010 $\mu$ s   | 9.966 $\mu$ s    | 1 ms         |                         |
| 10000     | 10 $\mu$ s   | 0.013 $\mu$ s   | 130 $\mu$ s      | 100 ms       |                         |
| 100000    | 0.10 ms      | 0.017 $\mu$ s   | 1.67 ms          | 10 s         |                         |
| 1000000   | 1 ms         | 0.020 $\mu$ s   | 19.93 ms         | 16.7 m       |                         |
| 10000000  | 0.01 s       | 0.023 $\mu$ s   | 0.23 s           | 1.16 days    |                         |
| 100000000 | 0.10 s       | 0.027 $\mu$ s   | 2.66 s           | 115.7 days   |                         |

In Table 9.3,  $1 \mu\text{s} = 10^{-6}$  seconds and  $1 \text{ ms} = 10^{-3}$  seconds.

Figure 9.5 shows the growth rate of functions in Table 9.3.

In the remainder of this section, we develop a notation that shows how a function  $f(n)$  grows as  $n$  increases without bound; that is, we develop a notation that is useful in describing the behavior of the algorithm, in that it gives us the most useful information about the algorithm. First, we define the term *asymptotic*.

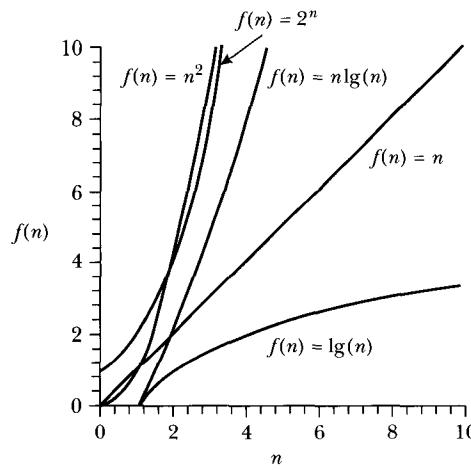


FIGURE 9.5 Growth rate of various functions

**DEFINITION 9.1.5** ▶ Let  $f$  be a function of  $n$ . By the term **asymptotic** we mean the study of the function  $f$  as  $n$  becomes larger and larger without bound.

Consider the functions  $g(n) = n^2$  and  $f(n) = n^2 + 4n + 20$ . Clearly, the function  $g$  does not contain a linear term; that is, the coefficient of  $n$  in  $g$  is zero. Consider Table 9.4.

Table 9.4 Growth of  $g(n)$  and  $f(n)$ 

| $n$   | $g(n) = n^2$ | $f(n) = n^2 + 4n + 20$ |
|-------|--------------|------------------------|
| 10    | 100          | 160                    |
| 50    | 2500         | 2720                   |
| 100   | 10000        | 10420                  |
| 1000  | 1000000      | 1004020                |
| 10000 | 100000000    | 100040020              |

Clearly, as  $n$  becomes larger and larger, the term  $4n + 20$  in  $f(n)$  becomes insignificant, and the term  $n^2$  becomes the dominant term. For large values of  $n$ , we can predict the behavior of  $f(n)$  by looking at the behavior of  $g(n)$ . In algorithm analysis, if the complexity of a function can be described by the complexity of a quadratic function without the linear term, we say that the function is of  $O(n^2)$ , called “big- $O$  of  $n^2$ .”

**DEFINITION 9.1.6** ▶ Let  $f(x)$  and  $g(x)$  be real-valued functions, i.e., their range is a subset of  $\mathbb{R}$ . We say that  $f(x)$  is **big- $O$**  of  $g(x)$  written  $f(x) = O(g(x))$ , if there exist positive constants  $c$  and  $x_0$  such that

$$|f(x)| \leq c|g(x)| \quad \text{for all } x \geq x_0.$$

**EXAMPLE 9.1.7**

Let  $f(n) = n^2 + 4n$  and  $g(n) = n^2$ ,  $n \geq 0$ . Notice that

$$4n \leq n^2 \quad \text{for all } n \geq 4.$$

This implies that

$$n^2 + 4n \leq n^2 + n^2 \quad \text{for all } n \geq 4$$

or

$$n^2 + 4n \leq 2n^2 \quad \text{for all } n \geq 4.$$

Let  $c = 2$  and  $n_0 = 4$ . Then

$$f(n) \leq cg(n) \quad \text{for all } n \geq n_0.$$

Because both  $f(n)$  and  $g(n)$  are nonnegative,  $|f(n)| = f(n)$  and  $|g(n)| = g(n)$ . Thus,

$$|f(n)| \leq c|g(n)| \quad \text{for all } n \geq n_0.$$

Hence,

$$f(n) = O(g(n)).$$

---

**REMARK 9.1.8** ▶ Note that to show that  $f(n) = O(g(n))$ , we only need to find positive constants  $c$  and  $n_0$  such that

$$|f(n)| \leq c|g(n)| \quad \text{for all } n \geq n_0.$$

The choice of  $c$  and  $n_0$  is not unique. For example, in Example 9.1.7, we could have chosen  $c = 5$  and  $n_0 = 1$  because

$$n^2 + 4n \leq n^2 + 4n^2 = 5n^2 \quad \text{for all } n \geq 1.$$

### EXAMPLE 9.1.9

Let  $f(n) = \frac{n(n+1)}{2}$  and  $g(n) = n^2$ ,  $n \geq 0$ . Notice that

$$\frac{n+1}{2} \leq n \quad \text{for all } n \geq 1.$$

This implies that

$$\frac{n(n+1)}{2} \leq n \cdot n \leq n^2 \quad \text{for all } n \geq 1.$$

Choose  $c = 1$  and  $n_0 = 1$ . Then

$$f(n) \leq cg(n) \quad \text{for all } n \geq n_0.$$

Because both  $f(n)$  and  $g(n)$  are nonnegative,  $|f(n)| = f(n)$  and  $|g(n)| = g(n)$ . Thus,

$$|f(n)| \leq c|g(n)| \quad \text{for all } n \geq n_0.$$

Hence,

$$f(n) = O(g(n)).$$

### EXAMPLE 9.1.10

Because  $n \leq n^2$  for all  $n \geq 0$ , it follows that  $n = O(n^2)$ .

### EXAMPLE 9.1.11

Let  $f(n) = n \cdot \lg n$ ,  $n > 0$ . Because  $\lg n \leq n$  for all  $n \geq 1$ , it follows that

$$n \cdot \lg n \leq n \cdot n = n^2 \quad \text{for all } n \geq 1.$$

| This shows that  $n \cdot \lg n = O(n^2)$ .

**REMARK 9.1.12** ► In Example 9.1.11, we can, in fact, show that  $n \cdot \log_a n = O(n^2)$ ,  $a > 1$ .

**Theorem 9.1.13:** Let  $f(n)$  be a real-valued function. Let  $f(n) = a_m n^m + a_{m-1} n^{m-1} + \dots + a_1 n + a_0$ ,  $a_m \neq 0$ ,  $n \geq 0$  and let  $m$  be a nonnegative integer. Then

$$f(n) = O(n^m).$$

**Proof:** Now,

$$\begin{aligned}|f(n)| &= |a_m n^m + a_{m-1} n^{m-1} + \dots + a_1 n + a_0| \\&\leq |a_m| n^m + |a_{m-1}| n^{m-1} + \dots + |a_1| n + |a_0| \\&\leq |a_m| n^m + |a_{m-1}| n^m + \dots + |a_1| n^m + |a_0| n^m \\&= (|a_m| + |a_{m-1}| + \dots + |a_1| + |a_0|) n^m\end{aligned}$$

for all  $n \geq 1$ . Let  $c = |a_m| + |a_{m-1}| + \dots + |a_1| + |a_0|$ . Then

$$|f(n)| \leq cn^m \quad \text{for all } n \geq 1.$$

Hence  $f(n) = O(n^m)$ . ■

### EXAMPLE 9.1.14

Let  $n \in \mathbb{R}$ . By Theorem 9.1.13,

- (i)  $f(n) = n^2 + 5n + 1 = O(n^2)$ .
- (ii)  $f(n) = 4n^6 + 3n^3 + 1 = O(n^6)$ .
- (iii)  $f(n) = an + b = O(n)$ , where  $a$  is nonzero.

Table 9.5 shows some common big- $O$  functions that appear in algorithm analysis. Let  $f(n) = O(g(n))$ , where  $n$  is the problem size.

**Table 9.5** Some common big- $O$  functions

| Function $g(n)$  | Growth rate of $f(n)$                                                                                                                             |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| $g(n) = 1$       | The growth rate is constant, so it does not depend on $n$ .                                                                                       |
| $g(n) = \lg n$   | The growth rate is a function of $\lg n$ . Because a logarithm function grows slowly, the growth rate of $f$ is also slow.                        |
| $g(n) = n$       | The growth rate is linear. The growth rate of $f$ is directly proportional to the size of the problem.                                            |
| $g(n) = n \lg n$ | The growth rate is faster than the linear algorithm.                                                                                              |
| $g(n) = n^2$     | The growth rate of such functions increases rapidly with the size of the problem. The growth rate is quadrupled when the problem size is doubled. |
| $g(n) = 2^n$     | The growth rate is exponential. The growth rate is squared when the problem size is doubled.                                                      |

Using the preceding notations, we can conclude that Equation (9.1) is of  $O(n)$ , and Equation (9.2) is of  $O(n^2)$ . Moreover, the algorithm in Example 9.1.1 is of  $O(1)$ , and the algorithm in Example 9.1.3 is of  $O(n)$ .

The big- $O$  puts an upper bound on a function. In other words, big- $O$  tells how bad a function can become. To determine how good a function can become, we introduce omega notation.

---

**DEFINITION 9.1.15** ▶ Let  $f(x)$  and  $g(x)$  be real-valued functions. The function  $f(x)$  is **omega** of  $g(x)$ , written  $f(x) = \Omega(g(x))$ , if there exist positive constants  $c$  and  $x_0$  such that

$$c|g(x)| \leq |f(x)| \quad \text{for all } x \geq x_0.$$

To simultaneously determine an upper bound and a lower bound on a complexity function, we introduce theta notation.

---

**DEFINITION 9.1.16** ▶ Let  $f(x)$  and  $g(x)$  be real-valued functions. The function  $f(x)$  is **theta** of  $g(x)$ , written  $f(x) = \Theta(g(x))$ , if there exist positive constants  $c_1$ ,  $c_2$  and  $x_0$  such that

$$c_1|g(x)| \leq |f(x)| \leq c_2|g(x)| \quad \text{for all } x \geq x_0.$$

### EXAMPLE 9.1.17

Let  $f(n) = 4n + 6$ ,  $n \geq 0$ . Now  $4n + 6 \geq 4n$  for all  $n \geq 3$  and  $4n + 6 \leq 6n$  for all  $n \geq 3$ . Let  $c_1 = 4$ ,  $c_2 = 6$ , and  $n_0 = 3$ . Then

$$c_1 n \leq f(n) \leq c_2 n \quad \text{for all } n \geq n_0.$$

Because  $f(n)$  is nonnegative and for  $n \geq 0$ ,  $|n| = n$ , we have

$$c_1|n| \leq |f(n)| \leq c_2|n| \quad \text{for all } n \geq n_0.$$

Hence,  $f(n) = \Theta(n)$ .

We leave the proof of the following theorem as an exercise.

**Theorem 9.1.18:** Let  $f(n)$  be a real-valued function. Let  $f(n) = a_m n^m + a_{m-1} n^{m-1} + \cdots + a_1 n + a_0$ ,  $a_m \neq 0$ ,  $n \geq 0$  and let  $m$  be a nonnegative integer. Then

$$f(n) = \Theta(n^m).$$

Theorem 9.1.19 follows from Definitions 9.1.6, 9.1.15, and 9.1.16. We leave its proof as an exercise.

**Theorem 9.1.19:** Let  $f(n)$  and  $g(n)$  be nonnegative real-valued functions. Then  $f(n) = \Theta(g(n))$  if and only if  $f(n) = O(g(n))$  and  $f(n) = \Omega(g(n))$ .

We leave the proof of the following theorem as an exercise.

**Theorem 9.1.20:** Let  $f(n)$ ,  $g(n)$ , and  $h(n)$  be nonnegative real-valued functions. Then the following assertions hold:

- (i)  $f(n) = O(f(n))$ .
- (ii)  $f(n) = \Theta(f(n))$ .
- (iii)  $f(n) = \Omega(f(n))$ .
- (iv)  $f(n) = \Theta(g(n))$  if and only if  $g(n) = \Theta(f(n))$ .
- (v) If  $f(n) = O(g(n))$  and  $g(n) = O(h(n))$ , then  $f(n) = O(h(n))$ .
- (vi) If  $f(n) = \Theta(g(n))$  and  $g(n) = \Theta(h(n))$ , then  $f(n) = \Theta(h(n))$ .

---

**REMARK 9.1.21** ▶ Because theta notation simultaneously imposes an upper bound and a lower bound, when we analyze an algorithm, we characterize its behavior (for the most part) in terms of theta notation.

In Worked-Out Exercise 4 (Chapter 8, p. 542), we considered the following recurrence relation:

$$w_n = 2w_{n/2} + n - 1, \quad n > 1, \quad n = 2^k \text{ for some integer } k, \quad (9.3)$$

with the initial condition  $w_1 = 1$ . When  $n$  is a power of 2, we showed that the solution of this recurrence relation is:

$$w_n = 1 - n + n \lg n = n \lg n - (n - 1),$$

where  $n = 2^k$  for some integer  $k$ .

Suppose the time complexity of an algorithm is given by the following function:

$$T(n) = 2T\left(\frac{n}{2}\right) + n - 1, \quad \text{if } n > 1 \quad (9.4)$$

and  $T(1) = 1$ . We want to describe the behavior of  $T$  using big- $O$  or theta notation in terms of some known functions.

Notice that if we substitute  $T(n) = w_n$ , then we obtain the recurrence relation given in (9.3). Therefore, if  $n = 2^k$  for some positive integer  $k$ , we can show that

$$T(n) = n \lg n - (n - 1) = \Theta(n \lg n). \quad (9.5)$$

What about the behavior of  $T(n)$ , when  $n$  is not a power of 2? Next we develop results that will help answer such questions.

In algorithm analysis—especially when the algorithm uses a divide-and-conquer technique—we often come across algorithm complexity functions such as the one in (9.4). Solutions of such complexity functions can be quickly obtained when  $n$  is some power of a positive integer, say  $b$ , i.e.,  $n = b^k$  for some integer  $k \geq 1$ . Typically,  $b = 2$ . (Note that the function  $T(n)$  given in (9.4) in fact describes the worst-case behavior of the merge sort algorithm, discussed in the next section.)

After obtaining a solution for  $n = b^k$ , we obtain a general solution. In the remainder of this section, we discuss how this is done. Before proceeding further, we need to introduce certain definitions. First, however, we should note that functions that describe the behavior of an algorithm are commonly called *complexity functions*.

**DEFINITION 9.1.22** ▶ Let  $f(x)$  be a real-valued function. Then  $f(x)$  is called **strictly increasing** if for all  $x_1$  and  $x_2$

$$x_1 < x_2 \Rightarrow f(x_1) < f(x_2).$$

In other words,  $f(x)$  gets larger and larger.

**DEFINITION 9.1.23** ▶ Let  $f(x)$  be a real-valued function. Then  $f(x)$  is called **nonddecreasing** if for all  $x_1$  and  $x_2$

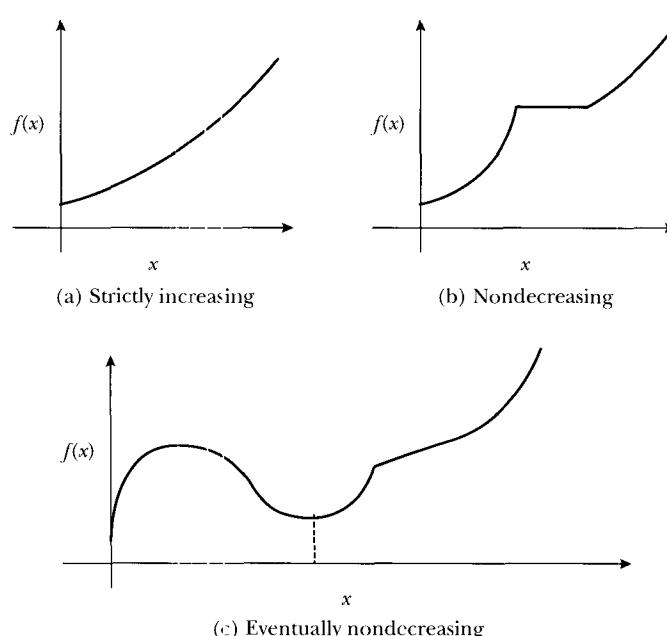
$$x_1 < x_2 \Rightarrow f(x_1) \leq f(x_2).$$

The time complexity of most algorithms is, typically, nonddecreasing. This is because when the input size increases, the time to process the input also usually increases.

**DEFINITION 9.1.24** ▶ Let  $f(x)$  be a real-valued function. Then  $f(x)$  is called **eventually nonddecreasing** if there exists a real number  $x_0$  such that for all  $x_1$  and  $x_2$

$$x_0 < x_1 < x_2 \Rightarrow f(x_1) \leq f(x_2).$$

Figure 9.6(a) shows a strictly increasing function, Figure 9.6(b) shows a nonddecreasing function, and Figure 9.6(c) shows an eventually nonddecreasing function.



**FIGURE 9.6** Graphs of various functions

**DEFINITION 9.1.25** ▶ Let  $f(x)$  be an eventually nonddecreasing function. Then  $f(x)$  is called **smooth** if  $f(2x) = \Theta(f(x))$ .

**EXAMPLE 9.1.26**

Let  $f(n) = \lg n$ . We know that  $f(n)$  is eventually nonddecreasing for  $n > 0$ . Now

$$f(2n) = \lg 2n = \lg n + \lg 2 = \lg n + 1 = \Theta(\lg n) = \Theta(f(n)).$$

Thus, the function  $\lg n$  is smooth.

**REMARK 9.1.27** ▶ As in Example 9.1.26, we can show that the functions  $f(n) = n$ ,  $g(n) = n \lg n$ , and  $h(n) = n^k$  are smooth. However, the function  $f(n) = 2^n$  is not smooth.

**Theorem 9.1.28:** Let  $f(x)$  be a real-valued function such that  $f(x)$  is smooth. Then  $f(bx) = \Theta(f(x))$  for all  $b \geq 2$ .

**Lemma 9.1.29:** Let  $b \geq 2$  be an integer. For any positive integer  $n$ , there exists a unique nonnegative integer  $k$  such that

$$b^k \leq n < b^{k+1}.$$

**Theorem 9.1.30:** Let  $f(n)$  be a smooth function and  $b > 1$  be an integer. Suppose that  $T(n)$  is a nonnegative eventually nondecreasing function such that

$$T(n) = O(f(n)), \quad \text{if } n = b^k, \text{ where } k \text{ is a positive integer.}$$

Then

$$T(n) = O(f(n)).$$

Moreover, this result also holds if big- $O$  is replaced with  $\Theta$  or  $\Omega$ .

**Proof:** Now  $T(n) = O(f(n))$ , if  $n = b^k$ , where  $k$  is a positive integer. Then there exist positive constants  $c$  and  $N_1$  such that

$$T(n) \leq c|f(n)|,$$

when  $n$  is a positive integral power of  $b$  and  $n > N_1$ .

By Theorem 9.1.28,  $f(bn) = \Theta(f(n))$ . Thus, there exist positive constants  $d$  and  $N_2$  such that

$$|f(bn)| \leq d|f(n)| \quad \text{for all } n > N_2.$$

This implies that for  $b^k \geq N_2$ ,

$$|f(b^{k+1})| = |f(bb^k)| \leq d|f(b^k)|.$$

Now  $T(n)$  and  $f(n)$  are eventually nondecreasing functions. So there exists a positive constant  $N_3$  such that  $N_3 < n_1 < n_2$  implies that

$$T(n_1) \leq T(n_2) \quad \text{and} \quad f(n_1) \leq f(n_2).$$

Choose  $s$  such that  $b^s > \max\{N_1, N_2, N_3\}$ . Let  $n$  be an integer such that  $n > b^s$ . By Lemma 9.1.29, there exists a unique nonnegative integer  $t$  such that  $b^t \leq n < b^{t+1}$ . Then  $b^t \geq b^s$ . Thus, for  $n > b^s$ ,

$$T(n) \leq T(b^{t+1}) \leq c|f(b^{t+1})| \leq cd|f(b^t)| \leq cd|f(n)|.$$

Hence,  $T(n) = O(f(n))$ . ■

**EXAMPLE 9.1.31**

Consider the complexity function given by the following recurrence relation:

$$T(n) = T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + 2, \quad \text{if } n > 1 \quad (9.6)$$

and  $T(1) = 1$ .

Let us solve (9.6), when  $n$  is a power of 2. Suppose  $n = 2^k$  for some positive integer  $k$ . Then

$$T(2^k) = T\left(\left\lfloor \frac{2^k}{2} \right\rfloor\right) + 2 = T(2^{k-1}) + 2. \quad (9.7)$$

Let us write  $a_k = T(2^k)$ . Then

$$a_k = a_{k-1} + 2, \quad k > 1. \quad (9.8)$$

We apply Theorem 8.3.6 to solve (9.8). Here  $d = 1$ ,  $b = 1$ , and  $u = 2$ . Thus,

$$a_k = c_0 + 2k, \quad (9.9)$$

where  $c_0$  is some constant. Now  $n = 2^k$  implies that  $\lg n = k$ , by taking  $\lg$  on both sides and simplifying. Also  $a_k = T(2^k) = T(n)$ . Substituting  $a_k$  and  $k$  in (9.9) we get

$$T(n) = c_0 + 2 \lg n. \quad (9.10)$$

We put  $n = 1$  to get  $1 = T(1) = c_0 + 2 \lg 1 = c_0$ . Hence,

$$T(n) = 2 \lg n + 1.$$

We can verify that this is a solution of (9.6), when  $n$  is a power of 2. Hence,

$$T(n) = 2 \lg n + 1 = \Theta(\lg n), \quad \text{when } n \text{ is power of 2.}$$

By Example 9.1.26,  $\lg n$  is smooth. Clearly, the function  $T(n)$  is nonnegative. Thus, to apply Theorem 9.1.30, we need to show that  $T(n)$ , as given in (9.6), is eventually nondecreasing. This we leave as an exercise.

Hence,

$$T(n) = \Theta(\lg n).$$

The following theorem, commonly called the *Master Theorem*, gives a general characterization of complexity that commonly appears in algorithms that use a divide-and-conquer technique. The proof of this theorem is left as an exercise.

**Theorem 9.1.32: Master Theorem.** Let  $f(n)$  be an eventually nondecreasing function such that

$$f(1) = d$$

$f(n) = af\left(\frac{n}{b}\right) + cn^k$ , if  $n > 1$  and  $n = b^m$ , where  $m$  is a positive integer,  $b \geq 2$ ,  $k \geq 0$  are constant integers, and  $a > 0$ ,  $c > 0$ , and  $d \geq 0$  are constants. Then

$$f(n) = \begin{cases} \Theta(n^k), & \text{if } a < b^k \\ \Theta(n^k \lg n), & \text{if } a = b^k \\ \Theta(n^{\log_b a}), & \text{if } a > b^k. \end{cases} \quad (9.11)$$

**EXAMPLE 9.1.33**

Let  $f(n)$  be defined by

$$f(1) = 5$$

$$f(n) = 9f\left(\frac{n}{2}\right) + 6n^4, \quad \text{if } n > 1 \text{ and } n = 2^m, \text{ where } m \text{ is a positive integer.}$$

Here  $a = 9$ ,  $b = 2$ ,  $c = 6$ , and  $k = 4$ . Now  $b^k = 2^4 = 16 > 9 = a$ . Hence,

$$f(n) = \Theta(n^4).$$

**EXAMPLE 9.1.34**

Let  $f(n)$  be defined by

$$f(1) = 2$$

$$f(n) = 10f\left(\frac{n}{3}\right) + 5n^2, \quad \text{if } n > 1 \text{ and } n = 3^m, \text{ where } m \text{ is a positive integer.}$$

Here  $a = 10$ ,  $b = 3$ ,  $c = 5$ , and  $k = 2$ . Now  $b^k = 3^2 = 9 < 10 = a$ . Hence,

$$f(n) = \Theta(n^{\log_3 10}).$$



## WORKED-OUT EXERCISES

**Exercise 1:** Let  $b > 1$  and  $c > 1$  be constant real numbers. Show that  $\log_b n = \Theta(\log_c n)$ .

**Solution:** Here we use the fact that  $\log_b a^n = n \log_b a$  and  $c^{\log_c n} = n$ . Now

$$\log_b n = \log_b(c^{\log_c n}) = (\log_c n)(\log_b c) = k \log_c n,$$

where  $k = \log_b c$  is a fixed positive constant. It follows that  $\log_b n = \Theta(\log_c n)$ .

**Exercise 2:** Let  $f(n) = 1 + 2 + \dots + n$ ,  $n \geq 1$ . Show that  $f(n) = \Theta(n^2)$ .

**Solution:** Notice that in the sum  $1 + 2 + \dots + n$ ,

$$1 \leq n, 2 \leq n, \dots, i \leq n, \dots, n \leq n.$$

Hence,

$$1 + 2 + \dots + n \leq n + n + \dots + n = n \cdot n = n^2$$

for all  $n \geq 1$ .

This implies that  $f(n) = O(n^2)$ .

Next, we obtain a lower bound on the sum. We have

$$\begin{aligned} 1 + 2 + \dots + n &= 1 + 2 + \dots + \left\lceil \frac{n}{2} \right\rceil + \dots + n \\ &\geq \left\lceil \frac{n}{2} \right\rceil + \dots + n \\ &\geq \left\lceil \frac{n}{2} \right\rceil + \left\lceil \frac{n}{2} \right\rceil + \dots + \left\lceil \frac{n}{2} \right\rceil \end{aligned}$$

$$\begin{aligned} &\geq \frac{n}{2} \cdot \frac{n}{2} \\ &= \frac{n^2}{4}. \end{aligned}$$

This implies that  $f(n) = \Omega(n^2)$ .

Because  $f(n) = O(n^2)$  and  $f(n) = \Omega(n^2)$ , by Theorem 9.1.19,  $f(n) = \Theta(n^2)$ .

**Exercise 3:** Let  $n \geq 1$  be a positive integer. Prove that  $\lg n! = \Theta(n \lg n)$ .

**Solution:** Let  $f(n) = \lg n!$  and  $g(n) = \Theta(n \lg n)$ . First we obtain an upper bound  $f(n)$ . By the properties of logarithms, we know that for positive real numbers  $x$  and  $y$ ,  $\lg xy = \lg x + \lg y$ . From this, it follows that

$$\begin{aligned} f(n) &= \lg n! = \lg(n(n-1)\cdots 2 \cdot 1) \\ &= \lg n + \lg(n-1) + \dots + \lg 2 + \lg 1. \end{aligned}$$

Now  $\lg$  is an increasing function, so  $i \leq n$  implies that  $\lg i \leq \lg n$  for all  $i = 1, 2, \dots, n$ . Thus, we have

$$\begin{aligned} \lg n! &= \lg n + \lg(n-1) + \dots + \lg 2 + \lg 1 \\ &\leq \lg n + \lg n + \dots + \lg n = n \lg n \end{aligned}$$

for all  $n \geq 1$ . Hence,  $\lg n! = O(n \lg n)$ .

Again

$$\begin{aligned}
 \lg n! &= \lg n + \lg(n-1) + \cdots + \lg 2 + \lg 1 \\
 &= \lg n + \lg(n-1) + \cdots + \lg \left\lceil \frac{n}{2} \right\rceil + \cdots + \lg 2 + \lg 1 \\
 &\quad \text{insert the middle term} \\
 &\geq \lg n + \lg(n-1) + \cdots + \lg \left\lceil \frac{n}{2} \right\rceil \\
 &\geq \lg \left\lceil \frac{n}{2} \right\rceil + \lg \left\lceil \frac{n}{2} \right\rceil + \cdots + \lg \left\lceil \frac{n}{2} \right\rceil, \\
 &\quad \text{because } \lg \text{ is an increasing function.} \\
 &= \left\lceil \frac{n+1}{2} \right\rceil \lg \left\lceil \frac{n}{2} \right\rceil \\
 &\geq \frac{n}{2} \lg \frac{n}{2} \\
 &\geq \frac{1}{4} n \lg n \quad \text{for all } n \geq 4.
 \end{aligned}$$

We show that  $\frac{n}{2} \lg \frac{n}{2} \geq \frac{1}{4} n \lg n$  for all  $n \geq 4$ . Notice that

$$\begin{aligned}
 \frac{n}{2} \lg \frac{n}{2} &\geq \frac{1}{4} n \lg n \\
 \Leftrightarrow \frac{1}{2} \lg \frac{n}{2} &\geq \frac{1}{4} \lg n, \quad \text{cancel } n \text{ from both sides} \\
 \Leftrightarrow \frac{1}{2} (\lg n - \lg 2) &\geq \frac{1}{4} \lg n
 \end{aligned}$$

$$\begin{aligned}
 &\Leftrightarrow \frac{1}{2} (\lg n - 1) \geq \frac{1}{4} \lg n \\
 &\Leftrightarrow 2(\lg n - 1) \geq \lg n \\
 &\Leftrightarrow 2 \lg n - 2 \geq \lg n \\
 &\Leftrightarrow \lg n \geq 2.
 \end{aligned}$$

Suppose  $4 \leq n$ . Because  $\lg$  is an increasing function, we have  $\lg 4 \leq \lg n$ ,  $2 \leq \lg n$ . Hence,  $\lg n \geq 2$  for all  $n \geq 4$ , i.e.,  $\frac{n}{2} \lg \frac{n}{2} \geq \frac{1}{4} n \lg n$  for all  $n \geq 4$ .

It now follows that

$$\lg n! \geq \frac{n}{2} \lg \frac{n}{2} \geq \frac{1}{4} n \lg n$$

for all  $n \geq 4$ . This implies that  $\lg n! = \Omega(n \lg n)$ . Hence, by Theorem 9.1.19,  $g(n) = \Theta(n \lg n)$ .

**Exercise 4:** Let  $f(n)$  be defined by

$$\begin{aligned}
 f(1) &= 1 \\
 f(n) &= 7f\left(\frac{n}{2}\right) + 4n^3, \quad \text{if } n > 1 \text{ and } n = 2^m \\
 &\quad \text{for some positive integer } m.
 \end{aligned}$$

Show that  $f(n) = \Theta(n^3)$ .

**Solution:** Let  $a = 7$ ,  $b = 2$ , and  $k = 3$ . Then  $a = 7 < 8 = 2^3 = b^k$ . Hence, by Theorem 9.1.32,  $f(n) = \Theta(n^3)$ .

## SECTION REVIEW

### Key Terms

|               |                     |                          |
|---------------|---------------------|--------------------------|
| asymptotic    | theta               | eventually nondecreasing |
| big- <i>O</i> | strictly increasing | smooth                   |
| omega         | nondecreasing       |                          |

### Some Key Definitions

- Let  $f(x)$  and  $g(x)$  be real-valued functions, i.e., their range is a subset of  $\mathbb{R}$ . We say that  $f(x)$  is **big-*O*** of  $g(x)$  written  $f(x) = O(g(x))$ , if there exist positive constants  $c$  and  $x_0$  such that  $|f(x)| \leq c|g(x)|$  for all  $x \geq x_0$ .
- Let  $f(x)$  and  $g(x)$  be real-valued functions. The function  $f(x)$  is **omega** of  $g(x)$ , written  $f(x) = \Omega(g(x))$ , if there exist positive constants  $c$  and  $x_0$  such that  $c|g(x)| \leq |f(x)|$  for all  $x \geq x_0$ .
- Let  $f(x)$  and  $g(x)$  be real-valued functions. The function  $f(x)$  is **theta** of  $g(x)$ , written  $f(x) = \Theta(g(x))$ , if there exist positive constants  $c_1$ ,  $c_2$  and  $x_0$  such that  $c_1|g(x)| \leq |f(x)| \leq c_2|g(x)|$  for all  $x \geq x_0$ .
- Let  $f(n)$  be a real-valued function. We call  $f(x)$  eventually nondecreasing if there exists a real number  $x_0$  such that for all  $x_1$  and  $x_2$ ,  $x_0 < x_1 < x_2 \Rightarrow f(x_1) \leq f(x_2)$ .

## Some Key Results

1. Let  $f(n)$  be a real-valued function. Let  $f(n) = a_m n^m + a_{m-1} n^{m-1} + \dots + a_1 n + a_0$ ,  $a_m \neq 0$ ,  $n \geq 0$  and let  $m$  be a nonnegative integer. Then  $f(n) = \Theta(n^m)$ .
2. Let  $f(n)$  be a smooth function and  $b > 1$  is an integer. Suppose that  $T(n)$  is a nonnegative eventually nondecreasing function such that  $T(n) = O(f(n))$ , if  $n = b^k$ , where  $k$  is a positive integer. Then  $T(n) = O(f(n))$ . Moreover, this result also holds, if big- $O$  is replaced with  $\Theta$  or  $\Omega$ .
3. Let  $f(n)$  be an eventually nondecreasing complexity function such that

$$f(1) = d$$

$f(n) = af\left(\frac{n}{b}\right) + cn^k$ , if  $n > 1$  and  $n = b^m$ , where  $m$  is a positive integer  $b \geq 2$ ,  $k \geq 0$  are constant integers and  $a > 0$ ,  $c > 0$ , and  $d \geq 0$  are constants. Then

$$f(n) = \begin{cases} \Theta(n^k), & \text{if } a < b^k \\ \Theta(n^k \lg n), & \text{if } a = b^k \\ \Theta(n^{\log_b a}), & \text{if } a > b^k. \end{cases}$$

## EXERCISES

---

1. Let  $f(n) = 2n^4 + 7n + 5$ ,  $n \geq 0$ . Show by using the definition that
  - a.  $f(n) = O(n^4)$ .
  - b.  $f(n) = \Theta(n^4)$ .
2. Let  $f(n) = (n^2 + 2)(25n^3 + 2n - 6)n^7$ ,  $n \geq 0$ . Characterize  $f(n)$  in terms of  $\Theta$  notation.
3. Let  $f(n) = n^2 + n \lg n$ ,  $n > 0$ . Characterize  $f(n)$  in terms of  $\Theta$  notation.
4. Let  $f(n) = 1^2 + 2^2 + \dots + n^2$ ,  $n \geq 0$ . Show that  $f(n) = \Theta(n^3)$ .
5. Let  $f(n) = 1^3 + 2^3 + \dots + n^3$ ,  $n \geq 0$ . Show that  $f(n) = \Theta(n^4)$ .
6. Show that  $\sum_{i=1}^n i(i+1) = \frac{n(n+1)(n+2)}{3} = \Theta(n^3)$ ,  $n \geq 1$ .
7. Show that  $\sum_{i=1}^{n-1} i(n-i) = \frac{(n-1)n(n+1)}{6} = \Theta(n^3)$ ,  $n \geq 2$ .
8. Let  $f(n) = 2 + 2^2 + 2^3 + \dots + 2^n$ ,  $n \geq 0$ . Characterize  $f(n)$  in terms of  $\Theta$  notation.
9. Show that  $f(n) = n! = O(n^n)$ , where  $n$  is a nonnegative integer.
10. Show that  $\sqrt{1} + \sqrt{2} + \dots + \sqrt{n} \leq n^{3/2}$ . Hence, show that  $\sqrt{1} + \sqrt{2} + \dots + \sqrt{n} = O(n^{3/2})$ .
11. Let  $f(n) = O(n)$  and  $g(n) = O(n^2)$ . Characterize  $f(n) + g(n)$  in terms of  $O$  notation.
12. Prove Theorem 9.1.18.
13. Characterize the following algorithm in terms of  $\Theta$  notation. Also find the exact number of additions executed by the loop.
 

```
for i := 1 to n do
    sum := sum + i * (i + 1);
```
14. Characterize the following algorithm in terms of  $\Theta$  notation. Also find the exact number of additions, subtractions, and multiplications executed by the loop.
 

```
for i := 5 to 2 * n do
    print 2 * n + i - 1;
```
15. Characterize the following algorithm in terms of  $\Theta$  notation.
 

```
for j := 1 to ⌊n/2⌋ do
  print j;
```
16. Characterize the following algorithm in terms of  $\Theta$  notation.
 

```
for i := 1 to 2 * n do
  for j := 1 to n do
    print 2 * i + j;
```
17. Characterize the following algorithm in terms of  $\Theta$  notation.
 

```
for i := 1 to n do
  for j := 1 to i do
    print i * (n - j);
```
18. Characterize the following algorithm in terms of  $\Theta$  notation.
 

```
for i := 1 to n do
  for j := 1 to n do
    for k := 1 to n do
      print i + j + k;
```
19. **Requires Calculus.** Let  $f(n) = 1 + \frac{1}{2} + \dots + \frac{1}{n}$ ,  $n > 0$ . Prove that  $f(n) = \Theta(\lg n)$ .
20. Prove Theorem 9.1.19.
21. Let  $f(n) = n$ ,  $n \geq 0$ . Prove that  $f(n)$  is smooth.
22. Let  $f(n) = n \lg n$ ,  $n > 0$ . Prove that  $f(n)$  is smooth.
23. Let  $k$  be a nonnegative integer and  $f(n) = n^k$ ,  $n \geq 0$ . Prove that  $f(n)$  is smooth.
24. Show that the function  $T(n) = T(\lfloor \frac{n}{2} \rfloor) + 2$ ,  $n > 1$ , and  $T(1) = 1$  is eventually nondecreasing.

25. Let  $f(n)$  be defined by

$$f(1) = 1$$

$$f(n) = 5f\left(\frac{n}{3}\right) + 4n^2, \quad \text{if } n > 1 \text{ and } n = 3^m, \\ \text{where } m \text{ is a positive integer.}$$

Show that  $f(n) = \Theta(n^2)$ .

26. Let  $f(n)$  be defined by

$$f(1) = 3$$

$$f(n) = 16f\left(\frac{n}{2}\right) + 5n^4, \quad \text{if } n > 1 \text{ and } n = 2^m, \\ \text{where } m \text{ is a positive integer.}$$

Show that  $f(n) = \Theta(n^4 \lg n)$ .

27. Let  $f(n)$  be defined by

$$f(1) = 2$$

$$f(n) = 35\left(\frac{n}{3}\right) + 7n^3, \quad \text{if } n > 1 \text{ and } n = 3^m, \\ \text{where } m \text{ is a positive integer.}$$

Show that  $f(n) = \Theta(n^{\log_3 35})$ .

28. Let  $a$ ,  $b$ , and  $c$  be integers such that  $a \geq 1$ ,  $b > 1$ , and  $c > 0$ . Let  $f : \mathbb{N} \rightarrow \mathbb{R}$  be function such that  $f(1) = c$  and

$$f(n) = af\left(\frac{n}{b}\right) + c \quad \text{for } n = b^k, \text{ where } k \text{ is an integer greater than 1.}$$

Prove the following when  $n = b^k$  where  $k$  is an integer greater than 1.

- (i) If  $a = 1$ , then  $f(n) = c(\log_b n + 1) = \Theta(\log_b n)$ .
- (ii) If  $a \neq 1$ , then  $f(n) = \frac{c(an^{\log_b a} - 1)}{a-1} = \Theta(n^{\log_b a})$ .

## 9.2 VARIOUS ALGORITHMS

In this section, we present a number of important algorithms. For each algorithm, we do an analysis in terms of big- $O$  or theta notation.

### Sequential Search

In Chapter 1, we discussed the sequential search algorithm but we did not analyze it. In this chapter, we analyze this algorithm and discuss other search algorithms as well. Analysis of algorithms enables programmers to decide which algorithm is best for a specific application.

In the analysis of an algorithm, the comparisons refer to comparing the search item with an item in the list. The number of comparisons refers to the number of times the search item (in algorithms such as searching and sorting) is compared with the items in the list.

For easy reference, we rewrite the **sequential search** algorithm here.

#### ALGORITHM 9.1: Sequential Search.

*Input:*  $L$ —list of  $n$  elements  
 $n$ —the size of  $L$   
 $x$ —the search item

*Output:* The index of the first element of  $L$  that is the same as  $x$ , otherwise  $-1$  (indicating an unsuccessful search).

```

1. function sequentialSearch( $L, n, x$ )
2.   begin
3.     for  $i := 1$  to  $n$  to
4.       if  $x = L[i]$  then
5.         return  $i$ ;
6.     return  $-1$ ;
7.   end
```

The search always starts at the first element in the list and continues until either the item is found in the list or the entire list is searched. If the search item is found, its index in the list is returned. If the search is unsuccessful,  $-1$  is returned. Note that the sequential search, as given above, does not require the list elements to be in any particular order.

The statements before and after the `for` loop are executed only once, and hence require very little computer time. The statements in the `for` loop are the ones that are repeated several times. For each iteration of the loop, the search item is compared with an element in the list, and a few other statements are executed, including some other comparisons. Clearly, the loop terminates as soon as the search item is found in the list. Therefore, the execution of the other statements in the loop is directly related to the outcome of the comparison. Also, different programmers might implement the same algorithm differently, although the number of comparisons would typically be the same. The speed of a computer can easily affect the time an algorithm takes to perform, but not the number of comparisons.

Therefore, when analyzing a search algorithm, we count the number of comparisons because this number gives us the most useful information. Furthermore, the criteria for counting the number of comparisons can be applied equally well to other search algorithms.

Suppose that the length of the list, say  $L$ , is  $n$ . We want to determine the number of key comparisons made by the sequential search when the list  $L$  is searched for a given item.

If the search item is not in the list, we then compare the search item with every element in the list, making  $n$  comparisons. This is an unsuccessful case.

Suppose that the search item is in the list. Then the number of *key comparisons* depends on where in the list the search item is located. If the search item is the first element of  $L$ , we make only one key comparison. This is the best case. On the other hand, if the search item is the last element in the list, the algorithm makes  $n$  comparisons. This is the worst case. The best and worst cases are not likely to occur every time we apply the sequential search on  $L$ , so it would be more helpful if we could determine the average behavior of the algorithm. That is, we need to determine the average number of key comparisons the sequential search algorithm makes in the successful case.

To determine the average number of comparisons in the successful case of the sequential search algorithm:

1. Consider all possible cases.
2. Find the number of comparisons for each case.
3. Add the number of comparisons and divide by the number of cases.

If the search item, called the *target*, is the first element in the list, one comparison is required. If the target is the second element in the list, two comparisons are required. Similarly, if the target is the  $k$ th element in the list,  $k$  comparisons are required. We assume that the target can be any element in the list; that is, all list elements are equally likely to be the target. Suppose that there are  $n$  elements in the list. The following expression gives the average number of comparisons:

$$\frac{1 + 2 + \dots + n}{n} = \frac{1}{n} \cdot \frac{n(n+1)}{2} = \frac{n+1}{2} = \Theta(n).$$

This expression shows that, on average, the sequential search searches half the list. It thus follows that if the list contains 1,000,000 elements, on average, the sequential search makes 500,000 comparisons. As a result, the sequential search is not efficient for large lists.

## Binary Search

As you can see, the sequential search is not efficient for large lists because, on average, the sequential search searches half the list. We therefore describe another search algorithm, called the **binary search**, which is very fast. However, a binary search can be performed only on ordered lists. We must therefore assume that the list is ordered. Later in this chapter, we describe several sorting algorithms.

The binary search algorithm uses a divide-and-conquer technique to search the list. First, the search item is compared with the middle element of the list. If the search item is less than the middle element of the list, we restrict the search to the first half of the list; otherwise, we search the second half of the list.

Consider the sorted list of length  $n = 12$  in Figure 9.7.

|     | [1] | [2] | [3] | [4] | [5] | [6] | [7] | [8] | [9] | [10] | [11] | [12] |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|
| $L$ | 4   | 8   | 19  | 25  | 34  | 39  | 45  | 48  | 66  | 75   | 89   | 95   |

FIGURE 9.7 List of length 12

Suppose that we want to determine whether 75 is in the list. Initially, the entire list is the search list (see Figure 9.8).

|     | [1] | [2] | [3] | [4] | [5] | [6] | [7] | [8] | [9] | [10] | [11] | [12] |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|
| $L$ | 4   | 8   | 19  | 25  | 34  | 39  | 45  | 48  | 66  | 75   | 89   | 95   |

FIGURE 9.8 Search list,  $L[1 \dots 12]$

First, we compare 75 with the middle element in this list,  $L[6]$  (which is 39). Because  $75 \neq L[6]$  and  $75 > L[6]$ , we then restrict our search to the list  $L[7 \dots 12]$ , as shown in Figure 9.9.

|     | [1] | [2] | [3] | [4] | [5] | [6] | [7] | [8] | [9] | [10] | [11] | [12] |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|
| $L$ | 4   | 8   | 19  | 25  | 34  | 39  | 45  | 48  | 66  | 75   | 89   | 95   |

FIGURE 9.9 Search list,  $L[7 \dots 12]$

This process is now repeated on the list  $L[7 \dots 12]$ , which is a list of length = 6.

Because we need to determine the middle element of the list frequently, the binary search algorithm is typically implemented for array-based lists. To determine the middle element of the list, we add the starting index, `first`, and the ending index, `last`, of the search list and then divide by 2 to calculate its index. That is,  $\lfloor \frac{\text{first} + \text{last}}{2} \rfloor$ .

Initially, `first` = 1 and `last` =  $n$ . The following function implements the binary search algorithm. If the item is found in the list, its location is returned; if the search item is not in the list, -1 is returned.

**ALGORITHM 9.2: Binary Search.**

*Input:*  $L[1 \dots n]$ —search list  
 $n$ —number of elements in the list  
 $x$ —search item

*Output:* If  $x$  is in  $L$ , its index in  $L$  is returned; otherwise  $-1$  is returned.

```

1. function binarySearch(L, n, x)
2. begin
3.   first := 1;
4.   last := n;
5.   while first <= last do
6.     begin
7.       mid :=  $\lfloor \frac{\text{first} + \text{last}}{2} \rfloor$ ;
8.       if list[mid] = x then
9.         return mid;
10.      else
11.        if list[mid] > x then
12.          last := mid - 1;
13.        else
14.          first := mid + 1;
15.      end
16.    return -1;
17.  end

```

**Binary Search Analysis**

Let  $T(n)$  denote the number of comparisons done by the binary search for a list of length  $n$ . Each time through the loop, we make two comparisons and also reduce the size of the search list by half for the next iteration. Then

$$T(n) = T\left(\frac{n}{2}\right) + 2, \quad \text{if } n > 1.$$

Also

$$T(1) = 1.$$

By Example 9.1.31,

$$T(n) = \Theta(\lg n).$$

**Selection Sort**

In the preceding sections, we discussed and analyzed sequential and binary search algorithms. We showed that the performance of a binary search algorithm is much better than that of sequential search. However, in a binary search the data must

be sorted. Recall the bubble sort algorithm, which we described in Chapter 1. In the next few sections, we describe various sorting algorithms. We begin with the selection sort algorithm.

The **selection sort** algorithm sorts a list by selecting the smallest element in the (unsorted portion of the) list and then moving this smallest element to the top of the (unsorted portion of the) list. The first time we locate the smallest item in the entire list, the second time we locate the smallest item in the list starting from the second element in the list, and so on.

For example, suppose that you have the list as shown in Figure 9.10.

|          |     |     |     |     |     |     |     |     |     |      |
|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|
|          | [1] | [2] | [3] | [4] | [5] | [6] | [7] | [8] | [9] | [10] |
| <i>L</i> | 16  | 30  | 24  | 7   | 25  | 62  | 45  | 5   | 65  | 50   |

FIGURE 9.10 List of 10 elements

Initially, the entire list is unsorted. So we find the smallest item in the list, which is at position 8, as shown in Figure 9.11.

|          |     |     |     |     |     |     |     |     |     |      |
|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|
|          | [1] | [2] | [3] | [4] | [5] | [6] | [7] | [8] | [9] | [10] |
| <i>L</i> | 16  | 30  | 24  | 7   | 25  | 62  | 45  | 5   | 65  | 50   |

←————— unsorted list —————→

FIGURE 9.11 Smallest element of the unsorted list

Because this is the smallest item, it must be moved to position 1. We therefore swap 16 (that is,  $L[1]$ ) with 5 (that is,  $L[8]$ ), as shown in Figure 9.12.

|          |     |     |     |     |     |     |     |     |     |      |
|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|
|          | [1] | [2] | [3] | [4] | [5] | [6] | [7] | [8] | [9] | [10] |
| <i>L</i> | 16  | 30  | 24  | 7   | 25  | 62  | 45  | 5   | 65  | 50   |

←————— unsorted list —————→

FIGURE 9.12 Swap elements  $L[1]$  and  $L[8]$

After swapping these elements, Figure 9.13 shows the resulting list.

|          |     |     |     |     |     |     |     |     |     |      |
|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|
|          | [1] | [2] | [3] | [4] | [5] | [6] | [7] | [8] | [9] | [10] |
| <i>L</i> | 5   | 30  | 24  | 7   | 25  | 62  | 45  | 16  | 65  | 50   |

←————— unsorted list —————→

FIGURE 9.13 List after swapping  $L[1]$  and  $L[8]$

Next we repeat the above process on the list  $L[2 \dots 10]$ .

The following three procedures and functions implement the selection sort algorithm.

**ALGORITHM 9.3:** Determine the position of the smallest element in a list.

*Input:*  $L$ —list  
           first—index of the first element in the sublist  
           last—index of the last element of the sublist

*Output:* The position of the smallest element in  $L[\text{first} \dots \text{last}]$  is returned.

1. **function** **minLocation**( $L$ , first, last)
2. **begin**
3.     minIndex := first;
4.     **for** loc := first + 1 **to** last **do**
5.         **if**  $L[\text{loc}] < L[\text{minIndex}]$  **then**
6.             minIndex := loc;
7.     **return** minIndex;
8. **end**

Given the locations in the list of the elements to be swapped, the following procedure, **swap**, swaps those elements.

**ALGORITHM 9.4:** Swap two elements of a list.

*Input:*  $L$ —list  
           first—index of the first element in the sublist  
           second—index of the second element of the sublist

*Output:* List after swapping  $L[\text{first}]$  and  $L[\text{second}]$

1. **procedure** **swap**( $L$ , first, second)
2. **begin**
3.     temp :=  $L[\text{first}]$ ;
4.      $L[\text{first}] := L[\text{second}]$ ;
5.      $L[\text{second}] := \text{temp}$ ;
6. **end**

We can now complete the definition of the procedure **selectionSort**.

**ALGORITHM 9.5: Selection Sort.**

*Input:*  $L$ —list  
            $n$ —the number of elements in  $L$

*Output:*  $L$  is sorted.

1. **procedure** **selectionSort**( $L$ ,  $n$ )
2. **begin**

```

3.   for loc := 1 to n - 1 do
4.     begin
5.       minIndex := minLocation(L, loc, n);
6.       swap(L, loc, minIndex);
7.     end
8.   end

```

### Selection Sort Analysis

In the case of search algorithms, our only concern has been with the number of item comparisons. A sorting algorithm makes item comparisons and also moves the data. Therefore, in analyzing the sorting algorithm, we look at the number of item comparisons as well as the number of data movements. Let us look at the performance of a selection sort.

Suppose that the length of the list is  $n$ . The function `swap` makes three item assignments and is executed  $n - 1$  times. Hence, the number of item assignments is  $3(n - 1)$ .

The key comparisons are made by the function `minLocation`. For a list of length  $k$ , the function `minLocation` makes  $k - 1$  key comparisons. Also, the function `minLocation` is executed  $n - 1$  times (by the procedure `selectionSort`). The first time, the function `minLocation` finds the index of the smallest key item in the entire list and therefore makes  $n - 1$  comparisons. The second time, the function `minLocation` finds the index of the smallest element in the sublist of length  $n - 1$  and makes  $n - 2$  comparisons, and so on. Hence the number of item comparisons is as follows:

$$\begin{aligned}
 (n - 1) + (n - 2) + \cdots + 2 + 1 &= \frac{n(n - 1)}{2} \\
 &= \frac{1}{2}n^2 - \frac{1}{2}n \\
 &= \Theta(n^2).
 \end{aligned}$$

It thus follows that if  $n = 1000$ , the number of key comparisons the selection sort algorithm makes is

$$\frac{1}{2}(1000)^2 - \frac{1}{2}(1000) = 499500 \approx 500000.$$

### Insertion Sort

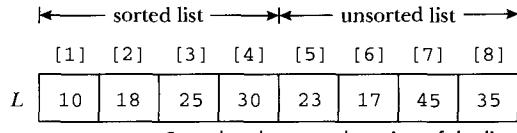
In the previous section, we described and analyzed the selection sort algorithm. We showed that if  $n = 1000$ , the number of item comparisons is approximately 500,000, which is quite high. In this section, we describe the sorting algorithm called the **insertion sort**, which tries to improve—that is, reduce—the number of key comparisons.

The insertion sort algorithm sorts the list by moving each element to its proper place. Consider the list in Figure 9.14.

|   | [1] | [2] | [3] | [4] | [5] | [6] | [7] | [8] |
|---|-----|-----|-----|-----|-----|-----|-----|-----|
| L | 10  | 18  | 25  | 30  | 23  | 17  | 45  | 35  |

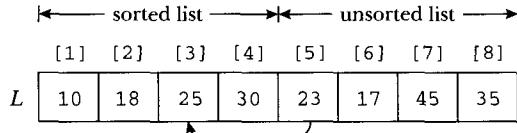
FIGURE 9.14 List

The length of the list is 8. In this list, the elements  $L[1]$ ,  $L[2]$ ,  $L[3]$ , and  $L[4]$  are in order. That is,  $L[1 \dots 4]$  is sorted (see Figure 9.15).



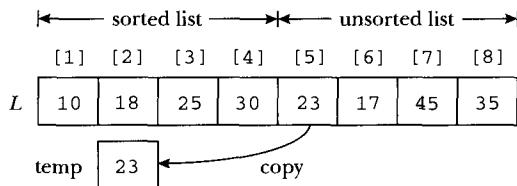
**FIGURE 9.15** Sorted and unsorted portion of the list

Next, we consider element  $L[5]$ , the first element of the unsorted list. Because  $L[5] < L[4]$ , we need to move element  $L[5]$  to its proper location. It thus follows that element  $L[5]$  should be moved to  $L[3]$  (see Figure 9.16).



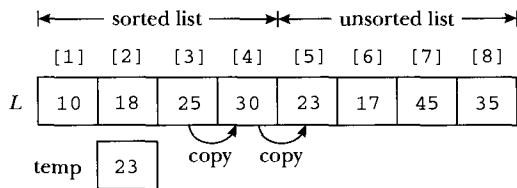
**FIGURE 9.16** Move  $L[5]$  into  $L[3]$

To move  $L[5]$  into  $L[3]$ , first we copy  $L[5]$  into `temp`, a temporary memory space (see Figure 9.17).



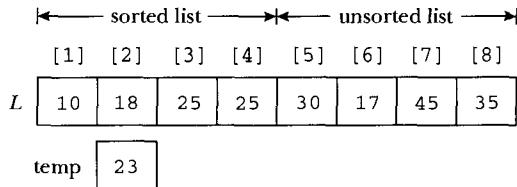
**FIGURE 9.17** Copy  $L[5]$  into `temp`

Next, we copy  $L[4]$  into  $L[5]$ , and then  $L[3]$  into  $L[4]$  (see Figure 9.18).



**FIGURE 9.18** List before copying  $L[4]$  into  $L[5]$  and then  $L[3]$  into  $L[4]$

After copying  $L[4]$  into  $L[5]$  and  $L[3]$  into  $L[4]$ , the list is as shown in Figure 9.19.



**FIGURE 9.19** List after copying  $L[4]$  into  $L[5]$  and then  $L[3]$  into  $L[4]$

We now copy  $\text{temp}$  into  $L[3]$ . Figure 9.20 shows the resulting list.

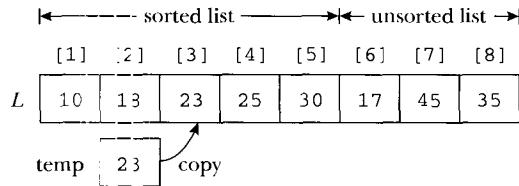


FIGURE 9.20 List after copying  $\text{temp}$  into  $L[2]$

Now  $L[1 \dots 5]$  is sorted and  $L[6 \dots 8]$  is unsorted. We repeat this process on the resulting list by moving the first element of the unsorted list into the sorted list in the proper place.

We see that during the sorting phase the array containing the list is divided into two sublists, lower and upper. Elements in the lower sublist are sorted; elements in the upper sublist are to be moved to the lower sublist in their proper places one at a time. We use an index—say, `firstOutOfOrder`—to point to the first element in the upper sublist; that is, `firstOutOfOrder` gives the index of the first element in the unsorted portion of the array. Initially, `firstOutOfOrder` is initialized to 2.

#### ALGORITHM 9.6: Insertion Sort.

*Input:*  $L[1 \dots n]$ —list  
 $n$ —the number of elements in  $L$

*Output:* Sorted  $L$

```

1. procedure insertionSort( $L, n$ )
2. begin
3.   for  $\text{firstOutOfOrder} := 2$  to  $n$  do
4.     if  $L[\text{firstOutOfOrder}] < L[\text{firstOutOfOrder} - 1]$  then
5.       begin
6.          $\text{temp} := L[\text{firstOutOfOrder}]$ ;
7.          $\text{loc} := \text{firstOutOfOrder}$ ;
8.         do
9.           begin
10.              $L[\text{loc}] := L[\text{loc} - 1]$ ;
11.              $\text{loc} := \text{loc} - 1$ ;
12.           end
13.           while ( $\text{loc} > 0$  and  $L[\text{loc} - 1] > \text{temp}$ );
14.            $L[\text{loc}] := \text{temp}$ ;
15.       end
16.   end

```

**REMARK 9.2.1** ▶ In Algorithm 9.6, consider the statement in Line 13. Suppose  $\text{loc} \leq 0$ . Then  $\text{loc} > 0$  is false. It follows that, in this case, the expression  $(\text{loc} > 0 \text{ and } L[\text{loc} - 1] > \text{temp})$  evaluates to false regardless of the value of the expression  $L[\text{loc} - 1]$ .

`- 1] > temp.` In cases such as these, most compilers would not evaluate the expression `L[loc - 1] > temp`, so this comparison will not be made. Therefore, when counting the exact number of comparisons to sort a list using insertion sort, if this situation occurs, we will not count this comparison.

### Insertion Sort Analysis

Consider the  $k$ th entry in the list. If the  $k$ th entry is moved, it could go to any of the first  $k - 1$  positions in the list. And if the  $k$ th entry is not moved, it stays in its current position. Thus, there are a total of  $k$  possibilities for the  $k$ th item:  $(k - 1)$  possibilities to move and one possibility of not moving. Assume all possibilities are equally likely. Then the probability of not moving is  $\frac{1}{k}$ , and the probability of moving is  $\frac{(k-1)}{k}$ .

If the  $k$ th entry is not moved, then the number of key comparisons is one and number of item assignments is zero.

Suppose that the  $k$ th entry is moved. Then the average number of key comparisons (executed by the loop) to move  $k$ th entry is:

$$\frac{1 + 2 + 3 + \cdots + (k - 1)}{k - 1} = \frac{k(k - 1)}{2(k - 1)} = \frac{k}{2}.$$

Now one key comparison is made before the `do/while` loop, one item assignment is done before the `do/while` loop, and one item assignment is done after the loop. It now follows that, if the  $k$ th entry is moved, on an average it requires  $\frac{k}{2} + 1$  key comparisons and  $\frac{k}{2} + 2$  item assignments.

Because the probability of moving the  $k$ th entry is  $\frac{k-1}{k}$  and the probability of not moving is  $\frac{1}{k}$ , the average number of key comparisons for the  $k$ th entry is:

$$\begin{aligned} \left(\frac{k-1}{k}\right)\left(\frac{k}{2} + 1\right) + \frac{1}{k}1 &= \frac{k-1}{k} \cdot \frac{k+2}{2} + \frac{1}{k} \\ &= \frac{(k-1)(k+2) + 2}{2k} \\ &= \frac{k(k+1)}{2k} \\ &= \frac{k+1}{2} \\ &= \frac{1}{2}k + \frac{1}{2} \end{aligned}$$

Similarly, the average number of assignments for the  $k$ th entry is:

$$\begin{aligned} \left(\frac{k-1}{k}\right)\left(\frac{k}{2} + 2\right) + \frac{1}{k}0 &= \frac{k-1}{k} \cdot \frac{k+4}{2} \\ &= \frac{(k-1)(k+4)}{2k} \\ &= \frac{k^2 + 3k - 4}{2k} \\ &= \frac{k^2}{2k} + \frac{3k}{2k} - \frac{4}{2k} \\ &= \frac{1}{2}k + \frac{3}{2} - \frac{2}{k} \\ &= \frac{1}{2}k + \Theta(1). \end{aligned}$$

Note that the average number of key comparisons and the average number of item assignments for the  $k$ th entry are similar.

To find the average number of key comparisons (item assignments) made by insertion sort, we add the average number of key comparisons made by list entries 2 through  $n$ . (Note that the `for` loop starts at the second entry of the list.) Thus, the average number of key comparisons is:

$$\begin{aligned}
 \sum_{k=2}^n \left[ \frac{1}{2}k + \frac{1}{2} \right] &= \frac{1}{2} \sum_{k=2}^n k + \sum_{k=2}^n \frac{1}{2} \\
 &= \frac{1}{2} \sum_{k=2}^n k + \frac{n-1}{2} \\
 &= \frac{1}{2} \sum_{k=1}^n k + \frac{n-1}{2} - \frac{1}{2} \\
 &= \frac{1}{2} \frac{n(n+1)}{2} + \frac{n-1}{2} - \frac{1}{2} \\
 &= \frac{n(n+1) + 2(n-1) - 2}{4} \\
 &= \frac{n^2 + n + 2n - 4}{4} \\
 &= \frac{n^2 + 3n - 4}{4} \\
 &= \frac{1}{4}(n^2 + 3n - 4) \\
 &= \Theta(n^2).
 \end{aligned}$$

In a similar manner, we can show that the average number of item assignments made by insertion sort is  $\Theta(n^2)$ .

---

**REMARK 9.2.2 ▶** Let  $L$  be a list of size 1000. Then, as shown earlier, we determine that the average number of key comparisons made by the selection sort algorithm to sort  $L$  is  $\approx 500000$ . However, if insertion sort is used to sort  $L$ , then the average number of key comparisons is  $\frac{1}{4}(1000000 + 3000 - 4) \approx 250000$ . Thus, on average, insertion makes fewer number of key comparisons. However, note that the number of item assignments made by selection sort is  $\approx 3000$ . The average number of item assignments made by insertion sort is considerably more than selection sort. We leave the details as an exercise.

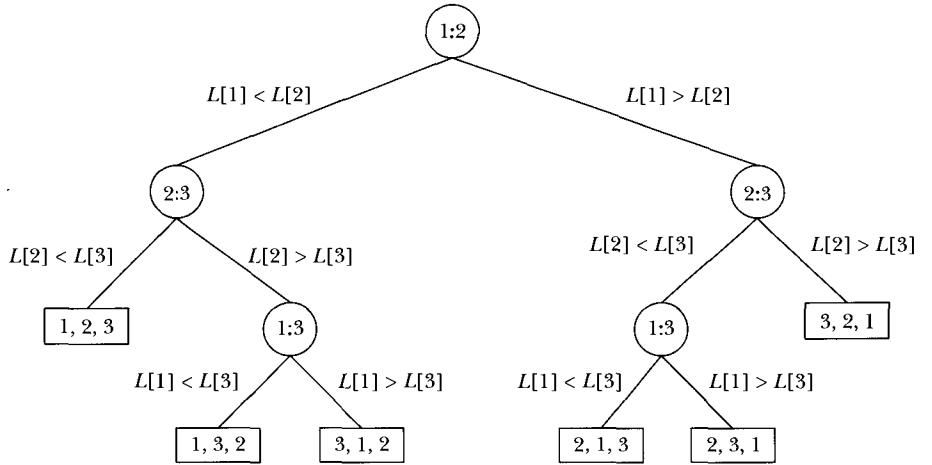
## Lower Bound on Comparison-Based Sort Algorithms

We can trace the execution of a comparison-based algorithm using a graph called a *comparison tree* or a *decision tree*. Let  $L$  be a list of  $n$  distinct elements,  $n > 0$ . For any  $j, k$ ,  $1 \leq j \leq n$ ,  $1 \leq k \leq n$ , either  $L[j] < L[k]$  or  $L[j] > L[k]$ . The comparison tree is a binary tree such that each internal node is labeled as  $j:k$  representing the comparison of  $L[j]$  with  $L[k]$ . If  $L[j] < L[k]$ , follow the left branch; otherwise follow the right branch. Figure 9.21 shows the comparison tree for a list of length 3.

The top node, with the label 1:2, is called the *root* node. The square nodes are called *leaves*.

---

**REMARK 9.2.3 ▶** Trees are discussed in detail in Chapter 11.



**FIGURE 9.21** Comparison tree for a list  $L$  of length 3

Associated with each path from the root to a leaf is a unique permutation of the elements of  $L$ . The uniqueness follows because the sort algorithm only moves data and makes comparisons. Further, the data movement on any path from the root to a leaf is the same regardless of what the initial inputs are. Now for a list of  $n$  elements,  $n > 0$ , there are  $n!$  different permutations. Any one of these  $n!$  permutations might be the correct ordering of  $L$ . Thus the comparison tree must have at least  $n!$  leaves.

Let us consider the worst case for all comparison-based sort algorithms. Let  $S(n)$  be the minimum number of comparisons required to sort  $L$  in the worst case. It can be shown that

$$n! \leq 2^{S(n)}.$$

This implies that because  $S(n)$  is an integer,

$$S(n) \geq \lceil (\lg n!) \rceil.$$

By Stirling approximation,

$$(\lg n!) = n \lg n - \frac{n}{\ln n} + \frac{1}{2} \lg n + O(1).$$

Hence,

$$S(n) = O(n \lg n).$$

We have thus proved the following theorem.

**Theorem 9.2.4:** Let  $L$  be a list of  $n$  distinct elements. Any comparison-based sort algorithm in its worst case must be at least  $O(n \lg n)$  to sort  $L$ .

## Merge Sort

In the previous section, we noted that the lower bound on comparison-based algorithms is  $O(n \lg n)$ . The selection sort and insertion sort algorithms, which we discussed earlier in this chapter, are both of the order  $\Theta(n^2)$ . In this section we discuss a sorting algorithm that is of the order  $\Theta(n \lg n)$ .

The **merge sort** algorithm uses the divide-and-conquer technique to sort a list. It partitions the list into two sublists, sorts the sublists, and then combines the sorted sublists into one sorted list. We assume that data are stored in an array.

Consider the list whose elements are as follows:

|       |    |    |    |    |    |    |    |    |
|-------|----|----|----|----|----|----|----|----|
| $L :$ | 35 | 28 | 18 | 45 | 62 | 48 | 30 | 38 |
|-------|----|----|----|----|----|----|----|----|

The merge sort algorithm partitions this list into two sublists as follows:

|                 |    |    |    |    |
|-----------------|----|----|----|----|
| First sublist:  | 35 | 28 | 18 | 45 |
| Second sublist: | 62 | 48 | 30 | 38 |

The two sublists are sorted using the same algorithm (that is, a merge sort) used on the original list. Suppose that we have sorted the sublist. That is, suppose that

|                 |    |    |    |    |
|-----------------|----|----|----|----|
| First sublist:  | 18 | 28 | 35 | 45 |
| Second sublist: | 30 | 38 | 48 | 62 |

Next, the merge sort algorithm combines—that is, merges—the two sorted sublists into one sorted list.

Figure 9.22 further illustrates the merge sort process.

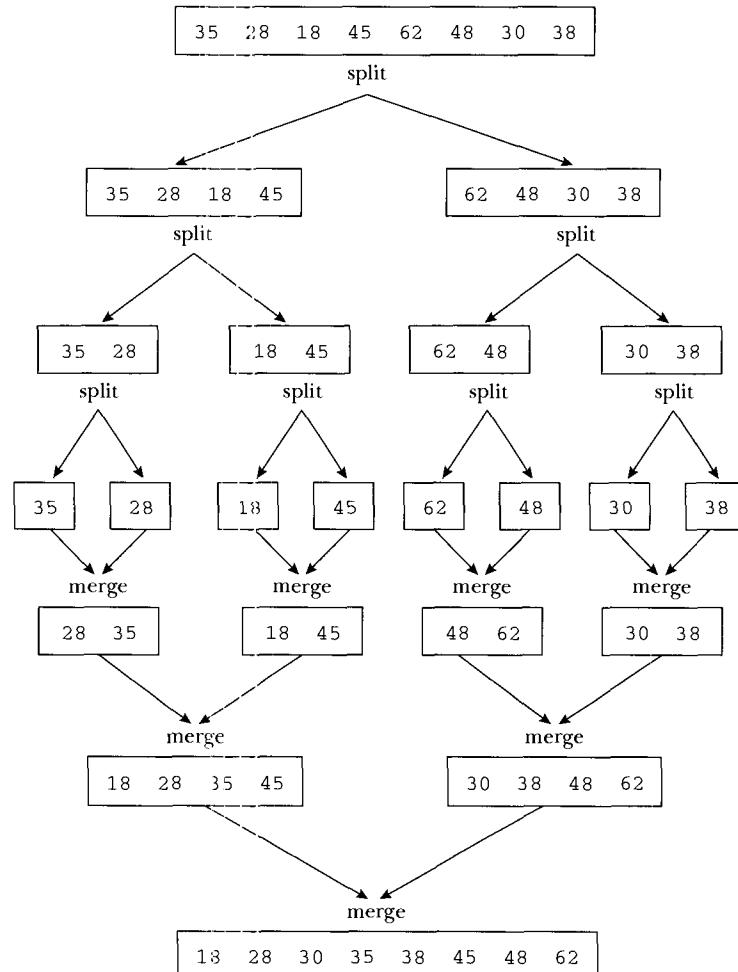


FIGURE 9.22 Merge sort process

From Figure 9.22, it follows that in the merge sort algorithm, most of the sorting work is done in merging the sorted sublists.

The general algorithm for the merge sort is as follows:

If the list is of size greater than 1, then

- a. Find the mid-position of the list.
- b. Merge sort the first sublist.
- c. Merge sort the second sublist.
- d. Merge the first sublist and the second sublist.

To state again what happens: After dividing the list into two sublists—the first sublist and the second sublist—the two sublists are sorted using the merge sort algorithm. In other words, we use recursion to implement the merge sort algorithm. The following algorithm implements the merge sort algorithm.

**ALGORITHM 9.7:** Merge sort algorithm to sort a list.

*Input:*  $L$ —list of length  $n$   
 $s$ —index of the first element in  $L$   
 $t$ —index of the last element in  $L$

*Output:* Sorted  $L[s \dots t]$ , i.e., elements  $L[s] \dots L[t]$  are in order.

```

1. procedure recMergeSort(L, s, t)
2. begin
3.   if s < t then
4.     begin
5.       m :=  $\lfloor \frac{s+t}{2} \rfloor$ ; //find the mid position of L[s...t]
6.       recMergeSort(L,s,m);      //Merge sort L[s...m]
7.       recMergeSort(L,m + 1,t); //Merge sort L[m+1...t]
8.       mergeLists(L,A,s,t,m); //Merge L[s...m] and L[m+1...t];
9.                               //A[s...t] contains merged lists
10.      for i := s to t do        //Copy A[s...t] into L[s...t]
11.        L[i] := A[i];
12.      end
13.    end

```

## Merge

Once the sublists are sorted, the next step in the merge sort algorithm is to merge the sorted sublists. Let us illustrate the merge process. Suppose  $L_1$  and  $L_2$  are two sorted lists as follows:

|       |   |    |    |    |
|-------|---|----|----|----|
| $L_1$ | 2 | 7  | 16 | 35 |
| $L_2$ | 5 | 20 | 25 | 40 |

We merge  $L_1$  and  $L_2$  into a third list, say  $L_3$ . The merge process is as follows: We repeatedly compare, using a loop, the elements of  $L_1$  with the elements of  $L_2$  and copy the smaller element into  $L_3$ .

First we compare  $L_1[1]$  with  $L_2[1]$  and see that  $L_1[1] < L_2[1]$ , so we copy  $L_1[1]$  into  $L_3[1]$  (see Figure 9.23). (Notice that  $i, j$ , and  $k$  are set to 1.)

|       |   |    |    |    |    |  |  |  |  |  |  |  |  |
|-------|---|----|----|----|----|--|--|--|--|--|--|--|--|
| $L_1$ | i | 2  | 7  | 16 | 35 |  |  |  |  |  |  |  |  |
|       | j |    |    |    |    |  |  |  |  |  |  |  |  |
| $L_2$ | 5 | 20 | 25 | 40 | 50 |  |  |  |  |  |  |  |  |
| $L_3$ | k |    |    |    |    |  |  |  |  |  |  |  |  |

Before Iteration 1

$$L_1[i] < L_2[j]$$

$$L_3[k] := L_1[i]$$

$$i := i + 1;$$

$$k := k + 1;$$

|       |   |    |    |    |    |  |  |  |  |  |  |  |  |
|-------|---|----|----|----|----|--|--|--|--|--|--|--|--|
| $L_1$ | i | 2  | 7  | 16 | 35 |  |  |  |  |  |  |  |  |
|       | j |    |    |    |    |  |  |  |  |  |  |  |  |
| $L_2$ | 5 | 20 | 25 | 40 | 50 |  |  |  |  |  |  |  |  |
| $L_3$ | k |    |    |    |    |  |  |  |  |  |  |  |  |

After Iteration 1

**FIGURE 9.23**  $L_1, L_2$ , and  $L_3$  before and after the first iteration

After the first iteration,  $i = 2, j = 1$ , and  $k = 2$ . Next we compare  $L_1[2]$  with  $L_2[1]$  as shown in Figure 9.24.

|       |   |    |    |    |    |  |  |  |  |  |  |  |  |
|-------|---|----|----|----|----|--|--|--|--|--|--|--|--|
| $L_1$ | i | 2  | 7  | 16 | 35 |  |  |  |  |  |  |  |  |
|       | j |    |    |    |    |  |  |  |  |  |  |  |  |
| $L_2$ | 5 | 20 | 25 | 40 | 50 |  |  |  |  |  |  |  |  |
| $L_3$ | k |    |    |    |    |  |  |  |  |  |  |  |  |

Before Iteration 2

$$L_1[i] > L_2[j]$$

$$L_3[k] := L_2[j]$$

$$j := j + 1;$$

$$k := k + 1;$$

|       |   |    |    |    |    |  |  |  |  |  |  |  |  |
|-------|---|----|----|----|----|--|--|--|--|--|--|--|--|
| $L_1$ | i | 2  | 7  | 16 | 35 |  |  |  |  |  |  |  |  |
|       | j |    |    |    |    |  |  |  |  |  |  |  |  |
| $L_2$ | 5 | 20 | 25 | 40 | 50 |  |  |  |  |  |  |  |  |
| $L_3$ | k |    |    |    |    |  |  |  |  |  |  |  |  |

After Iteration 2

**FIGURE 9.24**  $L_1, L_2$ , and  $L_3$  before and after the second iteration

After the second iteration,  $i = 2, j = 2$ , and  $k = 3$ . Next we compare  $L_1[2]$  with  $L_2[2]$  as shown in Figure 9.25.

|       |   |    |    |    |    |  |  |  |  |  |  |  |  |
|-------|---|----|----|----|----|--|--|--|--|--|--|--|--|
| $L_1$ | i | 2  | 7  | 16 | 35 |  |  |  |  |  |  |  |  |
|       | j |    |    |    |    |  |  |  |  |  |  |  |  |
| $L_2$ | 5 | 20 | 25 | 40 | 50 |  |  |  |  |  |  |  |  |
| $L_3$ | k |    |    |    |    |  |  |  |  |  |  |  |  |

Before Iteration 3

$$L_1[i] < L_2[j]$$

$$L_3[k] := L_1[i]$$

$$i := i + 1;$$

$$k := k + 1;$$

|       |   |    |    |    |    |  |  |  |  |  |  |  |  |
|-------|---|----|----|----|----|--|--|--|--|--|--|--|--|
| $L_1$ | i | 2  | 7  | 16 | 35 |  |  |  |  |  |  |  |  |
|       | j |    |    |    |    |  |  |  |  |  |  |  |  |
| $L_2$ | 5 | 20 | 25 | 40 | 50 |  |  |  |  |  |  |  |  |
| $L_3$ | k |    |    |    |    |  |  |  |  |  |  |  |  |

After Iteration 3

**FIGURE 9.25**  $L_1, L_2$ , and  $L_3$  before and after the third iteration

After the third iteration,  $i = 3, j = 2$ , and  $k = 4$ . Next we compare  $L_1[3]$  with  $L_2[2]$  as shown in Figure 9.26.

|       |   |    |    |    |    |  |  |  |  |  |  |  |  |
|-------|---|----|----|----|----|--|--|--|--|--|--|--|--|
| $L_1$ | i | 2  | 7  | 16 | 35 |  |  |  |  |  |  |  |  |
|       | j |    |    |    |    |  |  |  |  |  |  |  |  |
| $L_2$ | 5 | 20 | 25 | 40 | 50 |  |  |  |  |  |  |  |  |
| $L_3$ | k |    |    |    |    |  |  |  |  |  |  |  |  |

Before Iteration 4

$$L_1[i] < L_2[j]$$

$$L_3[k] := L_1[i]$$

$$i := i + 1;$$

$$k := k + 1;$$

|       |   |    |    |    |    |  |  |  |  |  |  |  |  |
|-------|---|----|----|----|----|--|--|--|--|--|--|--|--|
| $L_1$ | i | 2  | 7  | 16 | 35 |  |  |  |  |  |  |  |  |
|       | j |    |    |    |    |  |  |  |  |  |  |  |  |
| $L_2$ | 5 | 20 | 25 | 40 | 50 |  |  |  |  |  |  |  |  |
| $L_3$ | k |    |    |    |    |  |  |  |  |  |  |  |  |

After Iteration 4

**FIGURE 9.26**  $L_1, L_2$ , and  $L_3$  before and after the fourth iteration

After the fourth iteration,  $i = 4$ ,  $j = 2$ , and  $k = 4$ . Next we compare  $L_1[4]$  with  $L_2[2]$  as shown in Figure 9.27.

|       |  |   |  |  |  |  |  |  |  |  |  |
|-------|--|---|--|--|--|--|--|--|--|--|--|
| $L_1$ |  | i |  |  |  |  |  |  |  |  |  |
| $L_2$ |  | j |  |  |  |  |  |  |  |  |  |
| $L_3$ |  | k |  |  |  |  |  |  |  |  |  |

$L_1[i] > L_2[j]$   
 $L_3[k] := L_2[j]$   
 $j := j + 1;$   
 $k := k + 1;$

$L_1$       i  

|   |   |    |    |  |
|---|---|----|----|--|
| 2 | 7 | 16 | 35 |  |
|---|---|----|----|--|

  
 $L_2$       j  

|   |    |    |    |    |  |
|---|----|----|----|----|--|
| 5 | 20 | 25 | 40 | 50 |  |
|---|----|----|----|----|--|

  
 $L_3$       k  

|   |   |   |    |    |  |  |  |  |  |  |  |
|---|---|---|----|----|--|--|--|--|--|--|--|
| 2 | 5 | 7 | 16 | 20 |  |  |  |  |  |  |  |
|---|---|---|----|----|--|--|--|--|--|--|--|

Before Iteration 5

FIGURE 9.27  $L_1$ ,  $L_2$ , and  $L_3$  before and after the fifth iteration

After Iteration 5

After the fifth iteration,  $i = 4$ ,  $j = 3$ , and  $k = 5$ . We continue this process until all elements of one list are copied into the third list. We then copy the remaining elements of the list (that has elements left to be merged) into the merged list.

Next, using the preceding procedure, we write the algorithm to merge two sorted lists. We assume that both lists, say  $L_1$  and  $L_2$ , are stored in the same array, say  $L$ . Suppose  $L[s \dots m]$  represents the elements of  $L_1$  and  $L[m+1 \dots t]$  represents the elements of  $L_2$ . Therefore, we will initialize  $i$  to  $s$ ,  $j$  to  $m+1$ , and  $k$  to  $s$ . The array  $A$  contains the merged list, and the size of  $A$  is the same as the size of  $L$ .

The following procedure, `mergeLists`, merges the two sorted sublists.

#### ALGORITHM 9.8: Merge two sorted lists.

**Input:**  $L, A$ —arrays of the same size  
 $s, t, m$ —positive integers  
 $L[s \dots m]$ —contains the first sublist  
 $L[m+1 \dots t]$ —contains the second sublist

**Output:**  $A - A[s \dots t]$  contains the elements of  $L[s \dots t]$  in order.

```

1. procedure mergeLists( $L, A, s, t, m$ )
2. begin
3.    $i := s$ ;
4.    $j := m + 1$ ;
5.    $k := s$ ;
6.   while  $i \leq m$  and  $j \leq t$  do
7.     begin
8.       if  $L[i] < L[j]$  then
9.         begin
10.           $A[k] := L[i]$ ;
11.           $i := i + 1$ ;
12.        end
13.       else
14.         begin

```

```

15.      A[k] := L[j];
16.      j := j + 1;
17.      end
18.      k := k + 1;
19.      end
20.      if i ≤ m then
21.          while i ≤ m do
22.              begin
23.                  A[k] := L[i];
24.                  k := k + 1;
25.                  i := i + 1;
26.              end
27.          else
28.              while j ≤ t do
29.                  begin
30.                      A[k] := L[j];
31.                      k := k + 1;
32.                      j := j + 1;
33.                  end
34.      end

```

The procedure `recMergeSort` sorts a list between two indices in the list. For example, if a call to this procedure is, say

```
recMergeSort(L, 5, 10);
```

then list  $L[5 \dots 10]$  is sorted. Similarly, if the call is

```
recMergeSort(L, 1, n);
```

where  $n$  is the size of the list, then the entire list is sorted.

Next, we give the definition of the procedure `mergeSort`, which simply calls the function `recMergeSort` and passes the list and the length of the list as parameters.

#### ALGORITHM 9.9: Merge sort algorithm to sort a list.

*Input:*  $L$ —list  
 $n$ —length of  $L$

*Output:* Sorted  $L$

1. **procedure** `mergeSort`( $L, n$ )
2. **begin**

```

3. recMergeSort(L, 1, n);
4. end

```

## Merge Sort Analysis

Suppose  $L$  is a list of  $n$  elements,  $n > 0$ . Let  $A(n)$  denote the number of comparisons in the average case and  $W(n)$  denote the number of comparisons in the worst case to sort  $L$ . First we determine a formula for  $W(n)$ .

In merge sort all the comparisons are made in the procedure `mergeList`, which merges two sorted sublists. Suppose that one sublist is of size  $s$  and other sublist is of size  $t$ . It can be shown that merging these lists would require at most  $s + t - 1$  comparisons in the worst case. Hence,

$$W(n) = W(s) + W(t) + s + t - 1.$$

Note that  $s = \lfloor \frac{n}{2} \rfloor$ , and  $t = \lceil \frac{n}{2} \rceil$ . Suppose that  $n = 2^m$ . Then  $s = 2^{m-1}$  and  $t = 2^{m-1}$ . It follows that  $s + t = n$ . Hence,

$$\begin{aligned} W(n) &= W\left(\frac{n}{2}\right) + W\left(\frac{n}{2}\right) + n - 1 \\ &= 2W\left(\frac{n}{2}\right) + n - 1, \quad n > 0. \end{aligned}$$

Also,

$$W(1) = 0.$$

As shown in Worked-Out Exercise 4, Section 8.3, when  $n$  is a power of 2,  $W(n)$  is given by the following equation:

$$W(n) = n \lg n - (n - 1) = \Theta(n \lg n).$$

We can show that  $W(n)$  is eventually nondecreasing and conclude that  $W(n) = \Theta(n \lg n)$ .

In the average case, during merge one of the sublists will exhaust before the other list. From this, it follows that on an average merging of two sorted sublists of combined size  $n$ , the number of comparisons will be less than  $n - 1$ . On an average it can be shown that number of comparison for merge sort is given by the following equation: If  $n$  is a power of 2,

$$A(n) = n \lg n - 1.26n.$$

This is also a good approximation when  $n$  is not a power of 2.

## Smallest and Largest Elements in a List

Let  $L$  be a list of  $n$  elements, where  $n$  is a nonnegative integer. In Chapter 1, we described an algorithm to determine the smallest element in  $L$ . In a similar manner, we can design an algorithm to determine the largest element in  $L$ . In this section, we describe various algorithms to find the smallest and largest elements simultaneously. We also give an analysis of the algorithms to show the efficiency of each algorithm.

We begin with the following algorithm, which is similar to the algorithm of determining the smallest element in  $L$ , given in Chapter 1.

**ALGORITHM 9.10: Max-Min Algorithm 1.**

*Input:*  $L$ —a list of  $n$  elements  
 $n$ —the number of elements in  $L$

*Output:*  $\max$ —the maximum element of the list  
 $\min$ —the minimum element of the list

```

1. procedure maxMin1( $L, n, \max, \min$ )
2. begin
3.    $\max := L[1];$ 
4.    $\min := L[1];$ 
5.   for  $i := 2$  to  $n$  do
6.     begin
7.       if  $L[i] > \max$  then
8.          $\max := L[i];$ 
9.       if  $L[i] < \min$  then
10.         $\min := L[i];$ 
11.     end
12.   end

```

In this example, the statement in Lines 3 and 4 executes one time. The comparisons are made at Lines 7 and 9. The **for** loop at Line 5 executes  $n - 1$  times. Each time through the loop the statements in Lines 7 and 9 executes. Thus, the number of comparisons, in the best, average, and worst case is:

$$2(n - 1).$$

Note that the statement in Line 7 determines whether  $L[i]$ , the current element of the list, is greater than  $\max$ . If  $L[i] > \max$  is true, then of course  $L[i]$  cannot be less than  $\min$ . Thus, the statement in Line 9 should be executed only if the statement in Line 7 evaluates to false. Thus, we can improve algorithm 1 by incorporating this observation. Let us modify algorithm 1 as follows.

**ALGORITHM 9.11: Max-Min Algorithm 2.**

*Input:*  $L$ —a list of  $n$  elements  
 $n$ —the number of elements in  $L$

*Output:*  $\max$ —the maximum element of the list  
 $\min$ —the minimum element of the list

```

1. procedure maxMin2( $L, n, \max, \min$ )
2. begin
3.    $\max := L[1];$ 
4.    $\min := L[1];$ 
5.   for  $i := 2$  to  $n$  do
6.     begin

```

```

7.   if L[i] > max then
8.       max := L[i];
9.   else
10.      if L[i] < min then
11.          min := L[i];
12.      end
13.  end

```

If the elements of  $L$  are in increasing order, then the statement in Line 7 always evaluates true and the statement in Line 10 does not execute. It follows that, in this case, the number of comparisons is  $n - 1$  because the `for` loop executes  $n - 1$  times, each time making one comparison. This is the best case.

If the elements of  $L$  are in decreasing order, then the statement in Line 7 always evaluates to false, so each time through the loop the statement in Line 10 executes. It follows that, in this case, the number of comparisons is  $2(n - 1)$  because the `for` loop executes  $n - 1$  times, and each time through the loop the statements in Lines 7 and 10 both execute. This is the worst case.

In the average case, half of the time the elements of  $L$  are greater than  $\text{max}$  and the other half of the time they are less than  $\text{max}$ . Thus, statement 7 evaluates to true only  $\frac{n-1}{2}$  times. Now if the statement in Line 7 evaluates to false, then the statement in Line 10 executes, thus making two comparisons through the loop. It now follows that, in this case, the number of comparisons in the average case is:

$$\begin{aligned} \frac{n-1}{2} + 2\left(\frac{n-1}{2}\right) &= \frac{n-1}{2} + n-1 \\ &= \frac{3}{2}n - \frac{3}{2}. \end{aligned}$$

---

**REMARK 9.2.5** ► In Exercise 16, p. 600, we describe the divide-and-conquer technique to find the smallest and largest elements simultaneously.

## Strassen's Matrix Multiplication

Let  $A = [a_{ij}]$  and  $B = [b_{ij}]$  be two  $n \times n$  matrices of real numbers. The multiplication of  $A$  and  $B$ ,  $AB$ , is the  $n \times n$  matrix:

$$AB = [c_{ij}],$$

where

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{in}b_{nj} = \sum_{k=1}^n a_{ik}b_{kj}$$

for all  $i = 1, 2, \dots, n, j = 1, 2, \dots, n$ . Thus, the element  $c_{ij}$  is determined by multiplying the elements of the  $i$ th row of  $A$  with the corresponding elements of the  $j$ th column of  $B$ .

Here we are dealing with two types of multiplication, the multiplication of real numbers and the multiplication of matrices. To distinguish between them, we call the multiplication of real numbers *scalar* multiplication.

It follows that in order to compute  $c_{ij}$  we need to evaluate  $n$  scalar multiplications. Because the matrix  $AE$  has  $n^2$  elements, the number of scalar multiplications required to compute  $AB$  is, therefore,  $n^3$ . From this it follows that the standard algorithm for multiplying two matrices of sizes  $n \times n$  is of the order  $O(n^3)$ . For example, the following algorithm, given in Chapter 4, determines the elements of the matrix  $AB$ . (Let  $C$  denote the multiplication of  $A$  and  $B$ ;  $A$ ,  $B$ , and  $C$  are two-dimensional arrays of size  $n \times n$ .)

```

1. for i := 1 to n do
2.   for j := 1 to n do
3.     begin
4.       C[i, j] := 0.0;
5.       for k := 1 to n do
6.         C[i, j] := C[i, j] + A[i, k] * B[k, j];
7.     end
```

There are three nested **for** loops, in Lines 1, 2, and 5, each executing  $n$  times. Therefore, this algorithm is of  $O(n^3)$ .

For large matrices, this algorithm could be very expensive. In this section, we describe **Strassen's algorithm**, which is faster than this, to multiply two matrices. Before describing Strassen's algorithm, let us observe the following.

Consider two  $2 \times 2$  matrices,  $A$  and  $B$ :

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}.$$

Then,

$$AB = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix}.$$

Thus, to compute  $AB$  using the standard algorithm, we need 8 scalar multiplications and 4 additions. Strassen noted that if

$$\begin{aligned} m_1 &= (a_{11} + a_{22})(b_{11} + b_{22}) \\ m_2 &= (a_{21} + a_{22})b_{11} \\ m_3 &= a_{11}(b_{12} - b_{22}) \\ m_4 &= a_{22}(b_{21} - b_{11}) \\ m_5 &= (a_{11} + a_{12})b_{22} \\ m_6 &= (a_{21} - a_{11})(b_{11} + b_{12}) \\ m_7 &= (a_{12} - a_{22})(b_{21} + b_{22}), \end{aligned}$$

then,

$$\begin{aligned} a_{11}b_{11} + a_{12}b_{21} &= m_1 + m_4 - m_5 + m_7 \\ a_{11}b_{12} + a_{12}b_{22} &= m_3 + m_5 \\ a_{21}b_{11} + a_{22}b_{21} &= m_2 + m_4 \\ a_{21}b_{12} + a_{22}b_{22} &= m_1 + m_3 - m_2 + m_6. \end{aligned}$$

Thus,

$$AB = \begin{bmatrix} m_1 + m_4 - m_5 + m_7 & m_3 + m_5 \\ m_2 + m_4 & m_1 + m_3 - m_2 + m_6 \end{bmatrix}.$$

**EXAMPLE 9.2.6**

Suppose that

$$A = \begin{bmatrix} 2 & 1 \\ 3 & 5 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 1 & 7 \\ 4 & 8 \end{bmatrix}.$$

Then,  $a_{11} = 2, a_{12} = 1, a_{21} = 3, a_{22} = 5; b_{11} = 1, b_{12} = 7, b_{21} = 4, b_{22} = 8$ . Thus,

$$a_{11}b_{11} + a_{12}b_{21} = 2 \cdot 1 + 1 \cdot 4 = 6$$

$$a_{11}b_{12} + a_{12}b_{22} = 2 \cdot 7 + 1 \cdot 8 = 22$$

$$a_{21}b_{11} + a_{22}b_{21} = 3 \cdot 1 + 5 \cdot 4 = 23$$

$$a_{21}b_{12} + a_{22}b_{22} = 3 \cdot 7 + 5 \cdot 8 = 61.$$

On the other hand,

$$m_1 = (a_{11} + a_{22})(b_{11} + b_{22}) = (2 + 5)(1 + 8) = 63$$

$$m_2 = (a_{21} + a_{22})b_{11} = (3 + 5)1 = 8$$

$$m_3 = a_{11}(b_{12} - b_{22}) = 2(7 - 8) = -2$$

$$m_4 = a_{22}(b_{21} - b_{11}) = 5(4 - 1) = 15$$

$$m_5 = (a_{11} + a_{12})b_{22} = (2 + 1)8 = 24$$

$$m_6 = (a_{21} - a_{11})(b_{11} + b_{12}) = (3 - 2)(1 + 7) = 8$$

$$m_7 = (a_{12} - a_{22})(b_{21} + b_{22}) = (1 - 5)(4 + 8) = -48.$$

Hence,

$$m_1 + m_4 - m_5 + m_7 = 63 + 15 - 24 + (-48) = 6 = a_{11}b_{11} + a_{12}b_{21}$$

$$m_3 + m_5 = -2 + 24 = 22 = a_{11}b_{12} + a_{12}b_{22}$$

$$m_2 + m_4 = 8 + 15 = 23 = a_{21}b_{11} + a_{22}b_{21}$$

$$m_1 + m_3 - m_2 + m_6 = 63 + (-2) - 8 + 8 = 61 = a_{21}b_{12} + a_{22}b_{22}$$

Thus,

$$AB = \begin{bmatrix} 6 & 22 \\ 23 & 61 \end{bmatrix} = \begin{bmatrix} m_1 + m_4 - m_5 + m_7 & m_3 + m_5 \\ m_2 + m_4 & m_1 + m_3 - m_2 + m_6 \end{bmatrix}.$$

Computing  $m_1, m_2, m_3, m_4, m_5, m_6$ , and  $m_7$  requires 7 scalar multiplications and 10 additions and subtractions. Furthermore, computing  $m_1 + m_4 - m_5 + m_7, m_3 + m_5, m_2 + m_4$ , and  $m_1 + m_3 - m_2 + m_6$  requires 8 additions and subtractions. It follows that computing  $AB$  using  $m_1, m_2, m_3, m_4, m_5, m_6$ , and  $m_7$  requires 7 scalar multiplications and 18 additions and subtractions. However, computing  $AB$  using standard algorithms requires 8 multiplications and 4 additions. Thus, we reduced the number of multiplications by 1, but increased the number of additions and subtractions by 14. For small-size matrices, we did not gain much. Strassen's method is especially efficient for multiplying large matrices.

The technique for computing  $AB$ , the product of  $2 \times 2$  matrices, can be generalized to  $n \times n$  matrices as follows.

Let  $A$  and  $B$  be  $n \times n$  matrices such that

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1j} & \cdots & a_{1n} \\ \vdots & & \vdots & & \vdots \\ a_{i1} & & a_{ij} & & a_{in} \\ \vdots & & \vdots & & \vdots \\ a_{n1} & \cdots & a_{nj} & \cdots & a_{nn} \end{bmatrix}, \quad B = \begin{bmatrix} b_{11} & \cdots & b_{1j} & \cdots & b_{1n} \\ \vdots & & \vdots & & \vdots \\ b_{i1} & & b_{ij} & & b_{in} \\ \vdots & & \vdots & & \vdots \\ b_{n1} & \cdots & b_{nj} & \cdots & b_{nn} \end{bmatrix}$$

For simplicity we assume that  $n$  is some power of 2. We can partition matrix  $A$  as follows:

$$A = \left[ \begin{array}{ccc|ccc} a_{11} & \cdots & a_{1,\frac{n}{2}} & a_{1,\frac{n}{2}+1} & \cdots & a_{1,n} \\ \vdots & & \vdots & \vdots & & \vdots \\ \hline a_{\frac{n}{2},1} & \cdots & a_{\frac{n}{2},\frac{n}{2}} & a_{\frac{n}{2},\frac{n}{2}+1} & \cdots & a_{\frac{n}{2},n} \\ a_{\frac{n}{2}+1,1} & \cdots & a_{\frac{n}{2}+1,\frac{n}{2}} & a_{\frac{n}{2}+1,\frac{n}{2}+1} & \cdots & a_{\frac{n}{2}+1,n} \\ \vdots & & & & & \\ \hline a_{n1} & \cdots & a_{n,\frac{n}{2}} & a_{n,\frac{n}{2}+1} & \cdots & a_{nn} \end{array} \right] = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix},$$

where

$$A_{11} = \begin{bmatrix} a_{11} & \cdots & a_{1,\frac{n}{2}} \\ \vdots & & \vdots \\ a_{\frac{n}{2},1} & \cdots & a_{\frac{n}{2},\frac{n}{2}} \end{bmatrix}, \quad A_{12} = \begin{bmatrix} a_{1,\frac{n}{2}+1} & \cdots & a_{1,n} \\ \vdots & & \vdots \\ a_{\frac{n}{2},\frac{n}{2}+1} & \cdots & a_{\frac{n}{2},n} \end{bmatrix}$$

$$A_{21} = \begin{bmatrix} a_{\frac{n}{2}+1,1} & \cdots & a_{\frac{n}{2}+1,\frac{n}{2}} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{n,\frac{n}{2}} \end{bmatrix}, \quad A_{22} = \begin{bmatrix} a_{\frac{n}{2}+1,\frac{n}{2}+1} & \cdots & a_{\frac{n}{2}+1,n} \\ a_{n,\frac{n}{2}+1} & \cdots & a_{nn} \end{bmatrix}.$$

Matrices  $A_{11}$ ,  $A_{12}$ ,  $A_{21}$ , and  $A_{22}$  are called submatrices of matrix  $A$ .

Similarly, we can partition matrix  $B$  as follows:

$$B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix},$$

where the size of  $B_{11}$  is same as the size of  $A_{11}$ , the size of  $B_{12}$  is same as the size of  $A_{12}$ , the size of  $B_{21}$  is same as the size of  $A_{21}$ , and the size of  $B_{22}$  is same as the size of  $A_{22}$ .

Because  $n = 2^k$  for some nonnegative  $k$ , each of the submatrices is of the size  $\frac{n}{2} \times \frac{n}{2}$ . For example, if  $n = 8$ , then each of the submatrices is of the size  $4 \times 4$ .

Using the submatrices, we can determine  $AB$  as follows:

$$AB = \begin{bmatrix} A_{11}B_{11} + A_{12}B_{21} & A_{11}B_{12} + A_{12}B_{22} \\ A_{21}B_{11} + A_{22}B_{21} & A_{21}B_{12} + A_{22}B_{22} \end{bmatrix},$$

which requires 8 matrix multiplications and 4 additions of matrices.

Let

$$M_1 = (A_{11} + A_{22})(B_{11} + B_{22})$$

$$M_2 = (A_{21} + A_{22})B_{11}$$

$$M_3 = A_{11}(B_{12} - B_{22})$$

$$M_4 = A_{22}(B_{21} - B_{11})$$

$$M_5 = (A_{11} + A_{12})B_{22}$$

$$M_6 = (A_{21} - A_{11})(B_{11} + B_{12})$$

$$M_7 = (A_{12} - A_{22})(B_{21} + B_{22}).$$

Then,

$$A_{11}B_{11} + A_{12}B_{21} = M_1 + M_4 - M_5 + M_7$$

$$A_{11}B_{12} + A_{12}B_{22} = M_3 + M_5$$

$$A_{21}B_{11} + A_{22}B_{21} = M_2 + M_4$$

$$A_{21}B_{12} + A_{22}B_{22} = M_1 + M_3 - M_2 + M_6.$$

Thus,

$$AB = \begin{bmatrix} M_1 + M_4 - M_5 + M_7 & M_3 + M_5 \\ M_2 + M_4 & M_1 + M_3 - M_2 + M_6 \end{bmatrix},$$

which requires 7 matrix multiplications and 18 matrix additions and subtractions.

**EXAMPLE 9.2.7**

Let  $A$  and  $B$  be  $4 \times 4$  matrices as follows:

$$A = \left[ \begin{array}{cc|cc} 2 & 1 & 6 & 7 \\ 4 & 3 & 3 & 5 \\ \hline 1 & 1 & 9 & 1 \\ 5 & 4 & 8 & 7 \end{array} \right] \quad \text{and} \quad B = \left[ \begin{array}{cc|cc} 1 & 3 & 4 & 8 \\ 5 & 2 & 1 & 7 \\ \hline 8 & 7 & 9 & 2 \\ 6 & 2 & 2 & 4 \end{array} \right].$$

Then,

$$A_{11} = \begin{bmatrix} 2 & 1 \\ 4 & 3 \end{bmatrix}, \quad A_{12} = \begin{bmatrix} 6 & 7 \\ 3 & 5 \end{bmatrix}, \quad A_{21} = \begin{bmatrix} 1 & 1 \\ 5 & 4 \end{bmatrix}, \quad A_{22} = \begin{bmatrix} 9 & 1 \\ 8 & 7 \end{bmatrix},$$

$$B_{11} = \begin{bmatrix} 1 & 3 \\ 5 & 2 \end{bmatrix}, \quad B_{12} = \begin{bmatrix} 4 & 8 \\ 1 & 7 \end{bmatrix}, \quad B_{21} = \begin{bmatrix} 8 & 7 \\ 6 & 2 \end{bmatrix}, \quad B_{22} = \begin{bmatrix} 9 & 2 \\ 2 & 4 \end{bmatrix}.$$

Thus,

$$\begin{aligned} A_{11}B_{11} + A_{12}B_{21} &= \begin{bmatrix} 2 & 1 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 1 & 3 \\ 5 & 2 \end{bmatrix} + \begin{bmatrix} 6 & 7 \\ 3 & 5 \end{bmatrix} \begin{bmatrix} 8 & 7 \\ 6 & 2 \end{bmatrix} \\ &= \begin{bmatrix} 7 & 8 \\ 19 & 18 \end{bmatrix} + \begin{bmatrix} 90 & 56 \\ 54 & 31 \end{bmatrix} \\ &= \begin{bmatrix} 97 & 64 \\ 73 & 49 \end{bmatrix}. \end{aligned}$$

Similarly,

$$A_{11}B_{12} + A_{12}B_{22} = \begin{bmatrix} 77 & 63 \\ 56 & 79 \end{bmatrix}$$

$$A_{21}B_{11} + A_{22}B_{21} = \begin{bmatrix} 84 & 70 \\ 131 & 93 \end{bmatrix}$$

$$A_{21}B_{12} + A_{22}B_{22} = \begin{bmatrix} 88 & 37 \\ 110 & 112 \end{bmatrix}.$$

Now,

$$\begin{aligned} M_1 &= (A_{11} + A_{22})(B_{11} + B_{22}) \\ &= \begin{bmatrix} 11 & 2 \\ 12 & 10 \end{bmatrix} \begin{bmatrix} 10 & 5 \\ 7 & 6 \end{bmatrix} \\ &= \begin{bmatrix} 124 & 67 \\ 190 & 120 \end{bmatrix}. \end{aligned}$$

Similarly,

$$M_2 = (A_{21} + A_{22})B_{11} = \begin{bmatrix} 20 & 34 \\ 68 & 61 \end{bmatrix}$$

$$M_3 = A_{11}(B_{12} - B_{22}) = \begin{bmatrix} -11 & 15 \\ -23 & 33 \end{bmatrix}$$

$$\begin{aligned}
 M_4 &= A_{22}(B_{21} - B_{11}) & = \begin{bmatrix} 64 & 36 \\ 63 & 32 \end{bmatrix} \\
 M_5 &= (A_{11} + A_{12})B_{22} & = \begin{bmatrix} 88 & 48 \\ 79 & 46 \end{bmatrix} \\
 M_6 &= (A_{21} - A_{11})(B_{11} + B_{12}) & = \begin{bmatrix} -5 & -11 \\ 11 & 20 \end{bmatrix} \\
 M_7 &= (A_{12} - A_{22})(B_{21} + B_{22}) & = \begin{bmatrix} -3 & 9 \\ -101 & -57 \end{bmatrix}.
 \end{aligned}$$

Thus,

$$\begin{aligned}
 M_1 + M_4 - M_5 + M_7 &= \begin{bmatrix} 97 & 64 \\ 73 & 49 \end{bmatrix} \\
 M_3 + M_5 &= \begin{bmatrix} 77 & 63 \\ 56 & 79 \end{bmatrix} \\
 M_2 + M_4 &= \begin{bmatrix} 84 & 70 \\ 131 & 93 \end{bmatrix} \\
 M_1 + M_3 - M_2 + M_6 &= \begin{bmatrix} 88 & 37 \\ 110 & 112 \end{bmatrix}.
 \end{aligned}$$

A direct computation shows that:

$$AB = \begin{bmatrix} 97 & 64 & 77 & 63 \\ 73 & 49 & 56 & 79 \\ 84 & 70 & 88 & 37 \\ 131 & 93 & 110 & 112 \end{bmatrix}.$$

Hence,

$$\begin{aligned}
 AB &= \begin{bmatrix} A_{11}B_{11} + A_{12}B_{21} & A_{11}B_{12} + A_{12}B_{22} \\ A_{21}B_{11} + A_{22}B_{21} & A_{21}B_{12} + A_{22}B_{22} \end{bmatrix} \\
 &= \begin{bmatrix} M_1 + M_4 - M_5 + M_7 & M_3 + M_5 \\ M_2 + M_4 & M_1 + M_3 - M_2 + M_6 \end{bmatrix}.
 \end{aligned}$$

Let  $T(n)$  denote the time complexity of multiplying two  $n \times n$  matrices, say  $A$  and  $B$ , using Strassen's method. To calculate  $AB$  using Strassen's method, we need to calculate  $M_1, M_2, M_3, M_4, M_5, M_6$ , and  $M_7$ , and the size of each of these matrices is  $\frac{n}{2} \times \frac{n}{2}$ . In addition, we need 18 additions and subtractions as explained earlier. Thus, it follows that  $T(n)$  is given by the following recurrence relation:

$$T(n) = 7T\left(\frac{n}{2}\right) + 18\left(\frac{n}{2}\right)^2, \quad \text{if } n > 1, \quad (9.12)$$

with initial conditions

$$T(1) = 0$$

$$T(2) = 25.$$

Suppose that  $n = 2^k$  for some positive integer  $k$ . Then  $k = \lg n$ . We substitute  $n$  in the recurrence relation to get:

$$\begin{aligned}
 T(2^k) &= 7T(2^{k-1}) + 18(2^{k-1})^2 \\
 &= 7T(2^{k-1}) + \frac{18}{4}4^k.
 \end{aligned}$$

Let  $a_k = T(2^k)$ . Then

$$a_k = 7a_{k-1} + 4^k \cdot \frac{18}{4}.$$

By Theorem 8.3.6, we can conclude that  $a_k$  is of the form:

$$a_k = c_1 7^k - 6 \cdot 4^k,$$

for some constants  $c_1$ . We substitute

$$a_k = T(2^k) = T(n)$$

and  $k = \lg n$  in this equation to obtain

$$T(n) = c_1 7^{\lg n} - 6 \cdot 4^{\lg n},$$

for some constants  $c_1$  and  $c_2$ . Using the initial conditions, we can show that  $c_1 = 6$ . Thus,

$$T(n) = 6 \cdot 7^{\lg n} - 6 \cdot 4^{\lg n}.$$

We can verify that this is a solution of (9.12), when  $n$  is power of 2. Hence, when  $n$  is a power of 2,

$$T(n) = 6 \cdot 7^{\lg n} - 6 \cdot 4^{\lg n} \approx 6 \cdot n^{2.81} - 6n^2 = \Theta(n^{2.81}).$$

Now if  $f(n) = n^{2.81}$  for all  $n \geq 1$ , then  $f(n)$  is smooth. We can show that the complexity function  $T(n)$  is eventually nondecreasing. Hence,  $T(n) = \Theta(n^{2.81})$  by Theorem 9.1.30.

## Matrix Chain Multiplication

Let  $A$  and  $B$  be matrices of real numbers. We say that matrix  $A$  is **compatible** with matrix  $B$  if the number of columns in  $A$  is the same as the number of rows in  $B$ . Let  $A = [a_{ij}]$  be a matrix of size  $m \times n$  and  $B = [b_{jk}]$  be a matrix of size  $n \times p$ , where  $a_{ij}$  and  $b_{jk}$  are real numbers for all  $i, j, k$ ,  $1 \leq i \leq m$ ,  $1 \leq j \leq n$ ,  $1 \leq k \leq p$ . The multiplication of  $A$  and  $B$ , denoted by  $AB$ , is the  $m \times p$  matrix  $AB = [c_{ik}]$ , where  $c_{ik}$  is given by the following formula:

$$c_{ik} = \sum_{j=1}^n a_{ij} b_{jk}.$$

Notice that to define the product  $AB$ , matrix  $A$  must be compatible with matrix  $B$ .

Recall from the preceding section that the multiplication of element  $a_{ij}$  of  $A$  and element  $b_{jk}$  of  $B$  is the scalar multiplication.

### EXAMPLE 9.2.8

Suppose that

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 6 & -2 & 0 \\ 9 & 1 & 3 \\ -1 & 4 & 5 \\ -3 & 0 & 7 \end{bmatrix}.$$

Because  $A$  is a  $2 \times 4$  matrix and  $B$  is a  $4 \times 3$  matrix,  $AB$  is a  $2 \times 3$  matrix, given by:

$$\begin{aligned} AB &= \begin{bmatrix} 1 \cdot 6 + 2 \cdot 9 + 3 \cdot (-1) + 4 \cdot (-3) & 1 \cdot (-2) + 2 \cdot 1 + 3 \cdot 4 + 4 \cdot 0 & 1 \cdot 0 + 2 \cdot 3 + 3 \cdot 5 + 4 \cdot 7 \\ 5 \cdot 6 + 6 \cdot 9 + 7 \cdot (-1) + 8 \cdot (-3) & 5 \cdot (-2) + 6 \cdot 1 + 7 \cdot 4 + 8 \cdot 0 & 5 \cdot 0 + 6 \cdot 3 + 7 \cdot 5 + 8 \cdot 7 \end{bmatrix} \\ &= \begin{bmatrix} 9 & 12 & 49 \\ 53 & 24 & 109 \end{bmatrix}. \end{aligned}$$

Notice that to compute an element of  $AB$ , 4 scalar multiplications are required. For example, the element  $c_{11}$  at position  $(1, 1)$  is:

$$c_{11} = 1 \cdot 6 + 2 \cdot 9 + 3 \cdot (-1) + 4 \cdot (-3).$$

Because there are 6 entries in  $AB$ , the number of scalar multiplications required to compute  $AB$  is  $4 \cdot 6 = 24$ .

**Theorem 9.2.9:** Let  $A$ ,  $B$ , and  $C$  be matrices of sizes  $m \times n$ ,  $n \times p$ , and  $p \times q$ , respectively. Then:

- (i) The number of scalar multiplications required to compute  $A(BC)$  is  $npq + mnq$ .
- (ii) The number of scalar multiplications required to compute  $(AB)C$  is  $mnp + mpq$ .

### EXAMPLE 9.2.10

Suppose that  $A$  is a matrix of size  $30 \times 5$ ,  $B$  is a matrix of size  $5 \times 40$ ,  $C$  is a matrix of size  $40 \times 7$ , and  $D$  is a matrix of size  $7 \times 10$ . Suppose that we want to compute  $A(B(CD))$ . Because matrix multiplication is associative,

$$A(B(CD)) = (AB)(CD) = A((BC)D) = ((AB)C)D = (A(BC))D.$$

This implies that matrices  $A$ ,  $B$ ,  $C$ , and  $D$  can be multiplied in any of the following five ways:  $A(B(CD))$ ,  $(AB)(CD)$ ,  $A((BC)D)$ ,  $((AB)C)D$ , or  $(A(BC))D$ . Next we determine the number of scalar multiplications required to compute each of these matrix multiplications.

Consider the multiplication  $A(B(CD))$ . The number of scalar multiplications required to compute  $CD$  is  $40 \cdot 7 \cdot 10 = 2800$ . Now  $B$  is a matrix of size  $5 \times 40$  and  $CD$  is a matrix of size  $40 \times 10$ . Therefore, the number of scalar multiplications required to compute  $B(CD)$  is  $5 \cdot 40 \cdot 10 = 2000$ . Similarly, because  $A$  is a matrix of size  $30 \times 5$  and  $B(CD)$  is a matrix of size  $5 \times 10$ , the number of scalar multiplications required to compute  $A(B(CD))$  is  $30 \cdot 5 \cdot 10 = 1500$ . It now follows that the total number of scalar multiplications required to multiply matrices  $A$ ,  $B$ ,  $C$ , and  $D$  in the order  $A(B(CD))$  is  $2800 + 2000 + 1500 = 6300$ .

In a similar manner, we can determine the total number of scalar multiplications, as shown by the following table, required to compute  $(AB)(CD)$ ,  $A((BC)D)$ ,  $((AB)C)D$ , or  $(A(BC))D$ .

| Expression | Number of Multiplications                                                  |
|------------|----------------------------------------------------------------------------|
| $A(B(CD))$ | $40 \cdot 7 \cdot 10 + 5 \cdot 40 \cdot 10 + 30 \cdot 5 \cdot 10 = 6300$   |
| $(AB)(CD)$ | $30 \cdot 5 \cdot 40 + 40 \cdot 7 \cdot 10 + 30 \cdot 40 \cdot 10 = 20800$ |
| $A((BC)D)$ | $5 \cdot 40 \cdot 7 + 5 \cdot 7 \cdot 10 + 30 \cdot 5 \cdot 10 = 3250$     |
| $((AB)C)D$ | $30 \cdot 5 \cdot 40 + 30 \cdot 40 \cdot 7 + 30 \cdot 7 \cdot 10 = 16500$  |
| $(A(BC))D$ | $5 \cdot 40 \cdot 7 + 30 \cdot 5 \cdot 7 + 30 \cdot 7 \cdot 10 = 4550$     |

We can see from the table that the optimal way to multiply matrices  $A$ ,  $B$ ,  $C$ , and  $D$  is to use the expression  $A((BC)D)$ .

**EXAMPLE 9.2.11**

Suppose that  $A$  is a matrix of size  $5 \times 30$ ,  $B$  is a matrix of size  $30 \times 40$ ,  $C$  is a matrix of size  $40 \times 7$ , and  $D$  is a matrix of size  $7 \times 10$ . Suppose that we want to compute  $A(B(CD))$ . As in the previous example, the total number of scalar multiplications required to compute  $(AB)(CD)$ ,  $A((BC)D)$ ,  $((AB)C)D$ , and  $(A(BC))D$  are given in the following table.

| Expression | Number of Multiplications                                                  |
|------------|----------------------------------------------------------------------------|
| $A(B(CD))$ | $40 \cdot 7 \cdot 10 + 30 \cdot 40 \cdot 10 + 5 \cdot 30 \cdot 10 = 16300$ |
| $(AB)(CD)$ | $5 \cdot 30 \cdot 40 + 40 \cdot 7 \cdot 10 + 5 \cdot 40 \cdot 10 = 10800$  |
| $A((BC)D)$ | $30 \cdot 40 \cdot 7 + 30 \cdot 7 \cdot 10 + 5 \cdot 30 \cdot 10 = 12000$  |
| $((AB)C)D$ | $5 \cdot 30 \cdot 40 + 5 \cdot 40 \cdot 7 + 5 \cdot 7 \cdot 10 = 7750$     |
| $(A(BC))D$ | $30 \cdot 40 \cdot 7 + 5 \cdot 30 \cdot 7 + 5 \cdot 7 \cdot 10 = 9800$     |

The table shows that the optimal solution to multiplying matrices  $A$ ,  $B$ ,  $C$ , and  $D$  is to use the expression  $((AB)C)D$ .

Let  $A_1, A_2, \dots, A_n$  be matrices such that matrix  $A_i$  is of the size  $s_{i-1} \times s_i$  for all  $i = 1, 2, \dots, n$ . That is,  $A_1$  is of the size  $s_0 \times s_1$ ,  $A_2$  is of the size  $s_1 \times s_2$ , and so on. It follows that matrix  $A_1$  is compatible with  $A_2$ ,  $A_2$  is compatible with  $A_3$ , and so on. Thus we can form the multiplication:

$$A_1 A_2 \cdots A_n.$$

Because matrix multiplication is associative, this expression can be evaluated in various ways. Our objective is to evaluate it such that the number of scalar multiplications is optimal. To accomplish this, we need to parenthesize the multiplication so that the number of scalar multiplications is optimal.

One way to do this is to look at all possible combinations and then choose the optimal one.

Let  $t_n$  denote the number of ways the expression  $A_1 A_2 \cdots A_n$  can be parenthesized. It can be shown that if we look at each way of parenthesizing the expression  $A_1 A_2 \cdots A_n$  and then choose the expression that gives the optimal number of scalar multiplications, then  $t_n$  is at least exponential. (See Worked-Out Exercise 3, at the end of this section, p. 598.)

Next, we show how dynamic programming can be used to design an algorithm, which is of the order  $O(n^3)$ , to parenthesize the expression  $A_1 A_2 \cdots A_n$ .

Let  $M[1 \dots n, 1 \dots n]$  be a two-dimensional array of  $n$  rows and  $n$  columns. Suppose that for each  $i$  and  $j$ ,  $1 \leq i \leq j \leq n$ ,

$$M[i, j] = \begin{aligned} &\text{minimum number of scalar multiplications needed} \\ &\text{to multiply } A_i \cdots A_j, \quad \text{if } i < j. \end{aligned}$$

$$M[i, i] = 0.$$

Consider:

$$\begin{aligned} M[i, j] &= \min\{M[i, k] + M[k+1, j] + s_{i-1}s_k s_j \mid i \leq k < j\}, \quad \text{if } i < j, \\ M[i, i] &= 0. \end{aligned} \tag{9.13}$$

It can be shown that  $M[i, j]$ , as defined in (9.13), gives the minimum number of scalar multiplications required to multiply  $A_i \cdots A_j$ , if  $i < j$ .

Next we illustrate how the entries,  $M[i, j]$ , of the two-dimensional array  $M$  are calculated.

**EXAMPLE 9.2.12**

Suppose that  $n = 5$ . Consider the following matrices:

$A_1$  is of the size  $10 \times 5$

$A_2$  is of the size  $5 \times 8$

$A_3$  is of the size  $8 \times 15$

$A_4$  is of the size  $15 \times 20$

$A_5$  is of the size  $20 \times 6$

It follows that  $A_1 A_2 A_3 A_4 A_5$  is of the size  $10 \times 6$ .

$$M = \begin{bmatrix} 0 & & & & \\ & 0 & & & \\ & & 0 & & \\ & & & 0 & \\ & & & & 0 \end{bmatrix}, \quad s \begin{bmatrix} [0] & [1] & [2] & [3] & [4] & [5] \\ \hline 10 & 5 & 8 & 15 & 20 & 6 \end{bmatrix}$$

Compute diagonal 1:  $M[1, 2], M[2, 3], M[3, 4], M[4, 5]$ .

$$\begin{aligned} M[1, 2] &= \min\{M[1, k] + M[k+1, 2] + s[0]s[k]s[2] \mid 1 \leq k < 2\} \\ &= M[1, 1] + M[2, 2] + s[0]s[1]s[2] \\ &= 0 + 0 + 10 \cdot 5 \cdot 8 \\ &= 400. \end{aligned}$$

$$\begin{aligned} M[2, 3] &= \min\{M[2, k] + M[k+1, 3] + s[1]s[k]s[2] \mid 2 \leq k < 3\} \\ &= M[2, 2] + M[3, 3] + s[1]s[2]s[3] \\ &= 0 + 0 + 5 \cdot 8 \cdot 15 \\ &= 600. \end{aligned}$$

$$\begin{aligned} M[3, 4] &= \min\{M[3, k] + M[k+1, 4] + s[2]s[k]s[4] \mid 3 \leq k < 4\} \\ &= M[3, 3] + M[4, 4] + s[2]s[3]s[4] \\ &= 0 + 0 + 8 \cdot 15 \cdot 20 \\ &= 2400. \end{aligned}$$

$$\begin{aligned} M[4, 5] &= \min\{M[4, k] + M[k+1, 5] + s[3]s[k]s[5] \mid 4 \leq k < 5\} \\ &= M[4, 4] + M[5, 5] + s[3]s[4]s[5] \\ &= 0 + 0 + 15 \cdot 20 \cdot 6 \\ &= 1800. \end{aligned}$$

After computing entries of the first diagonal:

$$M = \begin{bmatrix} 0 & 400 & & & & \\ & 0 & 600 & & & \\ & & 0 & 2400 & & \\ & & & 0 & 1800 & \\ & & & & 0 & \end{bmatrix}, \quad s \begin{bmatrix} [0] & [1] & [2] & [3] & [4] & [5] \\ \hline 10 & 5 & 8 & 15 & 20 & 6 \end{bmatrix}$$

Compute diagonal 2:  $M[1, 3], M[2, 4], M[3, 5]$ .

$$\begin{aligned} M[1, 3] &= \min\{M[1, k] + M[k+1, 3] + s[0]s[k]s[3] \mid 1 \leq k < 3\} \\ &= \min\{M[1, 1] + M[2, 3] + s[0]s[1]s[3], \\ &\quad M[1, 2] + M[3, 3] + s[0]s[2]s[3]\} \\ &= \min\{0 + 600 + 10 \cdot 5 \cdot 15, 400 + 0 + 10 \cdot 8 \cdot 15\} \\ &= \min\{1350, 1600\} \\ &= 1350. \end{aligned}$$

$$\begin{aligned}
M[2, 4] &= \min\{M[2, k] + M[k+1, 4] + s[1]s[k]s[4] \mid 2 \leq k < 4\} \\
&= \min\{M[2, 2] + M[3, 4] + s[1]s[2]s[4], \\
&\quad M[2, 3] + M[4, 4] + s[1]s[3]s[4]\} \\
&= \min\{0 + 2400 + 5 \cdot 8 \cdot 20, 600 + 0 + 5 \cdot 15 \cdot 20\} \\
&= \min\{3200, 2100\} \\
&= 2100.
\end{aligned}$$

$$\begin{aligned}
M[3, 5] &= \min\{M[3, k] + M[k+1, 5] + s[2]s[k]s[5] \mid 3 \leq k < 5\} \\
&= \min\{M[3, 3] + M[4, 5] + s[2]s[3]s[5], \\
&\quad M[3, 4] + M[5, 5] + s[2]s[4]s[5]\} \\
&= \min\{0 + 1800 + 8 \cdot 15 \cdot 6, 2400 + 0 + 8 \cdot 20 \cdot 6\} \\
&= \min\{2520, 3360\} \\
&= 2520.
\end{aligned}$$

After computing entries of the second diagonal:

$$M = \begin{bmatrix} 0 & 400 & 1350 & & & \\ & 0 & 600 & 2100 & & \\ & & 0 & 2400 & 2520 & \\ & & & 0 & 1800 & \\ & & & & 0 & \end{bmatrix}, \quad s \begin{bmatrix} [0] & [1] & [2] & [3] & [4] & [5] \\ \boxed{10} & 5 & 8 & 15 & 20 & 6 \end{bmatrix}$$

Compute diagonal 3:  $M[1, 4], M[2, 5]$ .

$$\begin{aligned}
M[1, 4] &= \min\{M[1, k] + M[k+1, 4] + s[0]s[k]s[4] \mid 1 \leq k < 4\} \\
&= \min\{M[1, 1] + M[2, 4] + s[0]s[1]s[4], \\
&\quad M[1, 2] + M[3, 4] + s[0]s[2]s[4], \\
&\quad M[1, 3] + M[4, 4] + s[0]s[3]s[4]\} \\
&= \min\{0 + 2100 + 10 \cdot 5 \cdot 20, 400 + 2400 + 10 \cdot 8 \cdot 20, \\
&\quad 1350 + 0 + 10 \cdot 15 \cdot 20\} \\
&= \min\{3100, 4400, 4350\} \\
&= 3100.
\end{aligned}$$

$$\begin{aligned}
M[2, 5] &= \min\{M[2, k] + M[k+1, 5] + s[1]s[k]s[5] \mid 2 \leq k < 5\} \\
&= \min\{M[2, 2] + M[3, 5] + s[1]s[2]s[5], \\
&\quad M[2, 3] + M[4, 5] + s[1]s[3]s[5], \\
&\quad M[2, 4] + M[5, 5] + s[1]s[4]s[5]\} \\
&= \min\{0 + 2520 + 5 \cdot 8 \cdot 6, 600 + 1800 + 5 \cdot 15 \cdot 6, \\
&\quad 2100 + 0 + 5 \cdot 20 \cdot 6\} \\
&= \min\{2760, 2850, 2700\} \\
&= 2700.
\end{aligned}$$

After computing entries of the fourth diagonal:

$$M = \begin{bmatrix} 0 & 400 & 1350 & 3100 & & \\ & 0 & 600 & 2100 & 2700 & \\ & & 0 & 2400 & 2520 & \\ & & & 0 & 1800 & \\ & & & & 0 & \end{bmatrix}, \quad s \begin{bmatrix} [0] & [1] & [2] & [3] & [4] & [5] \\ \boxed{10} & 5 & 8 & 15 & 20 & 6 \end{bmatrix}$$

Compute diagonal 4:  $M[1, 5]$ .

$$\begin{aligned}
 M[1, 5] &= \min\{M[1, k] + M[k+1, 5] + s[0]s[k]s[5] \mid 1 \leq k < 5\} \\
 &= \min\{M[1, 1] + M[2, 5] + s[0]s[1]s[5], \\
 &\quad M[1, 2] + M[3, 5] + s[0]s[2]s[5], \\
 &\quad M[1, 3] + M[4, 5] + s[0]s[3]s[5], \\
 &\quad M[1, 4] + M[5, 5] + s[0]s[4]s[5]\} \\
 &= \min\{0 + 2700 + 10 \cdot 5 \cdot 6, 400 + 2520 + 10 \cdot 8 \cdot 6, \\
 &\quad 1350 + 1800 + 10 \cdot 15 \cdot 6, 3100 + 0 + 10 \cdot 20 \cdot 6\} \\
 &= \min\{3000, 3400, 4050, 4300\} \\
 &= 3000.
 \end{aligned}$$

After computing entries of the fourth diagonal:

$$M = \begin{bmatrix} 0 & 400 & 1350 & 3100 & 3000 \\ 0 & 600 & 2100 & 2700 & \\ 0 & 2400 & 2520 & & \\ 0 & 1800 & & & \\ 0 & & & & \end{bmatrix}, \quad s = \begin{bmatrix} [0] & [1] & [2] & [3] & [4] & [5] \\ 10 & 5 & 8 & 15 & 20 & 6 \end{bmatrix}$$

Now  $M[1, 5] = 3000$ . Hence, the optimal number of scalar multiplications is 3000.

### ALGORITHM 9.12: Chained Matrix Multiplication.

*Input:*  $n$ —the number of matrices  
 $s[0 \dots n]$ , where  $s[i-1] \times s[i]$  specifies the size of matrix  $A_i$ ,  
 $1 \leq i \leq n$ .

*Output:* Minimum number of scalar multiplications and the two-dimensional array  $P[1 \dots n, 1 \dots n]$ . The array  $P$  is used to obtain the optimal order to multiply matrices  $A_1, A_2, \dots, A_n$ .

```

1. function chainedMatrixMultiplication(P, s, n)
2. begin
3.   for i := 1 to n do
4.     M[i, i] := 0;
5.   for d := 1 to n - 1 do
6.     for i := 1 to n - d do
7.       begin
8.         j := i + d;
9.         min := i;
10.        M[i, j] := infinity;
11.        for k = i to j - 1 do
12.          begin
13.            p := M[i, k] + M[k+1, j] + s[i-1] * s[k] * s[j];

```

```

14.      if p < M[i, j] then
15.          begin
16.              M[i, j] := p;
17.              min := k;
18.          end
19.      end
20.      P[i, j] := min;
21.  end
22. return M[1,n]; //Optimal number of scalar multiplications
23. end

```

## Function chainedMatrixMultiplication Analysis

Let  $T(n)$  denote the time complexity of the function `chainedMatrixMultiplication`. The function `chainedMatrixMultiplication` consists of three nested for loops. Let us count the number of iterations of these loops. The number of statements executed inside the innermost loop depends on whether the body of the `if` statement executes or not. However, regardless of whether the body of the `if` statements executes, the number of statements executed is finite. Let  $a$  be the maximum number of statements executed in the innermost loop. Now

$$\begin{aligned}
\sum_{d=1}^{n-1} \sum_{i=1}^{n-d} \sum_{k=i}^{j-1} a &= \sum_{d=1}^{n-1} \sum_{i=1}^{n-d} \sum_{k=i}^{(d+i)-1} a \\
&= \sum_{d=1}^{n-1} \sum_{i=1}^{n-d} a((d+i)-1 - i + 1) \\
&= \sum_{d=1}^{n-1} \sum_{i=1}^{n-d} ad \\
&= \sum_{d=1}^{n-1} ad(n-d) \\
&= a \left( \sum_{d=1}^{n-1} d(n-d) \right) \\
&= a \frac{n(n-1)(n+1)}{6} \quad \text{by Exercise 7, page 563.}
\end{aligned}$$

It now follows that

$$T(n) = \Theta(n^3).$$

Algorithm 9.12 gives the optimal number of scalar multiplications, but it does not give the order in which the matrices should be multiplied. We can use the two-dimensional array  $P$  to find that information.

Suppose that  $P[i, j] = k$ , where  $i < j$ . Then  $i \leq k \leq j$ , and it follows from the function `chainedMatrixMultiplication` that the optimal order for multiplying matrices  $A_i, \dots, A_j$  is

$$(A_i \cdots A_k)(A_{k+1} \cdots A_j).$$

That is, first multiply  $A_i, \dots, A_k$  and  $A_{k+1}, \dots, A_j$ , then multiply the result. For the matrices in Example 9.2.12, the two-dimensional array  $P$  is:

$$P = \begin{bmatrix} 1 & 1 & 1 & 1 \\ & 2 & 3 & 4 \\ & & 3 & 3 \\ & & & 4 \end{bmatrix}$$

Now  $P[3, 5] = 3$ . So the optimal order for multiplying  $A_3, A_4$ , and  $A_5$  is

$$A_3(A_4A_5).$$

Similarly, because  $P[2, 5] = 4$ , the optimal order for multiplying  $A_2, \dots, A_5$  is

$$(A_2A_3A_4)A_5.$$

The multiplication  $A_2A_3A_4$ , inside the parentheses, is obtained according to the optimal order. In other words, because  $P[2, 5] = 4$ , the multiplication  $A_2A_3A_4A_5$  is split at 4.

To find the optimal order for multiplying matrices  $A_1, \dots, A_n$ , we start with  $P[1, n]$  to obtain the top level factorization. We then find the factorization for each factor and continue the process until the complete factorization is obtained. We demonstrate this process for the matrices in Example 9.2.12.

We start with  $P[1, 5]$ . Because  $P[1, 5] = 1$ , we obtain the factorization

$$A_1(A_2A_3A_4A_5).$$

Next, we determine the optimal order to determine  $A_2A_3A_4A_5$ . Now  $P[2, 5] = 4$ , so we obtain the factorization

$$A_1((A_2A_3A_4)A_5).$$

This requires us to determine the optimal order to multiply  $A_2, A_3$ , and  $A_4$ . Now  $P[2, 4] = 3$ , so we obtain the factorization

$$A_1(((A_2A_3)A_4)A_5).$$

This gives the optimal order for multiplying  $A_1, A_2, A_3, A_4$ , and  $A_5$ .

Given the two-dimensional array  $P$  and the starting and ending indices,  $i$  and  $j$ , of the chain of matrices, the following recursive function `printOptimalOrder` outputs the optimal parenthesization of matrices  $A_i, \dots, A_j$ .

### ALGORITHM 9.13: Print Optimal Order.

*Input:* The matrix  $P$  and the indices  $i$  and  $j$

*Output:* Parenthetical expression specifying the order in which to multiply the matrices

1. **procedure** `printOptimalOrder`( $P, i, j$ )
2. **begin**
3.   **if**  $i = j$  **then**
4.     **print** "A",  $i$ ;
5.   **else**
6.     **begin**

```

7.      k := P[i, j];
8.      print "(";
9.      printOptimalOrder(P, i, k);
10.     printOptimalOrder(P, k + 1, j);
11.     print ")";
12.   end
13. end

```

To obtain an optimal parenthesization of matrices  $A_1, \dots, A_n$ , a call to the function **printOptimalOrder** is:

```
printOptimalOrder(P, 1, n);
```

It can be shown that the procedure **printOptimalOrder** is of the order  $O(n)$ . We leave the details as an exercise.

## WORKED-OUT EXERCISES

**Exercise 1:** Let  $n$  be a positive integer and  $r$  be an integer such that  $0 \leq r \leq n$ . Recall that  $C(n, r)$  or  $\binom{n}{r}$  denotes the  $r$ -combinations of  $n$  objects. (In this exercise, we use the notation  $\binom{n}{r}$  to denote  $r$ -combinations.) In Chapter 7, we gave an algorithm that uses the divide-and-conquer technique to determine  $\binom{n}{r}$ . Let  $T(n, r)$  denote the number of terms computed to determine  $\binom{n}{r}$ . Show that  $T(n, r) = 2\binom{n}{r} - 1$ .

**Solution:** We prove the result by induction on  $n$ .

*Basis step:* Suppose  $n = 1$ . Then  $r = 0$  or  $1$ . In this case, we can directly compute  $\binom{n}{r}$ , so only one term is needed to compute  $\binom{n}{r}$ . Thus,  $T(n, r) = 1$ . Also,  $\binom{1}{1} = 1$  and  $\binom{1}{0} = 1$ . Thus,  $2\binom{n}{r} - 1 = 2 \cdot 1 - 1 = 2 - 1 = 1 = T(n, r)$ . Therefore, the result is true for  $n = 1$ .

*Inductive hypothesis:* Suppose  $T(k, r) = 2\binom{k}{r} - 1$  for some positive integer  $n = k$ , and for all  $r$ ,  $0 \leq r \leq k$ .

*Inductive step:* Let  $n = k + 1$  and  $0 \leq r \leq k + 1$ .

Suppose  $r = 0$  or  $r = k + 1$ . Then we can directly determine  $\binom{k+1}{r}$ . That is,  $T(k + 1, r) = 1$ . Also in this case,  $\binom{k+1}{r} = 1$ . Thus,  $2\binom{k+1}{r} - 1 = 2 \cdot 1 - 1 = 2 - 1 = 1 = T(k + 1, r)$ .

Suppose  $0 < r < k + 1$ . By Theorem 7.6.7,

$$\binom{k+1}{r} = \binom{k}{r-1} + \binom{k}{r}.$$

This implies that to determine  $\binom{k+1}{r}$ , we first determine  $\binom{k}{r-1}$  and  $\binom{k}{r}$ . One more operation is needed to add  $\binom{k}{r-1}$  and  $\binom{k}{r}$ . It follows that

$$T(k + 1, r) = T(k, r - 1) + T(k, r) + 1.$$

By the induction hypothesis,  $T(k, r - 1) = 2\binom{k}{r-1} - 1$

and  $T(k, r) = 2\binom{k}{r} - 1$ . Therefore,

$$\begin{aligned} T(k + 1, r) &= T(k, r - 1) + T(k, r) + 1 \\ &= \left(2\binom{k}{r-1} - 1\right) + \left(2\binom{k}{r} - 1\right) + 1 \\ &= 2\binom{k}{r-1} - 1 + 2\binom{k}{r} - 1 + 1 \\ &= 2\left\{\binom{k}{r-1} + \binom{k}{r}\right\} - 1 \\ &= 2\binom{k+1}{r} - 1. \end{aligned}$$

This implies that the result is true for  $n = k + 1$ . Hence, by induction the result is true.

**Exercise 2: Horner's method for evaluating a polynomial at a given value.** In Chapter 1, we described an algorithm to evaluate a polynomial at a given value. In this exercise, we describe Horner's method to accomplish the same thing. Let  $p(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$  be a polynomial of degree  $n$ . Horner's method of evaluating  $p(x)$  at a given value is based on the following observation: The polynomial  $p(x)$  can be written as

$$p(x) = a_0 + (a_1 + (a_2 + (a_3 + \dots + (a_{n-1} + a_nx)x)\dots)x)x.$$

For example, if  $p(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4$ , then

$$p(x) = a_0 + (a_1 + (a_2 + (a_3 + a_4x)x)x)x.$$

Using Horner's method, design an algorithm to evaluate a polynomial at a given value. What is the order of the algorithm?

**Solution:** In Horner's method, first  $a_n$  is multiplied by  $x$  and then  $a_{n-1}$  is added to the result. Thus, after these operations, we have computed  $a_{n-1} + a_n x$ . Next, this result is multiplied by  $x$  and  $a_{n-2}$  is added to the result. So at this point, we have computed  $a_{n-2} + (a_{n-1} + a_n x)x = a_{n-2} + a_{n-1}x + a_n x^2$ . This process is continued. The following loop implements this algorithm. (We assume that the coefficients of the polynomial  $p$  are stored in the array  $A[0 \dots n]$ .)

```
value := A[n];
for i := 0 to n - 1 do
    value = value * x + A[n - i - 1];
```

Each time through the loop, there is one multiplication, one addition, and one assignment; a total of three operations. Because the **for** loop executes  $n$  times, it follows that this algorithm is of the order  $\Theta(n)$ .

**Exercise 3:** Let  $A_1, A_2, \dots, A_n$  be matrices such that  $A_i$  is compatible with  $A_{i+1}$ ,  $i = 1, 2, \dots, n-1$ . Let  $t_n$  denote the number of ways the expression  $A_1 A_2 \cdots A_n$  can be parenthesized. Show that if we look at each way to parenthesize the expression  $A_1 A_2 \cdots A_n$  and then choose the expression that gives the optimal number of scalar multiplications, then  $t_n$  is at least exponential.

**Solution:** Let  $t_n$  denote the number of ways the expression  $A_1 A_2 \cdots A_n$  can be parenthesized.

Consider the expression  $A_1(A_2 \cdots A_n)$ . Here, we first multiply  $A_2 \cdots A_n$  and then multiply  $A_1$  by the result. Because  $A_2 \cdots A_n$  is a multiplication of  $n-1$  matrices, the number of ways the expression  $A_2 \cdots A_n$  can be parenthesized is  $t_{n-1}$ . This implies that the number of ways the expression  $A_1(A_2 \cdots A_n)$  can be parenthesized is  $t_{n-1}$ . Moreover, the

number of ways the expression  $A_1(A_2 \cdots A_n)$  can be parenthesized is a subset of the number of ways  $A_1 A_2 \cdots A_n$  can be parenthesized.

Now consider the expression  $(A_1 A_2 \cdots A_{n-1}) A_n$ . Here, we first multiply  $A_1 A_2 \cdots A_{n-1}$  and then multiply the result by  $A_n$ . Because  $A_1 A_2 \cdots A_{n-1}$  is a multiplication of  $A_1 A_2 \cdots A_{n-1}$  matrices, the number of ways the expression  $A_1 A_2 \cdots A_{n-1}$  can be parenthesized is  $t_{n-1}$ . This implies that the number of ways the expression  $(A_1 A_2 \cdots A_{n-1}) A_n$  can be parenthesized is  $t_{n-1}$ . Moreover, the number of ways the expression  $(A_1 A_2 \cdots A_{n-1}) A_n$  can be parenthesized is a subset of the number of ways  $A_1 A_2 \cdots A_n$  can be parenthesized.

It now follows that

$$t_n \geq t_{n-1} + t_{n-1} = 2t_{n-1}.$$

Because there is only one way to multiply  $A_1$  and  $A_2$ , it follows that

$$t_2 = 1.$$

Thus, we have the following recurrence relations:

$$t_n \geq 2t_{n-1}$$

with the initial condition  $t_2 = 1$ .

Solving these recurrence relations, it can be shown that

$$t_n \geq 2^n.$$

This shows that if we look at each way to parenthesize the expression  $A_1 A_2 \cdots A_n$  and then choose the expression that gives the optimal number of scalar multiplications, then  $t_n$  is at least exponential.

## SECTION REVIEW

### Key Terms

sequential search  
binary search  
selection sort

insertion sort  
merge sort  
Strassens's algorithm

compatible matrices

### Some Key Results

1. Sequential search is  $\Theta(n)$ .
2. Binary search is  $\Theta(\lg n)$ .
3. Selection sort is  $\Theta(n^2)$ .
4. Insertion sort is  $\Theta(n^2)$ .
5. Let  $L$  be a list of  $n$  distinct elements. Any comparison-based sort algorithm in its worst case must be of the order  $O(n \lg n)$  to sort  $L$ .
6. Merge Sort is  $\Theta(n \lg n)$ .

7. Strassen's multiplication method is  $\Theta(n^{2.81})$ .
8. The function `chainedMatrixMultiplication` is  $\Theta(n^3)$ .

## EXERCISES

---

1. Consider the following list:

|    |    |    |    |    |    |    |     |
|----|----|----|----|----|----|----|-----|
| 63 | 45 | 32 | 98 | 46 | 57 | 28 | 100 |
|----|----|----|----|----|----|----|-----|

Using the sequential search algorithm, how many comparisons are required to find whether the following items are in the list? (Recall that by comparisons we mean item comparisons, not index comparisons.)

- a. 90      b. 57      c. 63      d. 120

2. Consider the following list:

|   |    |    |    |    |    |    |    |    |    |     |
|---|----|----|----|----|----|----|----|----|----|-----|
| 2 | 10 | 17 | 45 | 49 | 55 | 68 | 85 | 92 | 98 | 110 |
|---|----|----|----|----|----|----|----|----|----|-----|

Using the binary search algorithm as described in this chapter, how many comparisons are required to find whether the following items are in the list? Show the values of first, last, and middle and the number of comparisons after each iteration of the loop.

- a. 15      b. 49      c. 98      d. 99

3. Consider the following list:

|   |    |    |    |    |    |
|---|----|----|----|----|----|
| 5 | 18 | 21 | 10 | 55 | 20 |
|---|----|----|----|----|----|

The first three keys are in order. To move 10 to its proper position using the insertion sort as described in this chapter, exactly how many key comparisons are executed?

4. Consider the following list:

|   |    |    |    |   |    |
|---|----|----|----|---|----|
| 7 | 28 | 31 | 40 | 5 | 20 |
|---|----|----|----|---|----|

The first four keys are in order. To move 5 to its proper position using the insertion sort algorithm as described in this chapter, exactly how many key comparisons are executed?

5. Consider the following list:

|    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|
| 28 | 18 | 21 | 10 | 25 | 30 | 12 | 71 | 32 | 58 | 15 |
|----|----|----|----|----|----|----|----|----|----|----|

This list is to be sorted using the insertion sort algorithm as described in this chapter for array-based lists. Show the resulting list after six passes of the sorting phase, that is, after six iterations of the `for` loop.

6. Recall the insertion sort algorithm as discussed in this chapter. Consider the following list:

|    |   |    |   |    |    |    |    |    |    |    |    |    |   |
|----|---|----|---|----|----|----|----|----|----|----|----|----|---|
| 18 | 8 | 11 | 9 | 15 | 20 | 32 | 61 | 22 | 48 | 75 | 83 | 35 | 3 |
|----|---|----|---|----|----|----|----|----|----|----|----|----|---|

Exactly how many key comparisons are executed to sort this list using the insertion sort algorithm?

7. Consider the following two lists:

|   |   |    |    |    |    |
|---|---|----|----|----|----|
| 5 | 8 | 12 | 20 | 25 | 28 |
|---|---|----|----|----|----|

and

|   |   |    |    |    |    |
|---|---|----|----|----|----|
| 7 | 9 | 13 | 30 | 35 | 48 |
|---|---|----|----|----|----|

These lists are to be merged into a sorted list using the procedure `mergeLists`. Exactly how many key comparisons are executed in order to merge these two lists?

8. Consider the following two lists:

|   |    |    |    |
|---|----|----|----|
| 4 | 10 | 20 | 55 |
|---|----|----|----|

and

|   |   |    |    |    |
|---|---|----|----|----|
| 7 | 9 | 15 | 30 | 35 |
|---|---|----|----|----|

These lists are to be merged into a sorted list, say  $M$ , using the procedure `mergeLists`. Illustrate how the `mergeLists` procedure works. Show the lists after merging each item, as illustrated in the Merge subsection (of the Merge Sort section).

9. Let

$$A = \begin{bmatrix} 1 & -1 \\ -4 & 2 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 2 & 3 \\ 11 & -1 \end{bmatrix}.$$

Multiply  $AB$  using Strassen's method.

10. Let

$$A = \begin{bmatrix} 1 & -1 & 2 & 5 \\ 2 & 1 & 0 & 2 \\ 0 & 3 & -4 & 6 \\ 0 & -3 & 9 & 8 \end{bmatrix}$$

and

$$B = \begin{bmatrix} 4 & 1 & -3 & 0 \\ 3 & 11 & 5 & 3 \\ 0 & -2 & 0 & 2 \\ 5 & 3 & 2 & 0 \end{bmatrix}.$$

Multiply  $AB$  using Strassen's method.

11. Let  $T(n) = 7T(\frac{n}{2}) + 18(\frac{n}{2})^2$ , if  $n > 1$  and  $T(1) = 1$ . Show that the function  $T(n)$  is eventually nondecreasing.
12. Suppose that  $A$  is a matrix of size  $20 \times 15$ ,  $B$  is a matrix of size  $15 \times 30$ ,  $C$  is a matrix of size  $30 \times 27$ , and  $D$  is a matrix of size  $27 \times 10$ . Find the optimal number of scalar multiplications to multiply  $ABCD$ . Also parenthesize the expression  $ABCD$  corresponding to the optimal number of scalar multiplications.

13. Consider the following matrices:  $A_1$  of size  $11 \times 15$ ,  $A_2$  of size  $15 \times 28$ ,  $A_3$  of size  $28 \times 13$ ,  $A_4$  of size  $13 \times 30$ ,  $A_5$  of size  $30 \times 16$ . Using dynamic programming, parenthesize the expression  $A_1 A_2 A_3 A_4 A_5$  so that the number of scalar multiplications is optimal. Show the matrix  $M$  after each iteration, as illustrated in this chapter.
14. Let  $n$  be a positive integer and  $r$  be an integer such that  $0 \leq r \leq n$ . Prove that the algorithm `combDynamicProg` (given in Chapter 7), which uses dynamic programming to determine  $C(n, r)$ , is of the order  $\Theta(nr)$ .
15. Prove that the bubble sort algorithm given in Chapter 1 is of the order  $O(n^2)$ .
16. Let  $L$  be a list of  $n$  elements. The following algorithm uses the divide-and-conquer technique to find the smallest and largest elements in  $L$  simultaneously. The divide-and-conquer Max Min Algorithm follows.

**Algorithm: Max-Min Algorithm 3**

*Input:*  $L$ —a list of  $n$  elements  
*n*—the number of elements in  $L$

*Output:*  $max$ —the maximum element of the list  
 $min$ —the minimum element of the list

```

1. procedure maxMin3( $L, i, j, max, min$ )
2. begin
3.   if  $i = j$  then
4.     begin
5.        $max := L[i]$ ;
6.        $min := L[j]$ ;
7.     end
8.   else
9.     if  $i = j - 1$  then

```

```

10.      if  $L[i] < L[j]$  then
11.        begin
12.           $max := L[j]$ ;
13.           $min := L[i]$ ;
14.        end
15.      else
16.        begin
17.           $max := L[i]$ ;
18.           $min := L[j]$ ;
19.        end
20.      else
21.        begin
22.           $mid := (i + j) / 2$ ;
23.          maxMin3( $L, i, mid, max, min$ );
24.          maxMin3( $L, mid+1, j, tempMax, tempMin$ );
25.          if  $max < tempMax$  then
26.             $max := tempMax$ ;
27.          if  $min > tempMin$  then
28.             $min := tempMin$ ;
29.        end
30.    end

```

Let  $T(n)$  denote the number of comparisons required to find the smallest and largest elements. Show that  $T(n)$  satisfies the following recurrence relation:

$$T(n) = T\left(\left\lceil \frac{n}{2} \right\rceil\right) + T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + 2,$$

with the initial conditions  $T(1) = 0$  and  $T(2) = 1$ . Moreover, show that if  $n$  is a power of 2, then  $T(n) = \frac{3}{2}n - 2$ .

## ► PROGRAMMING EXERCISES

1. Write a program to implement a sequential search algorithm.
2. Write a program to implement a binary search algorithm.
3. Write a program to implement a selection sort algorithm.
4. Write a program to implement an insertion sort algorithm.
5. Write a program to implement a merge sort algorithm.
6. Write a program to implement the various algorithms given in this chapter to determine the largest and smallest elements in a list simultaneously.
7. Write a program to implement the dynamic programming technique, as given in this chapter, to determine the optimal order in which to multiply matrices.

## Graph Theory

**The objectives of this chapter are to:**

- Learn the basic properties of graph theory
- Learn about walks, trails, paths, circuits, and cycles in a graph
- Explore how graphs are represented in computer memory
- Learn about Euler and Hamilton circuits
- Learn about isomorphism of graphs
- Explore various graph algorithms
- Examine planar graphs and graph coloring

In 1736, the following problem was posed: In the town of Königsberg (now called Kaliningrad), the river Pregel (Pregolya) flows around the island of Kneiphof and then divides in two. See Figure 10.1(a).

The river has four land areas ( $A$ ,  $B$ ,  $C$ ,  $D$ ), as shown in Figure 10.1(a). These land areas are connected using seven bridges, also shown in Figure 10.1(a). The bridges are labeled  $a$ ,  $b$ ,  $c$ ,  $d$ ,  $e$ ,  $f$ , and  $g$ . The Königsberg bridge problem is as follows: Starting at one land area, is it possible to walk across all of the bridges exactly once and return to the starting land area? In 1736, Euler represented the Königsberg bridge problem as a graph, as shown in Figure 10.1(b), and answered the question in the negative. This marked (as recorded) the birth of graph theory.

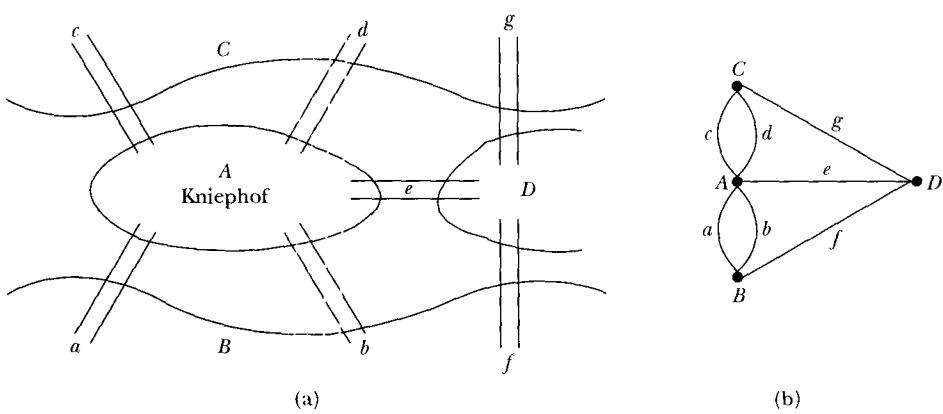


FIGURE 10.1 Königsberg bridge problem

Over the past 200 years, graph theory has been used in a variety of applications. Graphs are used to model electrical circuits, chemical compounds, highway maps, and so on. They are also used in the analysis of electrical circuits, finding the shortest route, project planning, linguistics, genetics, and social science. In this chapter, we discuss graphs and their applications in computer science. The first text on graph theory appeared in 1936.

In 1852, Francis Guthrie, a student of geography, noticed that a map of the counties of England could be colored with only four different colors in such a way that each county had exactly one color and adjacent counties each had different colors. He wondered if four colors would be enough to color the map of a country such that each state has exactly one color and adjacent states each have a different color. He discussed this with his brother, a student of DeMorgan. DeMorgan could not answer the problem and he communicated it to Hamilton. Hamilton thought that a proof might be possible, but he said he had no time to work on it.

In 1878, Cayley gave a seminar on the four-color problem at a meeting of the London Mathematical Society. At last, the problem attracted the attention of



### HISTORICAL NOTES

#### **Leonhard Euler** (1707–1783)

Euler was born the son of a Protestant minister in Basel, Switzerland.

He attended school in Basel, but his father, who had been well educated, was the first to introduce mathematics to him. At the age of 14, Euler entered the University of Basel with plans to follow his father into the church. He discovered, however, that even though he was a religious man, his interest in theology was eclipsed by his love and gift for mathematics. Johann Bernoulli, a lifelong family friend, persuaded Euler's father to allow Leonhard to change his course of study to mathematics.

Upon graduation, Euler accepted a position at the St. Petersburg Academy

where he taught physics as well as mathematics. As a junior professor, Euler was required to serve in the Russian Navy, which he did between 1727 and 1730. Upon discharge, he was awarded the position of full professor. As a member of the Academy, Euler was expected to perform projects for the state, and he worked in areas as diverse as cartography, magnetism, ship-building, and science education. In addition, he completed research on number theory, differential equations, calculus of variations, and rational mechanics.

In 1740, Euler left St. Petersburg to become director of mathematics at the newly formed Academy of Science in Berlin at the invitation of Frederick the Great. In his 25 years there, he performed administrative duties as

well as teaching and research, and by the end of his tenure was directing the Academy. When the office of the president was not offered to him, he returned to St. Petersburg.

By 1771, Euler was completely blind, yet he managed to complete over 50 percent of his life's work without the aid of sight owing to his incredible memory and the help of his sons and other mathematicians. The magnitude of Euler's work can be best appreciated by the fact that after his death in 1783, the Academy continued to publish his unpublished works for the next 50 years.

mathematicians and work on it accelerated the growth of graph theory. In 1976, the problem was finally solved by Kenneth Appel and Wolfgang Haken proving that four colors are sufficient to color the map of a country such that each state has exactly one color and adjacent states each have a different color. The proof required 1000 hours of computer time using fast, large computers. Appel and Haken worked on the problem for 10 years. While we shall not attempt to prove the four-color problem in this chapter, we will show how graph theory can be used to solve various color problems.

## 10.1 GRAPH DEFINITION AND NOTATIONS

In this section, we give various definitions related to graphs and establish some notations. Unfortunately there are a formidable number of definitions that must be presented before we begin. Let us consider some problems that can be solved easily with the help of graph theory.

Consider the following problem related to an old children's game. Using a pencil, can we trace each of the diagrams in Figure 10.2 satisfying the following conditions?

1. The tracing must start at point  $A$  and come back to point  $A$ .
2. While tracing the figure, the pencil cannot be lifted from the figure.
3. A line cannot be traced twice.

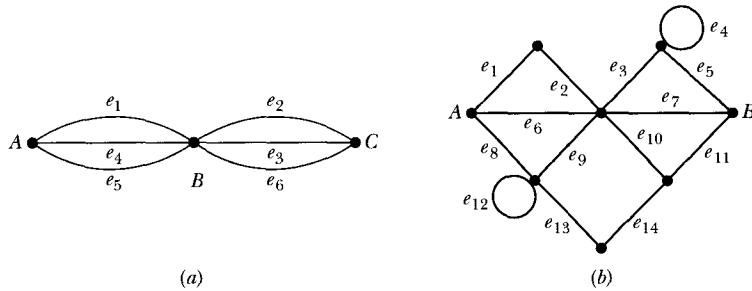


FIGURE 10.2 Various graphs

To answer this problem we will consider each of the diagrams in Figure 10.2 as a pictorial representation of graphs. As in geometry, each of points  $A$ ,  $B$ , and  $C$  is called a *vertex* of the graph and the line joining two vertices is called an *edge*.

Next let us consider the following problems, which we will answer using the concepts developed in this chapter.

- Is it true that in any gathering of  $n > 1$  people, there are at least two people with exactly the same number of friends?
- A party consists of six people. Is it true that it is always possible to find either three people who know each other or three people such that no one knows each other?
- Suppose there are three houses, which are to be connected to three services—water, telephone, and electricity—by means of underground pipelines. The services are to be connected subject to the following condition: The pipes must be laid so that they do not cross each other. Consider three distinct points,  $A$ ,  $B$ ,  $C$ , as three houses and three other distinct points,  $W$ ,

$T$ , and  $E$ , which represent the water source, the telephone connection point, and the electricity connection point. Try to join  $W$ ,  $T$ , and  $E$  with each of  $A$ ,  $B$ , and  $C$  by drawing lines (they may not be straight lines) so that no two lines intersect each other (see Figure 10.3). This is known as the *three utilities problem*.

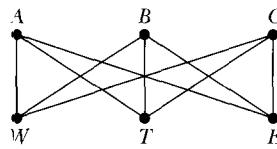


FIGURE 10.3 Three utilities problem

Notice that Figure 10.3 is not a solution of the three utilities problem.

Before we can answer such graph problems, we need to formally define a graph.

**DEFINITION 10.1.1** ▶ A graph  $G$  is a triple  $(V, E, g)$ , where

- (i)  $V$  is a finite nonempty set, called the **set of vertices**;
- (ii)  $E$  is a finite set (may be empty), called the **set of edges**; and
- (iii)  $g$  is a function, called an **incidence function**, that assigns to each edge,  $e \in E$ , a one-element subset  $\{v\}$  or a two-element subset  $\{v, w\}$ , where  $v$  and  $w$  are vertices.

For convenience, we will write  $g(e) = \{v, w\}$ , where  $v$  and  $w$  may be the same.

Let  $G = (V, E, g)$  be a graph. Suppose that  $e$  is an edge of this graph. Then there are vertices  $v$  and  $w$  such that  $g(e) = \{v, w\}$ ; the vertices  $v$  and  $w$  are called the **end vertices**, or **endpoints**, of the edge  $e$ . When a vertex  $v$  is an endpoint of some edge  $e$ , we say that  $e$  is **incident** with the vertex  $v$  and that  $v$  is **incident** with the edge  $e$ . Two vertices  $v$  and  $w$  of  $G$  are said to be **adjacent** if there exists an edge  $e \in E$  such that  $g(e) = \{v, w\}$ . Two edges are said to be **adjacent** if they have a common end vertex. If  $e$  is an edge such that  $g(e) = \{v, w\}$ , where  $v = w$ , then  $e$  is an edge from the vertex  $v$  to itself; such an edge is called a **loop** on the vertex  $v$  or at the vertex  $v$ . If there is a loop on  $v$ , then  $v$  is adjacent to itself.

**REMARK 10.1.2** ▶ Let  $G = (V, E, g)$  be a graph. If no confusion arises, we will write  $G$  as  $(V, E)$ , or simply as  $G$ .

**EXAMPLE 10.1.3**

Let  $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$ ,  $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$ , and  $g$  be defined by

$$\begin{aligned} g(e_1) &= g(e_2) = \{v_1, v_2\} \\ g(e_3) &= \{v_4, v_3\} \\ g(e_4) &= g(e_6) = g(e_7) = \{v_6, v_3\} \\ g(e_5) &= \{v_2, v_4\}. \end{aligned}$$

Then  $G = (V, E, g)$  is a graph.

Let  $G = (V, E, g)$  be a graph. The incidence function,  $g$ , can be defined by a two-row table, called an **incidence table**, whose columns are indexed by the edges. The vertices adjacent to an edge are placed in the second row below the edge.

For example, consider the graph of Example 10.1.3. Now vertices  $v_1$  and  $v_2$  are adjacent to edges  $e_1$  and  $e_2$ , vertices  $v_4$  and  $v_3$  are adjacent to edge  $e_3$ , and so on. The incidence table corresponding to the incidence function  $g$  is:

| edge         | $e_1$      | $e_2$      | $e_3$      | $e_4$      | $e_5$      | $e_6$      | $e_7$      |
|--------------|------------|------------|------------|------------|------------|------------|------------|
| end vertices | $v_1, v_2$ | $v_1, v_2$ | $v_4, v_3$ | $v_6, v_3$ | $v_2, v_4$ | $v_6, v_3$ | $v_6, v_3$ |

The set of vertices and the set of edges of a graph are finite. Therefore, one of the features that makes the study of graph theory interesting and attractive is that a graph can be represented pictorially. In other words, corresponding to a graph we can draw a diagram that helps us visualize the facts. This is possible because both the set of vertices and the set of edges of a graph are finite.

In a pictorial representation of a graph, each vertex is drawn as a large dot or a small circle in a plane, which is labeled by the vertex itself. If two vertices are adjacent, i.e., if there is an edge between two vertices, it is drawn as a line, which may not be straight, connecting the vertices. Moreover, the line is labeled by the edge.

For example, graph  $G$  of Example 10.1.3 may be represented as in Figure 10.4.

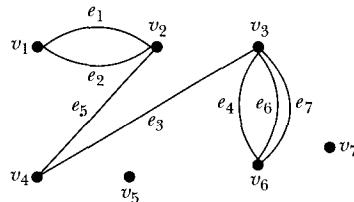


FIGURE 10.4 Graph of Example 10.1.3

The way the diagram of a graph is drawn is not important. The important thing is to show the relationships between the vertices. From this pictorial representation we find that  $v_4, v_2$  are end vertices of edge  $e_5$ , edge  $e_3$  is incident with vertices  $v_4, v_3$ , and  $v_3, v_6$  are adjacent vertices, but  $v_4, v_6$  are not adjacent vertices. Also we see that  $e_3, e_5$  are adjacent edges, but  $e_1, e_6$  are not adjacent edges.

Let  $G = (V, E, g)$  be a graph. The incidence function  $g$  need not be one-one. Therefore, there may exist edges  $e_1, e_2, \dots, e_{n-1}, e_n, n \geq 2$  such that

$$g(e_1) = g(e_2) = \dots = g(e_n) = \{v, w\}.$$

Such edges are called **parallel edges**.

In the graph in Figure 10.4,  $g(e_1) = g(e_2) = \{v_1, v_2\}$ . Thus, the edges  $e_1, e_2$  are parallel edges. Also in this graph,

$$g(e_4) = g(e_6) = g(e_7) = \{v_6, v_3\}.$$

Hence, edges  $e_4, e_6$ , and  $e_7$  are also parallel.

---

**DEFINITION 10.1.4** ▶ Let  $G$  be a graph and  $v$  be a vertex in  $G$ . We say that  $v$  is an **isolated vertex** if it is not incident with any edge.

In the graph of Figure 10.4,  $v_5$  and  $v_7$  are isolated vertices.

#### EXAMPLE 10.1.5

Let  $V = \{v_1, v_2, v_3, v_4\}$ ,  $E = \{e_1, e_2, e_3\}$ , and  $g(e_1) = \{v_1, v_2\}$ ,  $g(e_2) = \{v_3, v_2\}$ ,  $g(e_3) = \{v_3, v_4\}$ . Then  $G = (V, E, g)$  is a graph. This graph is represented by the diagram in Figure 10.5.

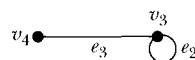
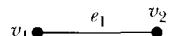


FIGURE 10.5 A graph

Now  $G$  does not contain any parallel edges. However,  $e_2$  is a loop in  $G$ .

In the graph in Figure 10.4 we find that the number of edges incident on  $v_2$  is 3. This number is called the degree of the vertex  $v_2$ . More formally, we have the following definition.

**DEFINITION 10.1.6** ▶ Let  $G$  be a graph and  $v$  be a vertex of  $G$ . The **degree** of  $v$ , written  $\deg(v)$  or  $d(v)$ , is the number of edges incident with  $v$ .

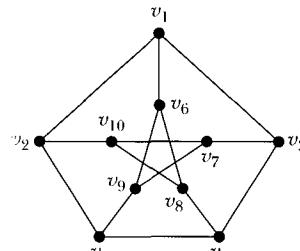
We make the convention that each loop on a vertex  $v$  contributes 2 to the degree of  $v$ . With this convention we find that the degree of  $v_3$  in Figure 10.5 is 3.

**REMARK 10.1.7** ▶ From the definition of the degree of a vertex it follows that a vertex  $v$  is an isolated vertex if and only if  $\deg(v) = 0$ .

There are graphs in which all vertices may have the same degree. These types of graph are called **regular graphs**.

**DEFINITION 10.1.8** ▶ Let  $G$  be a graph and  $k$  be a nonnegative integer.  $G$  is called a  **$k$ -regular graph** if the degree of each vertex of  $G$  is  $k$ .

An interesting  $k$ -regular graph is the *Petersen 3-regular graph*, shown in Figure 10.6.

FIGURE 10.6 Petersen  
3-regular graph

**DEFINITION 10.1.9** ▶ Let  $G$  be a graph and  $v$  be a vertex in  $G$ .  $v$  is called an **even (odd) degree vertex** if the degree of  $v$  is even (odd).

**EXAMPLE 10.1.10**

Consider the graph in Figure 10.7.

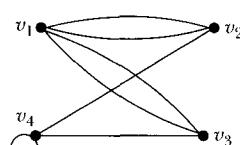
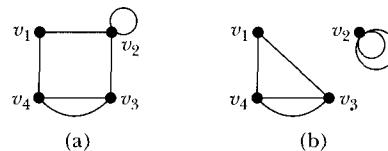


FIGURE 10.7 A graph

For this graph  $\deg(v_1) = 4$ ,  $\deg(v_2) = 3$ ,  $\deg(v_3) = 3$ , and  $\deg(v_4) = 4$ . Here  $v_2$  and  $v_3$  are odd degree vertices, and  $v_1$  and  $v_4$  are even degree vertices.

**DEFINITION 10.1.11** ▶ Let  $n_1, n_2, n_3, \dots, n_k$  be the degrees of vertices of a graph  $G$  such that  $n_1 \leq n_2 \leq n_3 \leq \dots \leq n_k$ . Then the finite sequence  $n_1, n_2, n_3, \dots, n_k$  is called the **degree sequence** of the graph.

Clearly, every graph has a unique degree sequence. However, we can construct completely different graphs having same degree sequence. For example, see the graphs in Figure 10.8. The degree sequence of both of these graphs is 2, 3, 3, 4, but the graphs are different.



**FIGURE 10.8** Different graphs with the same degree sequence

**EXAMPLE 10.1.12** The degree sequence of graph  $G$  in Example 10.1.10 (see Figure 10.7) is 3, 3, 4, 4.

Now consider the graph given in Example 10.1.10 (see Figure 10.7). For this graph

$$\deg(v_1) = 4, \quad \deg(v_2) = 3, \quad \deg(v_3) = 3, \quad \deg(v_4) = 4.$$

The sum of the degrees of all of the vertices is

$$\deg(v_1) + \deg(v_2) + \deg(v_3) + \deg(v_4) = 14,$$

which is an even integer. This is true for any graph, and it is one of the basic properties of a graph. We prove this result in the following theorem, due to Euler.

**Theorem 10.1.13: Euler.** The sum of the degrees of all vertices of a graph is twice the number of edges.

**Proof:** Let  $G$  be a graph with  $n$  edges and  $m$  vertices,  $v_1, v_2, \dots, v_m$ . We want to determine

$$\deg(v_1) + \deg(v_2) + \dots + \deg(v_{m-1}) + \deg(v_m).$$

Now the degree,  $\deg(v_i)$ , of  $v_i$  is the number of edges incident with  $v_i$ . Each edge  $e$  is either a loop or incident with two distinct vertices. If  $e$  is a loop on a vertex  $v$ , then  $e$  contributes 2 to the degree of  $v$ . On the other hand, if  $e$  is incident with two distinct vertices  $v$  and  $w$ , then  $e$  contributes 1 to the degree of each vertex. Thus we find that when we compute the sum  $\deg(v_1) + \deg(v_2) + \dots + \deg(v_{m-1}) + \deg(v_m)$ , each edge contributes 2 in this sum. Because there are  $n$  edges, the total contribution in the above sum is  $2n$ . Hence,

$$\deg(v_1) + \deg(v_2) + \dots + \deg(v_{m-1}) + \deg(v_m) = 2n. \blacksquare$$

**Corollary 10.1.14:** The sum of the degrees of all of the vertices of a graph is an even integer.

**Proof:** Because  $2n$  is an even integer, the corollary follows from Theorem 10.1.13. ■

**Corollary 10.1.15:** In a graph, the number of odd degree vertices is even.

**Proof:** Suppose a graph  $G$  has  $k$  odd degree vertices,  $v_1, v_2, \dots, v_k$ , and  $t$  even degree vertices,  $u_1, u_2, \dots, u_t$ . By Theorem 10.1.13,

$$\deg(v_1) + \deg(v_2) + \cdots + \deg(v_k) + \deg(u_1) + \deg(u_2) + \cdots + \deg(u_t) = 2n,$$

where  $n$  is the number of edges.

Because each  $\deg(u_j)$  is even, it follows that  $\deg(u_1) + \deg(u_2) + \cdots + \deg(u_t)$  is an even integer. Also,  $2n$  is an even integer. Hence,  $\deg(v_1) + \deg(v_2) + \cdots + \deg(v_k)$  must be an even integer. Now the sum of an odd number of odd integers is an odd integer. Because each number  $\deg(v_i)$  is odd and  $\deg(v_1) + \deg(v_2) + \cdots + \deg(v_k)$  is an even integer it follows that the number  $k$  cannot be odd, so  $k$  is even. This completes the proof. ■

From Corollary 10.1.15, it follows that the sum of the members of the degree sequence of a graph is an even integer. We now show that for any finite nondecreasing sequence of nonnegative integers whose sum is an even integer, there exists a graph such that its degree sequence is the given sequence. This follows from the method of construction for such graphs: Suppose  $n_1, n_2, n_3, \dots, n_k$  is a sequence of nonnegative integers such  $n_1 + n_2 + n_3 + \cdots + n_k$  is an even integer. We draw a graph with vertices  $v_1, v_2, v_3, \dots, v_k$  such that  $\deg(v_i) = n_i, i = 1, 2, \dots, k$ .

First we take  $k$  vertices  $v_1, v_2, v_3, \dots, v_k$ . For each  $i$ , if  $n_i$  is even, we draw  $\frac{n_i}{2}$  loops at  $v_i$ ; and if  $n_i$  is odd, we draw  $\frac{n_i-1}{2}$  loops at  $v_i$ . Now in the given sequence  $n_1, n_2, n_3, \dots, n_k$  there must be an even number of odd integers. We identify the vertices that we have drawn for these odd integers and pair these vertices such that no two pairs have a common vertex. We then join each two members of a pair by an edge. Because each loop at a vertex contributes 2 to the degree of that vertex and each edge different from a loop with end vertices  $u$  and  $v$  contributes 1 to the degree of  $u$  and 1 to the degree of  $v$ , it follows that we obtain a graph with vertices  $v_1, v_2, v_3, \dots, v_k$  such that  $\deg(v_i) = n_i, i = 1, 2, \dots, k$ . (See Worked-Out Exercise 3 at the end of this section.)

## Directed Graphs

In Chapter 3, we discussed directed graphs in the context of binary relations. Here we introduce directed graphs in general form.

---

**DEFINITION 10.1.16** ► A **directed graph** (or **digraph**)  $G$  is a triple  $(V, E, g)$ , where

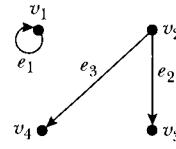
- (i)  $V$  is a *finite nonempty* set of vertices;
- (ii)  $E$  is a finite set (may be empty) of **directed edges**, or **arcs**; and
- (iii)  $g : E \rightarrow V \times V$  is a function that assigns to each arc  $e$  an ordered pair  $(v, w)$ , where  $v$  and  $w$  are vertices ( $v$  and  $w$  may be the same).

As described in Chapter 3, we can draw a diagram corresponding to a directed graph. The vertices are represented as small or large dots or a small circle, which is labeled by the vertex itself. Moreover, we draw an arrow from a vertex  $v$  to a vertex  $w$  if and only if there is an arc  $e$  such that  $g(e) = (v, w)$ .

If  $g(e) = (v, w)$ , then  $v$  is called the **starting vertex** and  $w$  is called the **terminating vertex** of the arc  $e$ . The **in-degree** of a vertex  $v$  is the number of arcs with  $v$  as the terminating vertex. The **out-degree** of a vertex  $v$  is the number of arcs with  $v$  as the starting vertex. In computing the in-degree and out-degree of a vertex, we assume that each loop on a vertex contributes 1 to the in-degree and also 1 to the out-degree of  $v$ .

#### EXAMPLE 10.1.17

Let  $V = \{v_1, v_2, v_3, v_4\}$ ,  $E = \{e_1, e_2, e_3\}$ , and  $g(e_1) = (v_1, v_1)$ ,  $g(e_2) = (v_2, v_3)$ ,  $g(e_3) = (v_2, v_4)$ . Then  $G = (V, E, g)$  is a directed graph. This diagram of  $G$  is shown in Figure 10.9.



**FIGURE 10.9**  
Graph of  $G$

In digraph  $G$ , the in-degrees of  $v_1, v_2, v_3$ , and  $v_4$  are 1, 0, 1, and 1, respectively, and the out-degrees of  $v_1, v_2, v_3$ , and  $v_4$  are 1, 2, 0, and 0, respectively. Notice that the sum of the in-degrees of all vertices = the sum of the out-degrees of all vertices = the number of arcs = 3. This result is true for any digraph and is stated in the next theorem.

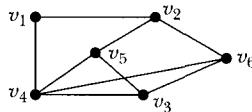
**Theorem 10.1.18:** In any digraph  $G = (V, E, g)$ , the following three numbers are equal.

- (i) The sum of the in-degrees of all the vertices
- (ii) The sum of the out-degrees of all the vertices
- (iii) The number of arcs

**Proof:** The proof is similar to the proof of Theorem 10.1.13. Here we consider the fact that each arc  $e$  with starting vertex  $u$  and terminating vertex  $v$  contributes 1 to the out-degree of  $u$  and 1 to the in-degree of  $v$ . We leave the details as an exercise for the reader. ■

## Simple Graphs

Consider the graph in Figure 10.10.



**FIGURE 10.10** A simple graph

This graph has no loops and no parallel edges. This type of graph is called a *simple graph*.

**DEFINITION 10.1.19** ▶ A graph  $G$  is called a **simple graph** if  $G$  does not contain any parallel edges and any loops.

Earlier we remarked that for any finite nondecreasing sequence of nonnegative integers whose sum is an even integer, there exists a graph such that its degree sequence is the given sequence. However, we note that for a finite nondecreasing sequence of nonnegative integers whose sum is an even integer, there may not exist a simple graph such that its degree sequence is the given sequence. (See Worked-Out Exercise 5 at the end of this section.)

The graph in Figure 10.10 contains at least two vertices of same degree. In the following theorem we prove this result for any simple graph.

**Theorem 10.1.20:** Let  $G$  be a simple graph with at least two vertices. Then  $G$  has at least two vertices of same degree.

**Proof:** Let  $G$  be a simple graph with  $n \geq 2$  vertices. Graph  $G$  has no loops and no parallel edges. Therefore, the degree of a vertex  $v$  is the same as the number of vertices adjacent to  $v$ . Now graph  $G$  has  $n$  vertices. Thus, a vertex  $v$  has at most  $n - 1$  adjacent vertices, because vertex  $v$  is not adjacent to itself. Hence, for any vertex  $v$ , the degree of  $v$  is one of the following integers:  $0, 1, 2, 3, \dots, n - 1$ .

We now show that if there exists a vertex  $v$  such that degree of  $v$  is 0, then for each vertex  $u$  of  $G$ ,  $\deg(u) < n - 1$ . On the contrary, suppose that in  $G$ ,  $v$  is a vertex with degree 0 and  $u$  is a vertex with degree  $n - 1$ . Then  $v$  is an isolated vertex and  $u$  has  $n - 1$  adjacent vertices. Because  $G$  is a simple graph,  $u$  is not adjacent to itself. From this and the fact that  $G$  is simple and  $\deg(u) = n - 1$ , it follows that every vertex of  $G$  other than  $u$  is adjacent to  $u$ . This implies that  $v$  is adjacent to  $u$ , so  $v$  cannot be an isolated vertex, which is a contradiction. This proves our claim.

In a similar manner, we can prove that if there exists a vertex  $v$  in  $G$  such that the degree of  $v$  is  $n - 1$ , then for each vertex  $u$  in  $G$ ,  $\deg(u) > 0$ .

We can now conclude that the degrees of all the vertices in  $G$  are either in the set  $\{0, 1, 2, 3, \dots, n - 2\}$  or in the set  $\{1, 2, 3, \dots, n - 1\}$ .

Let  $v_1, v_2, \dots, v_n$  be the  $n$  vertices of  $G$ . Then either for all  $i = 1, 2, \dots, n$ ,  $\deg(v_i) \in \{0, 1, 2, 3, \dots, n - 2\}$ , or for all  $i = 1, 2, \dots, n$ ,  $\deg(v_i) \in \{1, 2, 3, \dots, n - 1\}$ . Thus, by the pigeonhole principle there exist  $i$  and  $j$ ,  $1 \leq i \leq n$ ,  $1 \leq j \leq n$ ,  $i \neq j$  such that  $\deg(v_i) = \deg(v_j)$ . Hence, there are at least two vertices of same degree. ■

**REMARK 10.1.21** ▶ Note that the converse of Theorem 10.1.20 is not true. For example, consider the graph in Figure 10.7. Vertices  $v_2$  and  $v_3$  are of the same degree. However, the graph is not simple.

In the next example, we use Theorem 10.1.20 to answer one of the problems posed in the beginning of this section.

**EXAMPLE 10.1.22**

We show that in a gathering of  $n > 1$  people, there are at least two people with exactly the same number of friends.

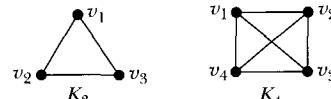
We convert this problem into a graph theory problem as follows: Let  $G = (V, E)$  be a graph with  $n > 1$  vertices such that the  $n$  people are represented by the  $n$  vertices of  $G$ . Now two vertices  $u$  and  $v$  of  $G$  are to be considered adjacent if and only if  $u$  and  $v$  are distinct and the people represented by  $u$  and  $v$  are friends. Thus, we obtain a simple graph with  $n > 1$  vertices. By Theorem 10.1.20, it follows that  $G$  has at least two vertices of same degree. This implies that  $G$  has at least two vertices with the same number of adjacent vertices. Hence, there are at least two people with exactly the same number of friends.

**DEFINITION 10.1.23** ▶ A simple graph with  $n$  vertices in which there is an edge between every pair of distinct vertices is called a **complete graph** on  $n$  vertices. This is denoted by  $K_n$ .

A complete graph on 3 vertices is called a **triangle**.

**EXAMPLE 10.1.24**

The complete graphs  $K_3$  and  $K_4$  are shown in Figure 10.11.

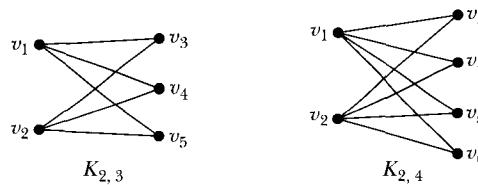


**FIGURE 10.11** Graphs of  $K_3$  and  $K_4$

**DEFINITION 10.1.25** ▶ A simple graph  $G$  is called a **bipartite graph** if the vertex set  $V$  of  $G$  can be partitioned into nonempty subsets  $V_1$  and  $V_2$  such that each edge of  $G$  is incident with one vertex in  $V_1$  and one vertex in  $V_2$ .  $V_1 \cup V_2$  is called a **bipartition** of  $G$ .

**DEFINITION 10.1.26** ▶ A bipartite graph  $G$  with bipartition  $V_1 \cup V_2$  is called a **complete bipartite graph** on  $m$  and  $n$  vertices if the subsets  $V_1$  and  $V_2$  contain  $m$  and  $n$  vertices, respectively, such that there is an edge between each pair of vertices  $v_1$  and  $v_2$ , where  $v_1 \in V_1$  and  $v_2 \in V_2$ . A complete bipartite graph on  $m$  and  $n$  vertices is denoted by  $K_{m,n}$ .

The complete bipartite graph on 2 and 3 vertices  $K_{2,3}$  and the complete bipartite graph on 2 and 4 vertices  $K_{2,4}$  are shown in Figure 10.12:



**FIGURE 10.12** Graphs of  $K_{2,3}$  and  $K_{2,4}$

Note that the number of edges in the graph of  $K_{m,n}$ ,  $m \geq 1$ ,  $n \geq 1$ , is  $mn$ . We now prove a basic property of a complete graph.

**Theorem 10.1.27:** The number of edges in a complete graph with  $n$  vertices is  $\frac{n(n-1)}{2}$ .

**Proof:** Let  $G$  be a complete graph with  $n$  vertices. Then  $G$  is a simple graph such that there exists an edge between any two distinct vertices. Therefore, for any vertex  $v$  of  $G$ , each of the remaining  $n - 1$  vertices is adjacent to  $v$ . Hence, the degree of each vertex in  $G$  is  $n - 1$ . Because  $G$  has  $n$  vertices, the sum of the degrees of the vertices is  $n(n - 1)$ . By Theorem 10.1.13,  $n(n - 1)$  is twice the total number of edges in  $G$ . We can now conclude that in a complete graph the number of edges is  $\frac{n(n-1)}{2}$ . ■

**REMARK 10.1.28** ▶ In a complete graph, there is an edge between two distinct vertices. Hence, the number of edges of this graph is equal to the number of selections of two distinct vertices from  $n$  distinct vertices, which is  $C(n, 2)$ .

## Subgraph

Consider the graphs  $G = (V, E)$  and  $G_1 = (V_1, E_1)$  shown in Figure 10.13.

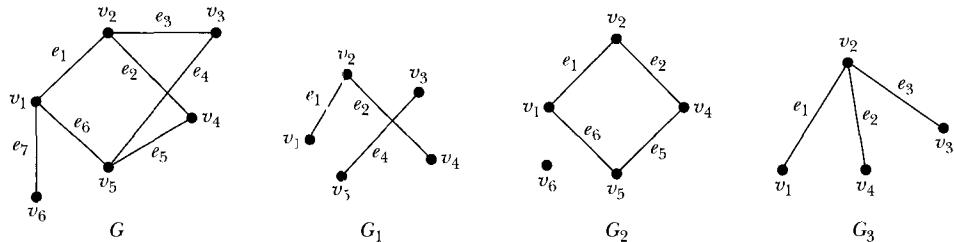


FIGURE 10.13 Various graphs

Now  $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ ,  $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$ ,  $V_1 = \{v_1, v_2, v_3, v_4, v_5\}$ , and  $E_1 = \{e_1, e_2, e_3\}$ . We thus see that  $V_1 \subseteq V$  and  $E_1 \subseteq E$ . In other words, every vertex in  $G_1$  is a vertex in  $G$ , and if  $e$  is an edge in  $G_1$ ,  $e$  is an edge in  $G$ . We say that graph  $G_1$  is a subgraph of  $G$ . In a similar manner, graphs  $G_2$  and  $G_3$  are also subgraphs of  $G$ .

Formally, we give the following definition of subgraph.

**DEFINITION 10.1.29** ▶ Let  $G = (V, E, g)$  be a graph. A triple  $G_1 = (V_1, E_1, g_1)$  is called a **subgraph** of  $G$  if  $V_1$  is a nonempty subset of  $V$ ,  $E_1$  is a subset of  $E$ , and  $g_1$  is the restriction of  $g$  to  $E_1$  such that for all  $e \in E_1$ , if  $g_1(e) = g(e) = \{u, v\}$ , then  $u, v \in V_1$ .

**REMARK 10.1.30** ▶ Let  $G = (V, E)$  be a graph and  $G_1 = (V_1, E_1)$  be a subgraph of  $G$ . From Definition 10.1.29, it follows that if  $e \in E$  and  $u, v$  are the end vertices of  $e$  in  $G$ , then  $u, v \in V_1$ .

Let  $G$  be a graph with vertex set  $V$  and edge set  $E$ . Suppose that  $V$  contains more than one vertex. Then for any vertex  $v \in V$ ,  $G - \{v\}$  denotes the subgraph whose vertex set is  $V_1 = V - \{v\}$  and the edge set  $E_1 = \{e \in E \mid v \text{ is not an end vertex of } e\}$ .

The graph  $G - \{v\}$  is called the *subgraph obtained from  $G$  by deleting the vertex  $v$* .

Let  $e \in E$ ,  $G - \{e\}$  denote the subgraph whose edge set  $E_1 = E - \{e\}$  and the vertex set  $V_1 = V$ . Then  $G - \{e\}$  is called the *subgraph obtained from G by deleting the edge e*.

**REMARK 10.1.31** ▶ Note that  $G - \{v\}$  is obtained from  $G$  by deleting vertex  $v$  and at the same time deleting all the edges that have  $v$  as one of the end vertices. However, the graph  $G - \{e\}$  is obtained from  $G$  by deleting only edge  $e$ , without deleting any vertices of  $G$ .

Let  $G = (V, E)$  be a simple graph with  $n$  vertices. We convert this graph into a complete graph  $H = (V, E_1)$  with  $n$  vertices as follows: The vertex set of  $H$  is the same as the vertex set of  $G$ . Let  $u$  and  $v$  be distinct vertices in  $H$ . If there is an edge, say  $e$ , from  $u$  to  $v$  in  $G$ , then  $e$  is also an edge from  $u$  to  $v$  in  $H$ . If there is no edge from  $u$  to  $v$  in  $G$ , then we add a new edge from  $u$  to  $v$  in  $H$ .

It follows that  $G$  is a subgraph of  $H$ . Now from this complete graph we delete all the old edges without deleting any vertex and form a new graph  $G' = (V', E')$ ; i.e., we construct a subgraph  $G' = (V', E')$  of  $H$  as follows:  $V' = V_1$  and

$$E' = E_1 - E.$$

This graph  $G'$  is called the **complement** of graph  $G$ . We note that if  $G$  is a simple graph with  $n$  vertices, then the extended complete graph is  $K_n$  and  $E \cup E' = E_n$  and  $E \cap E' = \emptyset$ , where  $E_n$  is the set of edges of  $K_n$ .

**Theorem 10.1.32:** For any simple graph  $G$  with six vertices either  $G$  or its complement  $G'$  contains a triangle as a subgraph.

**Proof:** Let  $u_1, u_2, u_3, u_4, u_5, u_6$  be the six vertices of graph  $G$ . Now they are also vertices of  $G'$ . Now each of the five vertices  $u_2, u_3, u_4, u_5, u_6$  is adjacent with  $u_1$  in  $G$  or in  $G'$ . Among these five vertices there are three vertices  $x, y, z$ , which are either adjacent with  $u_1$  in  $G$  or in  $G'$ . Suppose  $u_2, u_3, u_4$  are not adjacent with  $u_1$  in  $G$ . Then they are adjacent in  $G'$ . Among these vertices suppose  $u_3, u_4$  are not adjacent with  $u_1$  in  $G$ . Then they are adjacent in  $G'$ . Now if one of  $u_5, u_6$  is adjacent with  $u_1$  in  $G'$ , we are done. If neither of  $u_5, u_6$  are adjacent with  $u_1$  in  $G'$ , then they are adjacent with  $u_1$  in  $G$ . Then we obtain three vertices  $u_2, u_5, u_6$ , which are adjacent with  $u_1$  in  $G$ . Let  $V_1 = \{u, x, y, z\}$  and  $E_1 = \{e_1, e_2, e_3\}$ , where  $u, x$  are the end vertices of  $e_1$ ,  $u, y$  are the end vertices of  $e_2$ , and  $u, z$  are the end vertices of  $e_3$ . Then  $G_1 = (V_1, E_1)$  (see Figure 10.14), is a subgraph of  $G$  or  $G'$ .

Suppose  $G_1 = (V_1, E_1)$  is a subgraph of  $G$ . Let  $x, y$  be the end vertices of edge  $e_4$ ,  $y$  and  $z$  be the end vertices of  $e_5$ , and  $x, z$  be the end vertices of  $e_6$ . If one of edges  $e_4, e_5, e_6$  belongs to  $G$ —for example, if  $e_4$  belongs to  $G$ —then the triangle formed by  $(\{u, x, y\}, \{e_1, e_2, e_4\})$  belongs to  $G$  (see Figure 10.15).

Suppose that none of  $e_4, e_5, e_6$  belongs to  $G$ . Then  $e_4, e_5, e_6$  belong to  $G'$ . This implies that the triangle formed by  $(\{x, y, z\}, \{e_4, e_5, e_6\})$  belongs to  $G'$ . Hence, either  $G$  or its complement,  $G'$ , contains a triangle as a subgraph. Similarly, if  $G_1 = (V_1, E_1)$  is a subgraph of  $G'$  we can prove that either  $G$  or its complement,  $G'$ , contains a triangle as a subgraph. ■

From Theorem 10.1.32, we find that in any simple graph  $G$  with six vertices either  $G$  or its complement,  $G'$ , contains the complete graph  $K_3$  as a subgraph. Theorem 10.1.32 motivates the following problem.

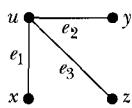


FIGURE 10.14  
Graph of  $G_1 = (V_1, E_1)$

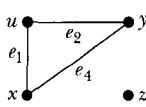


FIGURE 10.15  
Graph of  $\{u, x, y\}, \{e_1, e_2, e_4\}$ )

Let  $m$  and  $n$  be positive integers. What is the smallest integer  $r(m, n)$  such that every simple graph,  $G$ , with  $r(m, n)$  vertices contains  $K_m$  or its complement,  $G'$ , contains  $K_n$ ?

The values  $r(m, n)$  are called **Ramsey numbers**. Of course,  $r(m, n) = r(n, m)$ . The determination of Ramsey numbers is an unsolved problem in graph theory.

Let us now solve the problem stated at the beginning of the section.

### EXAMPLE 10.1.33

In this example, we solve the problem posed at the beginning of this section. That is, in a party that consists of six people, is it true that it is always possible to find either three people who know each other or three people such that no one knows each other?

We first convert the problem into a problem of graph theory. We construct a graph  $G = (V, E)$  with six vertices. Each of the six people in the party is identified with six vertices of the graph, respectively. Two vertices of  $G$  are to be considered adjacent if and only if they are distinct and the corresponding people know each other. Thus, we obtain a simple graph with six vertices. From Theorem 10.1.32, either  $G$  or its complement,  $G'$ , contains a triangle formed by  $(\{x, y, z\}, \{e_4, e_5, e_6\})$  as a subgraph. If  $G$  contains this subgraph, then the three people corresponding to vertices  $x, y, z$  know each other. On the other hand, if  $G'$  contains this subgraph, then the three people corresponding to vertices  $x, y, z$  do not know each other, because they are not adjacent in  $G$ .

## WORKED-OUT EXERCISES

**Exercise 1:** Find the degree of each vertex in the graphs in Figure 10.16.

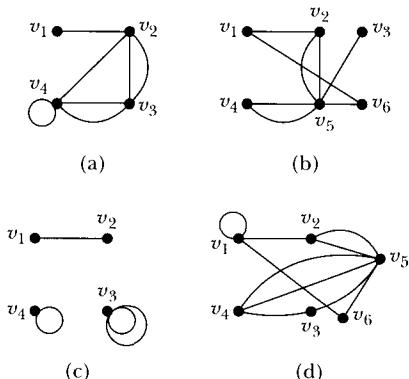


FIGURE 10.16 Various graphs

### Solution:

- Only one edge is incident on vertex  $v_1$ . Therefore,  $\deg(v_1) = 1$ . Four edges are incident on vertex  $v_2$ . Therefore,  $\deg(v_2) = 4$ . Three edges and one loop are incident on vertex  $v_4$ . Therefore,  $\deg(v_4) = 5$ . Similarly,  $\deg(v_3) = 4$ .
- Proceeding as in part (a), we find that  $\deg(v_1) = 2$ ,  $\deg(v_2) = 3$ ,  $\deg(v_3) = 1$ ,  $\deg(v_4) = 2$ ,  $\deg(v_5) = 6$ , and  $\deg(v_6) = 2$ .
- Proceeding as in part (a), we find that  $\deg(v_1) = 1$ ,  $\deg(v_2) = 1$ ,  $\deg(v_3) = 4$ , and  $\deg(v_4) = 2$ .

- Proceeding as in part (a), we find that  $\deg(v_1) = 4$ ,  $\deg(v_2) = 3$ ,  $\deg(v_3) = 2$ ,  $\deg(v_4) = 3$ ,  $\deg(v_5) = 6$ , and  $\deg(v_6) = 2$ .

**Exercise 2:** Determine which of the graphs in Figure 10.17 are simple.

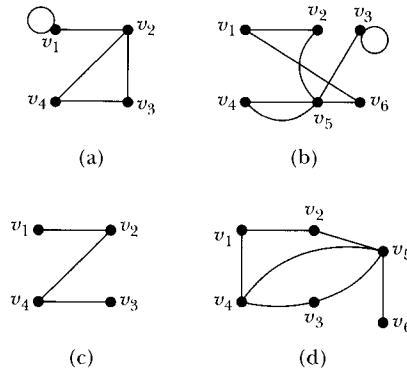


FIGURE 10.17 Various graphs

### Solution:

- There is a loop at vertex  $v_1$ . Hence, this graph is not simple.
- This is not a simple graph, because it contains a parallel edge and a loop.
- There are no loops and parallel edges in this graph. Hence, this graph is a simple graph.
- There are no loops and parallel edges in this graph. Hence, this graph is a simple graph.

**Exercise 3:** Draw a graph with degree sequence 0, 3, 4, 4, 5, 5, 5.

**Solution:** Let  $G$  be a graph with degree sequence 0, 3, 4, 4, 5, 5, 5. This is a graph with seven vertices. Let  $v_1, v_2, \dots, v_7$  be the vertices of this graph such that  $\deg(v_1) = 0$ ,  $\deg(v_2) = 3$ ,  $\deg(v_3) = 4$ ,  $\deg(v_4) = 4$ ,  $\deg(v_5) = 5$ ,  $\deg(v_6) = 5$ , and  $\deg(v_7) = 5$ . Here  $v_1, v_3, v_4$  are even degree vertices and  $v_2, v_5, v_6$ , and  $v_7$  are odd degree vertices. Draw  $\frac{0}{2} = 0$  loops at  $v_1$ ,  $\frac{4}{2} = 2$  loops at  $v_3$ , and  $\frac{4}{2} = 2$  loops at  $v_4$ . Draw  $\frac{3-1}{2} = 1$  loop at  $v_2$ ,  $\frac{5-1}{2} = 2$  loops at  $v_5$ ,  $\frac{5-1}{2} = 2$  loops at  $v_6$ , and  $\frac{5-1}{2} = 2$  loops at  $v_7$ . Next draw an edge between  $v_2, v_5$  and an edge between  $v_6, v_7$ . We know that each loop at vertex  $v$  contributes 2 to the degree of the vertex at  $v$ . Thus, we get the graph shown in Figure 10.18(a) with the degree sequence 0, 3, 4, 4, 5, 5, 5.

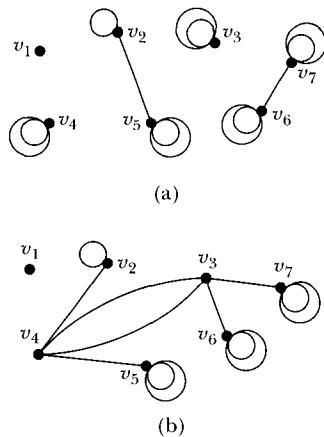


FIGURE 10.18 Graphs with degree sequence 0, 3, 4, 4, 5, 5, 5

The graph with the given degree sequence may not be unique. We can draw a different graph as follows: Let  $G_1$  be a graph with degree sequence 0, 3, 4, 4, 5, 5, 5. This is a graph with seven vertices. Let  $v_1, v_2, \dots, v_7$  be the vertices of this graph such that  $\deg(v_1) = 0$ ,  $\deg(v_2) = 3$ ,  $\deg(v_3) = 4$ ,  $\deg(v_4) = 4$ ,  $\deg(v_5) = 5$ ,  $\deg(v_6) = 5$ , and  $\deg(v_7) = 5$ . Here  $v_2, v_5, v_6$ , and  $v_7$  are odd degree vertices. Draw  $\frac{3-1}{2} = 1$  loops at  $v_2$ ,  $\frac{5-1}{2} = 2$  loops at  $v_5$ ,  $\frac{5-1}{2} = 2$  loops at  $v_6$ , and  $\frac{5-1}{2} = 2$  loops at  $v_7$ . Next draw an edge between  $v_2, v_4$ ; an edge between  $v_4, v_5$ ; two parallel edges between  $v_3, v_4$ ; an edge between  $v_3, v_6$ ; and an edge between  $v_3, v_7$ . Thus, we get the graph shown in Figure 10.18(b) with the degree sequence 0, 3, 4, 4, 5, 5, 5.

**Exercise 4:** Draw a graph having the given properties or explain why no such graph exists.

- Simple graph, five vertices each of degree 2
- Simple graph having the degree sequence 3, 3, 3, 3, 4
- Six edges and having the degree sequence 1, 2, 3, 4, 6

**Solution:**

- The graph is shown in Figure 10.19(a).
- The graph is shown in Figure 10.19(b).

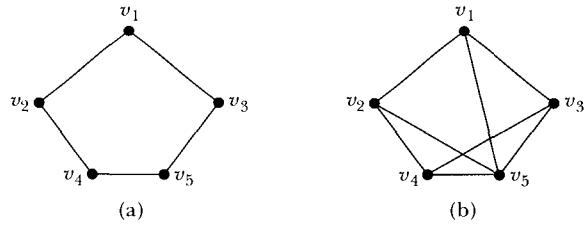


FIGURE 10.19 Various graphs

(c) Suppose there exists a graph  $G$  with the given properties. Then the sum of the degrees of the vertices of  $G$  is 16. In any graph, we know that the sum of the degrees is twice the number of edges. Thus,  $G$  must have eight edges. Hence, it follows that there does not exist any graph with the given properties.

**Exercise 5:** Does there exist a simple graph with six vertices having degrees 2, 2, 2, 4, 5, 5? Justify your answer.

**Solution:** There does not exist any simple graph satisfying the given properties.

Suppose there exists a simple graph with six vertices  $v_1, v_2, v_3, v_4, v_5$ , and  $v_6$  having degrees 2, 2, 2, 4, 5, and 5, respectively. Because the graph does not contain any loops and any parallel edges, it follows that  $v_5$  and  $v_6$  must have five adjacent vertices different from  $v_5$  and  $v_6$ , respectively. Hence,  $v_1, v_2$ , and  $v_3$  are adjacent vertices of  $v_5$  and  $v_6$ . This implies that the degrees of  $v_1, v_2$ , and  $v_3$  are at least 2. Again, the degree of  $v_4$  is 4. Hence,  $v_4$  must have four adjacent vertices different from  $v_4$ . Then at least two of  $v_1, v_2$ , and  $v_3$  are adjacent vertices of  $v_4$ , which implies that the degrees of at least two of  $v_1, v_2$ , and  $v_3$  must be greater than or equal to 3. This is a contradiction to our assumption. Hence, there is no such graph.

**Exercise 6:** How many vertices are there in a graph with 20 edges if each vertex is of degree 5?

**Solution:** Let there be  $n$  vertices. Because each vertex is of degree 5, the sum of the degrees of  $n$  vertices is  $5n$ . This sum is twice the number of edges. Hence,  $5n = 2 \cdot 20$ . So we find that the number of vertices is 8.

**Exercise 7:** Which graphs in Figure 10.20 are bipartite graphs?

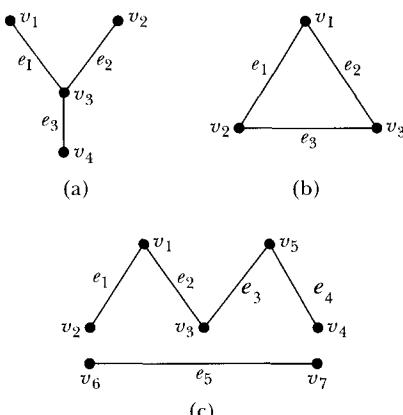


FIGURE 10.20 Various graphs

**Solution:**

- (a) Let  $G$  denote the graph in Figure 10.20(a). Let  $U = \{v_1, v_2, v_4\}$  and  $W = \{v_3\}$ . Then  $V$ , the vertex set of  $G$ , is the union of  $U$  and  $W$ , and  $U \cap W = \emptyset$ . Moreover, each edge in  $G$  has one of its end vertices in  $U$  and the other in  $W$ . Hence,  $G$  is a bipartite graph.
- (b) Let  $G$  denote the graph in Figure 10.20(b). Suppose  $G$  is a bipartite graph. Then the vertex set is the union of two disjoint sets, say  $U$  and  $W$ . Now edge  $e_1$  connects  $v_1$  and  $v_2$ . So one of  $v_1$  and  $v_2$  is in  $U$  and the other is in  $W$ . To be specific, suppose  $v_1 \in U$  and  $v_2 \in W$ . Now consider  $v_3$ . Because  $e_2$  is an edge that connects  $v_1$  and  $v_3$  and  $v_1 \in U$ , we must have  $v_3 \in W$ . Now  $v_2$  and  $v_3 \in W$  and  $G$  is bipartite. Thus, it follows that there cannot be any edge connecting  $v_2$  and  $v_3$ . This is a contradiction because  $e_3$  is an edge connecting  $v_2$  and  $v_3$ . Hence,  $G$  is not a bipartite graph.
- (c) Let  $G$  denote the graph in Figure 10.20(a). Let  $U = \{v_1, v_5, v_6\}$  and  $W = \{v_2, v_3, v_4, v_7\}$ . Then  $V$ , the vertex set of  $G$ , is the union of  $U$  and  $W$ , and  $U \cap W = \emptyset$ . Moreover, each edge in  $G$  has one of its endpoints in  $U$  and the other in  $W$ . Hence,  $G$  is a bipartite graph.

**Exercise 8:** Find three subgraphs of graph  $G$  in Figure 10.21.

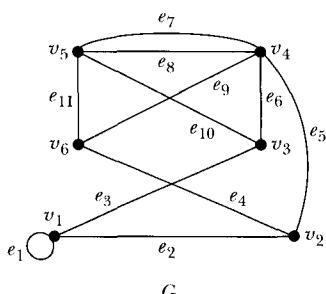


FIGURE 10.21 A graph

**Solution:** Graphs  $G_1$ ,  $G_2$ , and  $G_3$ , shown in Figure 10.22, are subgraphs of graph  $G$ .

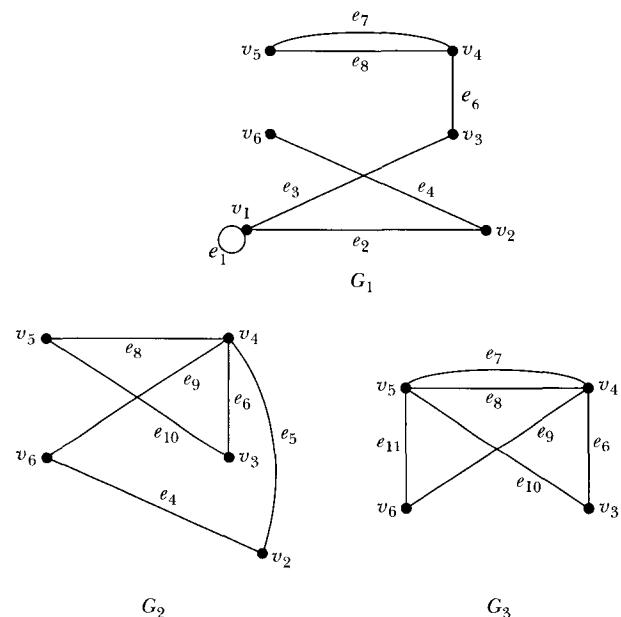


FIGURE 10.22 Subgraphs of the graph in Figure 10.21

## SECTION REVIEW

### Key Terms

|                    |                    |                    |
|--------------------|--------------------|--------------------|
| graph              | parallel edges     | arc                |
| set of vertices    | isolated vertex    | starting vertex    |
| set of edges       | degree             | terminating vertex |
| incidence function | $k$ -regular graph | in-degree          |
| end vertices       | even degree vertex | out-degree         |
| endpoints          | odd degree vertex  | simple graph       |
| incident           | degree sequence    | complete graph     |
| adjacent           | directed graph     | triangle           |
| loop               | digraph            | bipartite graph    |
| incidence table    | directed edge      | bipartition        |

|                                |                                         |
|--------------------------------|-----------------------------------------|
| complete bipartite<br>subgraph | complement of a graph<br>Ramsey numbers |
|--------------------------------|-----------------------------------------|

## Some Key Definitions

1. A graph  $G$  is a triple  $(V, E, g)$ , where
  - (i)  $V$  is a finite nonempty set, called the set of vertices;
  - (ii)  $E$  is a finite set (may be empty), called the set of edges; and
  - (iii)  $g$  is a function, called an incidence function, that assigns to each edge,  $e \in E$ , a one-element subset  $\{v\}$  or a two-element subset  $\{v, w\}$ , where  $v$  and  $w$  are vertices.
2. Let  $G$  be a graph and  $v$  be a vertex of  $G$ . The degree of  $v$ , written  $\deg(v)$  or  $d(v)$ , is the number of edges incident with  $v$ .
3. Let  $n_1, n_2, n_3, \dots, n_k$  be the degrees of vertices of a graph  $G$  such that  $n_1 \leq n_2 \leq n_3 \leq \dots \leq n_k$ . Then the finite sequence  $n_1, n_2, n_3, \dots, n_k$  is called the degree sequence of the graph.
4. A directed graph (or digraph)  $G$  is a triple  $(V, E, g)$ , where
  - (i)  $V$  is a finite nonempty set of vertices;
  - (ii)  $E$  is a finite set (may be empty) of directed edges, or arcs; and
  - (iii)  $g : E \rightarrow V \times V$  is a function that assigns to each arc  $e$  an ordered pair  $(v, w)$ , where  $v$  and  $w$  are vertices ( $v$  and  $w$  may be the same).
5. A graph  $G$  is called a simple graph if  $G$  contains no parallel edges and no loops.
6. A simple graph  $G$  is called a bipartite graph if the vertex set  $V$  of  $G$  can be partitioned into nonempty subsets  $V_1$  and  $V_2$  such that each edge of  $G$  is incident with one vertex in  $V_1$  and one vertex in  $V_2$ .  $V_1 \cup V_2$  is called a bipartition of  $G$ .
7. Let  $G = (V, E, g)$  be a graph. A triple  $G_1 = (V_1, E_1, g_1)$  is called a subgraph of  $G$  if  $V_1$  is a nonempty subset of  $V$ ;  $E_1$  is a subset of  $E$ ; and  $g_1$  is the restriction of  $g$  to  $E_1$  such that for all  $e \in E_1$  if  $g_1(e) = g(e) = \{u, v\}$ , then  $u, v \in V_1$ .

## Some Key Results

1. The sum of the degrees of all vertices of a graph is twice the number of edges.
2. The sum of the degrees of all vertices of a graph is an even integer.
3. In a graph, the number of odd degree vertices is even.
4. The number of edges in a complete graph with  $n$  vertices is  $\frac{n(n-1)}{2}$ .

## EXERCISES

---

1. Draw a graph with three vertices such that two vertices are of degree 1 and one is of degree 2. How many edges are there in this graph?
2. Draw a graph with five vertices such that two vertices are of degree 4, one is of degree 5, and another is of degree 1. How many edges are there in this graph?

3. Draw a graph with four vertices such that two vertices are of degree 4, one is of degree 5, and another is of degree 3. How many edges are there in this graph?
4. Draw a graph  $G = (V, E, g)$ , where  $V = \{v_1, v_2, v_3, v_4, v_5\}$ ,  $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$ , and  $g$  is defined by

$$\begin{aligned}g(e_1) &= g(e_2) = \{v_1, v_2\} \\g(e_3) &= \{v_4, v_4\} \\g(e_4) &= g(e_5) = \{v_1, v_3\} \\g(e_6) &= \{v_2, v_4\} \\g(e_7) &= \{v_4, v_5\}.\end{aligned}$$

Find the degree of each vertex. Find all odd degree vertices.

5. Draw a graph  $G = (V, E, g)$ , where  $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$  with incidence table

| Edge         | $e_1$      | $e_2$      | $e_3$      | $e_4$      | $e_5$      | $e_6$      | $e_7$      |
|--------------|------------|------------|------------|------------|------------|------------|------------|
| End vertices | $v_1, v_1$ | $v_1, v_2$ | $v_4, v_3$ | $v_6, v_1$ | $v_6, v_3$ | $v_2, v_4$ | $v_6, v_8$ |

6. Draw a graph with five vertices  $v_1, v_2, v_3, v_4, v_5$  such that  $\deg(v_1) = 3$ ,  $\deg(v_2) = 2$ ,  $\deg(v_3) = 2$ ,  $\deg(v_4) = 3$ ,  $\deg(v_5) = 2$ , and  $v_1$  and  $v_2$  are adjacent to  $v_5$ .
7. Draw a simple graph with five vertices  $v_1, v_2, v_3, v_4, v_5$  such that  $\deg(v_1) = 3$ ,  $\deg(v_2) = 2$ ,  $\deg(v_3) = 2$ ,  $\deg(v_4) = 3$ ,  $\deg(v_5) = 2$ , and  $v_1$  and  $v_2$  are adjacent to  $v_5$ .
8. List the degrees of the vertices of the graphs in Figure 10.23. Find the number of odd degree vertices.

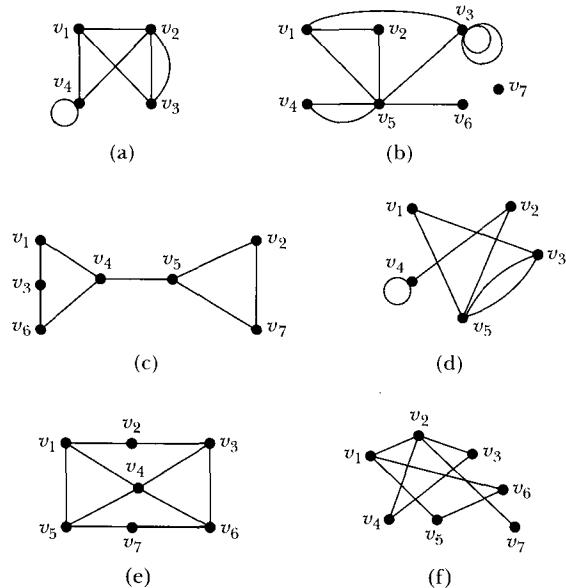


FIGURE 10.23 Various graphs

9. Write the degree sequences of the graphs of Exercise 8 of this section.
10. Draw a graph with degree sequence 1, 1, 4, 6, 6.
11. State which graphs of Exercise 8, of this section, are simple. If the graph is simple, then find the complement.

12. How many vertices are there in a graph with 15 edges if each vertex is of degree 3?
13. How many vertices are there in a graph with 20 edges if each vertex has degree 4?
14. A graph has a degree sequence 1, 1, 2, 2, 2, 2, 4. Find the number of edges of this graph and draw the graph.
15. Does there exist a graph with 20 edges if each vertex is of degree 3?
16. Find degree sequences of all possible simple graphs with three vertices.
17. Find degree sequences of all possible simple graphs with five vertices and three edges.
18. Draw a simple graph such that every vertex is adjacent to two vertices and every edge is adjacent to two edges.
19. Let  $G$  be a graph with  $n$  vertices and  $n - 1$  edges. Show that  $G$  has either an isolated vertex or a vertex of degree 1.
20. Does there exist a graph with four edges and degree sequence 1, 2, 3, 4? Justify your answer.
21. Does there exist a graph with degree sequence 1, 2, 3, 4, 5? Justify your answer.
22. Does there exist a simple graph with degree sequence 1, 2, 4, 5? Justify your answer.
23. Does there exist a simple graph with degree sequence 2, 2, 2, 4, 5, 5? Justify your answer.
24. Does there exist a simple graph with degree sequence 2, 3, 3, 4, 4, 4? Justify your answer.
25. Let  $G = (V, E, g)$  be a graph,  $V = \{1, 2, 3, \dots, 8\}$ ,  $E = \{e_1, e_2, \dots, e_n\}$ , and  $g(E) = \{\{x, y\} \mid x, y \in V, x \neq y \text{ and } x \text{ divides } y \text{ or } y \text{ divides } x\}$ . Find  $n$ , the number of edges. Draw this graph. Is it a simple graph?
26. Let  $G$  be a graph with eight vertices having degrees 5, 5, 4, 3, 3, 2, 2, 2. Find the number of edges of  $G$ .
27. Suppose in a graph  $G$  each vertex is of degree 5. Prove that the number of edges in  $G$  is a multiple of 5.
28. Draw a graph having the given properties or explain why no such graph exists.
- a. (i) Five vertices each of degree 2  
(ii) Five vertices each of degree 4
  - b. Simple graph with four vertices each of degree 2
  - c. Simple graph with seven edges and nine vertices such that each vertex is of degree at least 1
  - d. Six vertices and four edges
  - e. Six edges; six vertices having degrees 1, 1, 2, 4, 5, 5
  - f. Simple graph; four vertices having degrees 3, 3, 3, 1
  - g. Simple graph; six vertices having degrees 2, 2, 3, 3, 3, 3
  - h. Seven vertices having degrees 3, 5, 2, 7, 4, 6, 8
  - i. Simple graph with five vertices such that every vertex is incident with at least one edge but no two edges are adjacent.
29. Draw a digraph with three vertices  $u, v, w$  such that

| Vertices   | $u$ | $v$ | $w$ |
|------------|-----|-----|-----|
| In-degree  | 1   | 1   | 2   |
| Out-degree | 3   | 1   | 0   |

30. Does there exist a digraph with four vertices  $u, v, w, x$  satisfying the following condition?

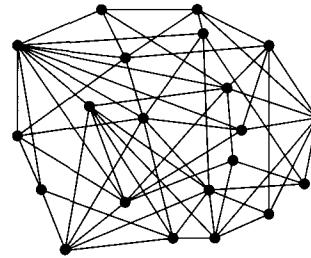
| Vertices   | $u$ | $v$ | $w$ | $x$ |
|------------|-----|-----|-----|-----|
| In-degree  | 1   | 1   | 1   | 2   |
| Out-degree | 2   | 0   | 1   | 1   |



33. Draw the complement of  $K_5$ .
  34. Draw a complete bipartite graphs on 3,4 vertices.
  35. How many edges are there in each of the following graphs?
    - a.  $K_{2,3}$
    - b.  $K_{4,3}$
    - c.  $K_{4,4}$
    - d.  $K_{n,n}$
  36. For each of the graphs of Exercise 8 of this section, draw the subgraphs  $G - \{v_2\}$  of graph  $G$  of part (a),  $G - \{v_3\}$  of graph  $G$  of part (b), and  $G - \{v_2\}$  of graph  $G$  of part (f).
  37. Draw the complement of  $K_{4,3}$ .

## 10.2 WALKS, PATHS, AND CYCLES

Suppose there are 100 small towns in a country. From each town there is a direct bus route to at least 50 towns. Is it possible to go from one town to any other town by bus, possibly changing from one bus in one town and taking another bus to another town? Figure 10.24 shows a graph of some of the cities and bus routes.



**FIGURE 10.24** Bus routes between cities

Let us consider another problem. Suppose there are 200 telephone exchanges in a city. Suppose each telephone exchange has direct lines to 100 other exchanges. Is it always possible to make calls between any two telephone exchanges, perhaps through other exchanges?

In this section, we answer these problems using graph theory. First, however, we introduce more basic concepts of graph theory. Unfortunately, for some of the concepts introduced here there is no universally agreed-upon terminology, so readers should refer to the definitions given in this section while working on a specific problem.

**DEFINITION 10.2.1** ▶ Let  $u$  and  $v$  be two vertices in a graph  $G$ . A **walk** from  $u$  to  $v$ , in  $G$ , is an alternating sequence of  $n + 1$  vertices and  $n$  edges of  $G$

$$(u = v_1, e_1, v_2, e_2, v_3, e_3, \dots, v_{n-1}, e_{n-1}, v_n, e_n, v_{n+1} = v)$$

beginning with vertex  $u$ , called the **initial vertex**, and ending with vertex  $v$ , called the **terminal vertex**, in which  $v_i$  and  $v_{i+1}$  are endpoints of edge  $e_i$  for  $i = 1, 2, \dots, n$ .

**DEFINITION 10.2.2** ▶ Let  $u$  and  $v$  be two vertices in a directed graph  $G$ . A **directed walk** from  $u$  to  $v$  in  $G$  is an alternating sequence of  $n + 1$  vertices and  $n$  arcs of  $G$

$$(u = v_1, e_1, v_2, e_2, v_3, e_3, \dots, v_{n-1}, e_{n-1}, v_n, e_n, v_{n+1} = v)$$

beginning with vertex  $u$  and ending with vertex  $v$ , in which each edge  $e_i$ , for  $i = 1, 2, \dots, n$ , is an arc from  $v_i$  to  $v_{i+1}$ .

**DEFINITION 10.2.3** ► The **length of a walk (directed walk)** is the total number of occurrences of edges (arcs) in the walk (directed walk). A walk or a directed walk of length 0 is just a single vertex.

A (directed) walk from a vertex  $u$  to a vertex  $v$  in  $G$  is also called a  **$u - v$  (directed) walk**. If  $u$  and  $v$  are the same, then a  $u - v$  (directed) walk is called a **closed (directed) walk**. If  $u$  and  $v$  are different, then a  $u - v$  (directed) walk is called an **open (directed) walk**.

**DEFINITION 10.2.4** ► A walk with no repeated edges is called a **trail**, and a walk with no repeated vertices except possibly the initial and terminal vertices is called a **path**.

**REMARK 10.2.5** ► Let  $(u = v_1, e_1, v_2, e_2, v_3, e_3, \dots, v_{n-1}, e_{n-1}, v_n, e_n, v_{n+1} = v)$  be a  $u - v$  walk. If all the edges  $e_1, e_2, e_3, \dots, e_{n-1}, e_n$  are distinct, then this  $u - v$  walk is a trail. If all the vertices  $u = v_1, v_2, v_3, \dots, v_{n-1}, v_n, v_{n+1} = v$  except possibly  $u$  and  $v$  are distinct, then this  $u - v$  walk is a path. Thus, from the preceding definition, it follows that in a path no edge can be repeated. Hence, every path is a trail, but not every trail is a path.

**DEFINITION 10.2.6** ► A walk, path, or trail is called **trivial** if it has only one vertex and no edges. A walk, path, or trail that is not trivial is called **nontrivial**.

**DEFINITION 10.2.7** ► A nontrivial closed trail from a vertex  $u$  to itself is called a **circuit**.

Hence, a circuit is a closed walk of nonzero length from a vertex  $u$  to  $u$  with no repeated edges.

### EXAMPLE 10.2.8

Consider the graph in Figure 10.25.

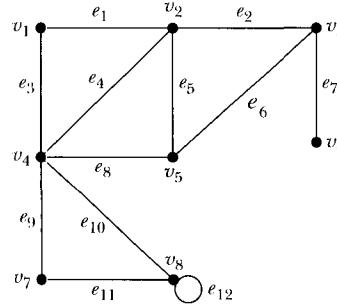


FIGURE 10.25 A graph

In this graph,

$$(v_1, e_1, v_2, e_2, v_3, e_3, v_6, e_7, v_3, e_2, v_2, e_4, v_4)$$

is a walk of length 5. It is an open walk from vertex  $v_1$  to vertex  $v_4$ . This is a walk with no repeated edges. Hence, this walk is a trail. Because  $v_2$  appears twice, this walk is not a path. But

$$(v_2, e_2, v_3, e_6, v_5, e_8, v_4, e_9, v_7)$$

is a path of length 4 from vertex  $v_2$  to vertex  $v_7$ .

**DEFINITION 10.2.9** ► A circuit that does not contain any repetition of vertices except the starting vertex and the terminal vertex is called a **cycle**.

**REMARK 10.2.10** ► From the definition of a cycle we find that a closed trail

$$(v_1, e_1, v_2, e_2, v_3, e_3, \dots, v_{n-1}, e_{n-1}, v_n, e_n, v_{n+1} = v_1)$$

of length  $n \neq 0$  is a cycle if and only if  $v_1, v_2, v_3, \dots, v_{n-1}$  are distinct.

**DEFINITION 10.2.11** ► A cycle of length  $k$  is called a  **$k$ -cycle**. A cycle is called **even (odd)** if it contains an even (odd) number of edges.

It follows from the definition that a 3-cycle is a triangle.

We summarize some of the important points regarding walks, trails, paths, circuits, and cycles in Table 10.1.

**Table 10.1** Some properties of walks, trails, paths, circuits, and cycles

|          | Vertices                                                                 | Edges                  |                                                                                                |
|----------|--------------------------------------------------------------------------|------------------------|------------------------------------------------------------------------------------------------|
| Walks    | Repetition allowed                                                       | Repetition allowed     |                                                                                                |
| Trails   | Repetition allowed                                                       | No repetition of edges |                                                                                                |
| Paths    | No repetition of vertices except possibly starting and terminal vertices | No repetition of edges |                                                                                                |
| Circuits | Repetition allowed                                                       | No repetition of edges | A nontrivial closed trail                                                                      |
| Cycles   | No repetition of vertices except starting and terminal vertices          | No repetition of edges | A nontrivial closed trail without repetition of vertices except starting and terminal vertices |

In the graph of Figure 10.25, Example 10.2.8, the walk

$$(v_2, e_2, v_3, e_3, v_5, e_5, v_8, e_8, v_4, e_4, v_2)$$

is a cycle, and the cycle

$$(v_2, e_2, v_3, e_3, v_6, e_6, v_5, e_5, v_2)$$

is a triangle. In the same graph, the walk

$$(v_4, e_{10}, v_8, e_{12}, v_8, e_{11}, v_7, e_9, v_4)$$

is a trail and also a circuit but not a cycle.

**REMARK 10.2.12** ► Directed trails, directed paths, directed circuits, and directed cycles are defined analogously to those of their counterparts in graphs.

**DEFINITION 10.2.13** ► Let  $P = (v_1, e_1, v_2, e_2, v_3, e_3, \dots, v_{n-1}, e_{n-1}, v_n)$  be a walk in a graph  $G$ . A **subwalk** of  $P$  is a subsequence of consecutive entries  $Q = (v_i, e_i, v_{i+1}, e_{i+1}, \dots, v_{k-1}, e_{k-1}, v_k)$ ,  $1 \leq i \leq k \leq n$ , that begins at a vertex and ends at a vertex.

From the definition of a subwalk, it follows that every subwalk is a walk.

**EXAMPLE 10.2.14**

In the graph of Figure 10.25, Example 10.2.8,  $(v_2, e_2, v_3, e_6, v_5)$  is a subwalk of the walk  $P : (v_1, e_1, v_2, e_2, v_3, e_6, v_5, e_5, v_2, e_4, v_4)$ . However,  $(v_1, e_1, v_2, e_4, v_4)$  is not a subwalk of  $P$ .

Let  $P = (v_1, e_1, v_2, e_2, v_3, e_3, \dots, v_{n-1}, e_{n-1}, v_n)$  be a walk in a graph  $G$  and  $Q = (v_i, e_i, v_{i+1}, e_{i+1}, \dots, v_{k-1}, e_{k-1}, v_k = v_i)$  be a closed subwalk of  $P$ . If we delete this subwalk,  $Q = (v_i, e_i, v_{i+1}, e_{i+1}, \dots, v_{k-1}, e_{k-1}, v_k = v_i)$ , from  $P$  except for vertex  $v_i$ , then we obtain a new walk. This walk is denoted by  $P - Q$  and is called the **reduction of  $P$  by  $Q$** .

**EXAMPLE 10.2.15**

Consider the graph of Figure 10.25, Example 10.2.8. In this graph, let  $Q = (v_2, e_2, v_3, e_6, v_5, e_5, v_2)$  and  $P = (v_1, e_1, v_2, e_2, v_3, e_6, v_5, e_5, v_2, e_4, v_4)$ . Then  $Q$  is a subwalk of  $P$ . Now  $P - Q = (v_1, e_1, v_2, e_4, v_4)$  is a reduction of  $P$  by  $Q$ .

The following theorem is proved for directed graphs in Chapter 3. Here we give a different proof for an undirected graph using the notion of a subwalk and the reduction of a walk.

**Theorem 10.2.16:** Let  $G$  be a graph and  $u, v$  be two vertices of  $G$ . If there is a walk from  $u$  to  $v$ , then there is a path from  $u$  to  $v$ .

**Proof:** Let  $P : (u = v_1, e_1, v_2, e_2, \dots, v_n = v)$  be a walk. If  $u = v$ , then this is a closed walk. In this case,  $(u)$  is a path from  $u$  to  $u$  consisting of a single vertex and no edges. Suppose that  $P : (u = v_1, e_1, v_2, e_2, \dots, v_n = v)$  is an open walk. If this is not a path, then  $v_i = v_j$  for some  $1 \leq i < j \leq n$ . This shows that there is a closed subwalk  $Q$  from  $v_i$  to  $v_j$ . We reduce  $P$  to  $P - Q$ . Now  $P - Q$  is a new walk from  $u$  to  $v$ . If this walk is not a path, we repeat this deletion process of subwalks. Because the number of closed subwalks in  $P$  is finite, we eventually obtain a path from  $u$  to  $v$ . ■

**EXAMPLE 10.2.17**

Consider the graph in Figure 10.26.

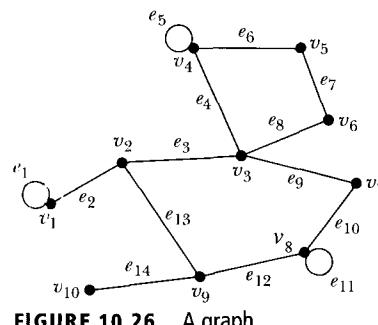


FIGURE 10.26 A graph

Let  $P = (v_1, e_2, v_2, e_3, v_3, e_4, v_4, e_6, v_5, e_7, v_6, e_8, v_3, e_9, v_7)$ . Then  $P$  is a walk from  $v_1$  to  $v_7$ . Here  $Q = (v_3, e_4, v_4, e_6, v_5, e_7, v_6, e_8, v_3)$  is a subwalk of  $P$ . Now  $P - Q = (v_1, e_2, v_2, e_3, v_3, e_9, v_7)$  is a path from  $v_1$  to  $v_7$ .

Proceeding as in Theorem 10.2.16, we can show that a circuit either is a cycle or can be reduced to a cycle, see Exercise 7 at the end of this section.

**Theorem 10.2.18:** Every circuit contains a subwalk that is a cycle.

**Proof:** Let  $T$  be a circuit. Let  $S$  be the collection of all closed nontrivial subwalks of  $T$ . Because  $T \in S$ ,  $S$  is nonempty. Now  $S$  is a finite set. Thus, we can find a member of  $S$  of minimum length. Let  $T_1$  be a nontrivial closed subwalk ( $u = v_1, e_1, v_2, e_2, \dots, v_n = u$ ) of  $T$  of minimum length. Because  $T_1$  is of the minimum length,  $T_1$  cannot contain a nontrivial closed subwalk other than  $T_1$ . This implies that  $T_1$  has no repeated vertices except the vertex  $u$ . Hence,  $T_1$  is a cycle. ■

**DEFINITION 10.2.19** ► A collection of cycles  $C_1, C_2, C_3, \dots, C_n$  is called a **decomposition** of a circuit  $T$  into edge disjoint cycles if

- (i) for  $i \neq j$ ,  $C_i$  and  $C_j$  have no common edges;
- (ii) each  $C_i$  is a subgraph of  $T$ ;
- (iii) the set of the edges of  $T$  is the union of the sets of the edges of  $C_1, C_2, C_3, \dots, C_n$ .

#### EXAMPLE 10.2.20

Consider the graph in Figure 10.26. In this graph, we have the following circuit:

$$(v_4, e_5, v_4, e_6, v_5, e_7, v_6, e_8, v_3, e_3, v_2, e_{13}, v_9, e_{12}, v_8, e_{11}, v_8, e_{10}, v_7, e_9, v_3, e_4, v_4).$$

This circuit can be decomposed into edge disjoint cycles:  $(v_4, e_5, v_4)$ ,  $(v_4, e_6, v_5, e_7, v_6, e_8, v_3, e_4, v_4)$ ,  $(v_3, e_3, v_2, e_{13}, v_9, e_{12}, v_8, e_{11}, v_7, e_9, v_3)$ , and  $(v_8, e_{10}, v_8)$ .

This result is true for any circuit and is proved in the next theorem.

**Theorem 10.2.21:** Every circuit  $T$  has a decomposition into edge disjoint cycles.

**Proof:** We prove this theorem by induction on  $n$ , where  $n$  is the number of edges in  $T$ .

*Basis step:* If  $n = 1$ , then the result trivially follows.

*Inductive hypothesis:* Assume that any circuit  $T$  with  $n$  or fewer edges has a decomposition into edge disjoint cycles.

*Inductive step:* Consider a circuit  $T$  with  $n + 1$  edges. By Theorem 10.2.18,  $T$  contains a cycle  $C$ . If  $T = C$ , then the result is true. Suppose  $T \neq C$ . Now we delete  $C$  from  $T$ . Then  $T - C$  is a circuit such that the number of edges in  $T - C$  is less than or equal to  $n$ . Hence, by the induction hypothesis  $T - C$  has a decomposition into edge disjoint cycles  $C_1, C_2, C_3, \dots, C_n$ . Then  $T$  has a decomposition into edge disjoint cycles  $C_1, C_2, C_3, \dots, C_n, C$ . ■

**DEFINITION 10.2.22** ► Let  $G$  be a graph. A vertex  $u$  is said to be **connected** to a vertex  $v$  if there is a  $u - v$  walk in graph  $G$ .

**DEFINITION 10.2.23** ► A graph  $G$  is called a **connected graph** if for any two vertices  $u, v$  of  $G$  there is a  $u - v$  walk in  $G$ , otherwise the graph is called a **disconnected graph**.

We can show that a graph  $G$  is a connected graph if and only if for any two vertices  $u, v$  of  $G$  there is a  $u - v$  path in  $G$ . We assume that a graph with only a single vertex and no edges is also a connected graph.

We now define a relation  $R$  on the vertex set  $V$  of a graph  $G$  by

$$R = \{(u, v) \in V \times V \mid \text{there exists a } u - v \text{ walk in } G\}.$$

Because the trivial walk  $(u)$  is a  $u - u$  walk for any vertex in  $G$ , the relation  $R$  is reflexive. Suppose there is a  $u - v$  walk  $(u = v_1, e_1, v_2, e_2, \dots, v_n = v)$ . Then  $(v = v_n, e_{n-1}, \dots, e_2, v_2, e_1, v_1 = u)$  is a  $v - u$  walk. Hence, the relation  $R$  is symmetric.

Now suppose that there is a  $u - v$  walk

$$(u = v_1, e_1, v_2, e_2, \dots, v_n = v)$$

from a vertex  $u$  to a vertex  $v$  and a  $v - w$  walk  $(v = u_1, f_1, u_2, f_2, \dots, u_m = w)$  from a vertex  $v$  to a vertex  $w$ . Then clearly

$$(u = v_1, e_1, v_2, e_2, \dots, v_n = v = u_1, f_1, u_2, f_2, \dots, u_m = w)$$

is a walk from a vertex  $u$  to a vertex  $w$ . Thus, the relation  $R$  is transitive. Consequently, the relation  $R$  is an equivalence relation on the vertex set  $V$ . Let  $V_1$  be an equivalence class of  $R$  and  $E_1$  be the set of edges joining the vertices in  $V_1$  in the graph  $G$ . Then  $G_1 = (V_1, E_1)$  is a subgraph of  $G$ . In this subgraph, we see that any two vertices are connected. This subgraph is called a *component* of  $G$ .

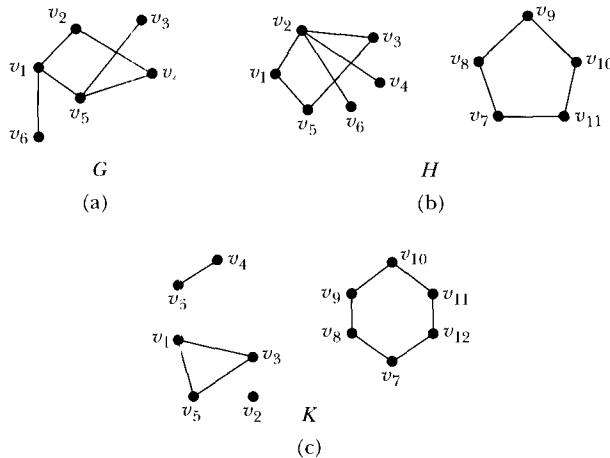
---

**DEFINITION 10.2.24** ► A subgraph  $H$  of a graph  $G$  is called a **component** of  $G$  if

- (i) any two vertices of  $H$  are connected in  $H$ , and
- (ii)  $H$  is not properly contained in any connected subgraph of  $G$ .

**EXAMPLE 10.2.25**

Consider the graphs in Figure 10.27.



**FIGURE 10.27** Various graphs

Graph  $G$ , in Figure 10.27(a), has only one component, which is  $G$  itself. Graph  $H$ , in Figure 10.27(b), has two components with vertices  $\{v_1, v_2, v_3, v_4, v_5, v_6\}$  and  $\{v_7, v_8, v_9, v_{10}, v_{11}\}$ . Graph  $K$ , in Figure 10.27(c), has four components with vertices  $\{v_1, v_3, v_5\}$ ,  $\{v_2\}$ ,  $\{v_4, v_6\}$ , and  $\{v_7, v_8, v_9, v_{10}, v_{11}, v_{12}\}$ .

From Definition 10.2.24, it follows that any component of a graph is a connected subgraph. Now every equivalence class of the equivalence relation  $R$ , defined earlier, gives a component of  $G$ . Hence, every graph can be partitioned into a finite number of components. It follows that a graph  $G$  is a connected graph if and only if  $G$  has only one component.

Graph  $G$ , in Figure 10.27(a), is a connected graph, but graph  $H$ , in Figure 10.27(b), is not a connected graph.

The following is a basic theorem of a connected graph.

**Theorem 10.2.26:** A connected graph with  $n$  vertices has at least  $n - 1$  edges.

**Proof:** We prove the result by induction on  $n$ .

*Basis step:* If  $n = 1$ , then the result is trivially true.

*Inductive hypothesis:* Assume that any connected graph with  $n$  vertices has at least  $n - 1$  edges.

*Inductive step:* Consider a connected graph  $G$  with  $n + 1$  vertices. Because  $G$  is a connected graph, the degree of each vertex of  $G$  is  $\geq 1$ . Suppose the degree of each vertex of  $G$  is  $\geq 2$ . Then the sum of the degrees of vertices of  $G$  is  $\geq 2(n + 1) > 2n$ . Thus, the number of edges of  $G$  is  $> n$ . Suppose now that  $G$  has a vertex  $v$  of degree 1. We construct a graph  $G_1$  by deleting the vertex  $v$  and the edge incident with  $v$ . The graph  $G_1$  is a connected graph with  $n$  vertices. By the induction hypothesis, the number of edges of  $G_1$  is at least  $n - 1$ . Therefore, the number of edges of  $G$  is at least  $n$ . Thus, the result is true for a graph with  $n + 1$  vertices.

Hence, by induction for any connected graph with  $n$  vertices the number of edges is at least  $n - 1$ . ■

---

**DEFINITION 10.2.27** ▶ Let  $G$  be a graph. The **distance** between two vertices  $u, v$  of  $G$ , written  $d(u, v)$ , is the length of a shortest path, if any exists, from  $u$  to  $v$ .

If  $G$  is a connected graph, then we can prove that

- (i)  $d(u, v) \geq 0$ , equality holds if and only if  $u = v$ ;
- (ii)  $d(u, v) = d(v, u)$ ;
- (iii)  $d(u, v) + d(v, w) \geq d(u, w)$  for all vertices  $u, v, w$  of  $G$ .

We conclude the section by proving two interesting theorems. Theorem 10.2.28 gives a characterization of the connected property of a graph by the degrees of vertices.

**Theorem 10.2.28:** Let  $G$  be a simple graph with at most  $2n$  vertices. If the degree of each vertex is at least  $n$ , then the graph is connected.

**Proof:** Suppose that  $G$  is not connected. Then  $G$  can be partitioned into components  $C_1, C_2, \dots, C_m$ ,  $m \geq 2$ . Because the degree of each vertex is at least  $n$  and

the graph is simple, we find that each vertex has at least  $n$  adjacent vertices. Then each component contains at least  $n + 1$  vertices. This implies that the number of vertices of the graph  $G$  is at least  $m(n + 1) \geq 2(n + 1) > 2n$ . This contradiction implies that the given graph is connected. ■

We are now in a position to solve a problem stated at the beginning of the section.

**EXAMPLE 10.2.29**

In this example, we solve the first problem posed at the beginning of this section. Suppose there are 100 small towns in a country. From each town there is a direct bus route to at least 50 towns. Is it possible to go from one town to any other town by bus possibly changing from one bus and then taking another bus to another town?

Consider a graph  $G = (V, E)$  with 100 vertices. The 100 small towns are identified with these vertices. Two vertices  $u$  and  $v$  are to be considered as adjacent vertices if and only if they are distinct and there is a direct bus route between cities  $u$  and  $v$ . Then the graph  $G = (V, E)$  is a simple graph with 100 vertices. Because from each town there is a direct bus route to at least 50 towns, the degree of each vertex is at least 50. Therefore, from Theorem 10.2.28, it follows that graph  $G$  is connected. Thus, there is a path between any two vertices. Hence, it is possible to go from one town to any other town by bus possibly changing from one bus to another bus to go to another town.

As in Example 10.2.29, we can solve the second problem posed at the beginning of this section: There are 200 telephone exchanges in a city. Suppose each telephone exchange has direct lines to 100 other exchanges. Is it always possible to make a call between any two exchanges perhaps through other exchanges?

The following theorem gives a characterization of a bipartite graph.

**Theorem 10.2.30:** A graph is bipartite if and only if it does not contain any cycle of odd length.

**Proof:** Let  $G = (V, E)$  be a bipartite graph with bipartition  $V = V_1 \cup V_2$ . Now each edge of  $G$  is incident with one vertex in  $V_1$  and one vertex in  $V_2$ . Let  $(v_1, e_1, v_2, e_2, \dots, v_k, e_k, v_1)$  be a cycle in  $G$ . Because  $v_i$  and  $v_{i+1}$  are end vertices of  $e_i$ , for  $i = 1, 2, \dots, k$  (assuming  $v_{k+1} = v_1$ ), it follows that for  $i = 1, 2, \dots, k$ , if  $v_i \in V_1$ , then  $v_{i+1} \in V_2$ . Suppose  $v_1 \in V_1$ . This implies that  $v_k \in V_2$ . Also it follows that  $v_i \in V_1$  if and only if  $i$  is odd. Now  $v_k \in V_2$ , which implies that  $k$  is even and hence the length of this cycle is even. This implies that the length of each cycle is even.

Conversely, let  $G$  be a graph such that  $G$  has no odd cycle. Suppose  $G$  is partitioned into components  $C_1, C_2, \dots, C_m$ ,  $m \geq 1$ . If we can show that each  $C_i$  is a bipartite subgraph, then  $G$  must be a bipartite graph. We may therefore assume that  $G$  is connected.

Let  $u$  be an arbitrary but fixed vertex of  $G$ . Define the subset  $V_1$  by

$$V_1 := \{v \in V \mid d(u, v) \text{ is even}\},$$

and define the subset  $V_2$  by

$$V_2 := \{w \in V \mid d(u, w) \text{ is odd}\}.$$

From our assumption that  $G$  is a connected graph, it follows that every vertex of  $G$  is either in  $V_1$  or in  $V_2$ . Then  $\{V_1, V_2\}$  is a partition of  $V$ . Because  $d(u, u) = 0$ ,

it follows that  $u \in V_1$ . Let  $v$  be an adjacent vertex of  $u$ . Then  $d(u, v) = 1$ . Hence,  $v \in V_2$ .

Suppose there are two distinct vertices  $v$  and  $w$  in  $V_1$  and suppose there exists an edge  $e$  with  $v, w$  as end vertices. Then there is a walk from  $u$  to  $v$  in  $G$  and hence there is a shortest path, say  $P_1$ , from  $u$  to  $v$ . Similarly, we have a shortest path,  $P_2$ , from  $u$  to  $w$ . Because  $v$  and  $w$  belong to  $V_1$ , these two shortest paths are of even length. Paths  $P_1$  and  $P_2$  may have several vertices and edges in common.

Now starting from  $u$ , let  $x$  be the last vertex common to both  $P_1$  and  $P_2$  (see Figure 10.28).

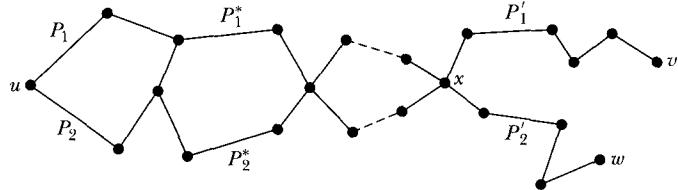


FIGURE 10.28 Paths  $P_1$ ,  $P_2$ ,  $P_1^*$ ,  $P_2^*$ ,  $P_1'$ , and  $P_2'$

Let  $P_1^*$  be the section of the path of  $P_1$  from  $u$  to  $x$  and let  $P_2^*$  be the section of the path of  $P_2$  from  $u$  to  $x$ . Because  $P_1$  and  $P_2$  are the shortest paths,  $P_1^*$  and  $P_2^*$  have equal lengths. Thus, the lengths of paths  $P_1^*$  and  $P_2^*$  are either both even or both odd. Let  $P_1'$  be the part of  $P_1$  from  $x$  to  $v$  and let  $P_2'$  be the part of  $P_2$  from  $x$  to  $w$ . It follows that the lengths  $P_1'$  and  $P_2'$  are either both even or both odd. Now the walk  $P_1'$  followed by  $e$  followed by  $P_2'$  forms a closed walk  $C$  from  $x$  to  $x$ . Moreover,  $C$  does not contain any repetitions of the vertices. Hence,  $C$  is a cycle. Because the lengths of paths  $P_1'$  and  $P_2'$  are either both even or both odd, cycle  $C$  must be of odd length, which is a contradiction. Thus,  $v$  and  $w$  cannot both be in  $V_1$ . Similarly, we can show that  $v$  and  $w$  cannot both belong to  $V_2$ . Hence, each edge of  $G$  connects one vertex from  $V_1$  with one vertex from  $V_2$ . Consequently,  $G$  is bipartite. ■

## Matching

Suppose the computer science department has five teachers,  $A_1, A_2, A_3, A_4$ , and  $A_5$ . In the spring semester, five courses,  $C_1, C_2, C_3, C_4$ , and  $C_5$ , are to be offered. Each teacher is qualified to teach one or more courses. The following table specifies the courses a teacher is qualified to teach.

| Teacher | Courses              |
|---------|----------------------|
| $A_1$   | $C_1, C_3$           |
| $A_2$   | $C_3, C_4$           |
| $A_3$   | $C_1, C_2$           |
| $A_4$   | $C_2, C_3, C_4, C_5$ |
| $A_5$   | $C_4, C_5$           |

A teacher is allowed to teach only one course in the spring semester. The chair of the department would like to assign the courses so that no course is assigned to more than one teacher. Can all these teachers be assigned a course they are qualified to teach?

We construct a graph  $G$  with vertices  $A_1, A_2, A_3, A_4, A_5, C_1, C_2, C_3, C_4, C_5$ . We divide the vertex set into two subsets,  $V_1 = \{A_1, A_2, A_3, A_4, A_5\}$  and  $V_2 = \{C_1, C_2, C_3, C_4, C_5\}$ .

$C_4, C_5\}$ . Then  $V_1 \cup V_2$  is a bipartition of the vertex set of  $G$ . An edge joins the vertices  $A_i$  and  $C_j$  if and only if  $A_i$  is qualified to teach the course  $C_j$ . Moreover, these are the only edges in this graph. It follows that the graph  $G$  is bipartite (see Figure 10.29).

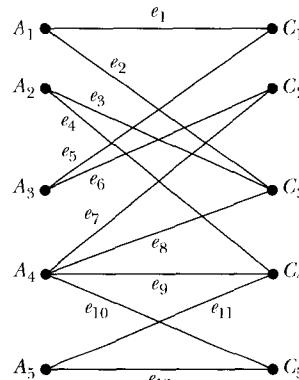


FIGURE 10.29 A graph

The course assignment problem can be solved if we can find a subset  $M$  of the set  $E$  of edges such that

- (i) no two distinct members of  $M$  have a common end vertex, and
- (ii) for each  $A_i$  in  $V_1$  there exists an edge in  $M$  with  $A_i$  as one end vertex.

Let us take  $M = \{e_2, e_4, e_5, e_7, e_{12}\}$ . This set  $M$  matches  $A_1$  with  $C_3$ ,  $A_2$  with  $C_4$ ,  $A_3$  with  $C_1$ ,  $A_4$  with  $C_2$ , and  $A_5$  with  $C_5$ . Such a set  $M$  is called a matching for the graph  $G$ .

Let us consider a similar problem. Suppose three persons,  $A_1, A_2$ , and  $A_3$ , apply for four jobs,  $P_1, P_2, P_3$ , and  $P_4$ , in a company. Each person is qualified for one or more jobs. The following information is obtained from the applications of the candidates.

| Applicant | Jobs qualified for |
|-----------|--------------------|
| $A_1$     | $P_2, P_4$         |
| $A_2$     | $P_1, P_2, P_3$    |
| $A_3$     | $P_1, P_3, P_4$    |

An applicant is required to fill only one position and a position cannot be filled by more than one candidate. Can each candidate be hired to fill one of the positions for which he or she is qualified?

We construct a graph  $G$  with vertices  $A_1, A_2, A_3, P_1, P_2, P_3, P_4$ . We divide the vertex set into two subsets,  $V_1 = \{A_1, A_2, A_3\}$  and  $V_2 = \{P_1, P_2, P_3, P_4\}$ . Then  $V_1 \cup V_2$  is a bipartition of the vertex set. An edge joins  $A_i$  to  $P_j$  if and only if  $A_i$  is qualified for the post  $P_j$ . Moreover, these are the only edges in this graph (see Figure 10.30). Notice that the graph in Figure 10.30 is a bipartite graph.

Let  $M = \{e_1, e_3, e_7\}$ . Then  $M$  matches  $A_1$  with  $P_2$ ,  $A_2$  with  $P_1$ , and  $A_3$  with  $P_3$ . Thus, each candidate can be hired to fill one of the positions for which he or she is qualified.

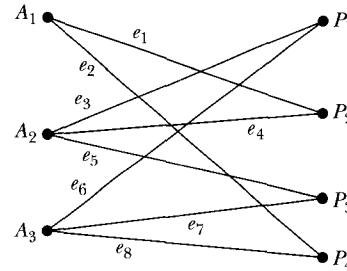


FIGURE 10.30 A graph

**DEFINITION 10.2.31** ▶ Let  $G = (V, E)$  be a graph. A subset  $M$  of  $E$  is called a **matching** in  $G$  if no two distinct members of  $M$  have a common end vertex.

Let  $M$  be a matching in graph  $G$ . If  $e$  is an edge in  $G$  with end vertices  $u$  and  $v$ , then we say that  $M$  matches  $u$  and  $v$ .

### EXAMPLE 10.2.32

Consider the graph in Figure 10.31.

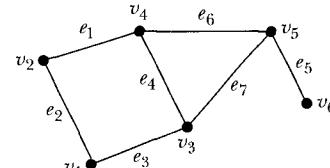


FIGURE 10.31 A graph

This is a graph with the set  $E$  of edges  $e_1, e_2, e_3, e_4, e_5, e_6$ , and  $e_7$ . Let  $M = \{e_1, e_3, e_5\}$ . Then no two edges in  $M$  have a common end vertex. Hence,  $M$  is a matching in  $G$ . In this matching,  $M$  matches  $v_1$  and  $v_3$ , but  $M$  does not match  $v_2$  and  $v_4$ .

**DEFINITION 10.2.33** ▶ Let  $G = (V, E)$  be a simple graph with a matching  $M$ .

- (i) A vertex  $v$  of  $G$  is said to be  **$M$ -saturated** if there exists an edge  $e$  in  $M$  with  $v$  as an end vertex.
- (ii) A vertex that is not  $M$ -saturated is called  **$M$ -unsaturated**.
- (iii) If every vertex of  $G$  is  $M$ -saturated, then  $M$  is called a **perfect matching**.
- (iv) A matching  $M$  is called **maximum** if there is no matching  $M_1$  in  $G$  such that the number of edges in  $M_1$  is more than those in  $M$ .

The matching  $M$  in  $G$  of Example 10.2.32 is a perfect matching, because each of the vertices of  $G$  is an end vertex of some member of  $M$ . We see that  $v_2, v_4$  are the end vertices of  $e_1$ ;  $v_1, v_3$  are the end vertices of  $e_3$ ; and  $v_5, v_6$  are the end vertices of  $e_5$ .

**REMARK 10.2.34** ▶ It follows that every perfect matching is a maximum matching.

In the graph in Figure 10.30, related to the job-assignment problem, we find that  $M = \{e_1, e_3, e_7\}$  is a matching such that it saturates the vertices of  $V_1$ . Note that  $M$  is also a maximum matching.

**EXAMPLE 10.2.35**

In this example, we consider the following problem. Suppose that five college students,  $A_1, A_2, A_3, A_4$ , and  $A_5$ , are members of five different committees,  $C_1, C_2, C_3, C_4$ , and  $C_5$ , as shown in the table.

| Student | Committees      |
|---------|-----------------|
| $A_1$   | $C_1, C_2$      |
| $A_2$   | $C_2, C_3$      |
| $A_3$   | $C_2, C_3$      |
| $A_4$   | $C_1, C_2, C_3$ |
| $A_5$   | $C_3, C_4, C_5$ |

Clearly, each of the students is a member of one or more committees. Each committee wants to send one representative to Africa. No student can represent more than one committee. Can all five students visit Africa? We will answer this question after stating Hall's Marriage Theorem.

To solve the types of problems described in this section, Philip Hall proved the theorem known as Hall's Marriage Theorem (1935), stated next. First, however, let us fix some notation.

**DEFINITION 10.2.36**

- Let  $G = (V, E)$  be a bipartite graph with vertex bipartition  $V = V_1 \cup V_2$ . Let  $A$  be a subset of  $V_1$ . Then  $N(A)$  denotes the subset of  $V_2$  consisting of all vertices in  $V_2$  that are adjacent to at least one vertex of  $A$ . The set  $N(A)$  is called the set of **neighbors** of  $A$ .

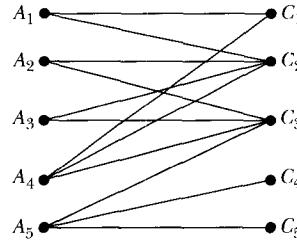
**Theorem 10.2.37: Hall's Marriage Theorem.** Let  $G = (V, E)$  be a bipartite graph with bipartition  $V = V_1 \cup V_2$ . Then there exists a matching  $M$  for  $G$  such that  $M$  saturates  $V_1$  if and only if for each subset  $A$  of  $V_1$ ,  $|A| \leq |N(A)|$ .

Let us now complete the solution of the problem of Example 10.2.35. Let us draw a graph describing the following information.

| Student | Committees      |
|---------|-----------------|
| $A_1$   | $C_1, C_2$      |
| $A_2$   | $C_2, C_3$      |
| $A_3$   | $C_2, C_3$      |
| $A_4$   | $C_1, C_2, C_3$ |
| $A_5$   | $C_3, C_4, C_5$ |

Let  $G$  be a graph with vertices  $A_1, A_2, A_3, A_4, A_5, C_1, C_2, C_3, C_4, C_5$ . We divide the vertex set into two subsets,  $V_1 = \{A_1, A_2, A_3, A_4, A_5\}$  and  $V_2 = \{C_1, C_2, C_3, C_4, C_5\}$ . Then  $V_1 \cup V_2$  is a bipartition of the vertex set. An edge joins  $A_i$  to  $C_j$  if and only if

$A_i$  is a member of  $C_j$ . Moreover, these are the only edges in this graph (see Figure 10.32).



**FIGURE 10.32** A graph representing students and committees

Notice that the graph in Figure 10.32 is a bipartite graph.

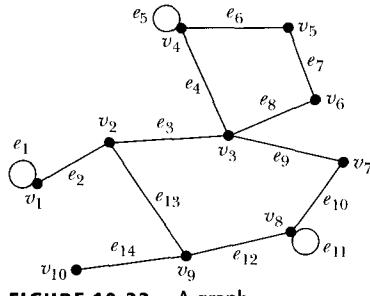
In this bipartite graph, the subset  $A = \{A_1, A_2, A_3, A_4\}$  of  $V_1$  is such that  $N(A) = \{C_1, C_2, C_3\}$ . Thus,  $|A| \not\leq |N(A)|$ . Hence, by Hall's Marriage Theorem, we find that  $G$  does not contain any matching that saturates  $V_1$ . Therefore, the answer to the question of sending all five students to Africa to represent the five committees is no.

---

**REMARK 10.2.38** ► In Chapter 11, we describe an algorithm to find a matching in a bipartite graph.

## WORKED-OUT EXERCISES

**Exercise 1:** Consider the graph in Figure 10.33.



**FIGURE 10.33** A graph

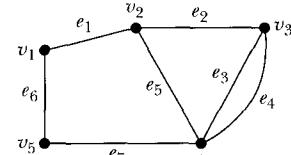
- Find an open walk of length 5. Is your walk a trail? Is your walk a path?
- Find a closed walk of length 5. Is your walk a circuit?
- Find a circuit of length 4. Is your circuit a cycle?

**Solution:**

- $(v_2, e_3, v_3, e_8, v_6, e_7, v_5, e_6, v_4, e_5, v_4)$  is a walk of length 5. Yes, it is a trail because it has no repeated edges. It is not a path because it has repeated vertices.
- $(v_3, e_8, v_6, e_7, v_5, e_6, v_4, e_5, v_4, e_4, v_3)$  is a closed walk of length 5. Yes, it is a circuit because it has no repeated edges.
- $(v_3, e_8, v_6, e_7, v_5, e_6, v_4, e_4, v_3)$  is a circuit of length 4. Yes, it is a cycle because it has no repeated vertices.

**Exercise 2:** Determine whether the following walks in the graph in Figure 10.34 are (i) a path, (ii) a trail, (iii) a closed walk, (iv) a circuit, or (v) a cycle.

- $(v_1, e_1, v_2)$
- $(v_2, e_2, v_3, e_3, v_4, e_4, v_3)$
- $(v_2, e_2, v_3, e_3, v_3, e_4, v_3, e_2, v_2)$
- $(v_4, e_7, v_5, e_6, v_1, e_1, v_2, e_2, v_3, e_3, v_4)$
- $(v_4, e_4, v_3, e_3, v_4, e_5, v_2, e_1, v_1, e_6, v_5, e_7, v_4)$ .



**FIGURE 10.34** A graph

**Solution:**

- A path
- A trail but not a path
- A closed walk but not a circuit
- A cycle
- A circuit but not a cycle

**Exercise 3:** Find the components of graph  $G$  in Figure 10.35.

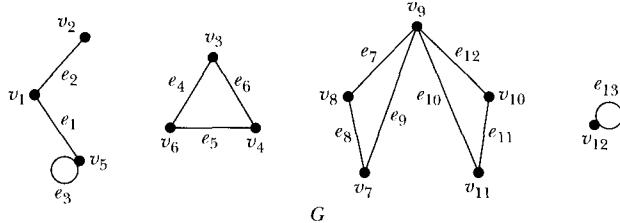


FIGURE 10.35 Graph  $G$

**Solution:** Graph  $G$  has four components:  $\{v_1, v_2, v_5\}$ ,  $\{v_3, v_4, v_6\}$ ,  $\{v_7, v_8, v_9, v_{10}, v_{11}\}$ , and  $\{v_{12}\}$ .

**Exercise 4:** Let  $G$  be a graph and  $u, v$  be two distinct vertices of  $G$ . If there is a trail from  $u$  to  $v$ , then prove that there is a path from  $u$  to  $v$ .

**Solution:** Let  $P = (u = v_1, e_1, v_2, e_2, \dots, v_n = v)$  be a trail. If this is not a path, then  $v_i = v_j$  for some  $1 \leq i < j \leq n$ . This shows that there is a closed walk  $Q$  from  $v_i$  to  $v_j$ . We reduce  $P$  to  $P - Q$ . Now  $P - Q$  is a new trail from  $u$  to  $v$ . If this trail is not a path, we repeat the above process. Because the number of closed trails in  $P$  is finite, we eventually obtain a path from  $u$  to  $v$ .

**Exercise 5:** If the degree of each vertex of a graph  $G$  is greater than or equal to 2, then show that  $G$  contains a cycle.

**Solution:** We choose a vertex  $u$  and an edge  $e_1$  with  $u$  as an end vertex and the other end vertex  $u_1$ . If  $u = u_1$ , then  $e_1$  is a loop at  $u$  and  $(u, e_1, u)$  is a cycle. Suppose  $u \neq u_1$ . Because  $\deg(u_1) \geq 2$ , there exists an adjacent vertex  $u_2$  of  $u_1$ . If  $u_2 = u$ , then there exist parallel edges  $e_2$  and  $e_1$  between  $u$  and  $u_1$  and then  $(u, e_1, u_1, e_2, u)$  is a cycle.

Assume that  $G$  is a simple graph. Choose a vertex  $u$  and an edge  $e_1$  with  $u$  as an end vertex and the other end vertex  $u_1$  different from  $u$ . Now  $\deg(u_1) \geq 2$ . Because  $G$  is a simple graph, we can choose an edge  $e_2$  with  $u_1$  as an end vertex and the other end vertex  $u_2$  different from  $u, u_1$ . Again  $\deg(u_2) \geq 2$ . Hence, we choose an edge  $e_3 \neq e_2$  with  $u_2$  as an end vertex and the other end vertex  $u_3$ . Because  $G$  is a simple graph,  $u_3$  is different from  $u_2, u_1$ . If  $u_3 = u$ , we obtain the cycle  $(u, e_1, u_1, e_2, u_2, e_3, u)$ . Now  $\deg(u_3) \geq 2$ . We repeat this process and choose an edge  $e_4$  different from  $e_3$  with  $u_3$  as an end vertex and the other end vertex  $u_4$ . Now  $u_4 \neq u_3$ . If  $u_4$  is one of  $u, u_1$ , we will get a cycle. If  $u_4$  is different from  $u, u_1$ , we repeat the process. Because the number of vertices is finite, repeating the above process we eventually find a sequence  $u, e_1, u_1, e_2, u_2, \dots, u_i, e_{i+1}, u_{i+1}$  such that  $u, u_1, u_2, \dots, u_i$  are distinct vertices,  $e_1, e_2, \dots, e_{i+1}$  are distinct edges, and  $u_{i+1} = u_j$  for some  $j$ , where  $1 \leq j < i$  and then we obtain a cycle  $u_j, e_{j+1}, u_{j+1}, u_{j+2}, e_{j+2}, \dots, u_i, e_i, u_{i+1} = u_j$ .

**Another Solution:** In graph  $G$ , we can find a path  $P$  such that length of  $P$  is greater than or equal to the length of any path  $Q$  in  $G$ . Because the number of vertices is finite, and the number of edges is finite, we can find such a path in  $G$ . Let

$$P : (u_0 = u, e_1, u_1, e_2, u_2, \dots, u_{n-1}, e_n, u_n = v).$$

Then  $P$  is a path from  $u$  to  $v$  of length  $n$ . Now  $\deg(v) \geq 2$ . Hence, there exists an edge  $e_{n+1}$  different from  $e_n$  with  $u_{n+1}$  as an end vertex. If  $u_{n+1} = v$ , then  $e_{n+1}$  is a loop at  $v$  and  $(v, e_{n+1}, v)$  is a cycle. Suppose  $u_{n+1} \neq v$ . If  $u_{n+1}$  is different from all the vertices on  $(u, e_1, u_1, e_2, u_2, \dots, u_{n-1}, e_n, u_n = v)$ , then we obtain a path  $(u, e_1, u_1, e_2, u_2, \dots, u_{n-1}, e_n, u_n = v, e_{n+1}, u_{n+1})$  of length  $n+1$ . This contradicts our assumption that the length of a maximal path  $P$  is  $n$ . Hence  $u_{n+1}$  is one of the vertices  $u_0, u_1, u_2, \dots, u_{n-1}$ . Let  $u_{n+1} = u_i$ ,  $0 \leq i \leq n-1$ . Then we obtain a cycle  $(v, e_{n+1}, u_{n+1} = u_i, e_{i+1}, u_{i+2}, \dots, u_{n-1}, e_n, u_n = v)$ .

**Exercise 6:** If a graph  $G$  contains exactly two vertices of odd degree, then show that there exists a path between these two vertices.

**Solution:** Let  $u$  and  $v$  be the odd degree vertices of  $G$ . Consider the component  $C$  to which  $u$  belongs. Now  $C$  is a connected subgraph of  $G$ . We know that the number of odd degree vertices in a graph is even. Because  $u$  and  $v$  are the only odd degree vertices in  $G$ , it follows that  $v \in C$ . Hence, there exists a path from  $u$  to  $v$ .

**Exercise 7:** Show that a simple graph with  $n$  vertices and  $m$  components can have at most  $\frac{(n-m)(n-m+1)}{2}$  edges.

**Solution:** Let  $G$  be a simple graph with  $m$  components  $C_1, C_2, C_3, \dots, C_m$ . Let  $n_i$  be the number of vertices in the component  $C_i$  for  $i = 1, 2, \dots, m$ . Then  $n_1 + n_2 + \dots + n_m = n$ . Because  $C_i$  is a simple graph, the maximum possible number of edges in  $C_i$  is  $\frac{n_i(n_i-1)}{2}$ . Hence, the number of edges in  $G$  is less than or equal to

$$\sum_{i=1}^m \frac{n_i(n_i-1)}{2} = \frac{1}{2} \sum_{i=1}^m n_i(n_i-1) = \frac{1}{2} \sum_{i=1}^m (n_i^2 - n_i).$$

Now for any  $m$  positive integers  $n_1, n_2, \dots, n_m$ , we have

$$(n_1 - 1) + (n_2 - 1) + \dots + (n_m - 1) = n_1 + n_2 + \dots + n_m - m.$$

Squaring both sides, we get

$$\begin{aligned} & (n_1 - 1)^2 + (n_2 - 1)^2 + \dots + (n_m - 1)^2 \\ & \quad + 2(n_1 - 1)(n_2 - 1) + 2(n_1 - 1)(n_3 - 1) + \dots \\ & = (n_1 + n_2 + \dots + n_m)^2 - 2(n_1 + n_2 + \dots + n_m)m + m^2. \end{aligned}$$

This implies that

$$\begin{aligned} & n_1^2 + n_2^2 + \dots + n_m^2 - 2(n_1 + n_2 + \dots + n_m) \\ & \quad + m + 2(n_1 - 1)(n_2 - 1) + 2(n_1 - 1)(n_3 - 1) + \dots \\ & = (n_1 + n_2 + \dots + n_m)^2 - 2(n_1 + n_2 + \dots + n_m)m + m^2. \end{aligned}$$

This implies that

$$\begin{aligned} & n_1^2 + n_2^2 + \dots + n_m^2 \leq (n_1 + n_2 + \dots + n_m)^2 \\ & \quad - 2(n_1 + n_2 + \dots + n_m)(m-1) + (m^2 - m). \end{aligned}$$

Hence, the number of edges in  $G$  is less than or equal to

$$\begin{aligned}
& \frac{1}{2} \sum_{i=1}^n (n_i^2 - n_i) \\
& \leq \frac{1}{2} (n_1 + n_2 + \cdots + n_m)^2 - (n_1 + n_2 + \cdots + n_m)(m-1) \\
& \quad + \frac{1}{2}(m^2 - m) - \frac{1}{2}n \\
& = \frac{1}{2}(n^2 - n) - n(m-1) + \frac{1}{2}(m^2 - m) \\
& = \frac{1}{2}(n-m)(n-m+1).
\end{aligned}$$

**Exercise 8:** Let  $G$  be a connected graph with at least two vertices. If the number of edges in  $G$  is less than the number of vertices, then prove that  $G$  has a vertex of degree 1.

**Solution:** Let  $G$  be a connected graph with  $n \geq 2$  vertices. Because graph  $G$  is connected,  $G$  has no isolated vertices. Suppose  $G$  has no vertex of degree 1. Then the degree of each vertex is at least 2. This implies that the sum of the degrees of vertices of  $G$  is at least  $2n$ . Hence, it follows that the number of edges is at least  $n$  (because the sum of the degrees of vertices in any graph is twice the number of edges), a contradiction. This contradiction implies that  $G$  contains at least one vertex of degree 1.

## SECTION REVIEW

### Key Terms

|                           |                  |                         |
|---------------------------|------------------|-------------------------|
| walk                      | path             | reduction of $P$ by $Q$ |
| initial vertex            | trivial walk     | decomposition           |
| terminal vertex           | trivial path     | connected               |
| directed walk             | trivial trail    | connected graph         |
| length of a walk          | nontrivial walk  | disconnected graph      |
| length of a directed walk | nontrivial path  | component               |
| $u - v$ walk              | nontrivial trail | distance                |
| $u - v$ directed walk     | circuit          | matching                |
| closed walk               | cycle            | $M$ -saturated          |
| closed directed walk      | $k$ -cycle       | $M$ -unsaturated        |
| open walk                 | even cycle       | perfect matching        |
| open directed walk        | odd cycle        | maximum matching        |
| trail                     | subwalk          | neighbors               |

### Some Key Definitions

- Let  $u$  and  $v$  be two vertices in a graph  $G$ . A walk from  $u$  to  $v$ , in  $G$ , is an alternating sequence of  $n+1$  vertices and  $n$  edges of  $G$  ( $u = v_1, e_1, v_2, e_2, v_3, e_3, \dots, v_{n-1}, e_{n-1}, v_n, e_n, v_{n+1} = v$ ) beginning with vertex  $u$ , called the initial vertex, and ending with vertex  $v$ , called the terminal vertex, in which  $v_i$  and  $v_{i+1}$  are the endpoints of the edge  $e_i$  for  $i = 1, 2, \dots, n$ .
- A walk with no repeated edges is called a trail and a walk with no repeated vertices except possibly the initial and terminal vertices is called a path.
- A nontrivial closed trail from a vertex  $u$  to itself is called a circuit.
- A circuit that does not contain any repetition of vertices is called a cycle.
- A cycle of length  $k$  is called a  $k$ -cycle. A cycle is called even (odd) if it contains an even (odd) number of edges.
- A graph  $G$  is called a connected graph if for any two vertices  $u, v$  of  $G$  there is a  $u - v$  walk in  $G$ ; otherwise the graph is called a disconnected graph.

7. A subgraph  $H$  of a graph  $G$  is called a component of  $G$  if
- any two vertices of  $H$  are connected in  $H$ , and
  - $H$  is not properly contained in any connected subgraph of  $G$ .
8. Let  $G = (V, E)$  be a graph. A subset  $M$  of  $E$  is called a matching in  $G$ , if no two distinct members of  $M$  have a common end vertex.
9. Let  $G = (V, E)$  be a bipartite graph with vertex bipartition  $V = V_1 \cup V_2$ . Let  $A$  be a subset of  $V_1$ . Then  $N(A)$  denotes the subset of  $V_2$  consisting of all vertices in  $V_2$  that are adjacent to at least one vertex of  $A$ . The set  $N(A)$  is called the set of neighbors of  $A$ .

## Some Key Results

- Let  $G$  be a graph and  $u, v$  be two vertices of  $G$ . If there is a walk from  $u$  to  $v$ , then there is a path from  $u$  to  $v$ .
- Every circuit contains a subwalk that is a cycle.
- Every circuit  $T$  has a decomposition into edge disjoint cycles.
- A connected graph with  $n$  vertices has at least  $n - 1$  edges.
- Let  $G$  be a simple graph with at most  $2n$  vertices. If the degree of each vertex in  $G$  is at least  $n$ , then  $G$  is connected.
- A graph is bipartite if and only if it does not contain any cycle of odd length.
- Let  $G = (V, E)$  be a bipartite graph with bipartition  $V = V_1 \cup V_2$ . Then there exists a matching  $M$  for  $G$  such that  $M$  saturates  $V_1$  if and only if for each subset  $A$  of  $V_1$ ,  $|A| \leq |N(A)|$ .

## EXERCISES

1. Consider the graph in Figure 10.36.
- Find an open walk of length 4. Is your walk a trail? Is your walk a path?
  - Find a closed walk of length 5. Is your walk a circuit?
  - Find a circuit of length 6. Is your circuit a cycle?
  - Find a 4-cycle.

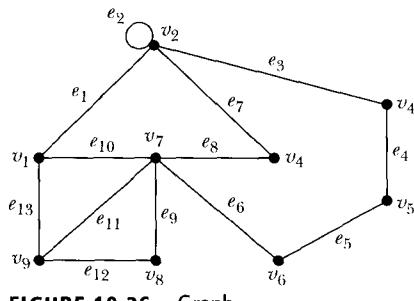


FIGURE 10.36 Graph

2. Determine whether each walk in (a)–(e) in the graph in Figure 10.37 is (i) a path, (ii) a trail, (iii) a closed walk, (iv) a circuit, or (v) a cycle.
- $(v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_5, e_5, v_4)$

- $(v_2, e_7, v_6, e_6, v_3, e_3, v_4, e_4, v_5, e_5, v_3)$
- $(v_6, e_7, v_2, e_2, v_3, e_3, v_4, e_4, v_5, e_5, v_3, e_6, v_6)$
- $(v_1, e_1, v_2, e_2, v_3, e_6, v_6, v_1)$
- $(v_2, e_2, v_3, e_6, v_6, e_8, v_1, e_1, v_2)$

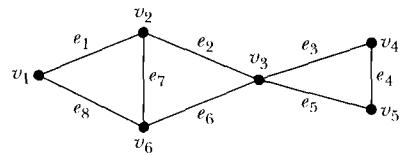


FIGURE 10.37 A graph

3. Find three subgraphs of the graph in Figure 10.38.

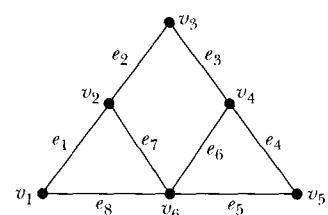


FIGURE 10.38 A graph

4. Find the components of the graph in Figure 10.39.

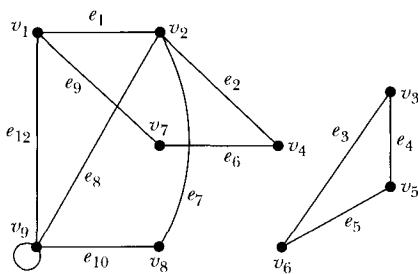


FIGURE 10.39 Graph

5. Prove that a connected graph is a circuit if the degree of each vertex is 2.  
 6. Prove that any cycle-free graph contains a vertex of degree 0 or 1.  
 7. Prove that a circuit either is a cycle or can be reduced to a cycle.  
 8. Suppose there are 90 small towns in a country. From each town there is a direct bus route to at least 50 towns. Is it possible to go from one town to any other town by bus possibly changing from one bus and then taking another bus to another town?  
 9. Suppose there are 200 telephone exchanges in a city. If each telephone exchange has direct lines to 110 other exchanges, is it always possible to make a call between any two exchanges perhaps through other exchanges?  
 10. Show that a simple graph  $G$  with  $n$  vertices is connected if  $G$  has more than  $\frac{(n-1)(n-2)}{2}$  edges.  
 11. Let  $G$  be a simple graph. If the degree of each vertex is at least  $n > 1$ , then show that  $G$  contains a circuit with at least  $n + 1$  edges.  
 12. Prove that for any graph  $G$ , either  $G$  or its complement,  $G'$ , is a connected graph.  
 13. Determine whether or not each of the graphs in Figure 10.40 is bipartite. Justify your answer.

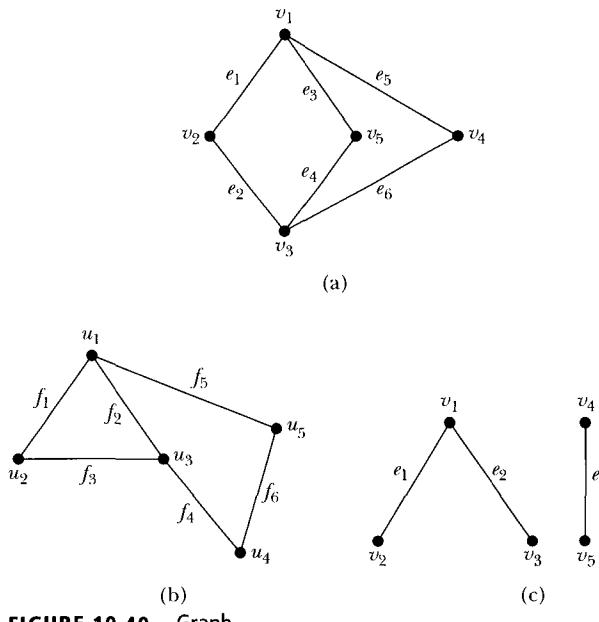


FIGURE 10.40 Graph

14. Prove that a simple graph with a cycle of length 3 cannot be a bipartite graph.

15. Find, if possible, a matching for the bipartite graph in Figure 10.41, which saturates the vertices  $v_1, v_2, v_3, v_4$ .

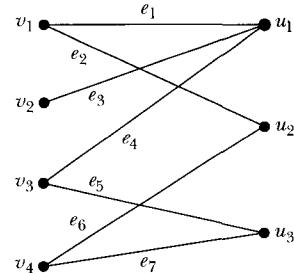


FIGURE 10.41 Graph

16. In the department of mathematics there are five teachers,  $A_1, A_2, A_3, A_4$ , and  $A_5$ . In the spring semester, six courses,  $C_1, C_2, C_3, C_4, C_5$ , and  $C_6$ , are to be offered. Each teacher is qualified to teach one or more courses. The department has the following information for each teacher.

| Teacher | Courses qualified for |
|---------|-----------------------|
| $A_1$   | $C_2, C_4$            |
| $A_2$   | $C_2, C_6$            |
| $A_3$   | $C_3, C_4$            |
| $A_4$   | $C_1, C_5, C_6$       |
| $A_5$   | $C_1, C_2, C_3$       |

Draw a bipartite graph displaying this information. Find a way in which each teacher can be assigned to teach a course or use Hall's Theorem to explain why no such way exists.

17. Four persons,  $A_1, A_2, A_3$ , and  $A_4$ , apply for five jobs,  $J_1, J_2, J_3, J_4$ , and  $J_5$ , in a company. Each person is qualified for one or more jobs. The following information is obtained from each candidate's application.

| Candidate | Jobs qualified for |
|-----------|--------------------|
| $A_1$     | $J_2, J_5$         |
| $A_2$     | $J_3, J_5$         |
| $A_3$     | $J_2, J_3, J_5$    |
| $A_4$     | $J_5$              |
| $A_5$     | $J_1, J_4, J_5$    |

Draw a bipartite graph describing this information. Either determine a way in which each candidate can be offered a job or use Hall's Theorem to explain why no such way exists.

## 10.3 MATRIX REPRESENTATION OF A GRAPH

To write programs that process and manipulate graphs, the graphs must be stored, that is, represented in computer memory. A graph can be represented (in computer memory) in several ways. In this section, we discuss how to describe a graph using adjacency matrices and incidence matrices, which in computer memory can be stored as a two-dimensional array.

### Adjacency Matrices

Let  $G$  be a graph with  $n$  vertices, where  $n > 0$ . Let  $V(G) = \{v_1, v_2, \dots, v_n\}$ . The **adjacency matrix**  $A_G$  with respect to the particular listing,  $v_1, v_2, \dots, v_n$ , of  $n$  vertices of  $G$  is an  $n \times n$  matrix  $[a_{ij}]$  such that the  $(i, j)$ th entry  $a_{ij}$  of  $A_G$  is the number of edges from  $v_i$  to  $v_j$ . That is,

$$a_{ij} = \text{the number of edges from } v_i \text{ to } v_j.$$

Because  $a_{ij}$  is the number of edges from  $v_i$  to  $v_j$ , the adjacency matrix  $A_G$  is a square matrix over the set of nonnegative integers.

---

**REMARK 10.3.1** ► If  $G$  is a digraph, then the adjacency matrix  $A_G$  with respect to the particular listing,  $v_1, v_2, \dots, v_n$ , of  $n$  vertices of  $G$  is an  $n \times n$  matrix  $[a_{ij}]$  such that the  $(i, j)$ th entry  $a_{ij}$  is the number of arcs from  $v_i$  to  $v_j$ .

Consider the graph in Figure 10.42.

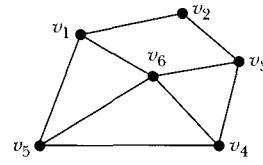


FIGURE 10.42 A graph

The vertices of the graph are listed as  $v_1, v_2, v_3, v_4, v_5$ , and  $v_6$ . The adjacency matrix of this graph with respect to this ordering of vertices is

$$A_G = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \quad (10.1)$$

Consider the graph in Figure 10.43.

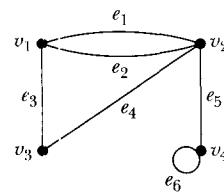


FIGURE 10.43 A graph

The vertices of the graph are listed as  $v_1, v_2, v_3$ , and  $v_4$ . The adjacency matrix of this graph with respect to this ordering of vertices is

$$A_G = \begin{bmatrix} 0 & 2 & 1 & 0 \\ 2 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

Sometimes for convenience we label the columns and rows by  $v_1, v_2, v_3, v_4, \dots$  as follows:

$$A_G = \begin{array}{c|cccc} & v_1 & v_2 & v_3 & v_4 \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{matrix} & \begin{bmatrix} 0 & 2 & 1 & 0 \\ 2 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \end{array} \quad (10.2)$$

Notice that the matrix  $A_G$  is a symmetric matrix, i.e.,  $a_{ij} = a_{ji}$ .

**REMARK 10.3.2** ▶ Note that if  $G$  is a digraph, then  $A_G$  need not be a symmetric matrix.

In the adjacency matrix, (10.1), of the graph of Figure 10.42, we find that all of the diagonal elements are zero. This is because this graph does not contain any loops. We also find that all of the entries of this matrix are either 0 or 1, because the graph has no parallel edges. In the adjacency matrix, (10.2), of the graph in Figure 10.43, we find that  $a_{44} = 1$ , because the vertex  $v_4$  has a loop. We also find that  $a_{12} = a_{21} = 2$ , because there are parallel edges between vertices  $v_1$  and  $v_2$ .

The adjacency matrix  $A_G$  of a graph has the following properties.

1. If  $G$  does not contain any loops and parallel edges, then each element of  $A_G$  is either 0 or 1.
2. If  $G$  does not contain any loops, then all of the diagonal elements of  $A_G$  are 0.

Suppose now that a symmetric square matrix  $A = [a_{ij}]$  of order  $n \times n$  over the set of nonnegative integers is given. We show that there exists a graph  $G$  such that  $A_G = [a_{ij}]$ . Let us illustrate this by the following example.

### EXAMPLE 10.3.3

Let  $A$  denote the  $5 \times 5$  symmetric matrix

$$\begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 2 & 0 & 1 \\ 1 & 2 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}.$$

We construct a graph  $G$  such that  $A_G = A$ . For this we denote the rows by  $v_1, v_2, v_3, v_4$ , and  $v_5$  and the columns by  $v_1, v_2, v_3, v_4$ , and  $v_5$ . Now we draw a graph with vertices  $v_1, v_2, v_3, v_4$ , and  $v_5$ . Because the (1, 1) and (4, 4) entries are 1 and all other

diagonal elements are 0, we draw loops only at vertices  $v_1$  and  $v_4$ . Now

(1, 2)th element = (2, 1)th element = 0  $\Rightarrow$  there is no edge between  $v_1$  and  $v_2$ .

(1, 3)th element = (3, 1)th element = 1  $\Rightarrow$  there is an edge between  $v_1$  and  $v_3$ .

(1, 4)th element = (4, 1)th element = 1  $\Rightarrow$  there is an edge between  $v_1$  and  $v_4$ .

(1, 5)th element = (5, 1)th element = 0  $\Rightarrow$  there is no edge between  $v_1$  and  $v_5$ .

(2, 3)th element = (3, 2)th element = 2  $\Rightarrow$  there are two parallel edges between  $v_2$  and  $v_3$ .

(2, 4)th element = (4, 2)th element = 0  $\Rightarrow$  there is no edge between  $v_2$  and  $v_4$ .

(2, 5)th element = (5, 2)th element = 1  $\Rightarrow$  there is an edge between  $v_2$  and  $v_5$ .

(3, 4)th element = (4, 3)th element = 0  $\Rightarrow$  there is no edge between  $v_3$  and  $v_4$ .

(3, 5)th element = (5, 3)th element = 0  $\Rightarrow$  there is no edge between  $v_3$  and  $v_5$ .

(4, 5)th element = (5, 4)th element = 1  $\Rightarrow$  there is an edge between  $v_4$  and  $v_5$ .

Thus, we obtain graph  $G$ , shown in Figure 10.44, such that  $A_G = A$ .

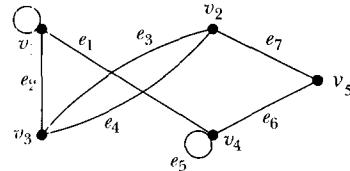


FIGURE 10.44 Graph  $G$

We now show how to draw a digraph for a given matrix.

#### EXAMPLE 10.3.4

Consider the following matrix.

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 2 \\ 1 & 0 & 0 \end{bmatrix}$$

We construct a digraph  $G$  such that  $A_G = A$ . For this we denote the rows by  $v_1$ ,  $v_2$ , and  $v_3$  and the columns by  $v_1$ ,  $v_2$ , and  $v_3$ . Now we draw a digraph with vertices  $v_1$ ,  $v_2$ , and  $v_3$ .

From the first row we find that there is an arc from  $v_1$  to  $v_1$ , an arc from  $v_1$  to  $v_2$ , and an arc from  $v_1$  to  $v_3$ .

From the second row we find that there is an arc from  $v_2$  to  $v_1$ , two arcs from  $v_2$  to  $v_3$ , and no arcs from  $v_2$  to  $v_2$ . From the third row we find that there is an arc from  $v_3$  to  $v_1$ . Thus, we obtain the digraph in Figure 10.45.

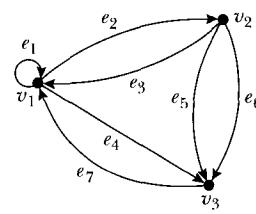


FIGURE 10.45 A graph

Consider graph  $G$  in Figure 10.43 and the matrix

$$A_G = [a_{ij}] = \begin{bmatrix} 0 & 2 & 1 & 0 \\ 2 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

of graph  $G$ . We can determine various properties of the graph in Figure 10.43 using matrix  $A_G$ .

Now in matrix  $A_G$ ,

$a_{12} = 2 \Rightarrow$  there exist two edges  $e_1, e_2$  from  $v_1$  to  $v_2$

$\Rightarrow$  there exist two walks  $(v_1, e_1, v_2)$  and  $(v_1, e_2, v_2)$  from  $v_1$  to  $v_2$  of length 1

$a_{13} = 1 \Rightarrow$  there exists one edge  $e_3$  from  $v_1$  to  $v_3$

$\Rightarrow$  there exists one walk  $(v_1, e_3, v_3)$  from  $v_1$  to  $v_3$  of length 1

$a_{14} = 0 \Rightarrow$  there exists no edges from  $v_1$  to  $v_4$

$\Rightarrow$  there exists no walk from  $v_1$  to  $v_4$  of length 1

Similar results hold for other entries.

Let us now find the number of different walks of length 2 from one vertex to another vertex. For example, let us find the number of walks of length 2 from  $v_1$  to  $v_2$ .

From the graph we find that  $(v_1, e_3, v_3, e_4, v_2)$  is the only walk of length 2 from  $v_1$  to  $v_2$ . Let us compute

$$(A_G)^2 = \begin{bmatrix} 0 & 2 & 1 & 0 \\ 2 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 2 & 1 & 0 \\ 2 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 5 & 1 & 2 & 2 \\ 1 & 6 & 2 & 1 \\ 2 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 \end{bmatrix}.$$

Let us write  $(A_G)^2 = [b_{ij}]$ . In this matrix, we have

$b_{12} = 1 =$  number of walks of length 2 from  $v_1$  to  $v_2$  in graph  $G$ .

Now choose any other  $b_{ij}$ , say  $b_{13}$ . Now  $b_{13} = 2$ . We expect two and only two walks of length 2 from  $v_1$  to  $v_3$  in graph  $G$ . In graph  $G$ , we see that

$$(v_1, e_2, v_2, e_4, v_3) \quad \text{and} \quad (v_1, e_1, v_2, e_4, v_3)$$

are the only two walks of length 2 from  $v_1$  to  $v_3$ .

Now choose  $b_{11}$ , which is 5. In graph  $G$  we find that  $(v_1, e_2, v_2, e_1, v_1)$ ,  $(v_1, e_1, v_2, e_2, v_1)$ ,  $(v_1, e_1, v_2, e_1, v_1)$ ,  $(v_1, e_2, v_2, e_2, v_1)$ , and  $(v_1, e_3, v_3, e_3, v_1)$  are the only five distinct walks of length 2 from  $v_1$  to  $v_1$ .

The preceding discussion suggests that the entry  $b_{ij}$  indicates the number of walks of length 2 from  $v_i$  to  $v_j$ . In a similar manner, the entries in  $(A_G)^k$  would indicate the number of walks between two vertices. In general, we have the following theorem.

**Theorem 10.3.5:** Let  $G$  be a graph with  $n$  vertices,  $v_1, v_2, \dots, v_n$ , and let  $A = [a_{ij}]$  denote the adjacency matrix with respect to this ordering of the vertices of  $G$ . For any positive integer  $k$ ,  $A^k = [b_{ij}]$  denotes the matrix multiplication of  $k$  copies of  $A$ . Then  $b_{ij}$  denotes the number of distinct walks of length  $k$  from vertex  $v_i$  to vertex  $v_j$  in graph  $G$ .

**Proof:** We prove the result by induction on  $k$ .

*Basis step:* Let  $k = 1$ . Then

$$\begin{aligned} b_{ij} &= a_{ij} \\ &= \text{the number of edges from } v_i \text{ to } v_j \\ &= \text{the number of distinct walks of length 1} \\ &\quad \text{from the vertex } v_i \text{ to the vertex } v_j. \end{aligned}$$

Thus, the theorem is true for  $k = 1$ .

*Inductive hypothesis:* Assume that the theorem is true for  $k - 1$ , where  $k$  is a positive integer greater than 1.

*Inductive step:* We prove the theorem for  $A^k = [b_{ij}]$ .

Now  $A^k = A^{k-1} \cdot A$ .

Let  $A^{k-1} = [c_{ij}]$ . By the inductive hypothesis, we find that  $c_{ij} =$  the number of distinct walks of length  $k - 1$  from vertex  $v_i$  to vertex  $v_j$  in graph  $G$ . Now

$$[b_{ij}] := A^k = A^{k-1} \cdot A = [c_{ij}] \cdot [a_{ij}]$$

implies that

$$b_{ij} = c_{i1}a_{1j} + c_{i2}a_{2j} + c_{i3}a_{3j} + \cdots + c_{ik}a_{kj}.$$

Now  $c_{ir}$  = the number of distinct walks of length  $k - 1$  from vertex  $v_i$  to vertex  $v_r$  in graph  $G$  and  $a_{rj}$  denotes the number of distinct walks of length 1 from vertex  $v_r$  to vertex  $v_j$  in graph  $G$  for  $r = 1, 2, \dots, k$ . Then  $c_{ir}a_{rj}$  denotes the number of distinct walks of length  $k$  from vertex  $v_i$  to vertex  $v_j$  through vertex  $v_r$  in graph  $G$  for  $r = 1, 2, \dots, k$ .

Again, any  $v_i$  to  $v_j$  walk of length  $k$  must be of the form

$$(v_i, e_{i1}, \dots, v_r, e_{ir}, v_j),$$

such that  $(v_i, e_{i1}, \dots, v_r)$  is of length  $k - 1$  and  $(v_r, e_{ir}, v_j)$  is of length 1. Because  $c_{ir}$  denotes the number of distinct walks of the form  $(v_i, e_{i1}, \dots, v_r)$  of length  $k - 1$  and  $a_{rj}$  denotes the number of distinct walks of the form  $(v_r, e_{ir}, v_j)$ , it therefore follows that

$$\begin{aligned} b_{ij} &= c_{i1}a_{1j} + c_{i2}a_{2j} + c_{i3}a_{3j} + \cdots + c_{ik}a_{kj} \\ &= \text{the total number of distinct walks of length } k \\ &\quad \text{from the vertex } v_i \text{ to the vertex } v_j. \end{aligned}$$

Thus, the result is true for  $k$ . Hence, by induction the theorem follows. ■

## Incidence Matrices

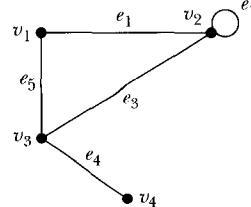
---

**DEFINITION 10.3.6** ▶ Let  $G$  be a graph with  $n$  vertices,  $v_1, v_2, \dots, v_n$ , where  $n > 0$  and  $m$  edges  $e_1, e_2, \dots, e_m$ . The **incidence matrix**  $I_G$  with respect to the ordering  $v_1, v_2, \dots, v_n$  of  $n$  vertices and  $m$  edges  $e_1, e_2, \dots, e_m$  is an  $n \times m$  matrix  $[a_{ij}]$  such that

$$a_{ij} = \begin{cases} 0 & \text{if } v_i \text{ is not an end vertex of } e_j, \\ 1 & \text{if } v_i \text{ is an end vertex of } e_j, \text{ but } e_j \text{ is not a loop,} \\ 2 & \text{if } e_j \text{ is a loop at } v_i. \end{cases}$$

**EXAMPLE 10.3.7**

Consider the graph in Figure 10.46.



**FIGURE 10.46** A graph

The vertices of this graph  $G$  are  $v_1, v_2, v_3$ , and  $v_4$  and the edges are  $e_1, e_2, e_3, e_4$ , and  $e_5$ . For incidence matrices we consider this ordering of vertices and edges. For incidence matrices we label the rows by  $v_1, v_2, v_3$ , and  $v_4$  and the columns by  $e_1, e_2, e_3, e_4$ , and  $e_5$ . Then the incidence matrix  $I_G$  with respect to the above ordering of vertices and edges is the following  $4 \times 5$  matrix.

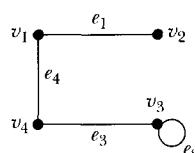
$$I_G = \begin{bmatrix} e_1 & e_2 & e_3 & e_4 & e_5 \\ v_1 & 1 & 0 & 0 & 0 & 1 \\ v_2 & 1 & 2 & 1 & 0 & 0 \\ v_3 & 0 & 0 & 1 & 1 & 1 \\ v_4 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Notice that the sum of the  $i$ th row is the degree of  $v_i$ . This is true for any incidence matrix.



## WORKED-OUT EXERCISES

**Exercise 1:** Find the adjacency matrices of the graph in Figure 10.47 with respect to the listing  $v_1, v_2, v_3$ , and  $v_4$  of the vertices.



**FIGURE 10.47** A graph

**Solution:** The vertices of the graph are listed as  $v_1, v_2, v_3$ , and  $v_4$ . The adjacency matrix of this graph with respect to this ordering of vertices is

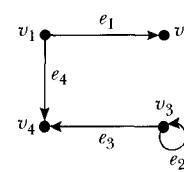
$$A_G = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

**Exercise 2:** Find the incidence matrix of the graph of Worked-Out Exercise 1 with respect to the listing  $v_1, v_2, v_3$ , and  $v_4$  of the vertices and the listing  $e_1, e_2, e_3$ , and  $e_4$  of the edges.

**Solution:** The vertices of the graph are listed as  $v_1, v_2, v_3$ , and  $v_4$  and the listing of the edges is  $e_1, e_2, e_3$ , and  $e_4$ . The incidence matrix of this graph with respect to this ordering of vertices and edges is

$$I_G = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

**Exercise 3:** Find the adjacency matrices of the digraph in Figure 10.48 with respect to the listing  $v_1, v_2, v_3$ , and  $v_4$  of the vertices.



**FIGURE 10.48** A graph

**Solution:** The vertices of the digraph are listed as  $v_1, v_2, v_3$ , and  $v_4$ . The adjacency matrix of this digraph with respect to

this ordering of vertices is

$$A_G = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

**Exercise 4:** Draw the graph of  $G$  represented by the given adjacency matrix.

$$A_G = \begin{bmatrix} 0 & 2 & 2 & 0 \\ 2 & 0 & 0 & 1 \\ 2 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

**Solution:** We construct a graph  $G$  such that  $A_G = A$ . For this we denote the rows by  $v_1, v_2, v_3$ , and  $v_4$  and the columns by  $v_1, v_2, v_3$ , and  $v_4$ . Now we draw a graph with vertices  $v_1, v_2, v_3$ , and  $v_4$  (see Figure 10.49).

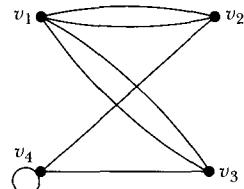


FIGURE 10.49 A graph

**Exercise 5:** Find the number of walks of length 2 from  $v_1$  to  $v_2$  and from  $v_2$  to  $v_3$  in the graph in Figure 10.50 and write these walks.

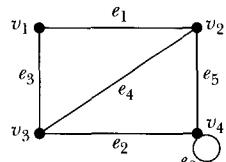


FIGURE 10.50 A graph

**Solution:** The vertices of the graph are listed as  $v_1, v_2, v_3$ , and  $v_4$ . The adjacency matrix of this graph with respect to this ordering of vertices is

$$A_G = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

$$(A_G)^2 = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 & 2 \\ 1 & 3 & 2 & 2 \\ 1 & 2 & 3 & 2 \\ 2 & 2 & 2 & 3 \end{bmatrix}.$$

Let us write  $(A_G)^2 = (b_{ij})$ . In this matrix, we find that

$$b_{12} = 1 = \text{number of walks of length 2 from } v_1 \text{ to } v_2 \text{ in graph } G.$$

In graph  $G$  we find that

$$(v_1, e_3, v_3, e_4, v_2)$$

is the only walk of length 2 from  $v_1$  to  $v_2$ .

Again,

$$b_{23} = 2 = \text{number of walks of length 2 from } v_2 \text{ to } v_3 \text{ in graph } G.$$

In graph  $G$ , we find that

$$(v_2, e_1, v_1, e_3, v_3) \text{ and } (v_2, e_5, v_4, e_2, v_3)$$

are the only two walks of length 2 from  $v_2$  to  $v_3$ .

## SECTION REVIEW

### Key Terms

adjacency matrix

incidence matrix

### Key Definition

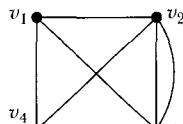
- Let  $G$  be a graph with  $n$  vertices, where  $n > 0$ . Let  $V(G) = \{v_1, v_2, \dots, v_n\}$ . The adjacency matrix  $A_G$  with respect to the particular listing,  $v_1, v_2, \dots, v_n$ , of  $n$  vertices of  $G$  is an  $n \times n$  matrix  $[a_{ij}]$  such that the  $(i, j)$ th entry  $a_{ij}$  of  $A_G$  is the number of edges from  $v_i$  to  $v_j$ . That is,  $a_{ij} =$  the number of edges from  $v_i$  to  $v_j$ .

## Key Result

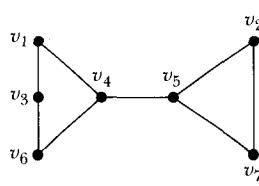
- Let  $G$  be a graph with  $n$  vertices,  $v_1, v_2, \dots, v_n$ , and let  $A = [a_{ij}]$  denote the adjacency matrix with respect to this ordering of the vertices of  $G$ . For any positive integer  $k$ ,  $A^k = [b_{ij}]$  denotes the matrix multiplication of  $k$  copies of  $A$ . Then  $b_{ij}$  denotes the number of distinct walks of length  $k$  from vertex  $v_i$  to vertex  $v_j$  in graph  $G$ .

## EXERCISES

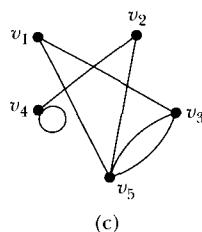
1. Find the adjacency matrices of the graphs in Figure 10.51 with respect to the listing  $v_1, v_2, v_3, v_4, v_5, v_6$ , and  $v_7$  of the vertices.



(a)



(b)



(c)

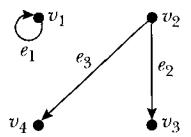
**FIGURE 10.51** Various graphs

2. Draw the graph represented by the given adjacency matrix.

$$(a) \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{bmatrix} \quad (b) \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$(c) \begin{bmatrix} 0 & 1 & 2 & 0 \\ 1 & 0 & 1 & 1 \\ 2 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

3. Find the adjacency matrices of the digraph in Figure 10.52 with respect to the listing  $v_1, v_2, v_3$ , and  $v_4$  of the vertices.

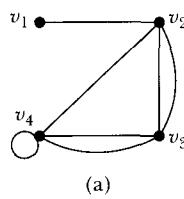
**FIGURE 10.52** A digraph

4. Draw the digraph represented by the given adjacency matrix.

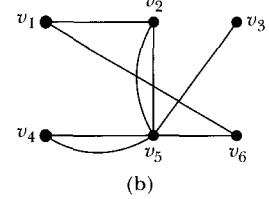
$$(a) \begin{bmatrix} 1 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{bmatrix} \quad (b) \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$(c) \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

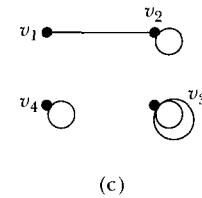
5. Find the number of walks of length 2 from  $v_2$  to  $v_4$  in the graphs in Figure 10.53.



(a)



(b)



(c)

**FIGURE 10.53** Various graphs

6. Find the adjacency matrices of graphs  $K_3$  and  $K_{2,3}$ .  
 7. Find the incidence matrices of graphs  $K_3$  and  $K_{2,3}$ .  
 8. Find the number of distinct paths of length 2 in graphs  $K_3$  and  $K_{2,3}$ .

## 10.4 SPECIAL CIRCUITS

In this section, we discuss Euler circuits and Hamiltonian cycles.

### Euler Circuits

Let us consider the Königsberg bridge problem stated at the beginning of the chapter. The problem is to determine whether it is possible to take a walk that crosses each bridge exactly once before returning to the starting point (see Figure 10.1(a)). As remarked earlier, Euler converted this problem into a graph theory problem as follows: Each of the islands,  $A$ ,  $B$ ,  $C$ , and  $D$ , is considered as a vertex of a graph and the bridges are considered as edges (see Figure 10.1(b)). Now the problem reduces to finding a circuit in the graph of Figure 10.1(b) such that it contains all the edges. In this section, we further describe properties of graphs, which will help us answer this question.

Notice that the graph in Figure 10.1(b) is a connected graph, and this graph has odd degree vertices as well as even degree vertices.

Let us consider a connected graph with more than one vertex such that every vertex has odd degree. For example, consider the graph in Figure 10.20(a). This is a connected graph, and every vertex of this graph is odd degree. This graph has no circuit, so it has no circuit that contains all the edges. Also the graph  $K_4$  contains 4 vertices and 6 edges. The degree of each vertex is 3. This graph has no circuit which contains all the edges.

Next consider a connected graph  $G$  such that every vertex has even degree. For example, graph  $G$  in Figure 10.54 is a connected graph such that every vertex is of even degree.

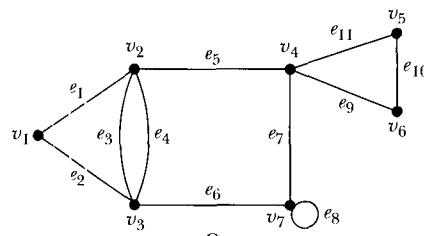


FIGURE 10.54 A connected graph

Let us see if we can find a circuit in the graph in Figure 10.54 such that it contains all the edges. To do so, we start with a vertex and then try to construct a path by alternatively selecting edges and vertices.

Let us choose a vertex in this graph, say  $v_2$ , and an edge,  $e_5$ , with  $v_2$  as one end vertex. The other end vertex of  $e_5$  is  $v_4$ . Now consider  $v_4$ . There are three edges,  $e_7$ ,  $e_9$ , and  $e_{11}$ , with  $v_4$  as one end vertex, and these edges are different from the edge  $e_5$ . We can choose any of these edges. Suppose we choose  $e_9$ . The other end vertex of  $e_9$  is  $v_6$ . Now  $e_{10}$  is the only edge different from  $e_9$  such that  $v_6$  is one of the end vertices of  $e_{10}$ . So we choose  $e_{10}$ . The other end vertex of  $e_{10}$  is  $v_5$ . Next we choose  $e_{11}$  with  $v_5$  and  $v_2$  as end vertices. In the graph, so far we have chosen edges  $e_5$ ,  $e_9$ ,  $e_{10}$ , and  $e_{11}$  and we are at vertex  $v_4$ . Next we choose an edge with one end vertex  $v_4$  different from the edges already chosen. So we choose  $e_7$  with  $v_4$  and  $v_7$  as end vertices. In a similar manner, we next select edge  $e_6$  with  $v_7$  and  $v_3$  as end vertices. Then we choose edge  $e_4$  with  $v_3$  and  $v_2$  as end vertices. Thus, we have formed the circuit

$$T_1 : (v_2, e_5, v_4, e_9, v_6, e_{10}, v_5, e_{11}, v_4, e_7, v_7, e_6, v_3, e_4, v_2)$$

However, the circuit  $T_1$  does not contain all the edges of  $G$ .

Construct graph  $G_1 = G - \{e_5, e_9, e_{10}, e_{11}, e_7, e_6, e_4\}$ . That is, graph  $G_1$  is constructed from  $G$  by deleting all the edges of the circuit  $T_1$  (see Figure 10.55).

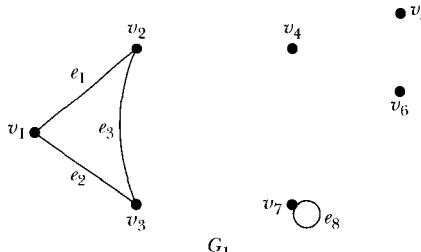


FIGURE 10.55 Graph  $G_1$

Next, we construct graph  $G_2$  from  $G_1$  by deleting all the isolated vertices, if any, from graph  $G_1$  (see Figure 10.56). Notice that graph  $G_2$  is not a connected graph, but every vertex is of even degree.

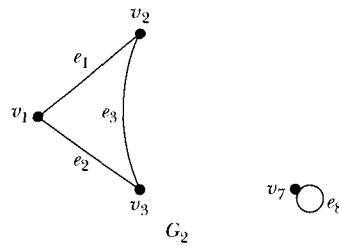


FIGURE 10.56 Graph  $G_2$

Next, we choose an edge  $e$  such that  $e$  is not an edge of circuit  $T_1$ , but one of the end vertices of  $e$  is a vertex of circuit  $T_1$ . Such an edge exists, because  $G$  is a connected graph. For example,  $T_1$  does not contain  $e_8$ , and  $v_7$  is a common vertex of  $T_1$  and  $e_8$ . We form the circuit

$$T_2 : (v_7, e_8, v_7)$$

We replace vertex  $v_7$  of  $T_1$  by circuit  $T_2$  to obtain the circuit

$$T_3 : (v_2, e_5, v_4, e_9, v_6, e_{10}, v_5, e_{11}, v_4, e_7, v_7, e_8, v_7, e_6, v_3, e_4, v_2).$$

Now circuit  $T_3$  also does not contain all the edges of  $G$ . For example, the edge  $e_2 \notin T_3$ .

Construct a new graph,  $G_3 = G_2 - \{e_8\}$ . That is,  $G_3$  is constructed from  $G_2$  by deleting all the edges of circuit  $T_2$  (see Figure 10.57).

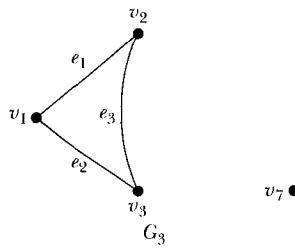
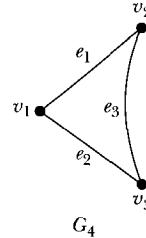


FIGURE 10.57 Graph  $G_3$

Next, we construct a graph  $G_4$  from  $G_3$  by deleting all the isolated vertices, if any, in graph  $G_3$  (see Figure 10.58).



**FIGURE 10.58**  
Graph  $G_4$

Notice that every vertex of  $G_4$  is of even degree. In  $G_4$ , we select vertex  $v_3$  and edge  $e_2$ , and then form the circuit

$$T_4 : (v_3, e_2, v_1, e_1, v_2, e_3, v_3).$$

Next, we replace vertex  $v_3$  of circuit  $T_3$  by circuit  $T_4$  and obtain the circuit

$$T_5 : (v_2, e_5, v_4, e_9, v_6, e_{10}, v_5, e_{11}, v_4, e_7, v_7, e_8, v_7, e_6, v_3, e_2, v_1, e_1, v_2, e_3, v_3, e_4, v_2).$$

It is easy to see that circuit  $T_5$  contains all the edges of  $G$ . Such a circuit is called an **Euler circuit**.

---

**DEFINITION 10.4.1** ► A circuit in a graph that includes all the edges of the graph is called an **Euler circuit**.

---

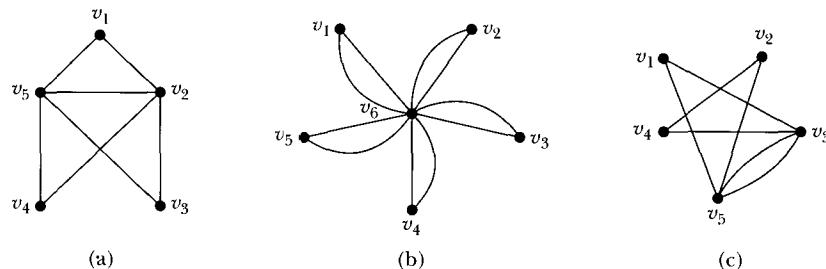
**DEFINITION 10.4.2** ► A graph  $G$  is said to be **Eulerian** if either  $G$  is a trivial graph or  $G$  has an Euler circuit.

By Definition 10.4.2, it follows that a graph  $G$  with only one vertex, say  $v$ , and no edges is Eulerian. In this graph  $G$ , we call the walk  $(v)$  an Euler circuit.

Notice that the graph of Figure 10.54 is Eulerian.

**EXAMPLE 10.4.3**

Each of the graphs in Figure 10.59 is an Eulerian graph.



**FIGURE 10.59** Eulerian graphs

In each of the preceding examples of Eulerian graphs, each vertex in these graphs is of even degree. This is, in fact, true for any Eulerian graph and is proved next.

**Theorem 10.4.4:** If a connected graph  $G$  is Eulerian, then every vertex of  $G$  has even degree.

**Proof:** Suppose that graph  $G$  is Eulerian.

First suppose  $G$  is a trivial graph. Then  $G$  has only one vertex  $v$  and no edges. Hence, the degree of  $v$  is 0, which is even.

Suppose that the graph  $G$  contains more than one vertex. Because  $G$  is Eulerian,  $G$  has an Euler circuit, say

$$C : (v_1, e_1, v_2, e_2, v_3, e_3, v_4, \dots, e_{n-1}, v_n = v_1)$$

from a vertex  $v_1$  to  $v_n = v_1$ . Now  $C$  contains all the vertices and all the edges of  $G$ . However, there are no repeated edges in  $C$ , though in  $C$  a vertex may appear more than once. Let  $u$  be a vertex of  $G$ . Because  $G$  is a connected graph,  $u$  is not an isolated vertex. So  $u$  is an end vertex of some edge. Because  $u$  is an end vertex of some edge and  $C$  contains all the edges, it follows that  $u$  is a member of  $C$ .

Suppose  $u$  is  $v_1$ . Let us say that this is the first appearance of  $u$  in  $C$ . Now  $u$  is also  $v_n$  and we say that  $v_n$  is the last appearance of  $u$  in  $C$ . For each of these two appearances of  $u$ , the edge  $e_1$  and the edge  $e_{n-1}$  together contribute 2 to the degree of  $u$ .

Suppose now  $u$  is  $v_i$  for some  $i$ ,  $1 < i < n$ . Then  $u$  is an end vertex of the edges of  $e_{i-1}$  and  $e_i$ . These edges together contribute 2 to the degree of vertex  $u$ . It now follows that the degree of any vertex in  $C$  is even. Hence, the degree of any vertex in  $G$  is even. ■

Suppose  $G$  is a connected graph in which every vertex is of even degree. We will show that  $G$  contains an Euler circuit. To do so, we first prove the following lemma.

**Lemma 10.4.5:** Let  $G$  be a connected graph with one or two vertices. If every vertex of  $G$  is of even degree, then  $G$  has an Euler circuit.

**Proof:** Suppose  $G$  is a graph with only one vertex, say  $u$ . Now there may exist 0 or more loops at  $u$ . However, the number of loops at  $u$  must be finite. If there is no loop at  $u$ , then  $(u)$  is an Euler circuit of  $G$ . Suppose that there are loops  $e_1, e_2, \dots, e_n$ ,  $n \geq 1$ , at  $u$ . Then  $(u, e_1, u, e_2, \dots, e_n, u)$  is an Euler circuit of  $G$ . Hence,  $G$  contains an Euler circuit.

Suppose now  $G$  has two vertices  $u$  and  $v$  such that both  $u$  and  $v$  are of even degree. Because  $G$  is connected,  $u$  and  $v$  are connected. So there exists an even number of parallel edges between  $u$  and  $v$ . Let  $\{f_1, f_2, \dots, f_{2k}\}$ ,  $k \geq 1$ , be the set of all edges between  $u$  and  $v$ . Let  $e_1, e_2, \dots, e_n$ ,  $n \geq 0$ , be the loops at  $u$  and  $g_1, g_2, \dots, g_m$ ,  $m \geq 0$ , be the loops at  $v$ . (If  $n = 0$ , then there is no loop at  $u$ . Similarly, if  $m = 0$ , then there is no loop at  $v$ ). Now

$$(u, e_1, u, e_2, \dots, u, e_n, u, f_1, v, g_1, v, g_2, v, \dots, g_m, v, f_2, u, f_3, v, f_4, \dots, f_{2k-1}, v, f_{2k}, u)$$

is a trail that begins at  $u$ , traverses all the loops incident with  $u$ , traverses one edge from  $u$  to  $v$ , traverses all the loops at  $v$ , then traverses one edge from  $v$  to  $u$ , and then traverses all the edges between  $u$  and  $v$ . This trail does not contain any repeated edges. Hence, it is a circuit from  $u$  to  $u$ . Because this circuit contains all the edges of  $G$ , it follows that the graph  $G$  has an Euler circuit. ■

**Theorem 10.4.6:** Let  $G$  be a connected graph such that every vertex of  $G$  is of even degree. Then  $G$  has an Euler circuit.

**Proof:** Suppose  $G$  has  $n$  edges. We prove by induction on the number of edges of  $G$  to show that  $G$  has an Euler circuit.

*Basis step:* Suppose  $n = 0$ . Because  $G$  has no edges, it follows that  $G$  has a single vertex, say  $u$ . Then  $(u)$  is an Euler circuit.

*Inductive hypothesis:* Let  $n$  be a positive integer. Assume that any connected graph with  $k$  edges,  $0 \leq k < n$ , in which every vertex has even degree has an Euler circuit.

*Inductive step:* Let  $G = (V, E)$  be a connected graph with  $n$  edges and the degree of each vertex of  $G$  is even. If the number of vertices of  $G$  is 1 or 2, then by Lemma 10.4.5 it follows that  $G$  has an Euler circuit. So assume that graph  $G$  has at least three vertices.

Because  $G$  is connected, there are vertices  $v_1, v_2, v_3$  and edges  $e_1, e_2$  such that  $v_1, v_2$  are the end vertices of  $e_1$  and  $v_2, v_3$  are the end vertices of  $e_2$ . Now consider the subgraph  $G_1 = (V_1, E_1)$ , where  $V_1 = V, E_1 = (E - \{e_1, e_2\})$ . Next, we add a new edge  $e$  with  $v_1, v_3$  as end vertices to the subgraph and obtain a new graph  $G_2 = (V_2, E_2)$ , where  $V_2 = V, E_2 = E_1 \cup \{e\}$ .

Notice that graph  $G_2$  is obtained from  $G$  by deleting edges  $e_1, e_2$ , but not removing any vertices, and adding a new edge  $e$  with end vertices  $v_1$  and  $v_3$ .

In  $G$ , suppose  $\deg(v_1) = r$ ,  $\deg(v_2) = m$ , and  $\deg(v_3) = t$ . Because we deleted edges  $e_1, e_2$ , in  $G_1$ ,  $\deg(v_1) = r - 1$ ,  $\deg(v_2) = m - 2$ , and  $\deg(v_3) = t - 1$ . Now in graph  $G_2$ , we add a new edge  $e$  with end vertices  $v_1$  and  $v_3$ . Hence, in graph  $G_2$ , we have  $\deg(v_1) = r, \deg(v_2) = m - 2, \deg(v_3) = t$ . While constructing  $G_1$  from  $G$  and  $G_2$  from  $G_1$ , the other vertices of  $G$  were not disturbed; i.e., their degree in  $G_2$  is the same as their degree in  $G$ . Thus, it follows that every vertex of  $G_2$  is of even degree.

Now graph  $G_2$  may not be a connected graph. We show that the number of components of  $G_2$  is less than or equal to two.

Because  $v_1$  and  $v_3$  are the end vertices of the edge  $e$  in  $G_2$ , it follows that  $v_1$  and  $v_3$  belong to the same component of  $G_2$ , say  $C_1$ . Now vertex  $v_2$  may not be in  $C_1$ . Let  $C_2$  be the component of  $G_2$  that contains  $v_2$ . Let  $v$  be a vertex of  $G_2$ . Then  $v$  is also a vertex of  $G$ . Because  $G$  is a connected graph, there is a path  $P$  from  $v$  to  $v_1$  in  $G$ .

If  $P$  contains one of the edges  $e_1$  or  $e_2$ , then  $P$  cannot be a path from  $v$  to  $v_1$  in  $G_2$ . Let  $P_1$  be the path in  $G_2$  that is a portion of the path  $P$  starting at  $v$  whose edges are also in  $G_2$ . Path  $P_1$  may terminate at  $v_1, v_2$ , or  $v_3$ . If  $P_1$  is a path from  $v$  to  $v_1$  in  $G_1$ , then  $v$  and  $v_1$  belong to the same component,  $C_1$ . If  $P_1$  ends at  $v_3$ , then  $(P_1, e, v_1)$  is a path from  $v$  to  $v_1$ . Hence, in this case  $v$  also belongs to the same component,  $C_1$ . Suppose  $P_1$  ends at  $v_2$ . Then  $v$  belongs to component  $C_2$ . Thus, any vertex  $v$  of  $G_2$  belongs to either  $C_1$  or  $C_2$ . Hence,  $G_2$  has one (if  $C_1 = C_2$ ) or two components.

Suppose  $G_2$  has only one component,  $C_1$ . Then  $G_2$  is a connected graph with  $n - 1$  edges. Thus, by the inductive hypothesis  $G_2$  has an Euler circuit, say  $T_1$ . From circuit  $T_1$ , we can construct an Euler circuit  $T$  in  $G$  by simply replacing the subpath  $(v_1, e, v_3)$  by the path  $(v_1, e_1, v_2, e_2, v_3)$ . Hence, in this case we find that  $G$  is Eulerian.

Suppose  $G_2$  has two components,  $C_1$  and  $C_2$ . Now each component  $C_i$ ,  $i = 1, 2$ , is a connected graph such that each vertex has even degree and the number of edges in  $C_i$  is  $n_i < n$ . Hence, by the inductive hypothesis  $C_i$  has an Euler circuit  $T_i$ ,  $i = 1, 2$ . Now  $T_1$  contains  $v_1, v_3$  and  $T_2$  contains  $v_2$ . Hence,  $(v_1, e, v_3)$  is a subpath of  $T$ . Moreover, we can assume that  $T_2$  is a circuit from  $v_2$  to  $v_2$ .

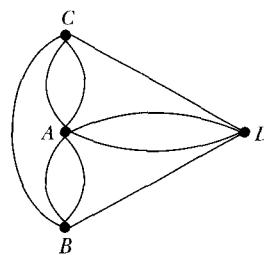
We now construct an Euler circuit in  $G$  by modifying  $T_1$  as follows: In  $T_1$ , replace  $(v_1, e, v_3)$  by  $(v_1, e_2, v_2)$ , followed by  $T_2$ , followed by  $(v_2, e_2, v_3)$ . Thus, we find that  $G$  has an Euler circuit.

The result now follows by induction. ■

We can effectively use Theorem 10.4.6 to determine whether a connected graph  $G$  has an Euler circuit by checking whether all of its vertices are of even degree.

Let us again consider the Königsberg bridge problem. Notice that the graph corresponding to this problem is a connected graph but has vertices of odd degree (see Figure 10.1(b)). Hence, by Theorem 10.4.6, the graph in Figure 10.1(b) has no Euler circuit. Therefore, starting at one land area, it is not possible to walk across all of the bridges exactly once and return to the starting land area.

Since 1736, two additional bridges have been constructed on the Pregel River, one between regions  $B$  and  $C$ , another between regions  $A$  and  $D$ . The graph with the additional two bridges is shown in Figure 10.60.



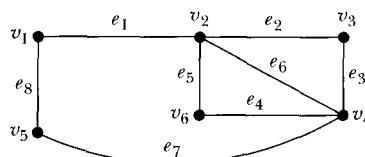
**FIGURE 10.60** Graph of the Königsberg bridge problem with two additional bridges

This is a connected graph with each vertex of even degree. Hence, this graph has an Euler circuit.

Consider the problem related to an old children's game stated in the introduction to the chapter. (See Figure 10.2.) Now we can answer those problems, because the solution is equivalent to finding an Euler circuit in the graph. Notice that the graphs in Figure 10.2 contain vertices of odd degree, and hence none of the graphs contain Euler circuits. We therefore cannot trace the diagrams satisfying the given conditions.

#### EXAMPLE 10.4.7

Consider the graph in Figure 10.61.



**FIGURE 10.61** A connected graph

This is a connected graph; each vertex has even degree. Hence, this graph has an Euler circuit. We can find an Euler circuit. Here

$$(v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_6, e_5, v_2, e_6, v_4, e_7, v_5, e_8, v_1)$$

is a circuit that contains all the vertices and all the edges of  $G$ . Hence, this is an Euler circuit.

In Theorem 10.4.6, we proved that every connected graph with only even degree vertices has an Euler circuit. However, Theorem 10.4.6 does not show how to obtain such a circuit. Next, we describe an algorithm which can be used to construct an Euler circuit in a connected graph with vertices of even degrees.

### Euler Circuit Algorithm

- Step 1.** Choose a vertex  $v$  as the starting vertex for the circuit.
- Step 2.** Construct a path from  $v$  to  $v$ , with distinct edges, as follows: Choose an edge, say  $e_1$ , with  $v$  as one of the end vertices. If the other end vertex, say  $u_1$ , of the edge  $e_1$  is also  $v$ , then go to Step 3. Otherwise choose an edge  $e_2$  different from  $e_1$  with  $u_1$  as one of the end vertices. If the other vertex, say  $u_2$ , of  $e_2$  is  $v$ , then go to Step 3, otherwise choose an edge  $e_3$  different from  $e_1$  and  $e_2$  with  $u_2$  as one of the end vertices. Thus, we have the sequence of vertices and edges  $(v, e_1, u_1, e_2, u_2, e_3, \dots, e_{i-1}, u_i)$ . Suppose we have constructed the sequence of vertices and edges  $(v, e_1, u_1, e_2, u_2, e_3, \dots, e_{i-1}, u_i)$ , where all the edges  $e_1, e_2, \dots, e_{i-1}$  are distinct,  $e_1$  is an edge from  $v$  to  $u_1$ , and  $e_j$  is an edge from  $u_j$  to  $u_{j+1}$  for all  $j = 2, \dots, i-1$ . If  $u_i = v$ , then we have constructed a path from  $v$  to  $v$  with distinct edges, so go to Step 3. Suppose  $u_i \neq v$ . Choose an edge, say  $e_i$  with  $u_i$  as one of the end vertices and  $e_i \neq e_j$ ,  $j = 1, \dots, i-1$ . If the other end vertex of  $e_i$ , say  $u_{i+1}$ , is  $v$ , then go to Step 3. Otherwise continue this process until the desired path from  $v$  to  $v$  is constructed.
- Step 3.** If the circuit  $T_1$  obtained in Step 2 contains all the edges, then stop. Otherwise choose an edge  $e_j$  different from the edges of  $T_1$  such that one of the end vertices of  $e_j$ , say  $w$ , is a member of circuit  $T_1$ .
- Step 4.** Construct a circuit  $T_2$  with starting vertex  $w$ , as in Steps 1 and 2, such that all the edges of  $T_2$  are different from the edges in circuit  $T_1$ .
- Step 5.** Construct circuit  $T_3$  by inserting circuit  $T_2$  at  $w$  of circuit  $T_1$ . Now go to Step 3 and repeat Step 3 with circuit  $T_3$ .

The following example illustrates how this algorithm works.

#### EXAMPLE 10.4.8

Consider the graph in Figure 10.54. This is a connected graph. The degree of each vertex is even. Let us apply the preceding algorithm to find an Eulerian circuit.

First select the vertex  $v_1$ . Then form the circuit

$$C_1 : (v_1, e_1, v_2, e_3, v_3, e_2, v_1).$$

Next select the vertex  $v_2$  and the edge  $e_4$ . Construct the circuit

$$C_2 : (v_2, e_4, v_3, e_6, v_7, e_7, v_4, e_5, v_2).$$

Then form the circuit

$$T_2 : (v_1, e_1, C_2, e_3, v_3, e_2, v_1).$$

Circuit  $T_2$  does not contain all the edges of the given graph. Now choose the vertex  $v_7$  and the edge  $e_8$  and form the circuit

$$C_3 : (v_7, e_8, v_7).$$

Now construct the circuit

$$T_3 : (v_1, e_1, v_2, e_4, v_3, e_6, C_3, e_7, v_4, e_5, v_2, e_3, v_3, e_2, v_1).$$

This circuit also does not contain all the edges. Select the vertex  $v_4$  and the edge  $e_{11}$ . Form the circuit

$$C_4 : (v_4, e_{11}, v_5, e_{10}, v_6, e_9, v_4)$$

and construct the circuit

$$T_4 : (v_1, e_1, v_2, e_4, v_3, e_6, C_3, e_7, C_4, e_5, v_2, e_3, v_3, e_2, v_1).$$

This circuit contains all the vertices and all the edges of the given graph, and hence  $T_4$  is an Euler circuit.

Next, we show that Euler Circuit algorithm does find an Euler circuit in a connected graph in which every vertex is of even degree.

Let  $G$  be a connected graph with more than one vertex such that every vertex is of even degree.

In  $G$ , choose a vertex  $u$  and an edge  $e_1$  with  $u$  as one of the end vertices and the other end vertex, say  $u_1$ . Now  $\deg(u_1) = \text{even}$ . Therefore, there exists an edge  $e_2 \neq e_1$  with one of the end vertices  $u_1$  and the other end vertex, say  $u_2$ .

If  $u_2 \neq u$ , then we choose an edge  $e_3$  different from  $e_2, e_1$  with one end vertex,  $u_2$ , and the other end vertex, say  $u_3$ . Because there are an even number of edges incident with a vertex and there are no vertices of degree 0, we can choose such an edge,  $e_3$ . If  $u_3 = u$ , then we obtain a circuit from  $u$  to  $u$ .

Suppose  $u_3 \neq u$ . Then we choose an edge  $e_4$  different from  $e_3, e_2, e_1$  with one end vertex,  $u_3$ , and the other end vertex,  $u_4$ . If  $u_4 \neq u$ , we repeat the process.

Because the number of edges in  $G$  is finite, using this process of choosing an edge different from the previously chosen edges, we can construct a sequence  $u, e_1, u_1, e_2, u_2, \dots, u_i, e_i, u_{i+1}$  such that  $e_1, e_2, \dots, e_i$  are distinct edges and  $u_{i+1} = u$ . Thus, we obtain the circuit

$$T_1 : (u, e_1, u_1, e_2, u_2, \dots, u_i, e_i, u_{i+1} = u_j).$$

If this circuit does not contain all the edges of  $G$ , then we construct a new graph  $G_1 = G - \{e_1, e_2, \dots, e_i\}$ . That is,  $G_1$  is constructed from  $G$  by deleting all the edges of circuit  $T_1$ . Next, we construct graph  $G_2$  by deleting all the isolated vertices, if any, from graph  $G_1$ . This graph  $G_2$  may not be a connected graph, but every vertex in  $G$  is of even degree.

We choose an edge  $e$  of  $G_2$  such that  $e$  is not an edge of circuit  $T_1$  but one of the end vertices of  $e$ , say  $v_1$ , is a vertex of circuit  $T_1$ . Such a vertex must exist, because  $G$  is a connected graph. In  $G_2$ , we can construct a circuit  $T_2$  from  $v_1$  to  $v_1$ . Clearly,  $T_1$  and  $T_2$  have no common edges.

Now  $T_1$  and  $T_2$  together produce a new circuit,  $T_3$ . If  $T_3$  contains all the edges, then this is an Euler circuit. Otherwise we repeat the process. Because the graph contains a finite number of edges, by repeating the preceding process, we will eventually obtain an Euler circuit.

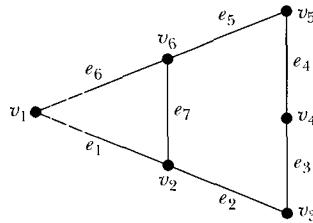
---

**DEFINITION 10.4.9** ► An open trail in a graph is called an **Euler trail** if it contains all the edges and all the vertices.

**EXAMPLE 10.4.10**

Consider the graph in Figure 10.62.

This is a connected graph. It has a vertex of odd degree. Thus, this graph has no Euler circuit, but the trail  $(v_2, e_7, v_6, e_5, v_5, e_4, v_4, e_3, v_3, e_2, v_2, e_1, v_1, e_6, v_6)$  contains all the edges of  $G$ . Hence, this is an Euler trail.



**FIGURE 10.62** A connected graph

**Theorem 10.4.11:** A connected graph  $G$  has an Euler trail if and only if  $G$  has only two vertices of odd degree.

**Proof:** Suppose that  $G$  has an Euler trail  $P$  from a vertex  $u$  to a vertex  $v$  of  $G$ . Now  $P$  contains all the vertices and all the edges of  $G$ . Construct a new graph,  $G_1$ , by adding a new edge,  $e$ , to  $G$  with  $u$  and  $v$  as the end vertices. In graph  $G_1$ , the trail  $P$  together with  $e$  forms an Euler circuit. Hence, every vertex of graph  $G_1$  is of even degree. In graph  $G_1$ , the edge  $e$  contributes 1 to the degree of  $u$  and 1 to the degree of  $v$ . Because graph  $G$  does not contain the edge  $e$ , it follows that  $u$  and  $v$  are the only vertices of odd degree in  $G$ .

Conversely, assume that a connected graph  $G$  has only two vertices, say  $u$  and  $v$ , of odd degrees. Construct a new graph,  $G_1$ , by adding a new edge,  $e$ , to  $G$  with  $u$  and  $v$  as the end vertices. Graph  $G_1$  is a connected graph such that every vertex is of even degree. Hence, by Theorem 10.4.6,  $G_1$  has an Euler circuit, say  $P$ .

Now  $(u, e, v)$  is a subpath of  $P$ . This subpath is not present in graph  $G$ . Hence, if we delete  $(u, e, v)$  from  $P$ , then we obtain an Euler trail  $P_1$  from  $v$  to  $u$  in  $G$ . ■

## Hamiltonian Cycle

In 1859, Sir William Rowan Hamilton, an Irish mathematician, marketed a game called *Around the world*. The game consisted of a regular dodecahedron made of wood. Each corner bore the name of a famous city of the world. The game was to find a path starting at any city, traveling along the edges of the dodecahedron, visiting each city exactly once and returning to the starting city. The diagram in Figure 10.63 represents the game in a plane.

This diagram in Figure 10.63 is a connected graph with 20 vertices. Each vertex represents a famous city. It follows that the game is equivalent to finding a cycle in the graph in Figure 10.63 that contains each vertex exactly once except for the starting and ending vertices, which appear twice.

**DEFINITION 10.4.12** ► A cycle in a graph  $G$  is called a **Hamiltonian cycle** if it contains each vertex of  $G$ .

From Definition 10.4.12, it follows that a Hamiltonian cycle is a closed trail that contains each vertex of the graph exactly once.

If a graph  $G$  has a Hamiltonian cycle, then  $G$  is called a **Hamiltonian graph**. A path in a graph  $G$  is called a **Hamiltonian path** if it contains each vertex of  $G$ .

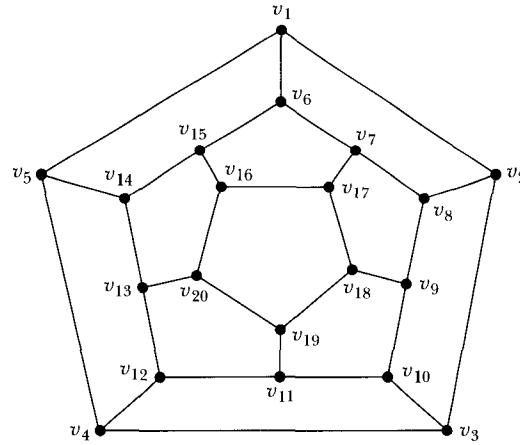


FIGURE 10.63 Around the world game in a plane

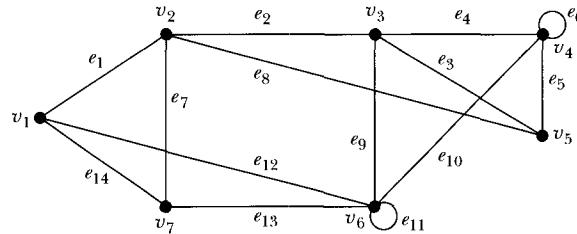
**EXAMPLE 10.4.13**Consider graph  $G$  in Figure 10.64.

FIGURE 10.64 A connected graph

In this graph,

$$(v_1, e_1, v_2, e_2, v_3, e_3, v_5, e_5, v_4, e_{10}, v_6, e_{13}, v_7, e_{14}, v_1)$$

is a cycle. This cycle contains all the vertices of  $G$ . Hence, this cycle is a Hamiltonian cycle.

Notice that this graph  $G$  is a connected graph and there are vertices (for example,  $v_7, v_5$ ) of odd degree. Hence, this graph has no Euler circuit.

**Sir William Rowan Hamilton**

(1805–1865)

Hamilton was born on August 4, 1805, in Dublin, Ireland. He was the fourth of nine children. His early education from the age of 3 was provided by his uncle, who was a gifted teacher, and by the age of 5, Hamilton was proficient in Latin, Greek, and Hebrew.

Hamilton started reading Newton's *Principia* when he was about 15 and became interested in astronomy. In 1822, he discovered an error in Laplace's *Mécanique Céleste*, which was conveyed to John Brinkley through a

**Historical Notes**

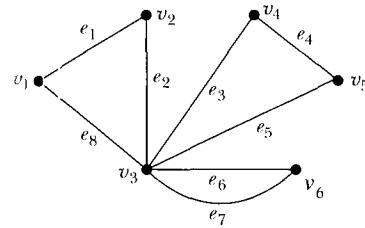
friend. On April 23, 1827, while still an undergraduate at Trinity College, Hamilton presented his paper, "Theory of Systems of Rays," to the Royal Irish Academy. This paper is responsible for creating the field of mathematical optics. Hamilton introduced the characteristic function, his first discovery. As a direct result of his work, Hamilton was considered for and then appointed to the position of Astronomer Royal at Dunsink Observatory, even though he did not have a degree.

Hamilton was interested in three-dimensional complex numbers, which he called "triplets." He had little success in this area, as he was able to add,

but could not find a suitable multiplication rule. He then considered the so-called quaternions. While he was walking along the Royal Canal on October 16, 1843, the discovery of the quaternions flashed in his mind. He immediately scratched the multiplication formula for the quaternions on the stone of a bridge over the canal. Hamilton discovered that he could give up the commutative law of multiplication and still have a meaningful algebraic system. The geometric significance of the quaternions was realized when Hamilton and Cayley independently showed that the quaternion operators rotated vectors about a given axis.

**EXAMPLE 10.4.14**

Consider graph  $G$  in Figure 10.65.

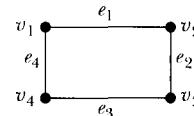


**FIGURE 10.65** A connected graph

Now  $G$  is a connected graph and each vertex of  $G$  has an even degree. Therefore,  $G$  has an Euler circuit. We show that  $G$  has no Hamiltonian cycle. Suppose  $G$  has a Hamiltonian cycle  $C$ . Then  $C$  contains each vertex of  $G$  exactly once except for the starting vertex, which is also the terminating vertex of  $C$ . Hence, the degree of each vertex in  $C$  is 2. Now each of the vertices  $v_i$ ,  $i = 1, 2, 4, 5, 6$  has degree 2. Thus,  $C$  must contain the edges  $e_i$ ,  $i = 1, 2, 3, 4, 5, 6, 7$ , and  $v_3$  is an end vertex of the edges  $e_i$ ,  $i = 2, 3, 5, 6, 7, 8$ . This implies that the degree of  $v_3$  in  $C$  is more than 2, a contradiction. Hence,  $G$  has no Hamiltonian cycle.

**EXAMPLE 10.4.15**

Consider graph  $G$  in Figure 10.66.



**FIGURE 10.66**  
A connected graph

This graph contains a Hamiltonian cycle

$$(v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_1).$$

This cycle is also an Euler circuit for this graph.

Let us now consider the following question: Is it possible to determine whether a graph has a Hamiltonian cycle?

This question appears similar to the problem of finding Euler circuits in a graph. However, the answer to this question has not been completely obtained. Finding necessary and sufficient conditions for a graph to have a Hamiltonian cycle is a major unsolved problem in graph theory. Next, we give some partial answers to this problem.

**Theorem 10.4.16:** Let  $G = (V, E)$  be a simple graph with  $n$  vertices such that  $G$  contains a Hamiltonian cycle. Let  $v_1, v_2, \dots, v_t \in V$ , where  $t < n$ . Then the number of components in the subgraph  $G_1 = G - \{v_1, v_2, \dots, v_t\}$  is less than or equal to  $t$ .

**Proof:** Let  $C$  be a Hamiltonian cycle in  $G$ . Now  $C$  contains all the vertices of  $G$ . Therefore, when the vertices  $v_1, v_2, \dots, v_t$  are deleted together with the edges

incident with these vertices, then they are also deleted from cycle  $C$ . As a result, cycle  $C$  is divided into at most  $t$  pieces, which contain all the vertices of  $G_1$ . Hence, the number of components of  $G_1$  will not exceed  $t$ . ■

Let us consider graph  $G$  of Example 10.4.14. Let us delete vertex  $v_3$  together with all the edges incident with  $v_3$ . Then we find that  $G_1 = G - \{v_3\}$  is a graph consisting of three components. If  $G$  is Hamiltonian, then by Theorem 10.4.16, the number of components in the subgraph  $G_1 = G - \{v_3\}$  is less than or equal to 1, which is not the case. Hence, the graph  $G$  is not Hamiltonian.

**Theorem 10.4.17:** Let  $G$  be a simple graph with  $n > 2$  vertices. If each vertex has degree at least  $\frac{n}{2}$ , then  $G$  has a Hamiltonian cycle.

**Proof:** Suppose  $n = 3$ . Let  $v_1$ ,  $v_2$ , and  $v_3$  be the vertices of  $G$ . Each vertex of  $G$  has a degree of at least 2. Because  $G$  is a simple graph, it follows that each vertex of  $G$  has degree 2. Let  $e_1$  be the edge with end vertices  $v_1$  and  $v_2$ ,  $e_2$  be the edge with end vertices  $v_2$  and  $v_3$ , and  $e_3$  be the edge with end vertices  $v_1$  and  $v_3$  (see Figure 10.67).

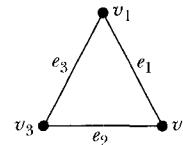


FIGURE 10.67  
Simple graph with  
three vertices each  
of degree 2

It follows that the graph is a cycle. Hence,  $G$  has a Hamiltonian cycle,

$$(v_1, e_1, v_2, e_2, v_3, e_3, v_1).$$

Suppose  $n \geq 4$ . Let  $T$  be the set of all paths in  $G$ . Because  $T$  is a finite set, there exists a path, say

$$P : (u_1, e_1, u_2, e_2, u_3, e_3, \dots, u_k, e_k, u_{k+1})$$

in  $T$  with maximal number of vertices. Suppose the path  $P$  is as shown in Figure 10.68.

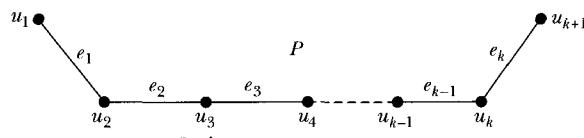


FIGURE 10.68 Path  $P$

Let  $v$  be a vertex adjacent to  $u_1$  in  $G$ . Suppose  $v \neq u_i$  for  $i = 1, 2, \dots, k + 1$ . Then  $G$  has a path with  $k + 2$  vertices. This contradicts the choice of path  $P$  with  $k + 1$  vertices.

Hence,  $v = u_i$  for some  $i = 1, 2, \dots, k + 1$ . Because  $G$  has no loop,  $v$  cannot be  $u_1$ . Now the degree of  $u_1$  is at least  $\frac{n}{2}$ . Hence,  $u_1$  has at least  $\frac{n}{2}$  adjacent vertices, which are all members of  $P$ . Consequently,  $P$  has at least  $1 + \frac{n}{2}$  vertices. Similarly, all the adjacent vertices of  $u_{k+1}$  are also members of  $P$ .

We now show that there is some vertex  $u_i$  of  $P$ , where  $2 \leq i \leq k+1$ , such that  $u_1$  is an adjacent vertex of  $u_i$ , whereas  $u_{i-1}$  is an adjacent vertex of  $u_{k+1}$ . Suppose this is not true. Then for each  $u_i$ , if  $u_i$  is adjacent to  $u_1$ , then  $u_{i-1}$  is not adjacent to  $u_{k+1}$ . Thus, there are at least  $\frac{n}{2}$  vertices of  $P$  not adjacent to  $u_{k+1}$ . Because  $G$  is a simple graph with  $n$  vertices, it follows that

$$\deg(u_{k+1}) \leq (n-1) - \frac{n}{2} = \frac{n}{2} - 1 < \frac{n}{2},$$

which is a contradiction. Hence, there exists a vertex  $u_i$ ,  $2 \leq i \leq k+1$  such that  $u_i$  is adjacent to  $u_1$  and  $u_{i-1}$  is adjacent to  $u_{k+1}$ .

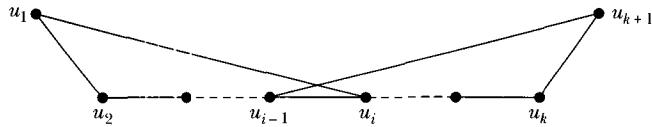


FIGURE 10.69 A path

Let  $P_1$  be the subpath of  $P$  from  $u_i$  to  $u_{k+1}$  and  $P_2$  be the subpath of  $P$  from  $u_{i-1}$  to  $u_1$ .

If  $i = k+1$ , then we have  $P_1 = (u_{k+1})$ .

We now construct the cycle

$$C = (u_1, e_{i1}, u_i, P_1, u_{k+1}, e_{i2}, u_{i-1}, P_2, u_1)$$

where  $u_1, u_i$  are end vertices of  $e_{i1}$  and  $u_{k+1}, u_{i-1}$  are end vertices of  $e_{i2}$ . Notice that  $C$  contains all the vertices of  $P$ . Let us next show that  $C$  contains all the vertices of  $G$ .

Let  $v$  be a vertex of  $G$  not contained in cycle  $C$ . Because cycle  $C$  contains at least  $1 + \frac{n}{2}$  vertices, there exist at most  $\frac{n}{2} - 1$  vertices of  $G$  that are not in  $C$ . Now  $\deg(v) \geq \frac{n}{2}$ . Thus, there must exist a vertex, say  $w$ , of  $C$  that is an adjacent vertex of  $v$ . We now relabel the vertices and edges of  $C$  so that  $C$  is

$$(v_1, e_1, v_2, e_2, v_3, \dots, v_{k+1}, e_{k+1}, v_{k+2} = v_1).$$

Suppose  $w = v_j$  ( $1 \leq j \leq k+1$ ). Let  $e$  be the edge from  $v$  to  $v_j$ . Then

$$P' : (v, e, v_j, e_j, v_{j-1}, e_{j+1}, \dots, v_{k+1}, e_{k+1}, v_{k+2} = v_1, e_1, v_2, \dots, v_{j-1})$$

is a path in  $G$  from  $v$  to  $v_{j-1}$ . Now the number of vertices in  $P'$  is one more than the number of vertices in  $P$ , a contradiction. Hence,  $C$  contains all the vertices of  $G$ . This implies that  $C$  is a Hamiltonian cycle of  $G$ . ■

Next we state another result that can be used to determine whether a simple connected graph has a Hamiltonian cycle.

**Theorem 10.4.18:** Let  $G$  be a simple connected graph with  $n$  vertices, where  $n > 2$ . If for any two vertices  $u$  and  $v$  of  $G$ , such that  $u$  and  $v$  are not adjacent,  $\deg(u) + \deg(v) \geq n$ , then  $G$  has a Hamiltonian cycle.

**REMARK 10.4.19** ▶ We can show that Theorem 10.4.17 follows from Theorem 10.4.18.

**REMARK 10.4.20** ▶ The directed Hamiltonian cycle and the directed Hamiltonian path of a directed graph are a directed path and a directed circuit of the graph, respectively, containing each vertex of the graph.

**EXAMPLE 10.4.21**

During a soccer tournament with  $n$  teams, where  $n \geq 2$ , each team has played against all the others exactly once and there were no tie matches. We show that all the teams can be listed in order so that each has defeated the team next on the list.

Let the teams and matches correspond to the vertices and to the arcs of a directed graph, respectively, in such a way that the initial and terminal vertices of an arc correspond to the winner and loser, respectively, of the corresponding match. The resulting graph is a simple digraph. Because in the tournament each team has played against all the other teams exactly once, in the graph  $G$ , for any two vertices  $u$  and  $v$  there exists an arc either from  $u$  to  $v$  or from  $v$  to  $u$ . Hence,  $G$  is a complete digraph. We show that this digraph has a directed Hamiltonian path. Let  $P$  denote a directed path of maximal length of  $G$  with  $m$  vertices.

Suppose

$$P : (v_1, e_1, v_2, e_2, \dots, e_{m-1}, v_m)$$

and  $P$  does not contain all the vertices. Then  $m < n$  and there exists a vertex  $u$  such that  $u \neq v_i, i = 1, 2, \dots, m$ .

Now either there exists an arc  $e$  from  $v_1$  to  $u$  or there exists an arc  $e$  from  $u$  to  $v_1$ . If there is an arc from  $u$  to  $v_1$ , then we get a directed path  $P_1$  from  $u$  to  $v_m$ , such that length of  $P_1$  is greater than length of  $P$ , a contradiction. This contradiction implies that there exists an arc from  $v_1$  to  $u$ . Similarly, we can show that there exists an arc from  $u$  to  $v_m$ . Now there exists a positive integer  $i$ ,  $1 \leq i \leq m$ , such that there exists an arc from  $v_{i-1}$  to  $u$  and an arc from  $u$  to  $v_{i+1}$ . This gives us the path

$$P_2 : (v_1, v_2, \dots, v_{i-1}, u, v_{i+1}, \dots, v_{m-1}, v_m)$$

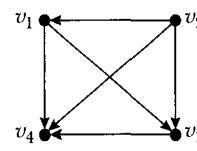
such that  $P_2$  is longer than  $P$ . This contradiction implies that  $P$  contains all the vertices of  $G$ . Hence,  $P$  is a directed Hamiltonian path.

Example 10.4.22 clarifies the solution of the preceding example.

**EXAMPLE 10.4.22**

During a certain soccer tournament with 4 teams, each team has played against all the others exactly once and there were no ties. We show that all the teams can be listed in order so that each has defeated the team next on the list.

Let the teams be denoted by  $v_1, v_2, v_3$ , and  $v_4$  and let the matches correspond to the vertices and the arcs of a directed graph, respectively, in such a way that the initial and terminal vertices of an arc correspond to the winner and loser, respectively, of the corresponding match. Regarding these matches, suppose we have the digraph in Figure 10.70.



**FIGURE 10.70** A digraph representing the outcome of the tournament

In this digraph we find that  $v_2 \rightarrow v_1 \rightarrow v_3 \rightarrow v_4$  is a Hamiltonian directed path.

## (E) WORKED-OUT EXERCISES

**Exercise 1:** Prove that a graph has a circuit if the degree of each vertex is an even positive integer.

**Solution:** Let  $G$  be a graph such that the degree of each vertex is an even positive integer. Let  $u$  be a vertex of  $G$  and  $C$  be the component of  $G$  that contains  $u$ . Then  $C$  is a maximal connected subgraph of  $G$  that contains  $u$ . Moreover, all the vertices adjacent to  $u$  are in  $C$ . Now  $C$  is a connected graph such that every vertex is of even ( $> 0$ ) degree. Hence,  $C$  has an Euler circuit. Because  $C$  contains at least one edge, this Euler circuit is of nonzero length. Hence, this Euler circuit is a circuit in graph  $G$ .

**Exercise 2:** Let  $G$  be a connected graph such that each vertex is of degree 2. Prove that  $G$  is a cycle.

**Solution:** Because  $G$  is a connected graph such that every vertex is of even degree, it follows that  $G$  has an Euler circuit. This circuit contains all the vertices and all the edges of  $G$ . Because the degree of each vertex is 2, it follows that in the above circuit, a vertex, except the starting vertex, cannot appear more than once. Hence, the above circuit is a cycle. This proves that graph  $G$  is a cycle.

**Exercise 3:** Determine whether the graphs in Figure 10.71 have Euler circuits. If a graph has an Euler circuit, describe one.

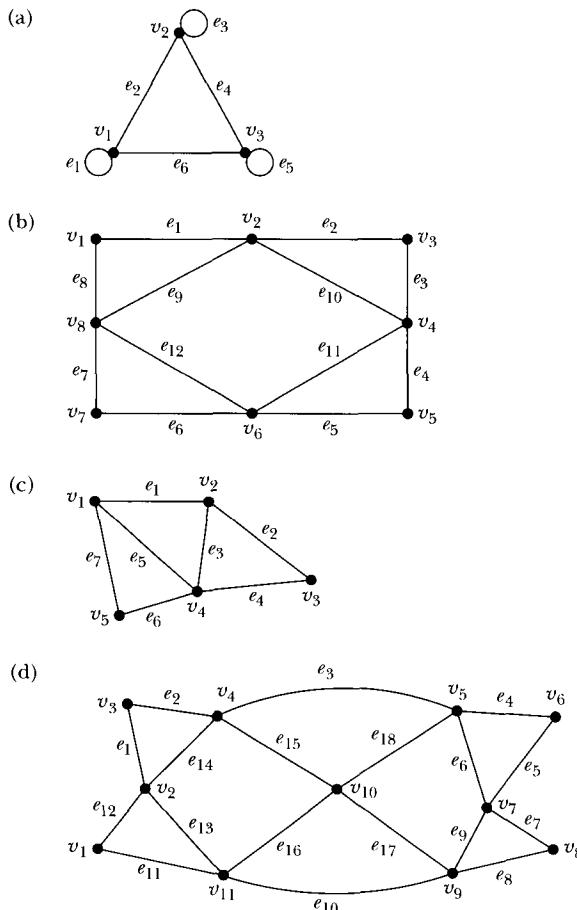


FIGURE 10.71 Various graphs

**Solution:**

- (a) The graph in Figure 10.71(a) is a connected graph. Every vertex is of even degree. Therefore, the given graph has an Euler circuit. Here

$$(v_1, e_1, v_1, e_2, v_2, e_3, v_2, e_4, v_3, e_5, v_3, e_6, v_1)$$

is a circuit that contains all the edges exactly once. Hence, this circuit is an Euler circuit.

- (b) The graph in Figure 10.71(b) is a connected graph. Every vertex is of even degree. Therefore, the given graph has an Euler circuit. Here  $(v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_5, e_5, v_6, e_6, v_7, e_7, v_8, e_8, v_1, e_{12}, v_2, e_9, v_3, e_{10}, v_4, e_{11}, v_5, e_3, v_6, e_5, v_7, e_4, v_8, e_7, v_9, e_{10}, v_{10}, e_{15}, v_4)$  is a circuit that contains all the edges exactly once. Hence, this circuit is an Euler circuit.  
(c) The graph in Figure 10.71(c) is a connected graph. Because  $v_1$  is a vertex of odd degree, this graph has no Euler circuit by Theorem 10.4.4.  
(d) The graph in Figure 10.71(d) is a connected graph. The degree of each vertex is even. Therefore, this graph has an Euler circuit. To find an Euler circuit, we use the algorithm given in this section. We consider the following cycles:

$$C_1 : (v_3, e_2, v_4, e_{14}, v_2, e_1, v_3)$$

$$C_2 : (v_4, e_3, v_5, e_{18}, v_{10}, e_{15}, v_4)$$

$$C_3 : (v_5, e_4, v_6, e_5, v_7, e_6, v_5)$$

$$C_4 : (v_7, e_7, v_8, e_8, v_9, e_9, v_7)$$

$$C_5 : (v_9, e_{10}, v_{11}, e_{16}, v_{10}, e_{17}, v_9)$$

$$C_6 : (v_{11}, e_{11}, v_1, e_{12}, v_2, e_{13}, v_{11})$$

Note that no two cycles have a common edge. Moreover, if  $e$  is an edge,  $e$  is a member of only one of the circuits  $C_1$ ,  $C_2$ ,  $C_3$ ,  $C_4$ ,  $C_5$ , and  $C_6$ . Next, we construct a circuit as follows:

Using  $C_5$  and  $C_6$ , construct the circuit

$$T_5 : (v_9, e_{10}, C_6, e_{16}, v_{10}, e_{17}, v_9),$$

by replacing  $v_{11}$  by  $C_6$ . Now, using  $C_4$  and  $T_5$ , construct the circuit

$$T_4 : (v_7, e_7, v_8, e_8, T_5, e_9, v_7),$$

by replacing  $v_9$  by  $T_5$ . Next, using  $C_3$  and  $T_4$ , construct the circuit

$$T_3 : (v_5, e_4, v_6, e_5, T_4, e_6, v_5),$$

by replacing  $v_7$  by  $T_4$ . We now use the circuits  $C_2$  and  $T_3$  to construct the circuit

$$T_2 : (v_4, e_3, T_3, e_{18}, v_{10}, e_{15}, v_4),$$

by replacing  $v_5$  by  $T_3$ . Finally, using  $C_1$  and  $T_2$ , we construct the circuit

$$T_1 : (v_3, e_2, T_2, e_{14}, v_2, e_1, v_3),$$

by replacing  $v_4$  by  $T_2$ . It follows that

$$\begin{aligned} T_1 : \quad & (v_3, e_2, T_2, e_{14}, v_2, e_1, v_3) \\ & = (v_3, e_2, v_4, e_3, T_3, e_{18}, v_{10}, e_{15}, v_4, e_{14}, v_2, e_1, v_3) \\ & = (v_3, e_2, v_4, e_3, v_5, e_4, v_6, e_5, T_4, e_6, v_5, e_{18}, v_{10}, \\ & \quad e_{15}, v_4, e_{14}, v_2, e_1, v_3) \\ & = (v_3, e_2, v_4, e_3, v_5, e_4, v_6, e_5, v_7, e_7, v_8, e_8, T_5, e_9, v_7, \\ & \quad e_6, v_5, e_{18}, v_{10}, e_{15}, v_4, e_{14}, v_2, e_1, v_3) \end{aligned}$$

$$\begin{aligned} & = (v_3, e_2, v_4, e_3, v_5, e_4, v_6, e_5, v_7, e_7, v_8, e_8, v_9, e_{10}, C_6, e_{16}, \\ & \quad v_{10}, e_{17}, v_9, e_9, v_7, e_6, v_5, e_{18}, v_{10}, e_{15}, v_4, e_{14}, v_2, e_1, v_3) \\ & = (v_3, e_2, v_4, e_3, v_5, e_4, v_6, e_5, v_7, e_7, v_8, e_8, v_9, e_{10}, v_{11}, \\ & \quad e_{11}, v_1, e_{12}, v_2, e_{13}, v_{11}, e_{16}, v_{10}, e_{17}, v_9, e_9, v_7, e_6, v_5, \\ & \quad e_{18}, v_{10}, e_{15}, v_4, e_{14}, v_2, e_1, v_3), \end{aligned}$$

is an Euler circuit.

## SECTION REVIEW

### Key Terms

Euler circuit

Euler trail

Hamiltonian graph

Eulerian graph

Hamiltonian cycle

Hamiltonian path

### Some Key Definitions

1. A circuit in a graph that includes all the edges of the graph is called an Euler circuit.
2. A graph  $G$  is said to be Eulerian if either  $G$  is a trivial graph or  $G$  has an Euler circuit.
3. An open trail in a graph is called an Euler trail if it contains all the edges and all the vertices.
4. A cycle in a graph  $G$  is called a Hamiltonian cycle if it contains each vertex of  $G$ .
5. If a graph  $G$  has a Hamiltonian cycle, then  $G$  is called a Hamiltonian graph. A path in a graph  $G$  is called a Hamiltonian path if it contains each vertex of  $G$ .

### Some Key Results

1. If a connected graph  $G$  is Eulerian, then every vertex of  $G$  has even degree.
2. A connected graph  $G$  with one or two vertices each of which has even degree has an Euler circuit.
3. Let  $G$  be a connected graph such that every vertex of  $G$  is of even degree. Then  $G$  has an Euler circuit.
4. A connected graph  $G$  has an Euler trail if and only if  $G$  has only two vertices of odd degree.
5. Let  $G = (V, E)$  be a simple graph with  $n$  vertices such that  $G$  contains a Hamiltonian cycle. Let  $v_1, v_2, \dots, v_t \in V$ , where  $t < n$ . Then the number of components in the subgraph  $G_1 = G - \{v_1, v_2, \dots, v_t\}$  is less than or equal to  $t$ .
6. Let  $G$  be a simple graph with  $n > 2$  vertices. If each vertex has degree at least  $\frac{n}{2}$ , then  $G$  has a Hamiltonian cycle.

7. Let  $G$  be a simple connected graph with  $n$  vertices, where  $n > 2$ . If for any two vertices  $u$  and  $v$  of  $G$  such that  $u$  and  $v$  are not adjacent,  $\deg(u) + \deg(v) \geq n$ , then  $G$  has a Hamiltonian cycle.

## EXERCISES

1. Describe whether the graphs in Figure 10.72 have an Euler circuit. If the graph has an Euler circuit, exhibit one.

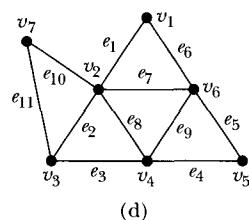
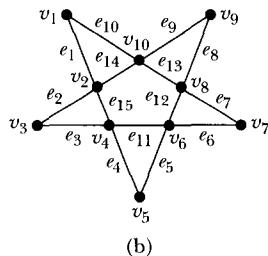
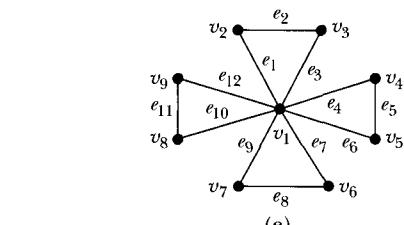
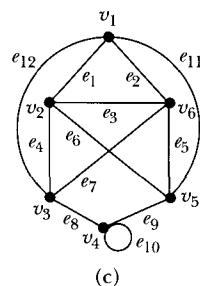
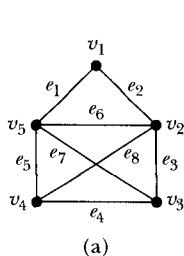


FIGURE 10.72 Various graphs

2. Determine whether the graphs in Figure 10.73 have Euler trails. If the graph has an Euler trail, exhibit one.

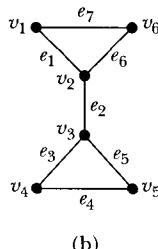
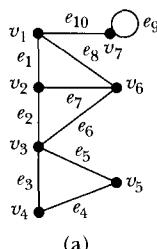


FIGURE 10.73 Various graphs

3. Prove that a complete graph  $K_n$  has an Euler circuit if  $n$  is odd.

4. When does a complete bipartite graph  $K_{n,m}$  contain an Euler circuit?
5. Prove that a connected graph  $G$  has an Euler circuit if and only if the set of edges can be partitioned into cycles.
6. Give an example of a graph that has an Euler circuit and a Hamiltonian cycle that are not identical.
7. Determine whether the graph in Figure 10.74 has a Hamiltonian cycle:

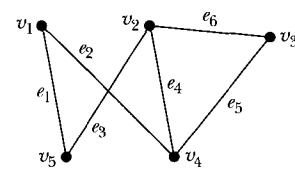
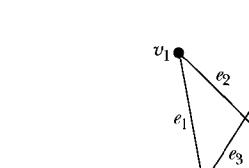
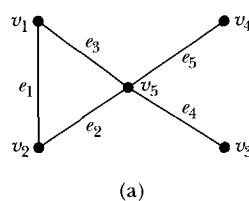


FIGURE 10.74 A graph

8. Give an example of a connected graph that has neither a Hamiltonian cycle nor an Euler circuit.
9. Let  $G$  be a connected simple graph with  $n > 2$  vertices and  $m$  edges. If  $m = \frac{1}{2}(n^2 - 3n + 6)$ , then show that  $G$  has a Hamiltonian cycle.
10. Show that the converse of Theorem 10.4.17 is not true.
11. Show that the complete graph  $K_n$ ,  $n > 2$  contains a Hamiltonian cycle.
12. If every member of a party of six people knows at least three people, prove that they can sit around a table in such a way that each of them knows both his neighbors.
13. Let  $G$  be a simple connected graph with  $n$  vertices. Suppose the degree of each vertex is at least  $n - 1$ . Does it imply the existence of a Hamiltonian cycle in  $G$ ?
14. Consider the game that asks the children to trace a figure with a pencil without either lifting the pencil from the figure or tracing a line more than once. Determine if this can be done for the diagram in Figure 10.75, assuming that the starting point and ending point must be the same.

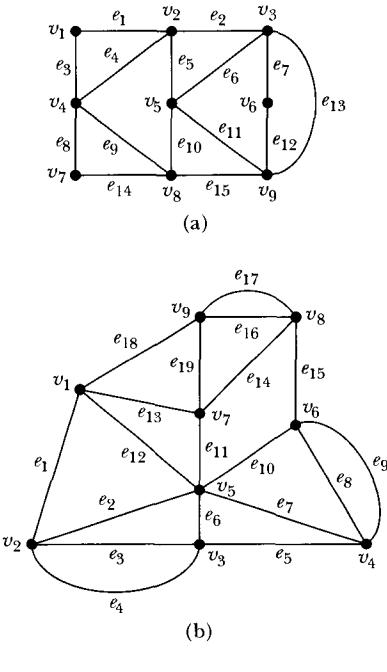


FIGURE 10.75 A graph

15. Suppose that the diagram in Figure 10.76 represents the floor plan for the ground floor of an art museum. Is it possible to tour the exhibit on the first floor so that you pass through each door exactly once?

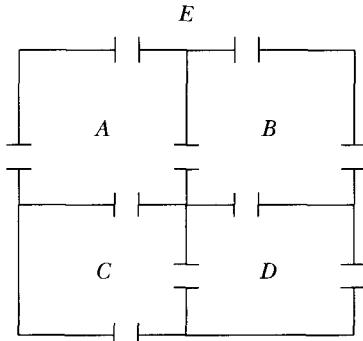
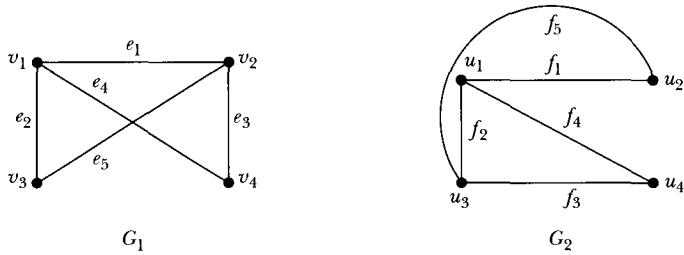


FIGURE 10.76 Floor plan

## 10.5 ISOMORPHISM

Consider graphs  $G_1$  and  $G_2$  in Figure 10.77.

FIGURE 10.77 Graphs  $G_1$  and  $G_2$ 

Both of these graphs have four vertices and five edges. The degree sequence of both of these graphs is 2, 2, 3, 3. The pictorial representation of  $G_1$  looks different from that of  $G_2$ . Are these two graphs the same? What is meant by the word “same”?

Both graphs in Figure 10.77 have the same number of vertices. So we can define a one-to-one correspondence  $f : V_1 \rightarrow V_2$ , where  $V_1 = \{v_1, v_2, v_3, v_4\}$  and  $V_2 = \{u_1, u_2, u_3, u_4\}$  are the set of vertices of  $G_1$  and  $G_2$ , respectively. Also,  $G_1$  and  $G_2$  have the same number of edges. Therefore, we can define a one-to-one correspondence  $h : E_1 \rightarrow E_2$ , where  $E_1 = \{e_1, e_2, e_3, e_4, e_5\}$  and  $E_2 = \{f_1, f_2, f_3, f_4, f_5\}$  are the set of edges of  $G_1$  and  $G_2$ , respectively. However, we want to find a one-to-one correspondence  $f : V_1 \rightarrow V_2$  and a one-to-one correspondence  $h : E_1 \rightarrow E_2$  in

such a way so that if two vertices  $v_i$  and  $v_j$  are end vertices of some edge  $e_k$  in  $G_1$ , then  $f(v_i)$  and  $f(v_j)$  are end vertices of the edge  $h(e_k)$  in  $G_2$ .

Let us define  $f : V_1 \rightarrow V_2$  by

$$\begin{aligned} f : v_1 &\mapsto u_1 \\ v_2 &\mapsto u_3 \\ v_3 &\mapsto u_4 \\ v_4 &\mapsto u_2 \end{aligned}$$

and  $h : E_1 \rightarrow E_2$  by

$$\begin{aligned} h : e_1 &\mapsto f_2 \\ e_2 &\mapsto f_4 \\ e_3 &\mapsto f_5 \\ e_4 &\mapsto f_1 \\ e_5 &\mapsto f_3 \end{aligned}$$

In Figure 10.77, we see that  $v_1$  and  $v_2$  are the end vertices of edge  $e_1$  in  $G_1$  and  $f(v_1) = u_1$  and  $f(v_2) = u_3$  are end vertices of edge  $h(e_1) = f_2$  in  $G_2$ . Also,  $f(v_3) = u_4$  and  $f(v_4) = u_1$  are the end vertices of edge  $f_4 = h(e_2)$  in  $G_2$  and  $v_3$  and  $v_1$  are end vertices of edge  $e_2$  in  $G_1$ .

Similarly, for other vertices, we can check that any two vertices  $v_i$  and  $v_j$  are end vertices of some edge  $e_k$  in  $G_1$  if and only if  $f(v_i)$  and  $f(v_j)$  are end vertices of edge  $h(e_k)$  in  $G_2$ . When this happens, we say that  $G_1$  is isomorphic to  $G_2$ . More formally, we have the following definition.

---

**DEFINITION 10.5.1** ▶ Let  $G_1 = (V_1, E_1, g_1)$  and  $G_2 = (V_2, E_2, g_2)$  be two graphs.  $G_1$  is said to be **isomorphic** to  $G_2$  if there exist a one-to-one correspondence  $f : V_1 \rightarrow V_2$  and a one-to-one correspondence  $h : E_1 \rightarrow E_2$  in such a way that for any edge  $e_k \in E_1$ ,  $g_1(e_k) = \{v_i, v_j\}$  in  $G_1$  if and only if  $g_2(h(e_k)) = \{f(v_i), f(v_j)\}$  in  $G_2$ .

---

**REMARK 10.5.2** ▶ Definition 10.5.1 can also be stated as follows:

Let  $G_1 = (V_1, E_1)$  and  $G_2 = (V_2, E_2)$  be two graphs.  $G_1$  is said to be *isomorphic* to  $G_2$  if there exist a one-to-one correspondence  $f : V_1 \rightarrow V_2$  and a one-to-one correspondence  $h : E_1 \rightarrow E_2$  such that for any edge  $e_k$  in  $E_1$ , vertices  $v_i, v_j$  are end vertices of  $e_k$  in  $G_1$  if and only if  $f(v_i), f(v_j)$  are end vertices of  $h(e_k)$  in  $G_2$ .

---

**REMARK 10.5.3** ▶ When we say that two graphs are the same we mean they are isomorphic to each other.

#### EXAMPLE 10.5.4

Consider graphs  $G_1$  and  $G_2$  in Figure 10.78.

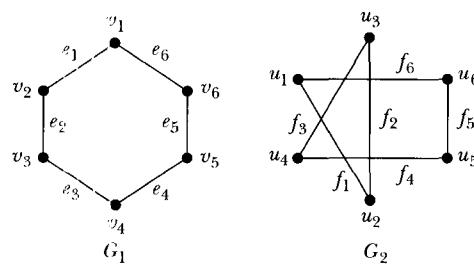


FIGURE 10.78 Graphs  $G_1$  and  $G_2$

Both of these graphs have six vertices and six edges. Moreover, both graphs are simple. The degree sequence of both of these graphs is 2, 2, 2, 2, 2, 2. The pictorial representation of  $G_1$  looks different from that of  $G_2$ .

Let us define  $f : V_1 \rightarrow V_2$  by

$$\begin{aligned} f : v_1 &\mapsto u_1, & v_4 &\mapsto u_4, \\ v_2 &\mapsto u_2, & v_5 &\mapsto u_5, \\ v_3 &\mapsto u_3, & v_6 &\mapsto u_6, \end{aligned}$$

and  $h : E_1 \rightarrow E_2$  by

$$\begin{aligned} h : e_1 &\mapsto f_1, & e_4 &\mapsto f_4, \\ e_2 &\mapsto f_2, & e_5 &\mapsto f_5, \\ e_3 &\mapsto f_3, & e_6 &\mapsto f_6. \end{aligned}$$

We find that if  $v_i$  and  $v_j$  are end vertices of edge  $e_k$  in  $G_1$ , then  $f(v_i) = u_i$  and  $f(v_j) = u_j$  are end vertices of edge  $h(e_k) = f_k$  in  $G_2$ . Hence, graph  $G_1$  is isomorphic to graph  $G_2$ .

We leave the proof of the following theorem as an exercise.

**Theorem 10.5.5:** Let  $G$ ,  $G_1$ ,  $G_2$ , and  $G_3$  be graphs. Then the following assertions hold.

- (i)  $G$  is isomorphic to itself.
- (ii) If  $G_1$  is isomorphic to  $G_2$ , then  $G_2$  is isomorphic to  $G_1$ .
- (iii) If  $G_1$  is isomorphic to  $G_2$  and  $G_2$  is isomorphic to  $G_3$ , then  $G_1$  is isomorphic to  $G_3$ .

---

**DEFINITION 10.5.6** ▶ Two graphs  $G_1$  and  $G_2$  are said to be *isomorphic*, written  $G_1 \simeq G_2$ , if  $G_1$  is isomorphic to  $G_2$ .

---

**DEFINITION 10.5.7** ▶ Two graphs  $G_1$  and  $G_2$  are said to be **different** if  $G_1$  is not isomorphic to  $G_2$ .

We leave the proof of the following theorem as an exercise.

**Theorem 10.5.8:** Let  $G_1 = (V_1, E_1)$  and  $G_2 = (V_2, E_2)$  be two simple graphs.  $G_1$  is isomorphic to  $G_2$  if there exists a one-to-one correspondence  $f : V_1 \rightarrow V_2$  such that vertices  $v_i, v_j$  are adjacent vertices in  $G_1$  if and only if  $f(v_i), f(v_j)$  are adjacent vertices in  $G_2$ .

We now prove the following theorem.

**Theorem 10.5.9:** Let  $G_1$  and  $G_2$  be two graphs such that  $G_1$  is isomorphic to  $G_2$ . Then  $G_1$  has a vertex of degree  $k$  if and only if  $G_2$  has a vertex of degree  $k$ .

**Proof:** Let  $G_1 = (V_1, E_1)$  and  $G_2 = (V_2, E_2)$ . Because  $G_1$  is isomorphic to  $G_2$ , there exist a one-to-one correspondence  $f : V_1 \rightarrow V_2$  and a one-to-one correspondence  $h : E_1 \rightarrow E_2$  such that if any two vertices  $v_i, v_j \in V_1$  are end vertices of some edge  $e_k$  in  $G_1$ , then  $f(v_i)$  and  $f(v_j)$  are end vertices of edge  $h(e_k)$  in  $G_2$ .

Let  $v$  be a vertex of  $G_1$ . Suppose there are  $k$  distinct edges  $e_1, e_2, \dots, e_k$  incident with  $v$ . Because  $h : E_1 \rightarrow E_2$  is a one-to-one correspondence, it follows that  $h(e_1), h(e_2), \dots, h(e_k)$  are distinct edges incident with  $f(v)$ .

Suppose  $f_1$  is an edge with  $f(v)$  as an end vertex. Because  $h : E_1 \rightarrow E_2$  is a one-to-one correspondence, it follows that  $h(e_t) = f_1$  for some  $e_t \in E_1$ . Suppose the end vertices of  $e_t$  are  $v_i$  and  $v_j$ . Then  $f(v_i)$  and  $f(v_j)$  are the end vertices of the edge  $h(e_t) = f_1$  in  $G_2$ . Because  $f(v)$  is an end vertex of  $h(e_t) = f_1$ , one of  $f(v_i)$  and  $f(v_j)$  is  $f(v)$ . Now because  $f$  is a one-to-one correspondence, it follows that  $v$  is either  $v_i$  or  $v_j$ . This implies that  $v$  is an end vertex of  $e_t$ , which implies that  $f(v)$  is an end vertex of  $h(e_t) = f_1$ . Moreover,  $e_t$  is a loop at  $v$  if and only if  $h(e_t)$  is a loop at  $f(v)$ . Hence,  $\deg(v) = \deg(f(v))$ .

Conversely, suppose that  $G_2$  has a vertex of degree  $k$ . Now  $G_2$  is isomorphic to  $G_1$  by Theorem 10.5.5. Hence, from the first part of the proof it follows that  $G_1$  has a vertex of degree  $k$ . ■

### EXAMPLE 10.5.10

Consider graphs  $G_1$  and  $G_2$  in Figure 10.79.

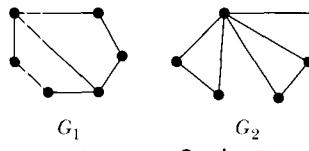


FIGURE 10.79 Graphs  $G_1$  and  $G_2$

Graphs  $G_1$  and  $G_2$  have the same number of vertices and the same number of edges.  $G_2$  has a vertex of degree 5, but  $G_1$  has no vertex of degree 5. Hence, graph  $G_1$  is not isomorphic to graph  $G_2$ .

### EXAMPLE 10.5.11

Consider graphs  $G_1$  and  $G_2$  in Figure 10.80.

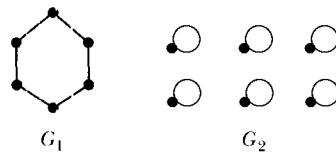


FIGURE 10.80 Graphs  $G_1$  and  $G_2$

Graphs  $G_1$  and  $G_2$  have the same number of vertices, the same number of edges, and also the degree of every vertex in both graphs is 2.  $G_2$  has 6 loops, but  $G_1$  has no loop. Hence, graph  $G_1$  is not isomorphic to graph  $G_2$ .

**Theorem 10.5.12:** Let  $G_1$  and  $G_2$  be two graphs such that  $G_1$  is isomorphic to  $G_2$ . Then  $G_1$  has a cycle of length  $k$  if and only if  $G_2$  has a cycle of length  $k$ .

**Proof:** Let  $G_1 = (V_1, E_1)$  and  $G_2 = (V_2, E_2)$ . Because  $G_1$  is isomorphic to  $G_2$ , there exist a one-to-one correspondence  $f : V_1 \rightarrow V_2$  and a one-to-one correspondence  $h : E_1 \rightarrow E_2$  such that if any two vertices  $v_i, v_j \in V_1$  are end vertices of some edge  $e_k$  in  $G_1$ , then  $f(v_i)$  and  $f(v_j)$  are end vertices of edge  $h(e_k)$  in  $G_2$ . Let

$$(v = v_1, e_1, v_2, e_2, v_3, e_3, \dots, v_{k-1}, e_{k-1}, v_k, e_k, v_{k+1} = v)$$

be a cycle of length  $k$  in  $G_1$ . Now  $v_1, v_2, v_3, \dots, v_{k-1}$  are distinct vertices and  $e_1, e_2, e_3, \dots, e_{k-1}, e_k$  are distinct edges in  $G_1$ . Hence,  $f(v_1), f(v_2), f(v_3), \dots, f(v_{k-1})$  are distinct vertices and  $h(e_1), h(e_2), h(e_3), \dots, h(e_{k-1}), h(e_k)$  are distinct edges in  $G_2$ . Because  $v_i, v_{i+1}$  are end vertices of  $e_i$  in  $G_1$  it follows that  $f(v_i), f(v_{i+1})$  are end vertices of  $h(e_i)$  in  $G_2$ . Thus,

$$(f(v) = f(v_1), h(e_1), f(v_2), h(e_2), f(v_3), h(e_3), \dots,$$

$$f(v_{k-1}), h(e_{k-1}), f(v_k), h(e_k), f(v_{k+1}) = f(v))$$

is a cycle of length  $k$  in  $G_2$ .

The converse follows because if  $G_1$  is isomorphic to  $G_2$ , then  $G_2$  is isomorphic to  $G_1$ . ■

As discussed, in the third section, Section 10.3, of this chapter the adjacency matrix of a graph shows the number of vertices of a graph and the adjacencies between the vertices. Hence, the following theorem follows easily.

**Theorem 10.5.13:** Two simple graphs are isomorphic if and only if their vertices can be labeled in such a way that the corresponding adjacency matrices are equal.

#### EXAMPLE 10.5.14

Consider graphs  $G_1$  and  $G_2$  in Figure 10.81.

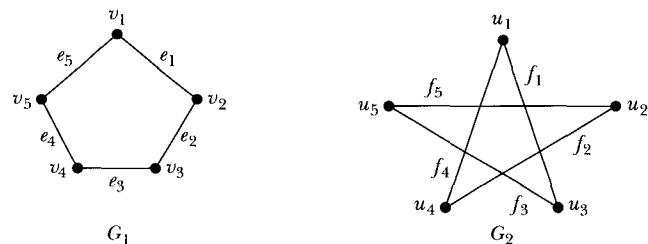


FIGURE 10.81 Isomorphic graphs  $G_1$  and  $G_2$

Notice that both graphs have the same number of vertices and the same number of edges. All of the vertices of both graphs have degree 2. Now define  $f : V_1 \rightarrow V_2$ , where  $V_1 = \{v_1, v_2, v_3, v_4, v_5\}$  and  $V_2 = \{u_1, u_2, u_3, u_4, u_5\}$  by

$$\begin{aligned} f : v_1 &\mapsto u_1, & v_4 &\mapsto u_2, \\ v_2 &\mapsto u_3, & v_5 &\mapsto u_4, \\ v_3 &\mapsto u_5, \end{aligned}$$

Then  $f$  is a one-to-one correspondence. To verify whether  $G_1$  is isomorphic to  $G_2$ , we examine the adjacency matrix  $A_{G_1}$  with rows and columns labeled in the order

$v_1, v_2, v_3, v_4, v_5$ , and the adjacency matrix  $A_{G_2}$  with rows and columns labeled in the order  $u_1, u_3, u_5, u_2, u_4$ .

$$A_{G_1} : \begin{matrix} v_1 & v_2 & v_3 & v_4 & v_5 \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix} \end{matrix}, \quad A_{G_2} : \begin{matrix} u_1 & u_3 & u_5 & u_2 & u_4 \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix} \end{matrix}$$

Because  $A_{G_1}$  and  $A_{G_2}$  are the same, it follows that graphs  $G_1$  and  $G_2$  are isomorphic.

## WORKED-OUT EXERCISES

**Exercise 1:** For each pair of graphs  $G_1$  and  $G_2$  in Figure 10.82, determine whether  $G_1$  is isomorphic to  $G_2$ . Justify your answer.

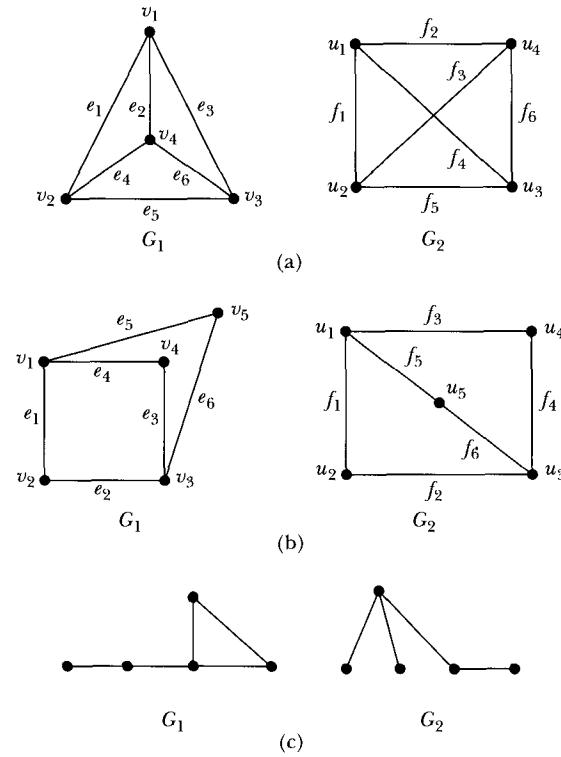


FIGURE 10.82 Various graphs

**Solution:**

- (a) Both graphs are simple and have the same number of vertices and the same number of edges. All of the vertices of both graphs have degree 3. Now define  $f : V_1 \rightarrow V_2$ , where  $V_1 = \{v_1, v_2, v_3, v_4\}$  and  $V_2 = \{u_1, u_2, u_3, u_4\}$  by

$$\begin{aligned} f : v_1 &\mapsto u_1, & v_3 &\mapsto u_3, \\ v_2 &\mapsto u_2, & v_4 &\mapsto u_4. \end{aligned}$$

Clearly,  $f$  is a one-to-one correspondence. To verify whether  $G_1$  and  $G_2$  are isomorphic, we examine the adjacency matrix  $A_{G_1}$  with rows and columns labeled in the order  $v_1, v_2, v_3, v_4$  and the adjacency matrix  $A_{G_2}$ , with rows and columns labeled in the order  $u_1, u_2, u_3, u_4$ .

$$A_{G_1} : \begin{matrix} v_1 & v_2 & v_3 & v_4 \\ \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \end{matrix}, \quad A_{G_2} : \begin{matrix} u_1 & u_2 & u_3 & u_4 \\ \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \end{matrix}$$

Because  $A_{G_1}$  and  $A_{G_2}$  are the same, it follows that  $G_1$  and  $G_2$  are isomorphic.

- (b) Both the graphs are simple and have the same number of edges and the same number of vertices. Now define  $f : V_1 \rightarrow V_2$ , where  $V_1 = \{v_1, v_2, v_3, v_4, v_5\}$  and  $V_2 = \{u_1, u_2, u_3, u_4, u_5\}$  by

$$\begin{aligned} f : v_1 &\mapsto u_1, & v_4 &\mapsto u_4, \\ v_2 &\mapsto u_2, & v_5 &\mapsto u_5, \\ v_3 &\mapsto u_3, \end{aligned}$$

Clearly,  $f$  is a one-to-one correspondence. To verify whether  $G_1$  and  $G_2$  are isomorphic, we examine the adjacency matrix  $A_{G_1}$ , with rows and columns labeled by the list  $v_1, v_2, v_3, v_4, v_5$  and the adjacency matrix  $A_{G_2}$ , with rows and columns labeled by the list  $u_1, u_2, u_3, u_4, u_5$ .

$$A_{G_1} : \begin{matrix} v_1 & v_2 & v_3 & v_4 & v_5 \\ \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{bmatrix} \end{matrix}$$

$$A_{G_2} : \begin{matrix} u_1 & u_2 & u_3 & u_4 & u_5 \\ \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{bmatrix} \end{matrix}$$

Because  $A_{G_1}$  and  $A_{G_2}$  are the same, it follows that  $G_1$  and  $G_2$  are isomorphic.

- (c) Notice that  $G_1$  has a cycle of length 3, but  $G_2$  has no cycle. Hence,  $G_1$  and  $G_2$  are not isomorphic.

**Exercise 2:** Let  $G_1$  and  $G_2$  be two graphs such that  $G_1$  is isomorphic to  $G_2$ . If  $G_1$  is a connected graph, then show that  $G_2$  is a connected graph.

**Solution:** Let  $G_1 = (V_1, E_1)$  and  $G_2 = (V_2, E_2)$ . Because  $G_1$  is isomorphic to  $G_2$ , there exist a one-to-one correspondence  $f : V_1 \rightarrow V_2$  and a one-to-one correspondence  $h : E_1 \rightarrow E_2$  such that if any two vertices  $v_i, v_j \in V_1$  are end vertices of some edge  $e_k$  in  $G_1$ , then  $f(v_i)$  and  $f(v_j)$  are end vertices of edge  $h(e_k)$  in  $G_2$ .

Let  $u$  and  $w$  be two vertices of  $G_2$ . Because  $f : V_1 \rightarrow V_2$  is onto, there exist vertices  $v$  and  $v_{k+1}$  in  $G_1$  such that  $f(v) = u$  and  $f(v_{k+1}) = w$ . Now  $G_1$  is a connected graph. Therefore, there exists a  $v - v_{k+1}$  walk

$$(v = v_1, e_1, v_2, e_2, v_3, e_3, \dots, v_{k-1}, e_{k-1}, v_k, e_k, v_{k+1})$$

in  $G_1$ . Because  $v_i, v_{i+1}$  are end vertices of  $e_i$  in  $G_1$ , it follows that  $f(v_i), f(v_{i+1})$  are end vertices of  $h(e_i)$  in  $G_2$ . Thus,  $(u = f(v_1), h(e_1), f(v_2), h(e_2), f(v_3), h(e_3), \dots, f(v_{k-1}), h(e_{k-1}), f(v_k), h(e_k), f(v_{k+1}) = w)$  is a walk in  $G_2$ . Hence,  $G_2$  is a connected graph.

**Exercise 3:** Draw all simple graphs with five vertices and three edges.

**Solution:** Let  $G$  be a simple graph with five vertices,  $v_1, v_2, v_3, v_4$ , and  $v_5$ , and three edges,  $e_1, e_2$ , and  $e_3$ . Then the sum of the degrees is 6. Hence,  $\deg(v_i) \leq 3$ ,  $i = 1, 2, 3, 4, 5$ . We consider the following cases.

**Case 1:**  $\deg(v_i) = 3$  for some vertex  $v_i$ .

Suppose  $\deg(v_1) = 3$ . Because the graph has no loops and no parallel edges,  $v_1$  has three distinct adjacent vertices, and  $v_1$  is not adjacent to itself. Then  $v_1$  is an end vertex of the three edges  $e_1, e_2, e_3$  and the other end vertices of these edges are from the set  $\{v_2, v_3, v_4, v_5, v_6\}$ . Suppose the other end ver-

tex of  $e_1$  is  $v_2$ , the end vertex of  $e_2$  is  $v_3$ , and the end vertex of  $e_3$  is  $v_4$ . It follows that  $\deg(v_1) = 3$ ,  $\deg(v_2) = 1$ ,  $\deg(v_3) = 1$ ,  $\deg(v_4) = 1$ , and  $\deg(v_5) = 0$ . Hence, in this case it follows that the degree sequence of a graph is 0, 1, 1, 1, 3. Moreover, any such simple graph is isomorphic to the graph in Figure 10.83.

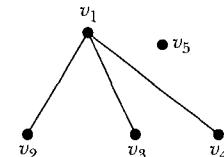


FIGURE 10.83

Simple graph

**Case 2:**  $\deg(v_i) \neq 3$  for all  $i = 1, 2, 3, 4, 5$ .

In this case, the possible degree sequences are: 0, 0, 2, 2, 2; 0, 1, 1, 2, 2; or 1, 1, 1, 1, 2.

Any simple graph  $G$  with degree sequence 0, 0, 2, 2, 2 is isomorphic to the graph in Figure 10.84.

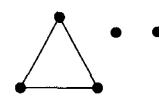


FIGURE 10.84

Simple graph

Any simple graph  $G$  with degree sequence 0, 1, 1, 2, 2 is isomorphic to the graph in Figure 10.85.



FIGURE 10.85

Simple graph

Any simple graph  $G$  with degree sequence 1, 1, 1, 1, 2 is isomorphic to the graph in Figure 10.86.



FIGURE 10.86

Thus, we find that there are four different simple graphs with five vertices and three edges.

## SECTION REVIEW

### Key Terms

isomorphic

different

### Some Key Definitions

- Let  $G_1 = (V_1, E_1, g_1)$  and  $G_2 = (V_2, E_2, g_2)$  be two graphs.  $G_1$  is said to be isomorphic to  $G_2$  if there exist a one-to-one correspondence  $f : V_1 \rightarrow V_2$

and a one-to-one correspondence  $h : E_1 \rightarrow E_2$  in such a way that for any edge  $e_k \in E_1$ ,  $g_1(e_k) = \{v_i, v_j\}$  in  $G_1$  if and only if  $g_2(h(e_k)) = \{f(v_i), f(v_j)\}$  in  $G_2$ .

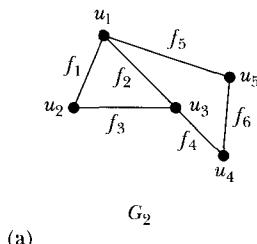
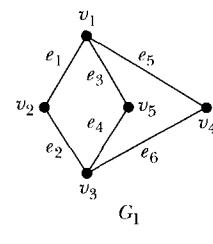
2. Two graphs  $G_1$  and  $G_2$  are said to be isomorphic, written  $G_1 \simeq G_2$ , if  $G_1$  is isomorphic to  $G_2$ .
3. Two graphs  $G_1$  and  $G_2$  are said to be different if  $G_1$  is not isomorphic to  $G_2$ .

## Some Key Results

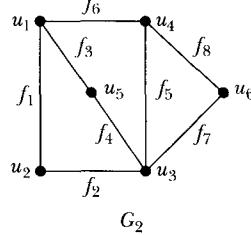
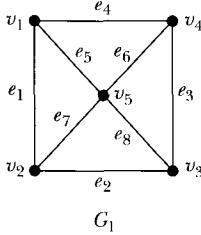
1. Let  $G$ ,  $G_1$ ,  $G_2$ , and  $G_3$  be graphs. Then the following assertions hold.
  - (i)  $G$  is isomorphic to itself.
  - (ii) If  $G_1$  is isomorphic to  $G_2$ , then  $G_2$  is isomorphic to  $G_1$ .
  - (iii) If  $G_1$  is isomorphic to  $G_2$  and  $G_2$  is isomorphic to  $G_3$ , then  $G_1$  is isomorphic to  $G_3$ .
2. Let  $G_1 = (V_1, E_1)$  and  $G_2 = (V_2, E_2)$  be two simple graphs.  $G_1$  is isomorphic to  $G_2$  if there exists a one-to-one correspondence  $f : V_1 \rightarrow V_2$  such that vertices  $v_i, v_j$  are adjacent vertices in  $G_1$  if and only if  $f(v_i), f(v_j)$  are adjacent vertices in  $G_2$ .
3. Let  $G_1$  and  $G_2$  be two graphs such that  $G_1$  is isomorphic to  $G_2$ . Then  $G_1$  has a vertex of degree  $k$  if and only if  $G_2$  has a vertex of degree  $k$ .
4. Let  $G_1$  and  $G_2$  be two graphs such that  $G_1$  is isomorphic to  $G_2$ . Then  $G_1$  has a cycle of length  $k$  if and only if  $G_2$  has a cycle of length  $k$ .

## EXERCISES

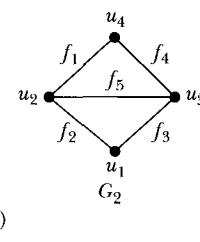
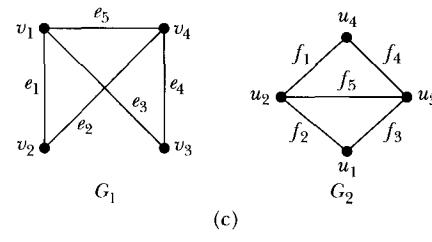
1. For each pair of graphs  $G_1$  and  $G_2$  in Figure 10.87, determine whether  $G_1$  is isomorphic to  $G_2$ . Justify your answer.



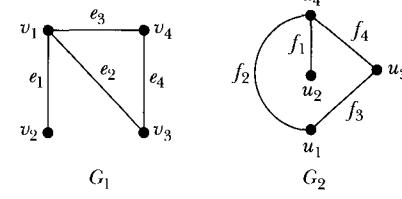
(a)



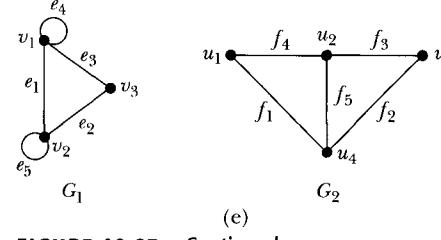
(b)

**FIGURE 10.87** Various graphs

(c)



(d)



(e)

**FIGURE 10.87** Continued

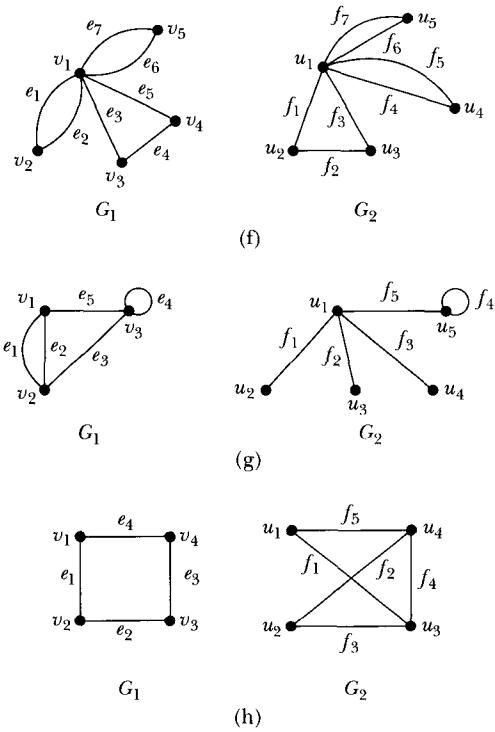


FIGURE 10.87 Continued

2. Let  $G_1 = (V_1, E_1)$  and  $G_2 = (V_2, E_2)$  be two simple graphs. Prove that  $G_1$  is isomorphic to  $G_2$  if there exists a one-to-one correspondence  $f : V_1 \rightarrow V_2$  such that

- vertices  $v_i, v_j$  are adjacent vertices in  $G_1$  if and only if  $f(v_i), f(v_j)$  are adjacent vertices in  $G_2$ .
3. Let  $G_1$  and  $G_2$  be two graphs such that  $G_1$  is isomorphic to  $G_2$ . If  $G_1$  is a simple graph, then show that  $G_2$  is a simple graph.
  4. Draw all different graphs with two vertices and two edges.
  5. Draw all different simple graphs with five vertices and four edges.
  6. Draw all different simple graphs with three vertices.
  7. Let  $G_1$  and  $G_2$  be two graphs such that  $G_1$  is isomorphic to  $G_2$ . If  $G_1$  is Eulerian, then show that  $G_2$  is Eulerian.
  8. Let  $G_1$  and  $G_2$  be two graphs such that  $G_1$  is isomorphic to  $G_2$ . If  $G_1$  has a Hamiltonian cycle, then show that  $G_2$  has a Hamiltonian cycle.
  9. Let  $G_1$  and  $G_2$  be two connected graphs with same number of vertices such that every vertex is of degree 2. Prove that  $G_1$  is isomorphic to  $G_2$ .
  10. Is  $K_{2,3}$  isomorphic to  $K_6$ ?
  11. Give an example of two nonisomorphic graphs with same degree sequences.
  12. Is the graph of  $K_{3,3}$  isomorphic to the graph of Figure 10.88?

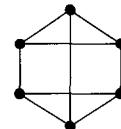


FIGURE 10.88 A graph

## 10.6 GRAPH ALGORITHMS

Graph theory has many applications. For example, we can use graphs to show how different chemicals are related or to show airline routes. They can also be used to show the highway structure of a city, state, or country. The edges connecting two vertices can be assigned a nonnegative real number, called the **weight** of the edge. If the graph represents a highway structure, the weight can represent the distance between two places, or the travel time from one place to another. Such graphs are called **weighted graphs**. The graph in Figure 10.89 is a weighted graph.

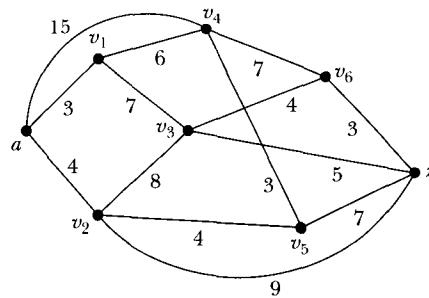


FIGURE 10.89 Weighted graph

Throughout this section, we consider only simple graphs. When we speak of a shortest path from one vertex to another, we mean a path with the shortest length

between the vertices. A simple graph does not contain loops and parallel edges. Hence, an edge  $e$  with end vertices  $u$  and  $v$  can also written  $uv$  or  $u - v$ .

Let  $G$  be a graph with  $n$  vertices, where  $n > 0$ . Let  $V = \{v_1, v_2, \dots, v_n\}$  be the vertex set of  $G$ . We list the vertices of  $G$  as  $v_1, v_2, \dots, v_n$ . Let  $W$  be an  $n \times n$  matrix such that its  $(i, j)$ th entry, for  $i \neq j$ ,  $W[i, j]$  is given by

$$W[i, j] = \begin{cases} w_{ij} & \text{if } v_i - v_j \text{ is an edge in } G \text{ and } w_{ij} \text{ is the weight of the edge } v_i - v_j, \\ \infty & \text{if there is no edge from } v_i \text{ to } v_j. \end{cases}$$

Also,  $W[i, i] = 0$  for all  $i$ . The matrix  $W$  is called the **weight matrix** of graph  $G$ .

For example, for the graph in Figure 10.89, the weight matrix  $W$  is: (Assume that the rows and columnus are labeled  $a, v_1, v_2, v_3, v_4, v_5, v_6, z$ .)

$$W = \begin{array}{ccccccccc} & c & v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & z \\ \begin{matrix} a \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ z \end{matrix} & \left[ \begin{matrix} 0 & 3 & 4 & \infty & 15 & \infty & \infty & \infty \\ 3 & 0 & \infty & 7 & 6 & \infty & \infty & \infty \\ \infty & \infty & 0 & 8 & \infty & 4 & \infty & 9 \\ \infty & 7 & 8 & 0 & \infty & \infty & 4 & 5 \\ 15 & 6 & \infty & \infty & 0 & 3 & 7 & \infty \\ \infty & \infty & 4 & \infty & 3 & 0 & \infty & 7 \\ \infty & \infty & \infty & 4 & 7 & \infty & 0 & 3 \\ \infty & \infty & 9 & 5 & \infty & 7 & 3 & 0 \end{matrix} \right] \end{array}$$

## Shortest Path Algorithm

Let  $G$  be a weighted graph. Let  $u$  and  $v$  be two vertices in  $G$ , and let  $P$  be a path in  $G$  from  $u$  to  $v$ . The **length** of path  $P$ , written  $l(P)$ , is the sum of the weights of all the edges on path  $P$ , which is also called the **length** of  $v$  from  $u$  via  $P$ .

Consider graph  $G$  of Figure 10.89. Let  $V = \{a = v_0, v_1, v_2, v_3, v_4, v_5, v_6, z\}$  be the vertex set of  $G$ . Suppose we need to find a path from  $a$  to  $z$ . Let  $P_1 : a - v_1 - v_4 - v_5 - z$ ,  $P_2 : a - v_2 - z$ , and  $P_3 : a - v_1 - v_3 - z$ . Then  $l(P_1) = 19$ ,  $l(P_2) = 13$ , and  $l(P_3) = 15$ . Now  $P_1$ ,  $P_2$ , and  $P_3$  are paths from  $a$  to  $z$ . However, among these paths, the length of  $P_2$  is the shortest. If the vertices in graph  $G$  represent cities and the weights of the edges represent the travel time between cities, then among paths  $P_1$ ,  $P_2$ , and  $P_3$  traveling from  $a$  to  $z$  via path  $P_2$  is the fastest.

In graph  $G$ , there are various other paths from  $a$  to  $z$ . Our objective is to find a path from  $a$  to  $z$  with the least length. One way to determine such a path is to determine all paths and their lengths from  $a$  to  $z$  and then choose a path of shortest length. However, determining a path from source to destination using this approach could be extremely time-consuming.

---

**REMARK 10.6.1** ► Throughout this section, by a shortest path from a vertex to another vertex, we mean a path with the shortest length between the vertices.

In this section, we describe the **shortest path algorithm**, also called the **greedy algorithm**, developed by Dijkstra. We assume that the graph under consideration is a simple and connected weighted graph. The weights are positive real numbers.

The inputs to the program are the graph and the weight matrix associated with the graph.

## Dijkstra's Shortest Path Algorithm

Let us again consider the graph in Figure 10.89. Our objective is to find the length of a shortest path from  $a$  to  $z$ .

Dijkstra's algorithm iteratively constructs the set  $S$  that consists of all the vertices of  $G$  for which the length of a shortest path has been determined. Initially,  $S = \emptyset$ . Let  $N = V - S$ , where  $V$  is the set of all vertices of  $G$ . It follows that initially  $N = V$ . Moreover,  $V = S \cup N$ .

For each vertex in  $v \in V$ , we assign the label  $L(v)$  as follows:

- Initially,  $L(a) = 0$  and  $L(v) = \infty$  for all other vertices of  $V$ .
- If  $v \in S$ , then  $L(v)$  gives the length of a shortest path from  $a$  to  $v$ .
- After each iteration of the algorithm, the value of  $L(v)$  for certain vertices of  $V$ , as described below, is updated.

At the termination of the algorithm,  $z \in S$  and  $L(z)$  gives the length of a shortest path from  $a$  to  $z$ .

In Dijkstra's shortest path algorithm, at each iteration of the algorithm, we choose a vertex  $v \in N$  such that

$$L(v) = \min\{L(u) \mid u \in N\}.$$

Next the vertex  $v$  is added to  $S$ , removed from  $N$ , and for all vertices  $w \in N$  that are adjacent to  $v$ , we check whether the path from  $a$  to  $w$  via  $v$  (using the current shortest path from  $a$  to  $v$ ) is shorter than the current path from  $a$  to  $w$ . This is done by checking whether

$$L(w) > L(v) + W[v, w].$$

If this is true, then the value of  $L(w)$  is updated as explained in the following steps.

The preceding discussion translates into the following shortest path algorithm due to Dijkstra.

- $S := \emptyset$
- $N := V$
- For all vertices  $u \in V$ ,  $u \neq a$ ,  $L(u) := \infty$



**Edsger Wybe Dijkstra**  
(1930–2002)

Dijkstra was born and raised in Rotterdam, The Netherlands. His father was a chemist and his mother a mathematician. As a child he attended the Gymnasium Erasmianum. Later he attended the University of Leyden, where he was awarded degrees in mathematics and theoretical physics. He earned a Ph.D. in computer science

### Historical Notes

from the University of Amsterdam. For many years he worked as a professor in both the United States and The Netherlands. Dijkstra also spent time working as a programmer and researcher. He earned many awards, including the prestigious Turing Award for computing.

Dijkstra was held in high esteem by his many friends and colleagues, whom he would constantly challenge and for whom he would play Mozart on

his piano. Dijkstra's impact is also felt in the language of computing. Many words and phrases, such as structured programming, synchronization, weakest precondition, and deadly embrace, are attributed to him. He is also known for his work in establishing a basis for computer programming construction that uses synchronized sequential processes, his implementation of mathematical methodology, and his creation of the efficient shortest path algorithm.

4.  $L(a) := 0$
5. while  $z \notin S$  do
  - 5.a Let  $v \in N$  be such that  $L(v) = \min\{L(u) \mid u \in N\}$
  - 5.b  $S := S \cup \{v\}$
  - 5.c  $N := N - \{v\}$
  - 5.d For all  $w \in N$  such that there is an edge from  $v$  to  $w$ 
    - 5.d.1. if  $L(v) + W[v, w] < L(w)$  then  
 $L(w) = L(v) + W[v, w].$

Let us illustrate this algorithm on the graph in Figure 10.89.

After the execution of the statements in Lines 1 through 4;

$$S = \emptyset, \quad N = \{a, v_1, v_2, v_3, v_4, v_5, v_6, z\}, \quad L(a) = 0,$$

$$L(v_i) = \infty \quad \text{for all } i = 1, 2, \dots, 6, \quad \text{and} \quad L(z) = \infty.$$

Figure 10.90 shows graph  $G$ , sets  $S$  and  $N$ , and the values of the labels of the vertices after the execution of the first four statements. (In this figure, the vertices labels are shown in red.)

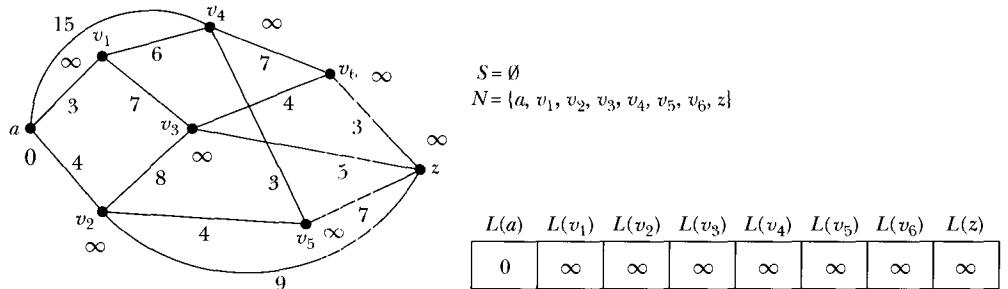


FIGURE 10.90 Graph  $G$ , sets  $S$  and  $N$ , and the labels before the first iteration

Consider the first iteration of the loop at Line 5. At Line 5.a, we choose the vertex  $a \in N$ , because  $L(a) = 0$  and the label of the other vertices is  $\infty$ . At Line 5.b,  $S = \{a\}$  and at Line 5.c,  $N = \{v_1, v_2, v_3, v_4, v_5, v_6, z\}$ . The vertices that are in  $N$  and adjacent to  $a$  are  $v_1, v_2$ , and  $v_4$ . The loop at Line 5.d updates the values of the labels of these vertices as  $L(v_1) = 3$ ,  $L(v_2) = 4$ , and  $L(v_4) = 15$ . After the first iteration of the loop at Line 5, graph  $G$ , sets  $S$  and  $N$ , and the values of the labels of the vertices are as shown in Figure 10.91. (Note that in the figure we have put a circle around vertex  $a$ , indicating that it is in set  $S$ . We will follow this convention after each iteration of the loop at Line 5.)

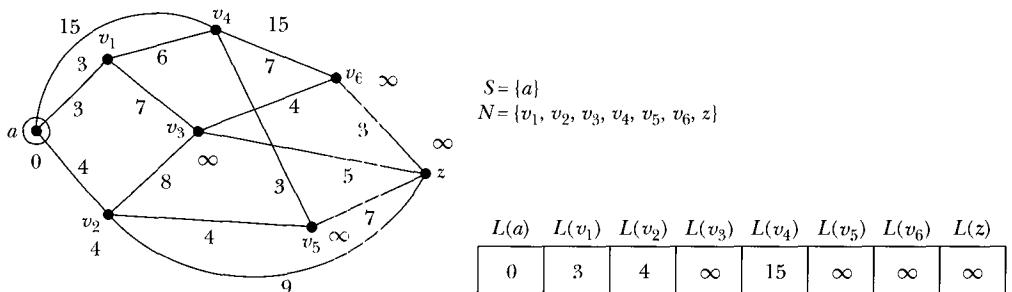


FIGURE 10.91 Graph  $G$ , sets  $S$  and  $N$ , and the labels after the first iteration

Now consider the second iteration of the loop at Line 5. At Line 5.a, we choose the vertex  $v_1 \in N$ , because

$$L(v_1) = 3 = \min\{L(u) \mid u \in N\}.$$

At Line 5.b,  $S = \{a, v_1\}$  and at Line 5.c,  $N = \{v_2, v_3, v_4, v_5, v_6, z\}$ . The vertices that are in  $N$  and adjacent to  $v_1$  are  $v_3$  and  $v_4$ . The loop at Line 5.d updates the values of the labels of these vertices as follows: Because

$$L(v_1) + W[v_1, v_3] = 3 + 7 = 10 < \infty = L(v_3),$$

the label of  $v_3$  is set to  $L(v_3) = L(v_1) + W[v_1, v_3] = 10$ . Next,

$$L(v_1) + W[v_1, v_4] = 3 + 6 = 9 < 15 = L(v_4).$$

Therefore, the label of  $v_4$  is set to  $L(v_4) = L(v_1) + W[v_1, v_4] = 9$ . After the second iteration of the loop at Line 5, graph  $G$ , sets  $S$  and  $N$ , and the values of the labels of the vertices are as shown in Figure 10.92.

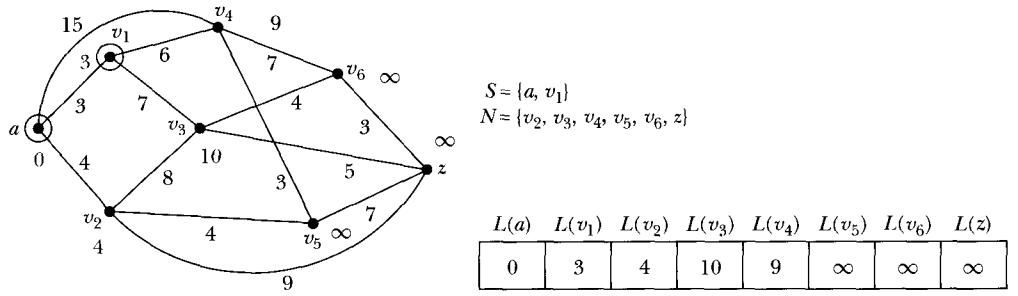


FIGURE 10.92 Graph  $G$ , sets  $S$  and  $N$ , and the labels after the second iteration

Now consider the third iteration of the loop at Line 5. At Line 5.a, we choose the vertex  $v_2 \in N$ , because

$$L(v_2) = 4 = \min\{L(u) \mid u \in N\}.$$

At Line 5.b,  $S = \{a, v_1, v_2\}$  and at Line 5.c,  $N = \{v_3, v_4, v_5, v_6, z\}$ . The vertices that are in  $N$  and adjacent to  $v_2$  are  $v_3, v_5$ , and  $z$ . The loop at Line 5.d updates the values of the labels of these vertices as follows: Because

$$L(v_2) + W[v_2, v_3] = 4 + 8 = 12 > 10 = L(v_3),$$

the label of  $v_3$  remains the same. Next,

$$L(v_2) + W[v_2, v_5] = 4 + 4 = 8 < \infty = L(v_5).$$

Therefore, the label of  $v_5$  is set to  $L(v_5) = L(v_2) + W[v_2, v_5] = 8$ . Also,

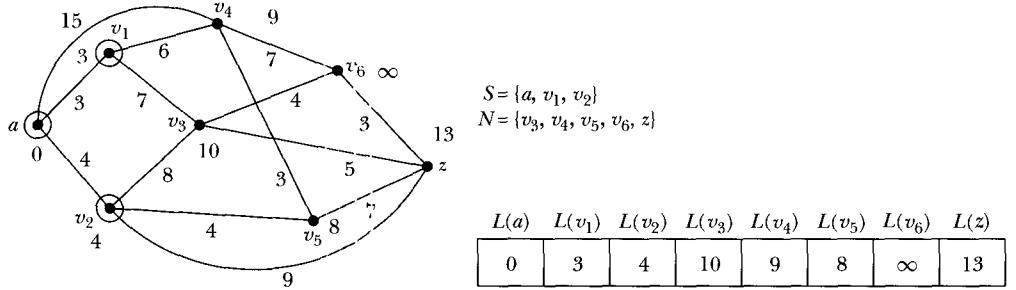
$$L(v_2) + W[v_2, z] = 4 + 9 = 13 < \infty = L(z).$$

Therefore, the label of  $z$  is set to  $L(z) = L(v_2) + W[v_2, z] = 13$ . After the third iteration of the loop at Line 5, graph  $G$ , sets  $S$  and  $N$ , and the values of the labels of the vertices are as shown in Figure 10.93.

Now consider the fourth iteration of the loop at Line 5. At Line 5.a, we choose the vertex  $v_5 \in N$ , because

$$L(v_5) = 8 = \min\{L(u) \mid u \in N\}.$$

At Line 5.b,  $S = \{a, v_1, v_2, v_5\}$  and at Line 5.c,  $N = \{v_3, v_4, v_6, z\}$ . The vertices that are in  $N$  and adjacent to  $v_5$  are  $v_4$  and  $z$ . The loop at Line 5.d updates the values

FIGURE 10.93 Graph  $G$ , sets  $S$  and  $N$ , and the labels after the third iteration

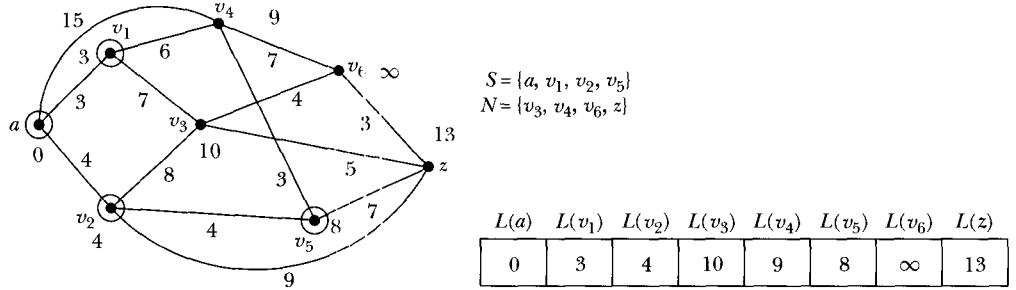
of the labels of these vertices as follows: Because

$$L(v_5) + W[v_5, v_4] = 8 + 3 = 11 > 9 = L(v_4),$$

the label of  $v_4$  remains the same. Next,

$$L(v_5) + W[v_5, z] = 8 + 7 = 15 > 13 = L(z).$$

Therefore, the label of  $z$  remains the same. After the fourth iteration of the loop at Line 5, graph  $G$ , sets  $S$  and  $N$ , and the values of the labels of the vertices are as shown in Figure 10.94.

FIGURE 10.94 Graph  $G$ , sets  $S$  and  $N$ , and the labels after the fourth iteration

Consider the fifth iteration of the loop at Line 5. At Line 5.a, we choose the vertex  $v_4 \in N$ , because

$$L(v_4) = 9 = \min\{L(u) \mid u \in N\}.$$

At Line 5.b,  $S = \{a, v_1, v_2, v_5, v_4\}$  and at Line 5.c,  $N = \{v_3, v_6, z\}$ . The vertex that is in  $N$  and adjacent to  $v_4$  is  $v_6$ . The loop at Line 5.d updates the value of the label of this vertex as follows: Because

$$L(v_4) + W[v_4, v_6] = 9 + 7 = 16 < \infty = L(v_6),$$

the label of  $v_6$  is set to  $L(v_6) = L(v_4) + W[v_4, v_6] = 16$ . After the fifth iteration of the loop at Line 5, graph  $G$ , sets  $S$  and  $N$ , and the values of the labels of the vertices are as shown in Figure 10.95.

Consider the sixth iteration of the loop at Line 5. At Line 5.a, we choose the vertex  $v_3 \in N$ , because

$$L(v_3) = 10 = \min\{L(u) \mid u \in N\}.$$

At Line 5.b,  $S = \{a, v_1, v_2, v_5, v_4, v_3\}$  and at Line 5.c,  $N = \{v_6, z\}$ . The vertices that are in  $N$  and adjacent to  $v_3$  are  $v_6$  and  $z$ . The loop at Line 5.d updates the values of the labels of these vertices as follows: Because

$$L(v_3) + W[v_3, v_6] = 10 + 4 = 14 < 16 = L(v_6),$$

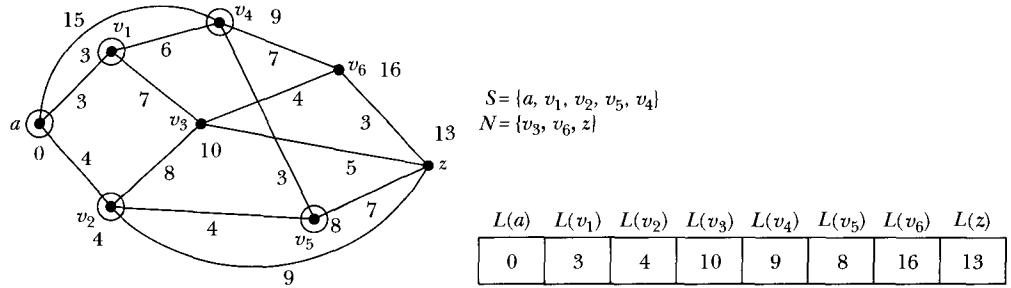


FIGURE 10.95 Graph  $G$ , sets  $S$  and  $N$ , and the labels after the fifth iteration

the label of  $v_6$  is set to  $L(v_6) = L(v_3) + W[v_3, v_6] = 14$ . Also,

$$L(z) + W[v_3, z] = 10 + 5 = 15 > 13 = L(z).$$

Therefore, the value of  $L(z)$  remains the same. After the sixth iteration of the loop at Line 5, graph  $G$ , sets  $S$  and  $N$ , and the values of the labels of the vertices are as shown in Figure 10.96.

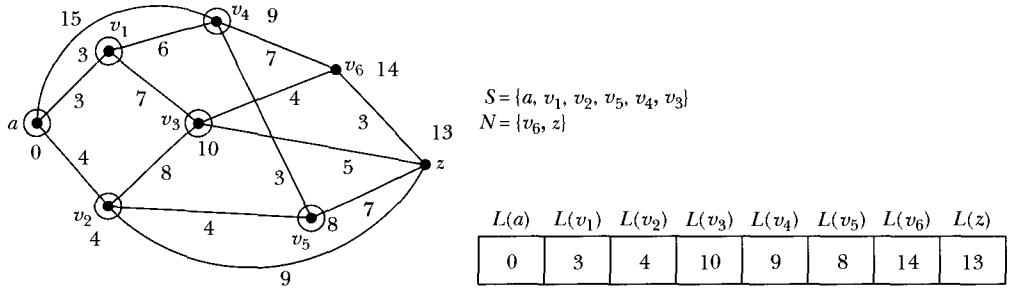


FIGURE 10.96 Graph  $G$ , sets  $S$  and  $N$ , and the labels after the sixth iteration

Consider the seventh iteration of the loop at Line 5. At Line 5.a, we choose the vertex  $z \in N$ , because

$$L(z) = 13 = \min\{L(u) \mid u \in N\}.$$

At Line 5.b,  $S = \{a, v_1, v_2, v_5, v_4, v_3, z\}$  and at Line 5.c,  $N = \{v_6\}$ . The vertex that is in  $N$  and adjacent to  $z$  is  $v_6$ . The loop at Line 5.d updates the value of the label of this vertex as follows: Because

$$L(z) + W[z, v_6] = 13 + 3 = 16 > 14 = L(v_6),$$

the label of  $v_6$  remains the same. After the seventh iteration the loop terminates because  $z \in S$  and graph  $G$ , sets  $S$  and  $N$ , and the values of the labels of the vertices are as shown in Figure 10.97.

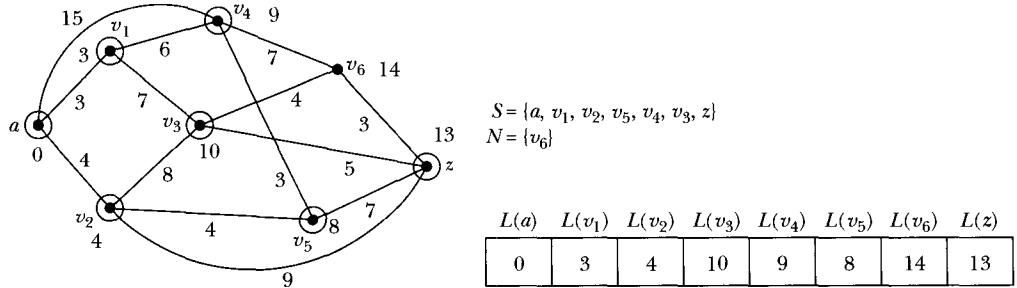


FIGURE 10.97 Graph  $G$ , sets  $S$  and  $N$ , and the labels after the seventh iteration

We can now formally write Dijkstra's shortest path algorithm as follows.

**ALGORITHM 10.1:** Dijkstra's shortest path algorithm.

*Input:*  $G$ —graph  
 $n$ —number of vertices in  $G$   
 $W$ —weight matrix  
 $a$ —source vertex  
 $z$ —destination vertex

*Output:*  $L(z)$ —the length of a shortest path from  $a$  to  $z$

```

1. function DijkstraSP( $G, W, a, z, n$ )
2. begin
3.    $S := \emptyset$ ;
4.    $N :=$  vertices in  $G$ ;
5.   for all  $u \in N$  do
6.      $L[u] := \infty$ ;
7.    $L[a] := 0$ ;
8.   while  $z \notin S$  do
9.     begin
10.     $\min := \infty$ ;
11.    for all  $u \in N$  do
12.      if  $L[u] < \min$  then
13.        begin
14.           $\min := L[u]$ ;
15.           $v := u$ ;
16.        end
17.         $S := S \cup \{v\}$ ;
18.         $N := N - \{v\}$ ;
19.        for all  $w \in N$  do
20.          if ( $v, w$ ) is an edge in  $G$ 
21.            and  $L[v] + W[v, w] < L[w]$  then
22.               $L[w] := L[v] + W[v, w]$ ;
23.        end
24.      return  $L[z]$ ;
25.    end

```

The next theorem shows that Dijkstra's algorithm correctly finds the length of a shortest path between two vertices.

**Theorem 10.6.2:** Let  $G$  be a weighted graph with  $n$  vertices,  $n > 0$ . Let  $a$  and  $z$  be two vertices in  $G$ . Dijkstra's algorithm correctly finds the length of a shortest path from vertex  $a$  to vertex  $z$ .

**Proof:** Notice that after each iteration of the while loop at Line 8, the values of  $S$ ,  $N$ , and the values of the labels of certain vertices change. Let  $S_k$  and  $N_k$  denote sets  $S$  and  $N$  after  $k$  iterations. Moreover, let  $L_k(v)$  denote the value of the label of the vertex  $v$  after  $k$  iterations. Note that once a vertex  $v$  is added to set  $S$ , in the successive iterations, the value of the label  $L(v)$  does not change, i.e., in the successive iterations, the value of  $L(v)$  becomes permanent.

By induction on  $k$ , where  $k$  is a nonnegative integer, we prove that after the  $k$ th iteration for each vertex  $v \in S$ ,  $L(v)$  is the length of a shortest path from  $a$  to  $v$ .

When  $k = 0$ , then  $S = \emptyset$ . So the result is trivially true.

Suppose  $k = 1$ . At the first iteration, the vertex  $a$  is added to  $S$ . Because  $L(a) = 0$ ,  $L(a)$  is the length of a shortest path from  $a$  to  $a$ . Moreover the shortest path from  $a$  to  $a$  consists of only the vertices from the set  $S$ . Hence, the result is true for  $k = 1$ .

*Inductive hypothesis:* Suppose the result is true for each iteration  $j$ ,  $0 \leq j \leq k - 1$ , where  $k > 1$ . Let  $S_{k-1} = \{v_{i_1}, v_{i_2}, \dots, v_{i_{k-1}}\}$ .

*Inductive step:* Consider the  $k$ th iteration.

Let  $v$  be the vertex added to  $S$  at the  $k$ th iteration. Then after  $k - 1$  iterations,  $L_{k-1}(v) = \min\{L(w) \mid w \in N\}$ . Moreover,  $S_k = S_{k-1} \cup \{v\} = \{v_{i_1}, v_{i_2}, \dots, v_{i_{k-1}}, v\}$ . By the inductive hypothesis,  $L_k(v_j)$  is the length of a shortest path from  $a$  to  $v_j$  for all  $j = 1, 2, \dots, k - 1$ .

Suppose  $L_k(v) = L_{k-1}(v)$  is not the length of a shortest path from  $a$  to  $v$ . Let  $P$  be a path from  $a$  to  $v$  such that  $l(P) < L_k(v)$ . If all the vertices of  $P$  are in  $S_k$ , then  $P$  is the path constructed by the algorithm and so  $l(P) = L_k(v)$ , because the algorithm constructs only one path from  $a$  to a vertex in  $S_k$ . So there is a vertex  $x$  on the path  $P$  such that  $x \neq v$  and  $x \notin S_k$ . Let  $u$  be the first vertex on  $P$  such that  $u \notin S_k$ . Then  $u \in N$ . Let  $y$  be the predecessor of  $u$  on  $P$ . Then  $y \in S_k$  and there is an edge from  $y$  to  $u$ . By the inductive hypothesis,  $L(y)$  is the length of a shortest path from  $a$  to  $v$ . Also, note that  $L(u) \leq L[y] + W[y, u]$ . Let us write  $P$  as  $P : a - \dots - y - u - \dots - v$ , where  $-$  denotes an edge. Now

$$L(u) \leq L[y] + W[y, u] \leq l(P) < L(v).$$

Also  $L(v) = L_k(v) = \min\{L(w) \mid w \in N\} \leq L(u)$ . Thus, we have a contradiction.

We can now conclude that the length of every path from  $a$  to  $v$  is at least  $L_k(v)$ . If  $R$  is the path from  $a$  to  $v$  constructed by the algorithm, then  $l(R) = L(v)$ .

The result now follows by induction. ■

---

**REMARK 10.6.3** ▶ Dijkstra's shortest path algorithm only gives the length of a shortest path from one vertex to another vertex. It does not output the shortest path itself. We leave it as an exercise for the reader to modify the algorithm so that it also outputs the shortest path.

**Theorem 10.6.4:** In the worst case, Dijkstra's shortest path algorithm is  $\Theta(n^2)$ .

## Topological Ordering

In college, before taking a particular course, students usually must take all of the prerequisite courses, if any. For example, before taking Programming II, a student must take Programming I. However, certain courses can be taken independently. The courses within a department can be represented as a directed graph. A directed edge from, say vertex  $u$  to vertex  $v$ , means that the course represented by vertex  $u$  is a prerequisite of the course represented by vertex  $v$ . Students need to know, before starting a major, the sequence in which they should take the courses so that before taking a course they take all its prerequisite courses and fulfill the graduation requirements on time. In this section, we describe an algorithm that can be used to output the vertices of a directed graph in such a sequence. Let us first introduce some terminology.

Let  $G$  be a directed graph and  $u$  and  $v$  be two vertices on  $G$ . If there is a path from  $u$  to  $v$ , then we say that  $u$  is a **predecessor** of  $v$  and  $v$  is a **successor** of  $u$ . If there is an edge from  $u$  to  $v$ , then we say that  $u$  is an **immediate predecessor** of  $v$  and  $v$  is an **immediate successor** of  $u$ .

Let  $G$  be a directed graph with the vertex set  $V = \{v_1, v_2, \dots, v_n\}$ , where  $n \geq 0$ . A **topological ordering** of  $V$  is a linear ordering  $v_{i1}, v_{i2}, \dots, v_{in}$  of the vertices such that, if  $v_{ij}$  is a predecessor of  $v_{ik}$ , then  $v_{ij}$  precedes  $v_{ik}$ ; that is,  $j < k$  in this linear ordering  $1 \leq j \leq n, 1 \leq k \leq n$ .

In this section, we describe an algorithm, topological order, which outputs the vertices of a directed graph in topological order. We assume that the graph has no cycles. We leave it for the reader, as an exercise, to modify the algorithm for the graphs that have cycles.

Because the graph has no cycles,

1. there exists a vertex  $v$  in  $G$  such that  $v$  has no successor, and
2. there exists a vertex  $u$  in  $G$  such that  $u$  has no predecessor.

Suppose that the array `topOrder` (of size  $n$ , the number of vertices) is used to store the vertices of  $G$  in topological order. Thus, if a vertex, say  $u$ , is a successor of the vertex  $v$  and  $\text{topOrder}[j] = v$  and  $\text{topOrder}[k] = u$ , then  $j < k$ .

The topological ordering algorithm can be implemented either using the depth-first traversal or the breadth-first traversal. This section discusses how to implement topological ordering using the breadth-first traversal. Programming Exercise 9 at the end of this chapter describes how to implement the topological ordering algorithm using the depth-first traversal.

In the breadth-first topological ordering, we first find a vertex that has no predecessor vertex and place it first in the topological ordering. We next find the vertex, say  $v$ , all of whose predecessors have been placed in the topological ordering and place  $v$  next in the topological ordering. To keep track of the number of the immediate predecessors of a vertex we use the array `predCount`. Initially,  $\text{predCount}[j]$  is the number of all of the immediate predecessors of vertex  $v_j$ .

After placing a vertex,  $v$ , in the topological ordering, the immediate predecessor count of all of its immediate successors is reduced by 1. When the immediate predecessor count of a vertex becomes 0, this vertex becomes a candidate to be

placed next in the topological ordering. When we reduce the immediate predecessor count of the immediate successors of  $v$ , it is possible that the immediate predecessor count of more than one vertex may become 0. However, we place one vertex at a time in the topological ordering. It is also possible, as we will illustrate, that when the immediate predecessor count of a vertex becomes 0, one of its predecessors may still be waiting to be placed in the topological order.

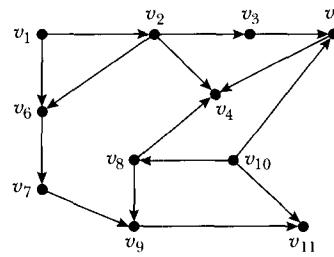
Therefore, to guide the breadth-first topological ordering, we use a data structure called a **queue**. Formally, a queue is a data structure in which elements are added at one end, called the **rear** of the queue, and removed from the other end, called the **front** of the queue. This allows us to process the items in the order they arrive.

When the immediate predecessor count of a vertex becomes 0, it is placed next in the queue. The vertex at the front of the queue is removed and placed in the topological order. It follows that when a vertex, say  $v$ , is placed in the queue, either all of its predecessors are already placed in the topological order or they are in the queue. Because vertex  $v$  is placed at the end of the queue before it is placed in the topological order, all of its predecessors are placed in the topological order.

The queue used to guide the breadth-first traversal is initialized to those vertices  $v_k$  such that  $\text{predCount}[k]$  is 0. In essence, the general algorithm is:

1. Create the array  $\text{predCount}$  and initialize it so that  $\text{predCount}[i]$  is the number of immediate predecessors of vertex  $v_i$ .
2. Initialize the queue, say  $\text{queue}$ , to all those vertices  $v_k$  so that  $\text{predCount}[k]$  is zero. (Clearly,  $\text{queue}$  is not empty because the graph has no cycles.)
3. **while** the  $\text{queue}$  is not empty
  - 3.1 Remove the front element,  $v$ , of the queue.
  - 3.2 Put  $v$  in the next available position, say  $\text{topOrder}[\text{topIndex}]$ , and increment  $\text{topIndex}$ .
  - 3.3 For all the immediate successors  $w$  of  $v$ , i.e., all the vertices  $w$  such that  $(v, w)$  is an edge
    - 3.3.1 Decrement the immediate predecessor count of  $w$  by 1.
    - 3.3.2 if the immediate predecessor count of  $w$  is zero, add  $w$  to  $\text{queue}$ .

Consider graph  $G$  in Figure 10.98. This graph has no cycles.



**FIGURE 10.98** A diagraph with no cycles

The vertices of  $G_3$  in a topological ordering are:  $v_1, v_{10}, v_2, v_8, v_3, v_6, v_5, v_7, v_4, v_9, v_{11}$ .

Next, we illustrate the use of breadth-first topological ordering to list the vertices of graph  $G$  in a topological order.

After Steps 1 and 2 execute, the arrays predCount, topOrder, and queue are as shown in Figure 10.99.

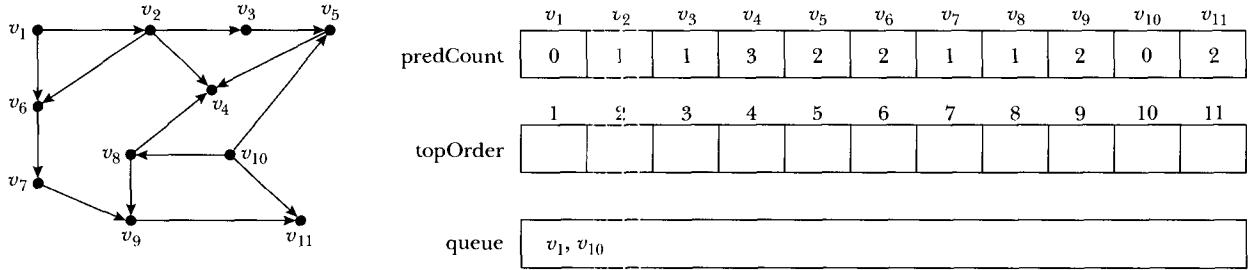


FIGURE 10.99 Arrays predCount, topOrder, and queue after Steps 1 and 2 execute

Step 3 executes as long as the queue is nonempty.

Step 3: Iteration 1: After Step 3.1 executes, the value of  $v$  is  $v_1$ . Step 3.2 stores the value of  $v$ , which is  $v_1$ , in the next available position in the array topOrder. Notice that  $v_1$  is stored at position 1 in this array. Step 3.3 reduces the predecessor count of all of the immediate successors of  $v_1$  by 1, and if the immediate predecessor count of any immediate successor of  $v_1$  reduces to 0, that vertex is pushed into the queue. The immediate successors of  $v_1$  are  $v_2$  and  $v_6$ . The immediate predecessor count of  $v_2$  reduces to 0, and the immediate predecessor count of  $v_6$  reduces to 1. Vertex  $v_2$  is pushed into the queue. After the first iteration of Step 3, the arrays predCount, topOrder, and queue are as shown in Figure 10.100.

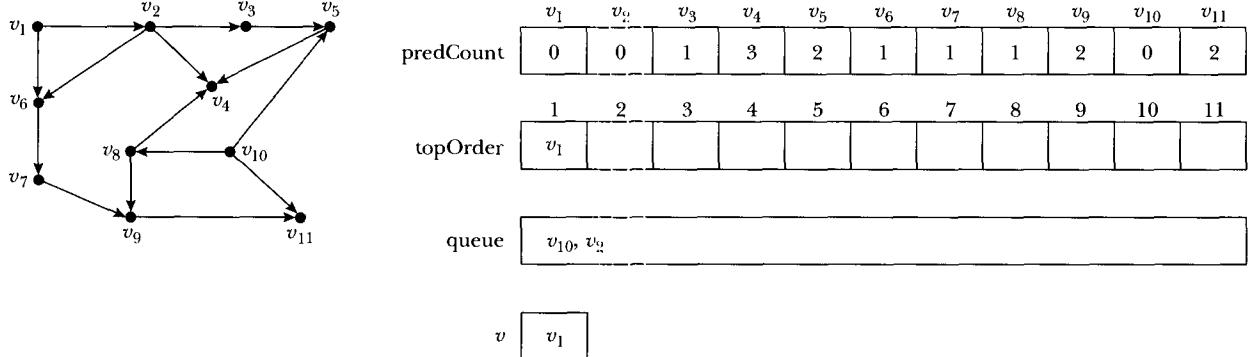


FIGURE 10.100 Arrays predCount, topOrder, and queue after the first iteration of Step 3

Step 3: Iteration 2: The queue is nonempty. After Step 3.1 executes, the value of  $v$  is  $v_{10}$ . Step 3.2 stores the value of  $v$ , which is  $v_{10}$ , in the next available position in the array topOrder. Notice that vertex  $v_{10}$  is stored at position 2 in this array. Step 3.3 reduces the immediate predecessor count of all of the immediate successors of  $v_{10}$  by 1, and if the immediate predecessor count of any immediate successor of  $v_{10}$  reduces to 0, that vertex is pushed into the queue. The immediate successors of  $v_{10}$  are  $v_5$ ,  $v_8$ , and  $v_{11}$ . The immediate predecessor count of  $v_8$  reduces to 0 and the immediate predecessor count of  $v_5$  and  $v_{11}$  reduces to 1. Vertex  $v_8$  is pushed into queue. After the second iteration of Step 3, the arrays predCount, topOrder, and queue are as shown in Figure 10.101.

Step 3: Iteration 3: The queue is nonempty. After Step 3.1 executes, the value of  $v$  is  $v_2$ . Step 3.2 stores the value of  $v$ , which is  $v_2$ , in the next available position in the array topOrder. Notice that  $v_2$  is stored at position 3 in this array.

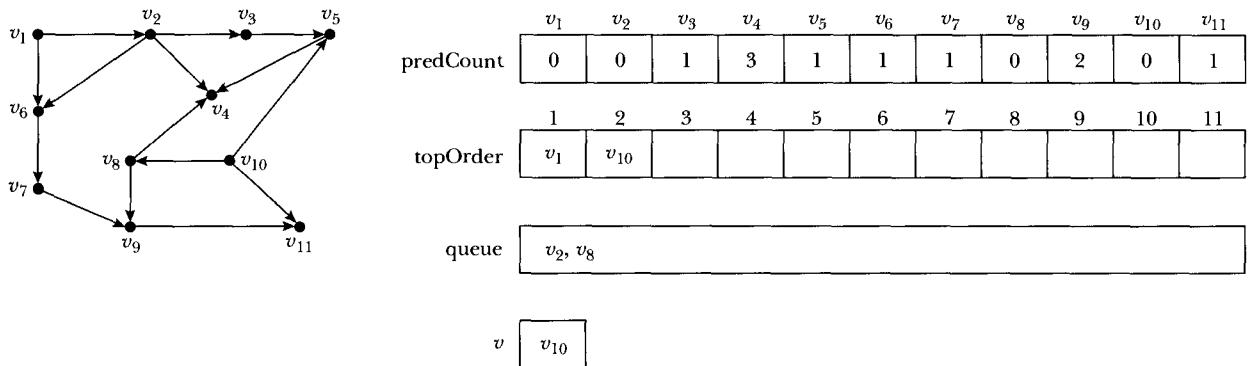


FIGURE 10.101 Arrays predCount, topOrder, and queue after the second iteration of Step 3

Step 3.3 reduces the immediate predecessor count of all the immediate successors of vertex  $v_2$  by 1 and if the immediate predecessor count of any immediate successors of  $v_2$  reduces to 0, that vertex is added into the queue. The immediate successors of  $v_2$  are  $v_3$ ,  $v_4$ , and  $v_6$ . The immediate predecessor count of  $v_3$  and  $v_6$  reduces to 0 and the immediate predecessor count of  $v_4$  reduces to 2. Vertices  $v_3$  and  $v_6$ , in this order, are pushed into the queue. After the third iteration of Step 3, the arrays predCount, topOrder, and queue are as shown in Figure 10.102.

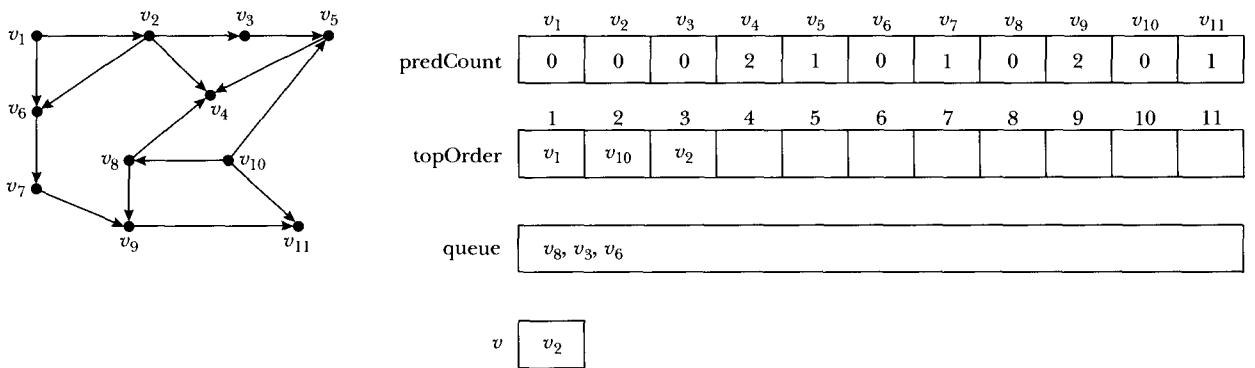


FIGURE 10.102 Arrays predCount, topOrder, and queue after the third iteration of Step 3

If you repeat Step 3 eight more times, the arrays predCount, topOrder, and queue are as shown in Figure 10.103.

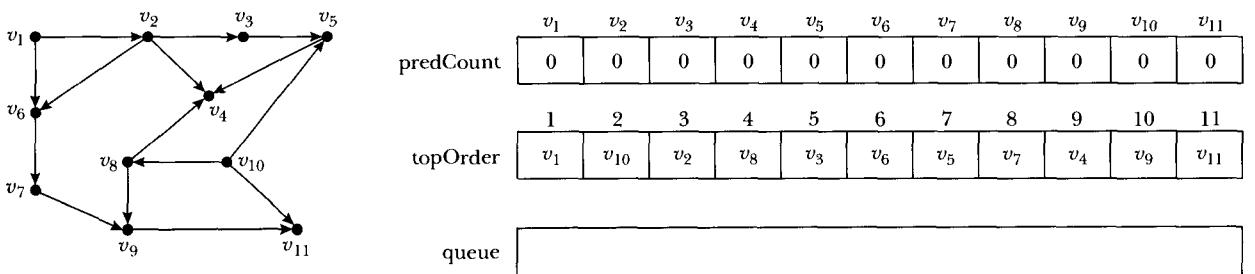


FIGURE 10.103 Arrays predCount, topOrder, and queue after Step 3 executes

In Figure 10.103, the array topOrder shows the breadth-first topological ordering of the nodes of graph  $G$ .

The following procedure implements the breadth-first topological ordering algorithm described in this section.

**ALGORITHM 10.2: Topological Ordering.**

*Input:*  $G$ —graph  
 $n$ —number of vertices in  $G$

*Output:* Vertices of  $G$  in topological order

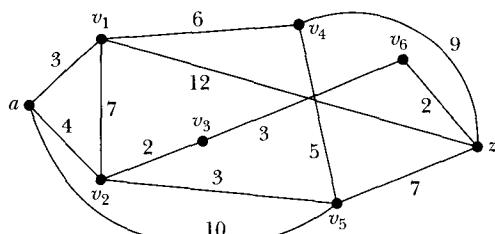
```

1. procedure bfTopologicalOrder( $G$ ,  $n$ )
2. begin
3.   topIndex := 1;
4.   for  $i := 1$  to  $n$  do
5.     predCount[ $i$ ] := 0;
6.   for  $v := 1$  to  $n$  do
7.     begin
8.       for each successor  $u$  of  $v$  do
9.         predCount[ $u$ ] := predCount[ $u$ ] + 1;
10.    end
11.   for  $v := 1$  to  $n$  do
12.     if predCount[ $v$ ] := 0 then
13.       addQueue(queue,  $v$ );
14.   while queue is not empty do
15.     begin
16.        $v :=$  first element of queue;
17.       remove the first element of queue;
18.       topOrder[topIndex] :=  $v$ ;
19.       topIndex := topIndex + 1;
20.       for each successor  $w$  of  $u$  do
21.         begin
22.           predCount[ $w$ ] := predCount[ $w$ ] - 1;
23.           if predCount[ $w$ ] = 0 then
24.             addQueue(queue,  $w$ );
25.         end
26.       end
27.       //output the vertices in topological order
28.     for  $i := 1$  to  $n$  do
29.       print topOrder[ $i$ ];
30.   end

```

# WORKED-OUT EXERCISES

**Exercise 1:** Find the weight matrix of the graph in Figure 10.104.



**FIGURE 10.104** A weighted graph

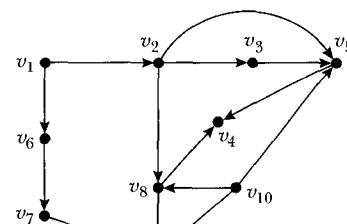
### Solution:

|       | $a$      | $v_1$    | $v_2$    | $v_3$    | $v_4$    | $v_5$    | $v_6$    | $z$      |
|-------|----------|----------|----------|----------|----------|----------|----------|----------|
| $a$   | 0        | 3        | 4        | $\infty$ | $\infty$ | 10       | $\infty$ | $\infty$ |
| $v_1$ | 3        | 0        | 7        | $\infty$ | 6        | $\infty$ | $\infty$ | 12       |
| $v_2$ | 4        | 7        | 0        | 2        | $\infty$ | 3        | $\infty$ | $\infty$ |
| $v_3$ | $\infty$ | $\infty$ | 2        | 0        | $\infty$ | $\infty$ | 3        | $\infty$ |
| $v_4$ | $\infty$ | 6        | $\infty$ | $\infty$ | 0        | 5        | $\infty$ | 9        |
| $v_5$ | 10       | $\infty$ | 3        | $\infty$ | 5        | 0        | $\infty$ | 7        |
| $v_6$ | $\infty$ | $\infty$ | $\infty$ | 3        | $\infty$ | $\infty$ | 0        | 2        |
| $z$   | $\infty$ | 12       | $\infty$ | $\infty$ | 9        | 7        | 2        | 0        |

**Exercise 2:** In the graph in Figure 10.104, find the length of a shortest path from vertex  $a$  to vertex  $z$ . What is such a shortest path?

**Solution:** The length of a shortest path from  $a$  to  $z$  is 11. The path  $a - v_2 - v_3 - v_6 - z$  is a shortest path from  $a$  to  $z$ .

**Exercise 3:** Consider the graph in Figure 10.105.



**FIGURE 10.105** A diagraph.

- (a) Consider the algorithm `bfTopologicalOrder` as given in this section. Show the array `predCount` after the statements in Lines 6 through 10 execute.
  - (b) List the vertices of this graph in breadth-first topological order.

**Solution:**

- (a) The array predCount is

| $v_1$ | $v_2$ | $v_3$ | $v_4$ | $v_5$ | $v_6$ | $v_7$ | $v_8$ | $v_9$ | $v_{10}$ |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|----------|
| 0     | 1     | 1     | 2     | 3     | 1     | 1     | 2     | 3     | 0        |

- (b) A breadth-first topological ordering of this graph is:  
 $v_1, v_{10}, v_2, v_6, v_3, v_8, v_7, v_5, v_9, v_4$ .

## SECTION REVIEW

### **Key Terms**

weight  
weighted graph  
weight matrix  
length of a path

shortest path algorithm  
greedy algorithm  
topological ordering  
immediate successor

queue  
rear  
front

## **Some Key Definitions**

- Let  $G$  be a graph with  $n$  vertices, where  $n > 0$ . Let  $V = \{v_1, v_2, \dots, v_n\}$  be the vertex set of  $G$ . We list the vertices of  $G$  as  $v_1, v_2, \dots, v_n$ . Let  $W$  be an  $n \times n$  matrix such that its  $(i, j)$ th entry, for  $i \neq j$ ,  $W[i, j]$  is given by

$$W[i, j] = \begin{cases} w_{ij} & \text{if } v_i - v_j \text{ is an edge in } G \text{ and } w_{ij} \text{ is the weight of the edge } v_i - v_j, \\ \infty & \text{if there is no edge from } v_i \text{ to } v_j. \end{cases}$$

Also,  $W[i, i] = 0$  for all  $i$ . Matrix  $W$  is called the weight matrix of graph  $G$ .

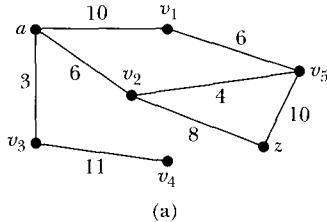
2. Let  $G$  be a weighted graph. Let  $u$  and  $v$  be two vertices in  $G$ , and let  $P$  be a path in  $G$  from  $u$  to  $v$ . The length of path  $P$ , written  $l(P)$ , is the sum of the weights of all of the edges on path  $P$ , which is also called the length of  $v$  from  $u$  via  $P$ .
3. Let  $G$  be a directed graph with the vertex set  $V = \{v_1, v_2, \dots, v_n\}$ , where  $n \geq 0$ . A topological ordering of  $V$  is a linear ordering  $v_{i1}, v_{i2}, \dots, v_{in}$  of the vertices such that, if  $v_{ij}$  is a predecessor of  $v_{ik}$ , then  $v_{ij}$  precedes  $v_{ik}$ ; that is,  $j < k$  in this linear ordering  $1 \leq j \leq n$ ,  $1 \leq k \leq n$ .

## Some Key Results

1. Let  $G$  be a graph with  $n$  vertices,  $n > 0$ . Let  $a$  and  $z$  be two vertices in  $G$ . Dijkstra's algorithm correctly finds the length of a shortest path between two vertices.
2. In the worst case, Dijkstra's shortest path algorithm is  $\Theta(n^2)$ .

## EXERCISES

1. Find the weight matrix of the graphs in Figure 10.106.



(a)

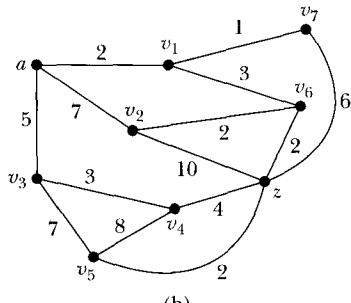


FIGURE 10.106 Weighted graphs

2. Use Dijkstra's shortest path algorithm to find the length of a shortest path from vertex  $a$  to vertex  $z$  in the graphs of Figure 10.106.
3. Trace the execution of Dijkstra's shortest path algorithm to find the length of a shortest path from vertex  $a$  to vertex  $z$  in the graph in Figure 10.107.

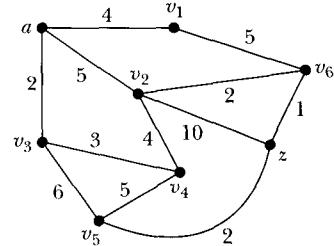


FIGURE 10.107 A weighted graph

4. Prove Theorem 10.6.4.
5. Dijkstra's shortest path algorithm as given in this section only finds the length of a shortest path between two vertices. Modify the algorithm so that it can also output the shortest path.
6. Dijkstra's shortest path algorithm as given in this section only finds the length of a shortest path between two vertices. Modify the algorithm so that it finds the shortest path from a given vertex to any other vertex in a simple connected graph.
7. List the vertices of the graphs in Figure 10.108 in breadth-first topological order.

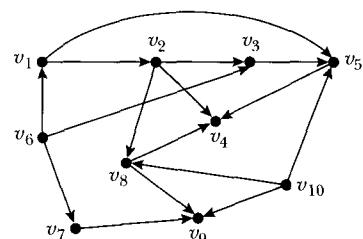
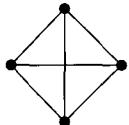


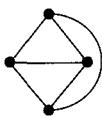
FIGURE 10.108 A digraph

## 10.7 PLANAR GRAPHS AND GRAPH COLORING

In this section, we discuss planer graphs and graph coloring.



**FIGURE 10.109**  
A graph



**FIGURE 10.110**  
Planar graph

### Planar Graphs

Recall the utility problem stated at the beginning of this chapter: Suppose there are three houses,  $H_1$ ,  $H_2$ , and  $H_3$ , each of which is to be connected to the centers of three companies,  $C_1$ ,  $C_2$ , and  $C_3$ , which supply water, telephone service, and electrical service. If a graph satisfying the given requirements is drawn, then the graph is a complete  $K_{3,3}$  graph. For example,  $H_1$  is connected with  $C_1$ ,  $C_2$ , and  $C_3$ . Now the problem is the following: Can we draw this graph in the plane such that no two edges intersect except at the vertices, which may be the common end vertices of the edges? In this section we consider such problems.

Let us consider the graph in Figure 10.109.

We can redraw the graph in Figure 10.109 as follows (see Figure 10.110).

In this graph, no two edges intersect except at the vertices, which may be the common end vertices of the edges.

**DEFINITION 10.7.1** ► A graph  $G$  is called a **planar graph** if it can be drawn in the plane such that no two edges intersect except at the vertices, which may be the common end vertices of the edges.

**DEFINITION 10.7.2** ► A graph drawn in the plane (on paper or a chalkboard) is called a **plane graph** if no two edges meet at any point except the common vertex, if they meet at all.

From Definitions 10.7.1 and 10.7.2, it follows that a graph is a planar graph if and only if it has a pictorial representation in a plane which is a plane graph. This pictorial representation of a planar graph  $G$  as a plane graph is called a **planar representation** of  $G$ .

Consider the planar representation, shown in Figure 10.111, of a planar graph.

Let  $G$  denote the plane graph in Figure 10.111. Graph  $G$ , in Figure 10.111, divides the plane into different regions, called the **faces** of  $G$ . Suppose  $x$  is a point in the plane that is not a vertex of  $G$  or a point on any edge of  $G$ . Then a face of  $G$  containing  $x$  is the set of all points on the plane that can be reached from  $x$  by a straight line or a curved line that does not cross any edge of  $G$  or pass through any vertex of  $G$ . Thus, it follows that a face is a region produced by a planar graph that is an area of the plane bounded by the edges and that is not further subdivided into subareas.

The set of edges that bound a region is called its **boundary**. Of course, there exists a region of infinite area in any plane graph  $G$ . This is the part of the plane that lies outside the planar representation of  $G$ . This region is called the **exterior face**. A face that is not exterior is called an **interior face**. We illustrate these concepts by the following planar representations of some planar graphs.

### EXAMPLE 10.7.3

Consider the graph in Figure 10.112.

This plane graph divides the plane into three regions.

Region 1.  $R_1$  : Bounded by the cycle  $(v_1, e_1, v_2, e_2, v_3, e_6, v_1)$ . The boundary of  $R_1$  consists of the edges  $e_1$ ,  $e_2$ , and  $e_6$ .

Region 2.  $R_2$  : Bounded by the cycle  $(v_4, e_4, v_5, e_5, v_3, e_3, v_4)$ . The boundary of  $R_2$  consists of the edges  $e_3$ ,  $e_4$ , and  $e_5$ .

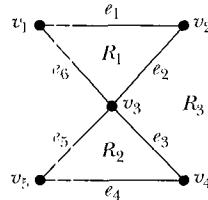


FIGURE 10.112 A graph

Both of these regions,  $R_1$  and  $R_2$ , are interior regions.

Region 3.  $R_3$  : The part of the plane outside this plane graph. The boundary of the region consists of the edges  $e_1, e_6, e_5, e_2, e_4$ , and  $e_3$ .

It follows that this planar graph has three faces,  $R_1, R_2$ , and  $R_3$ .

For this planar graph, the number of edges  $n_e = 6$ , the number of vertices  $n_v = 5$ , the number of faces  $n_f = 3$ , and we find that  $n_v - n_e + n_f = 5 - 6 + 3 = 2$ .

#### EXAMPLE 10.7.4

Consider the graph in Figure 10.113.

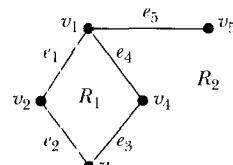


FIGURE 10.113 A graph

This is a plane graph. This plane graph divides the plane into two regions.

Region 1.  $R_1$  : Bounded by the cycle  $(v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_1)$ . This is an interior region. The boundary consists of edges  $e_1, e_2, e_3$ , and  $e_4$ .

Region 2.  $R_2$  : The exterior region. The boundary consists of all of the edges of the graph.

This planar graph has two faces,  $R_1$  and  $R_2$ .

For this planar graph, the number of edges  $n_e = 5$ , the number of vertices  $n_v = 5$ , and the number of faces  $n_f = 2$ . Notice that  $n_v - n_e + n_f = 5 - 5 + 2 = 2$ .

#### EXAMPLE 10.7.5

Consider graph  $G$  in Figure 10.114.

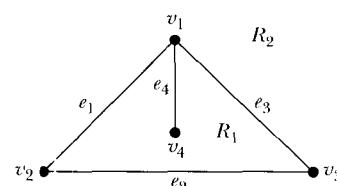


FIGURE 10.114 Planar graph

Graph  $G$  is a planar representation of a planar graph. This plane graph divides the plane into two regions.

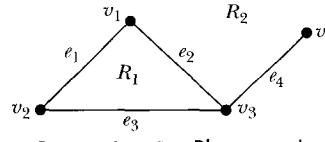
Region 1.  $R_1$  : This is an interior region. The boundary consists of the edges  $e_1, e_2, e_3$ , and  $e_4$ .

Region 2.  $R_2$  : The exterior region. The boundary consists of the edges  $e_1$ ,  $e_2$ , and  $e_3$ .

For this planar graph, the number of edges  $n_e = 4$ , the number of vertices  $n_v = 4$ , and the number of faces  $n_f = 2$ . Notice that  $n_v - n_e + n_f = 4 - 4 + 2 = 2$ .

**EXAMPLE 10.7.6**

Consider the graph in Figure 10.115.



**FIGURE 10.115** Planar graph

This graph is a planar representation of a planar graph. This plane graph divides the plane into two regions.

Region 1.  $R_1$  : This is an interior region. The boundary consists of the edges  $e_1$ ,  $e_2$ , and  $e_3$ .

Region 2.  $R_2$  : The exterior region. The boundary consists of the edges  $e_1$ ,  $e_2$ ,  $e_3$ , and  $e_4$ .

For this planar graph, the number of edges  $n_e = 4$ , the number of vertices  $n_v = 4$ , and the number of faces  $n_f = 2$ . Notice that

$$n_v - n_e + n_f = 4 - 4 + 2 = 2.$$

In the connected planar graphs of Examples 10.7.3 through 10.7.6, we find the relation  $n_v - n_e + n_f = 2$  among  $n_v$ ,  $n_e$ , and  $n_f$ . In 1752, this result, for any connected planar graph, was proved by Euler.

**Theorem 10.7.7: Euler.** Let  $G$  be a connected planar graph with  $n_v$  vertices,  $n_e$  edges, and  $n_f$  faces. Then  $n_v - n_e + n_f = 2$ .

**Proof:** We prove the theorem by induction on  $n_e$ .

*Basis step:* Let  $n_e = 0$ . Then it has only one vertex and one region, which is obviously the exterior region. Then  $n_v - n_e + n_f = 1 - 0 + 1 = 2$ .

*Inductive hypothesis:* Let  $k$  be a positive integer. Assume that  $n_v - n_e + n_f = 2$  for any connected planar graph with  $n_e = k - 1$ .

*Inductive step:* Let  $G$  be a connected planar graph with  $n_e = k$  edges and  $n_v = t$  vertices. Suppose  $G$  has no cycles. Then  $G$  has no interior region, which implies that the exterior region is the only region for this planar graph. Therefore,  $n_f = 1$ . We now show that  $G$  contains a vertex of degree 1. Choose a vertex  $v$  in  $G$ . If  $\deg(v) = 1$ , we are done. Suppose  $\deg(v) > 1$ . Let  $v_1$  be an adjacent vertex of  $v$ . Because  $G$  has no cycles,  $G$  is loop free and hence  $v_1$  is different from  $v$ . If  $\deg(v_1) = 1$ , we are done. Suppose  $\deg(v_1) > 1$ . Let  $v_2$  be an adjacent vertex of  $v_1$ . Because  $G$  has no cycles,  $v_2$  is different from  $v$  and  $v_1$ . If  $\deg(v_2) \neq 1$ , we find an adjacent vertex  $v_3$  of  $v_2$  different from  $v$ ,  $v_1$ , and  $v_2$ . Because  $G$  has a finite number of vertices, it follows that  $G$  has a vertex  $u$  of degree 1. We now delete this vertex and the only edge that is incident

with this vertex and thus we form a new connected planar graph  $H$  with  $k - 1$  edges and  $t - 1$  vertices. By the inductive hypothesis, for this graph  $H$ , we have  $n_v - n_e + n_f = 2$ . Hence,  $(t - 1) - (k - 1) + n_f = 2$ , which implies that  $t - k + n_f = 2$ , i.e.,  $n_v - n_e + n_f = 2$  holds in  $G$ .

Suppose now that  $G$  has a cycle  $C$ . Let  $e$  be an edge in  $C$ . Now construct a new graph,  $G_1 = G - \{e\}$ . This is still a connected planar graph. For this planar graph  $G_1$ , we compute  $n_v$ ,  $n_e$ , and  $n_f$ . Let  $n_f = m$ . In the construction of  $G_1$ , we delete only the edge  $e$  without deleting any vertices. Therefore,  $n_v = t$ ,  $n_e = k - 1$ . Now  $C - \{e\}$  is not a cycle in  $G_1$ . Therefore, the edges of  $C - \{e\}$  will not form a boundary in  $G_1$ . Thus, in  $G_1$ ,  $n_f = m - 1$ . Hence,  $G_1$  is a connected planar graph with  $n_v = t$  vertices,  $n_e = k - 1$  edges, and  $n_f = m - 1$  faces. By the inductive hypothesis, it follows that  $t - (k - 1) + (m - 1) = 2$ . This implies that  $t - k + m = 2$ . Hence,  $n_v - n_e + n_f = 2$ .

The result now follows by induction. ■

In the next corollary, we use Theorem 10.7.7 to prove that  $K_{3,3}$  is not a planar graph.

**Corollary 10.7.8:** The graph  $K_{3,3}$  is not a planar graph.

**Proof:** The graph of  $K_{3,3}$  is shown in Figure 10.116.

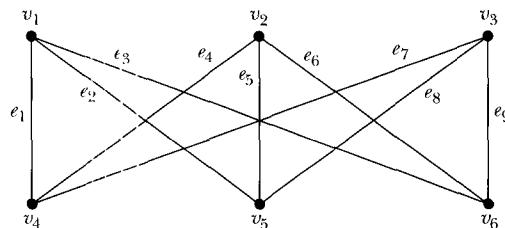


FIGURE 10.116  $K_{3,3}$

Suppose that  $K_{3,3}$  is a planar graph. Then it has a planar representation. This planar representation divides the plane into different regions. Note that an edge of the graph may appear at most in boundaries of two different regions. For any such planar representation, by Euler's theorem we have  $n_v - n_e + n_f = 2$ . For  $K_{3,3}$ ,  $n_v = 6$ ,  $n_e = 9$ . Then  $6 - 9 + n_f = 2$ , which implies that  $n_f = 5$ . Now  $K_{3,3}$  does not contain any triangles, but it contains cycles of length 4. For example  $(v_1, e_1, v_4, e_4, v_2, e_6, v_6, e_3, v_1)$ . Therefore, the total number of appearances of the edges in boundaries of five faces is  $\geq 5 \cdot 4 = 20$ . In counting these appearances, an edge may be counted at most two times. Thus, the total number of appearances of the nine edges in boundaries is  $\leq 18$ . Thus, we arrive at a contradiction. Hence,  $K_{3,3}$  is not a planar graph. ■

---

**REMARK 10.7.9** ► From Corollary 10.7.8, we find the answer to the utilities problem. It follows that the pipes cannot be laid so that they do not cross each other.

**Theorem 10.7.10:** Let  $G$  be a connected simple planar graph with  $n_v \geq 3$  vertices and  $n_e$  edges. Then

$$n_e \leq 3n_v - 6.$$

**Proof:** Because  $G$  is a planar graph, it has a planar representation. Consider a planar representation of  $G$ . Suppose  $n_v = 3$ . Because  $G$  is a simple connected graph with three vertices, it follows that  $n_e \leq 3$ . Then  $n_e \leq 3 \cdot 3 - 6$ , which implies that  $n_e \leq 3n_v - 6$ .

Suppose now  $n_v > 3$ . If  $G$  does not contain any cycles, then we can show that  $n_e = n_v - 1$ . Now  $3n_v - 6 = (n_v - 1) + (n_v - 2) + (n_v - 3) > (n_v - 1) = n_e$ .

Suppose  $G$  contains a cycle. Because  $G$  is simple, it may contain a cycle with three edges. Thus, the number of edges in the boundary of a face is  $\geq 3$ . Now there are  $n_f$  faces and every edge is a member of some boundary of the planar representation. Hence, the total number of appearances of the edges in boundaries of  $n_f$  faces is  $\geq n_f \cdot 3$ . In counting these appearances, an edge may be counted at most two times. Thus, the total number of appearances of the  $n_e$  edges in boundaries is  $\leq 2n_e$ . Hence,  $n_f \cdot 3 \leq 2n_e$ . Now by Euler's Theorem,

$$\begin{aligned} n_v - n_e + n_f &= 2 \\ \Rightarrow 3n_v - 3n_e + 3n_f &= 6 \\ \Rightarrow 3n_e &= 3n_v + 3n_f - 6 \\ \Rightarrow 3n_e &\leq 3n_v + 2n_e - 6 \\ \Rightarrow n_e &\leq 3n_v - 6. \quad \blacksquare \end{aligned}$$

**Corollary 10.7.11:** The graph  $K_5$  is not a planar graph.

**Proof:** The graph of  $K_5$  is shown in Figure 10.117.

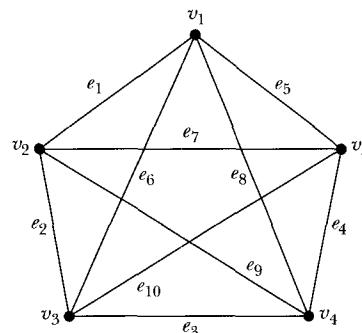


FIGURE 10.117  $K_5$

$K_5$  is a connected simple graph with 10 edges and five vertices. Suppose  $K_5$  is a planar graph. Then by Theorem 10.7.10,  $n_e \leq 3n_v - 6$ . Thus, for  $K_5$ , we find that  $10 \leq 3 \cdot 5 - 6 = 9$ . This is a contradiction. Hence,  $K_5$  is not a planar graph. ■

Let  $G = (V, E)$  be a graph. Suppose that  $e$  is an edge with  $v_1, v_2$  as end vertices. Construct the subgraph  $G_1 = G - \{e\}$ . To construct  $G_1$  we have deleted edge  $e$  without deleting any vertices from  $G$ . We now construct a new graph,  $G_2 = (V_2, E_2)$ , by taking  $V_2 = V \cup \{w\}$ ,  $E_2 = (E - \{e\}) \cup \{f_1, f_2\}$  such that  $w \notin V, f_1, f_2 \notin E, v_1, w$  are end vertices of  $f_1$ , and  $v_2, w$  are end vertices of  $f_2$ . The process of obtaining  $G_2$  from  $G$  is called a one-step subdivision of an edge of  $G$ .

---

**DEFINITION 10.7.12** ▶ A graph  $H$  is said to be a **subdivision of a graph**  $G$  if there exist graphs  $H_1, H_2, \dots, H_{n-1}, H_n$ , such that  $H_0 = G$ ,  $H_n = H$ , and  $H_i$  is obtained from  $H_{i-1}$  by a one-step subdivision of an edge of  $H_{i-1}$  for  $i = 1, 2, \dots, n$ .

If a graph  $H$  is a subdivision of a graph  $G$ , then we say that  $H$  is obtained from  $G$  by subdividing edges of  $G$ .

**EXAMPLE 10.7.13**

Consider graphs  $G$  and  $H$  in Figure 10.118.

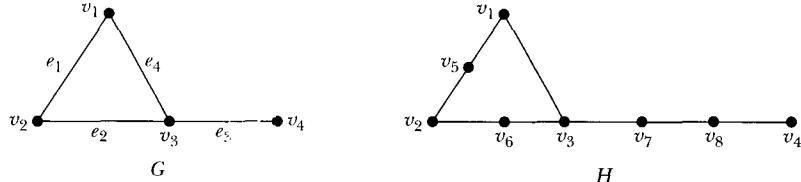


FIGURE 10.118 Graphs  $G$  and  $H$

We find that  $H$  is obtained from  $G$  by a finite sequence of subdivisions of edges.  $H$  is obtained from  $G$  by dividing edge  $e_1$  one time,  $e_2$  one time, and  $e_3$  two times.

**DEFINITION 10.7.14** ▶

Two graphs  $G$  and  $H$  are said to be **homeomorphic** graphs if there is an isomorphism from a subdivision of  $G$  to a subdivision of  $H$ .

We illustrate this definition by the following example.

**EXAMPLE 10.7.15**

Consider graphs  $G$  and  $H$  in Figure 10.119.

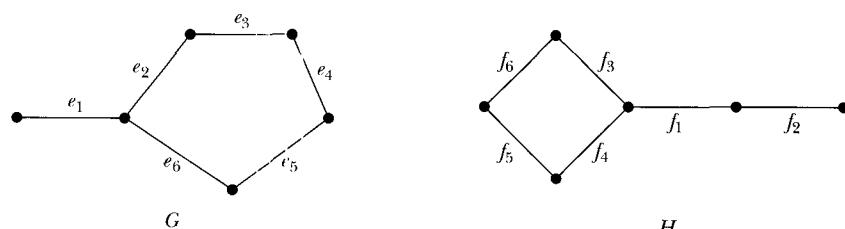
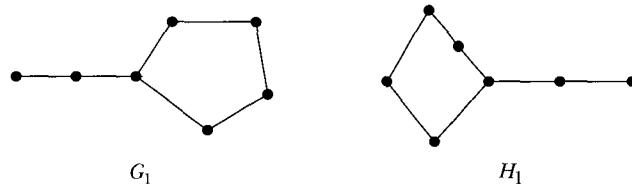


FIGURE 10.119 Graphs  $G$  and  $H$

We see that  $G$  contains a cycle of length 5, and  $H$  contains a cycle of length 4. Hence, these two graphs are not isomorphic. But we find a subdivision  $G_1$  of  $G$  and a subdivision  $H_1$  of  $H$  such that  $G_1$  and  $H_1$  are isomorphic. (See Figure 10.120.)

Hence,  $G$  and  $H$  are homeomorphic graphs.

FIGURE 10.120 Graphs  $G_1$  and  $H_1$ 

We proved that  $K_{3,3}$  and  $K_5$  are not planar graphs. In 1930, Kuratowski proved the following famous theorem, characterizing simple planar graphs in terms of these graphs.

**Theorem 10.7.16: Kuratowski.** A simple graph is planar if and only if it does not contain a subgraph homeomorphic to  $K_5$  or  $K_{3,3}$ .

Using Kuratowski's Theorem, we prove that the Petersen graph  $G$ , shown in Figure 10.121, is not a planar graph.

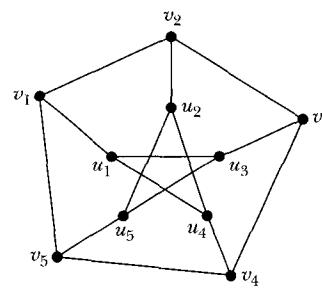


FIGURE 10.121 A graph

**Kazimierz Kuratowski**  
(1896–1980)

Kuratowski grew up in Poland during a highly volatile time.

As a child and young adult, the part of Poland he lived in was under Russian dominion. Education, beyond the primary level, was available only in Russian. A clandestine college, the Underground University of Warsaw, existed and provided a Polish education, but it was illegal and the risks of attendance were high. Many Poles traveled to Galicia (a part of Poland under Austrian control) or abroad for their education. Kuratowski traveled to the University of Glasgow in Scotland to study engineer-

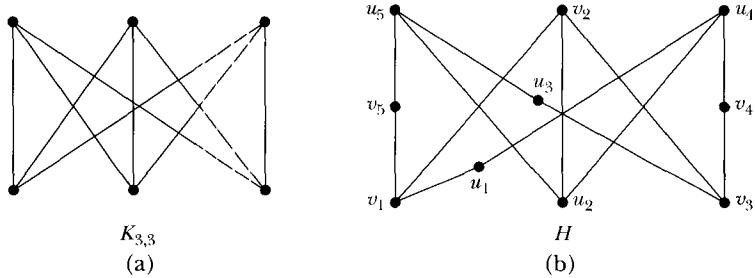
### Historical Notes

ing. However, in 1914, while home for summer recess, World War I commenced, which made his return to Scotland impossible.

In 1915, as war raged through Europe, the Russians withdrew from Poland, and the University of Warsaw openly flourished as a Polish university. It was here that Kuratowski continued his education, this time focused on mathematics. In 1921, Kuratowski earned his Ph.D. and accepted a post at the Technical University of Lvov. After teaching at Lvov for several years, Kuratowski resigned his position to teach at the University of Warsaw. In addition to teaching, Kuratowski became active in the organization of Polish mathematics abroad.

Kuratowski is best known for his work in topology, nonplanar graphs, and set theory.

For this we show that  $G$  contains a subgraph homeomorphic to  $K_5$  or  $K_{3,3}$ . Now  $K_5$  contains vertices of degree 4, but  $G$  contains no vertices of degree 4. Hence,  $G$  cannot contain any subgraph homeomorphic to  $K_5$ . Let us now try to find a subgraph of  $G$  homeomorphic to  $K_{3,3}$ . We find that graph  $H$  (see Figure 10.122) is a subgraph of  $G$  and  $H$  is a subdivision of graph  $K_{3,3}$ .



**FIGURE 10.122** Graphs  $K_{3,3}$  and  $H$

Hence, by Kuratowski's Theorem, we find that the Peterson graph  $G$  is not a planar graph.

## Graph Coloring

Let  $G = (V, E)$  be a simple graph and  $C = \{c_1, c_2, \dots, c_n\}$  be a set of  $n$  colors.

- (i) A **vertex coloring** of  $G$  using the colors of  $C$  is a function  $f : V \rightarrow C$ .
- (ii) Let  $f : V \rightarrow C$  be a vertex coloring of  $G$ . If for every adjacent vertices  $u, v \in V, f(u) \neq f(v)$ , then  $f$  is called a **proper vertex coloring**.

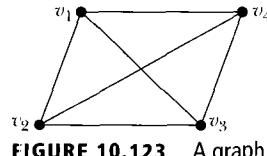
For each vertex  $v$ , its image  $f(v)$  is called the *color* of  $v$ .

It follows that a vertex coloring of a graph  $G$  is an assignment of the colors  $c_1, c_2, \dots, c_n$  to the vertices of graph  $G$ . Similarly, a proper vertex coloring of  $G$  is an assignment of the colors  $c_1, c_2, \dots, c_n$  to the vertices of graph  $G$  such that adjacent vertices have different colors.

### EXAMPLE 10.7.17

Consider the graph in Figure 10.123. This is a graph with four vertices,  $v_1, v_2, v_3$ , and  $v_4$ . Suppose  $C = \{R, B, Y, G\}$ , where  $R$  denotes red,  $B$  denotes blue,  $Y$  denotes yellow, and  $G$  denotes green. Define  $f : V \rightarrow C$  by

$$\begin{aligned} f : v_1 &\mapsto R \\ v_2 &\mapsto G \\ v_3 &\mapsto Y \\ v_4 &\mapsto B \end{aligned}$$



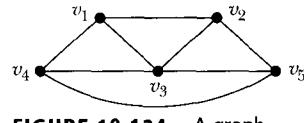
**FIGURE 10.123** A graph

Notice that  $f$  is a proper coloring with four colors.

The proper coloring of graph  $G$  of Example 10.7.17 uses four colors. We pose the following question: Is there a proper vertex coloring of  $G$  that uses less than four colors? Before we answer this question, let us consider a few more examples of vertex coloring.

**EXAMPLE 10.7.18**

Consider the graph in Figure 10.124 with five vertices  $v_1, v_2, v_3, v_4$ , and  $v_5$ .



**FIGURE 10.124** A graph

Let  $C = \{R, B, G\}$ . Define  $f : V \rightarrow C$  by

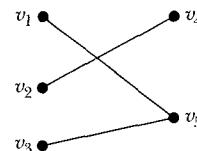
$$\begin{aligned} f : v_1 &\mapsto R \\ v_2 &\mapsto B \\ v_3 &\mapsto G \\ v_5 &\mapsto R \\ v_4 &\mapsto B \end{aligned}$$

Notice that  $f$  is a proper coloring of this graph with three colors. Is there a proper vertex coloring of this graph that uses less than three colors?

---

**DEFINITION 10.7.19** ► The smallest number of colors needed to make a proper vertex coloring of a simple graph  $G$  is called the **chromatic number** of  $G$ . The chromatic number of  $G$  is denoted by  $\chi(G)$ .

Next, we determine the chromatic number of bipartite graphs. Consider graph  $G$  in Figure 10.125.



**FIGURE 10.125**  
A bipartite graph

Now  $G$  is a bipartite graph with bipartition  $V_1 \cup V_2$ , where  $V_1 = \{v_1, v_2, v_3\}$  and  $V_2 = \{v_4, v_5\}$ . Let  $C = \{R, B\}$  be a set of two colors. Define a function  $f : V \rightarrow C$  such that  $f(v) = R$  if  $v \in V_1$  and  $f(v) = B$  if  $v \in V_2$ . Now no two vertices in  $V_1$  are adjacent and no two vertices in  $V_2$  are adjacent. It follows that  $f$  is a proper vertex coloring. Thus,  $\chi(G) \leq 2$ . Because  $G$  contains edges that are not loops, it follows that  $\chi(G) \neq 1$ . Hence,  $\chi(G) = 2$ .

We prove this result for any nontrivial bipartite graph.

**Theorem 10.7.20:** Let  $G$  be a nontrivial simple graph. Then  $\chi(G) = 2$  if and only if  $G$  is a bipartite graph.

**Proof:** Let  $G = (V, E)$  be a bipartite graph. Then vertex set  $V$  can be partitioned into two nonempty subsets  $V_1$  and  $V_2$  such that each edge of  $G$  is incident with one vertex in  $V_1$  and one vertex in  $V_2$ . Let  $C = \{c_1, c_2\}$  be a set of two colors.

Define a function  $f : V \rightarrow C$  such that  $f(v) = c_1$  if  $v \in V_1$  and  $f(v) = c_2$  if  $v \in V_2$ . Because  $V_1 \cap V_2 = \emptyset$ , it follows that  $f$  is well defined. Now no two vertices of  $V_1$  are adjacent. Therefore, all vertices can have the same color. Similarly, all vertices of  $V_2$  can have the same color. From the definition of  $f$ , it follows that two adjacent vertices of  $G$  have different colors. Thus,  $\chi(G) \leq 2$ . Because  $G$  contains at least one edge,  $\chi(G) > 1$ . Hence,  $\chi(G) = 2$ .

Conversely, suppose  $\chi(G) = 2$ . This implies that the graph contains at least one edge. Also, there exists a function  $f : V \rightarrow C = \{c_1, c_2\}$  such that no two adjacent vertices have the same image.

Let  $V_1 = \{v \in V \mid f(v) = c_1\}$  and  $V_2 = \{v \in V \mid f(v) = c_2\}$ . It follows that  $V_1 \cap V_2 = \emptyset$  and  $V_1 \cup V_2 = V$ . Let  $e$  be an edge with end vertices  $v_1$  and  $v_2$ . Because  $v_1$  and  $v_2$  cannot have the same color,  $v_1 \in V_1$  if and only if  $v_2 \in V_2$ . Hence,  $G$  is a bipartite graph. ■

---

**DEFINITION 10.7.21** ▶ Let  $G$  be a graph with vertices  $v_1, v_2, \dots, v_{n-1}, v_n$ . The maximum of the integers  $\deg(v_i)$ ,  $i = 1, 2, \dots, n$  is denoted by  $\Delta(G)$ . That is,

$$\Delta(G) := \max\{\deg(v_i) \mid i = 1, 2, \dots, n\}.$$

**EXAMPLE 10.7.22**

Consider graph  $G$  in Figure 10.126.

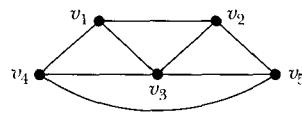


FIGURE 10.126 A graph

In  $G$ ,  $\deg(v_1) = 3$ ,  $\deg(v_2) = 3$ ,  $\deg(v_3) = 4$ ,  $\deg(v_4) = 3$ ,  $\deg(v_5) = 3$ . Hence,  $\Delta(G) = 4$ . As in Example 10.7.18,  $\chi(G) \leq 3$ .

The following theorem shows how the two numbers  $\chi(G)$  and  $\Delta(G)$  for a simple graph are related.

**Theorem 10.7.23:** For any simple graph  $G$ ,  $\chi(G) \leq \Delta(G) + 1$ .

**Proof:** We prove this theorem by induction on  $n$ , where  $n$  is the number of vertices of  $G$ .

*Basis step:* Let  $n = 1$ . Then  $G$  is a graph with only one vertex and  $G$  has no edge. Hence,  $\chi(G) = 1$  and  $\Delta(G) = 0$ . This implies that  $\chi(G) \leq \Delta(G) + 1$  for  $n = 1$ .

*Inductive hypothesis:* Suppose that  $k > 1$  is an integer such that for any simple graph  $G$ , with  $k - 1$  vertices,  $\chi(G) \leq \Delta(G) + 1$ .

*Inductive step:* Let  $G$  be a simple graph with  $k$  vertices. Consider a vertex  $v$  of  $G$  and construct the graph  $G_1 = G - \{v\}$ . The graph  $G_1$  is obtained from  $G$  by deleting the vertex  $v$  and also deleting all the edges incident with  $v$ . Clearly,  $\Delta(G_1) \leq \Delta(G)$ . This is a simple graph with  $k - 1$  vertices. Thus, by the inductive

hypothesis  $\chi(G_1) \leq \Delta(G_1) + 1$ . Then  $\chi(G_1) \leq \Delta(G) + 1$ . This implies that  $G_1$  can be properly colored by at most  $\Delta(G_1) + 1$  colors. Now  $v$  has at most  $\Delta(G)$  adjacent vertices. Because  $\Delta(G) < \Delta(G) + 1$ , it follows that not all the  $\Delta(G) + 1$  colors are needed to color these  $\Delta(G)$  adjacent vertices. Thus, from these  $\Delta(G) + 1$  colors one unused color is definitely available to color vertex  $v$ . Hence,  $\chi(G) \leq \Delta(G) + 1$ . ■

Let  $G = (V, E)$  be a simple graph and  $C = \{c_1, c_2, \dots, c_n\}$  be a set of  $n$  colors.

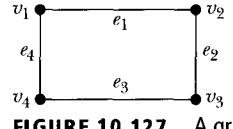
- (i) An **edge coloring** of  $G$  using the colors of  $C$  is a function  $f : E \rightarrow C$ .
- (ii) Let  $f : E \rightarrow C$  be an edge coloring of  $G$ . If for every two edges  $e_1$  and  $e_2$  meeting at a common vertex  $f(e_1) \neq f(e_2)$ , then  $f$  is called a **proper edge coloring**.

For each edge  $e$ , its image  $f(e)$  is called the *color* of  $e$ .

It follows that a proper edge coloring of a graph  $G$  is an assignment of the colors  $c_1, c_2, \dots, c_n$  to the edges of graph  $G$  such that any two edges meeting at a common vertex have different colors.

#### EXAMPLE 10.7.24

Consider the graph in Figure 10.127 with four edges,  $e_1, e_2, e_3$ , and  $e_4$ , and four vertices,  $v_1, v_2, v_3$ , and  $v_4$ .



**FIGURE 10.127** A graph

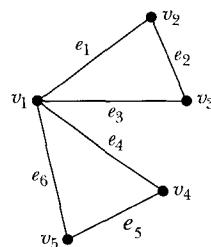
Suppose  $C = \{R, G\}$ , where  $R$  denotes red and  $G$  denotes green. Define  $f : E \rightarrow C$  by

$$\begin{aligned} f : e_1 &\mapsto R \\ e_2 &\mapsto G \\ e_3 &\mapsto R \\ e_4 &\mapsto G \end{aligned}$$

This is a proper edge coloring with two colors. Is there a proper edge coloring of this graph that uses less than two colors?

#### EXAMPLE 10.7.25

Consider the graph in Figure 10.128 with six edges,  $e_1, e_2, e_3, e_4, e_5$ , and  $e_6$ .



**FIGURE 10.128** A graph

Suppose  $C = \{R, B, Y, G\}$ , where  $R$  denotes red,  $B$  denotes blue,  $Y$  denotes yellow, and  $G$  denotes green. Define  $f : E \rightarrow C$  by

$$\begin{aligned} f : e_1 &\mapsto R \\ e_3 &\mapsto G \\ e_4 &\mapsto B \\ e_6 &\mapsto Y \\ e_2 &\mapsto B \\ e_5 &\mapsto R \end{aligned}$$

It follows that  $f$  is a proper edge coloring of the graph of Figure 10.128 with four colors. Is there a proper edge coloring of  $G$  that uses less than four colors?

---

**DEFINITION 10.7.26** ► The smallest number of colors needed to make a proper coloring of edges of a simple graph  $G$  is called the **chromatic index** of  $G$ . The chromatic index of  $G$  is denoted by  $\chi'(G)$ .

For a simple graph we have the following theorem.

**Theorem 10.7.27:** For any simple graph  $G$ ,  $\chi'(G) = \Delta(G)$  or  $\chi'(G) = \Delta(G) + 1$ .

Let us now consider the map-coloring problem mentioned in the beginning of this chapter.

Consider the map of a country consisting of 10 states (see Figure 10.129).

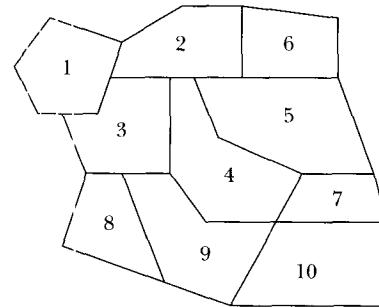


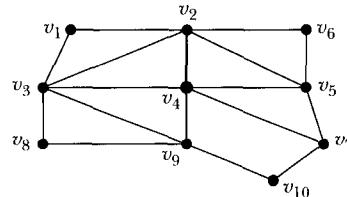
FIGURE 10.129 A map of 10 states

Can we color this map using only four colors so that no two states having a common boundary have the same color?

Corresponding to this map we can associate a graph  $G = (V, E)$  as follows: Let  $V = \{v_1, v_2, \dots, v_{10}\}$ , where  $v_i$  represents the  $i$ th state  $i = 1, 2, \dots, 10$ . Two vertices,  $v_i$  and  $v_j$ , are considered to be adjacent if  $i \neq j$ , and  $v_i$  and  $v_j$  have a common boundary. That is, two vertices,  $v_i$  and  $v_j$ ,  $i \neq j$ , are the end vertices of an edge  $e$ , if and only if the  $i$ th state and the  $j$ th state have a common boundary. Two states meeting at a single point are not considered adjacent. For example, in Figure 10.129, state 7 and state 9 are not adjacent. Similarly, state 4 and state 10 are not adjacent. Graph  $G$  is shown in Figure 10.130.

Notice that  $G$  is a simple graph.

The map-coloring problem (Figure 10.129) reduces to the vertex-coloring problem in the graph in Figure 10.130.

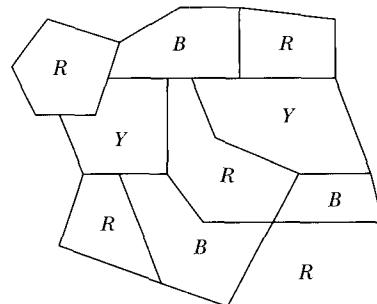


**FIGURE 10.130** A graph representing the states in Figure 10.129

First we directly show how the map in Figure 10.129 can be colored using just four colors. Then we accomplish the same result using vertex coloring of the corresponding graph.

Now coloring the map must satisfy the conditions that every state is colored by exactly one color and no two adjacent states are colored by the same color. Let us now try to color the given map. Let us assign the colors as follows (see Figure 10.131).

|                 |                   |                   |
|-----------------|-------------------|-------------------|
| Red to state 1  | Blue to state 2   | Yellow to state 3 |
| Red to state 4  | Yellow to state 5 | Red to state 6    |
| Blue to state 7 | Red to state 8    | Blue to state 9   |
| Red to state 10 |                   |                   |



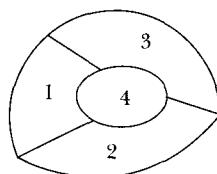
**FIGURE 10.131** Map with colors assigned to states

Thus, the map (Figure 10.129) can be colored by four different colors.

Now we consider the graph  $G = (V, E)$  in Figure 10.130. Let  $C = \{R, B, Y\}$  be a set of three different colors. Define  $f : V \rightarrow C$  by  $f(v_1) = R, f(v_2) = B, f(v_3) = Y, f(v_4) = R, f(v_5) = Y, f(v_6) = R, f(v_7) = B, f(v_8) = R, f(v_9) = B$ , and  $f(v_{10}) = R$ .

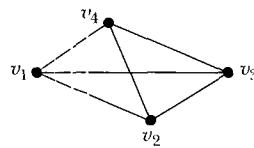
We can verify that no two adjacent vertices have the same color. Thus,  $f$  is a proper coloring of  $G = (V, E)$ . Hence,  $\chi(G) \leq 3$ .

Next consider the map shown in Figure 10.132.



**FIGURE 10.132** A map

Because states 1, 2, and 3 share common boundaries with state 4, it follows that four different colors are necessary to color this map. The graphical representation of this map is the graph in Figure 10.133.



**FIGURE 10.133** Graphical representation of the map in Figure 10.132

We can show that for this graph  $G$ ,  $\chi(G) = 4$ .

From these examples, it follows that solving the map-coloring problem is equivalent to showing that for any connected simple graph  $G$ ,  $\chi(G) = 4$ .

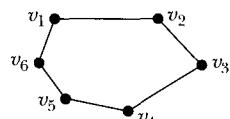
## WORKED-OUT EXERCISES

**Exercise 1:** In a connected simple planar graph, show that there exists a vertex  $v$  such that  $\deg(v) \leq 5$ .

**Solution:** In a connected simple graph we know that  $n_e \leq 3n_v - 6$ . Suppose  $\deg(v) \geq 6$  for all vertices  $v$ . Now  $\sum \deg(v) = 2n_e$ . Hence,  $2n_e \geq 6n_v$ . Again,  $2n_e \leq 3n_v - 6$ . This implies  $6n_v \leq 6n_v - 12$ . Thus, we find that  $0 \leq -12$ . This is a contradiction. Hence, there exists a vertex  $v$  such that  $\deg(v) \leq 5$ .

**Exercise 2:** For the cycle  $C_6$ , find  $\chi(C_6)$ .

**Solution:** The pictorial representation of  $C_6$  is shown in Figure 10.134.



**FIGURE 10.134**  $C_6$

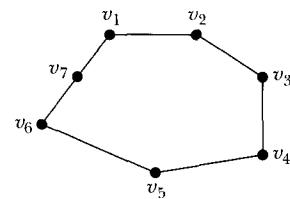
The vertices of  $C_6$  are  $v_1, v_2, v_3, v_4, v_5$ , and  $v_6$ . Let  $C = \{B, R, G\}$ . Define  $f : V \rightarrow C$  by

$$\begin{aligned} f : v_1 &\mapsto R \\ v_2 &\mapsto G \\ v_3 &\mapsto R \\ v_4 &\mapsto G \\ v_5 &\mapsto R \\ v_6 &\mapsto G \end{aligned}$$

This is a proper coloring of  $G$ . Thus,  $\chi(C_6) \leq 2$ . But  $\chi(C_6) > 1$ . Hence,  $\chi(C_6) = 2$ .

**Exercise 3:** For the cycle  $C_7$ , find  $\chi(C_7)$ .

**Solution:** The pictorial representation of  $C_7$  is shown in Figure 10.135.



**FIGURE 10.135**  $C_7$

The vertices of  $C_7$  are  $v_1, v_2, v_3, v_4, v_5, v_6$ , and  $v_7$ . Let  $C = \{B, R, G\}$ . Define  $f : V \rightarrow C$  by

$$\begin{aligned} v_1 &\mapsto R \\ v_2 &\mapsto G \\ v_3 &\mapsto R \\ v_4 &\mapsto G \\ v_5 &\mapsto R \\ v_6 &\mapsto G \\ v_7 &\mapsto B \end{aligned}$$

Because  $v_7$  and  $v_1$  are adjacent vertices, we cannot assign the same color to  $v_7$  and  $v_1$ . Thus, for  $v_7$  we need a color different from the colors assigned to  $v_6$  and  $v_1$ . Hence, we find that we need three and only three distinct colors for proper coloring of  $C_7$ . Hence,  $\chi(C_7) = 3$ .

**Exercise 4:** For the graph  $K_n$ , find  $\chi(K_n)$ .

**Solution:**  $K_n$  is a complete graph with  $n$  vertices. For any vertex  $v$  of  $K_n$  each of the remaining  $n - 1$  vertices is an adjacent vertex of  $v$ . Hence, we need  $n$  distinct colors for proper coloring of  $K_n$ . Then  $\chi(K_n) \geq n$ . But  $K_n$  has  $n$  vertices. Hence,  $\chi(K_n) = n$ .

**Exercise 5:** For the graph  $K_{2,3}$ , find  $\chi(K_{2,3})$ .

**Solution:**  $K_{2,3}$  is a complete bipartite graph with five vertices,  $v_1, v_2, u_1, u_2$ , and  $u_3$ . Let us draw this graph (see Figure 10.136).

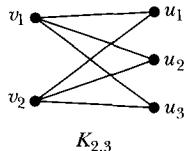


FIGURE 10.136  $K_{2,3}$

Here we find that  $u_1, u_2$ , and  $u_3$  are adjacent vertices of  $v_1$  and also that  $u_1, u_2$ , and  $u_3$  are adjacent vertices of  $v_2$ . Let  $C = \{G, R\}$  be the set of two colors. Define  $f : V \rightarrow C$  by

$$\begin{aligned} f : v_1 &\mapsto R \\ u_1 &\mapsto G \\ u_2 &\mapsto G \\ u_3 &\mapsto G \\ v_2 &\mapsto R \end{aligned}$$

This is a proper coloring of  $G$ . Hence,  $\chi(K_{2,3}) = 2$ .

**Exercise 6:** Find  $\chi(G)$  for the graph in Figure 10.137.

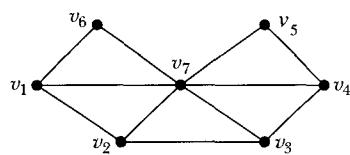


FIGURE 10.137

**Solution:** Let  $C = \{G, R, Y\}$  be the set of three colors. Define  $f : V \rightarrow C$  by

$$\begin{aligned} f : v_1 &\mapsto R \\ v_2 &\mapsto G \\ v_3 &\mapsto R \\ v_4 &\mapsto G \\ v_5 &\mapsto R \\ v_6 &\mapsto G \\ v_7 &\mapsto Y \end{aligned}$$

This is a proper coloring of  $G$ . Hence,  $\chi(G) \leq 3$ . Now  $v_7$  is adjacent to both  $v_1$  and  $v_2$ . We need a third color for  $v_7$ . Hence,  $\chi(G) = 3$ .

**Exercise 7:** Find the number of colors required to color the map in Figure 10.138 so that no two adjacent regions are colored by the same color.

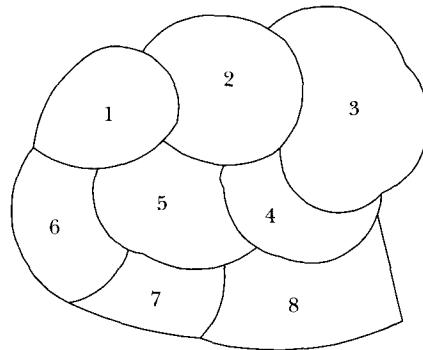


FIGURE 10.138  $G$   
A map

**Solution:** Corresponding to this map we associate a graph  $G = (V, E)$ . For this we let  $V = \{v_1, v_2, \dots, v_8\}$ , where  $v_i$  represents the  $i$ th state,  $i = 1, 2, \dots, 8$ . Now two vertices,  $v_i$  and  $v_j$ , are considered to be adjacent if  $i \neq j$  and  $v_i$  and  $v_j$  have a common boundary. Hence, two vertices,  $v_i$  and  $v_j$ , are the end vertices of an edge  $e$  if and only if the  $i$ th state and the  $j$ th state have a common boundary. Then we have the following pictorial representation of this graph (see Figure 10.139). This is a simple graph.

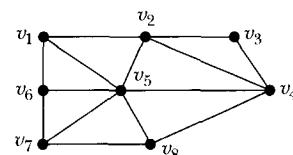


FIGURE 10.139 Graphic representation of the map in Figure 10.138

Now we consider the graph  $G = (V, E)$ . Let  $C = \{R, B, G\}$  be a set of three different colors. Define  $f : V \rightarrow C$  by  $f(v_1) = R, f(v_2) = B, f(v_3) = G, f(v_4) = R, f(v_5) = G, f(v_6) = B, f(v_7) = R$ , and  $f(v_8) = B$ .

Notice that no two adjacent vertices have the same color. Thus, we obtain a proper coloring of  $G = (V, E)$ . Hence,  $\chi(G) \leq 3$ . Now  $v_1, v_2, v_4, v_8, v_7, v_6$ , and  $v_1$  form a cycle of length 6. Therefore, by Worked-Out Exercise 2, we need two distinct colors for these vertices. Now  $v_5$  is adjacent to each of these vertices but  $v_5$  is not adjacent to  $v_3$ . Vertex  $v_3$  is adjacent to  $v_2$  and  $v_4$  and has been assigned a different color than the color assigned to  $v_2$  and  $v_4$ . Therefore, to  $v_5$  we can assign the same color as the color of  $v_3$ . Therefore, we need one more color. Hence,  $\chi(G) = 3$ .

## SECTION REVIEW

---

### Key Terms

|                                  |                        |                        |
|----------------------------------|------------------------|------------------------|
| planar graph                     | exterior face          | proper vertex coloring |
| plane graph                      | interior face          | chromatic number       |
| planar representation of a graph | subdivision of a graph | edge coloring          |
| faces                            | homeomorphic           | proper edge coloring   |
| boundary                         | vertex coloring        | chromatic index        |

### Some Key Definitions

1. A graph  $G$  is called a planar graph if it can be drawn in the plane such that no two edges intersect except at the vertices, which may be the common end vertices of the edges.
2. A graph drawn in the plane (on paper or a chalkboard) is called a plane graph if no two edges meet at a point except the common vertex, if they meet at all.
3. A graph  $H$  is said to be a subdivision of a graph  $G$  if there exist graphs  $H_1, H_2, \dots, H_{n-1}, H_n$ , such that  $H_0 = G$ ,  $H_n = H$ , and  $H_i$  is obtained from  $H_{i-1}$  by a one-step subdivision of an edge of  $H_{i-1}$  for  $i = 1, 2, \dots, n$ .
4. Two graphs  $G$  and  $H$  are said to be homeomorphic graphs if there is an isomorphism from a subdivision of  $G$  to a subdivision of  $H$ .
5. Let  $G = (V, E)$  be a simple graph and  $C = \{c_1, c_2, \dots, c_n\}$  a set of  $n$  colors.
  - (i) A vertex coloring of  $G$  using the colors of  $C$  is a function  $f : V \rightarrow C$ .
  - (ii) Let  $f : V \rightarrow C$  be a vertex coloring of  $G$ . If for every adjacent vertices  $u, v \in V$ ,  $f(u) \neq f(v)$ , then  $f$  is called a proper vertex coloring.
6. For each vertex  $v$ , its image  $f(v)$  is called the color of  $v$ .
7. Let  $G = (V, E)$  be a simple graph and  $C = \{c_1, c_2, \dots, c_n\}$  be a set of  $n$  colors.
  - (i) An edge coloring of  $G$  using the colors of  $C$  is a function  $f : E \rightarrow C$ .
  - (ii) Let  $f : E \rightarrow C$  be an edge coloring of  $G$ . If for every two edges  $e_1$  and  $e_2$  meeting at a common vertex  $f(e_1) \neq f(e_2)$ , then  $f$  is called a proper edge coloring.
8. The smallest number of colors needed to make a proper coloring of edges of a simple graph  $G$  is called the chromatic index of  $G$ . The chromatic index of  $G$  is denoted by  $\chi'(G)$ .
9. The smallest number of colors needed to make a proper vertex coloring of a simple graph  $G$  is called the chromatic number of  $G$ . The chromatic number of  $G$  is denoted by  $\chi(G)$ .
10. Let  $G$  be a graph with vertices  $v_1, v_2, \dots, v_{n-1}, v_n$ . Then the maximum of the integers  $\deg(v_i)$ ,  $i = 1, 2, \dots, n$  is denoted by  $\Delta(G)$ . That is,

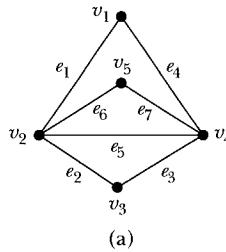
$$\Delta(G) = \max\{\deg(v_i) \mid i = 1, 2, \dots, n\}.$$

## Some Key Results

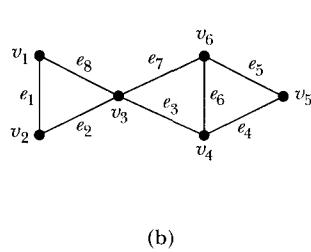
1. Let  $G$  be a connected planar graph with  $n_v$  vertices,  $n_e$  edges, and  $n_f$  faces. Then  $n_v - n_e + n_f = 2$ .
2. The graph  $K_{3,3}$  is not a planar graph.
3. Let  $G$  be a connected simple planar graph with  $n_v \geq 3$  vertices and  $n_e$  edges. Then  $n_e \leq 3n_v - 6$ .
4. The graph  $K_5$  is not a planar graph.
5. A simple graph is planar if and only if it does not contain a subgraph homeomorphic to  $K_5$  or  $K_{3,3}$ .
6. Let  $G$  be a nontrivial simple graph. Then  $\chi(G) = 2$  if and only if  $G$  is a bipartite graph.
7. For any simple graph  $G$ ,  $\chi'(G) = \Delta(G)$  or  $\chi'(G) = \Delta(G) + 1$ .

## EXERCISES

1. For the planar graphs in Figure 10.140 find the number of faces and list the edges of the boundaries.



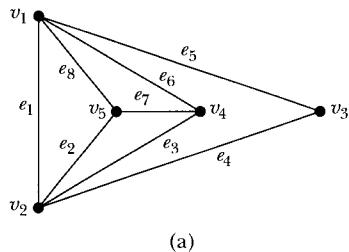
(a)



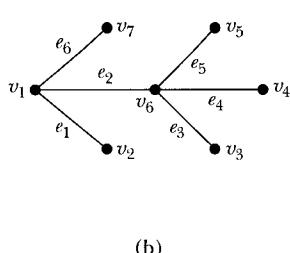
(b)

**FIGURE 10.140** Planar graphs

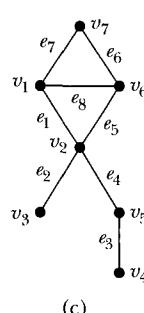
2. Does there exist a simple connected planar graph with 35 vertices and 100 edges?  
 3. For the planar graphs in Figure 10.141 find the number of faces and list the edges of the boundaries.



(a)



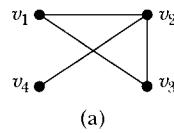
(b)



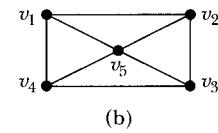
(c)

**FIGURE 10.141** Planar graphs

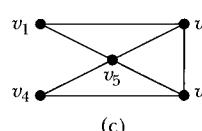
4. Find  $\chi(G)$  for each of the graphs in Figure 10.142.



(a)



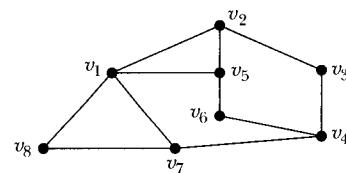
(b)



(c)

**FIGURE 10.142** Graphs

5. Find  $\chi(G)$  of graph in Figure 10.143.  
 6. For the cycle  $C_8$ , find  $\chi(C_8)$ .  
 7. For the cycle  $C_9$ , find  $\chi(C_9)$ .  
 8. For any integer  $n \geq 2$ , write a formula for  $\chi(C_n)$ , where  $C_n$  is a cycle of length  $n$ . Justify your answer.  
 9. Let  $G$  be a simple graph. Then prove that  $\chi(G) \geq 3$  if and only if  $G$  has an odd cycle.  
 10. For any simple connected planar graph  $G$ , prove that  $\chi(G) \leq 6$ .  
 11. Find the number of colors required to color the map in Figure 10.144 so that no two adjacent regions are colored by the same color.



**FIGURE 10.143** A graph

**FIGURE 10.144** Planar graphs

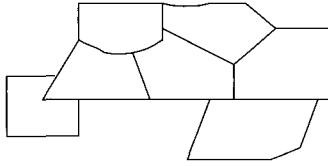


FIGURE 10.144 A map

12. Find  $\chi'(K_5)$  and  $\chi'(K_6)$ .
13. Find  $\chi'(K_{2,3})$  and  $\chi'(C_6)$ .
14. If  $G$  is a bipartite graph, then prove that  $\chi'(G) = \Delta(G)$ .

## ► PROGRAMMING EXERCISES

---

1. Write a program that takes as input a graph. The program outputs the degree of each vertex.
2. Write a program that takes as input a directed graph. The program outputs the indegree and outdegree of each vertex.
3. Write a program that takes as input a graph, a pair of vertices, and the length of a walk. The program outputs the number of walks of the specified length between the two vertices.
4. Write a program that takes as input a simple graph. The program then determines if the graph is bipartite. If the graph is bipartite, then the program outputs the sets  $V_1$  and  $V_2$  of the bipartition; otherwise the program outputs that the graph is not bipartite.
5. Write a program that takes as input a graph. The program then determines if the graph is Eularian. If the graph is Eularian, the program outputs an Euler circuit.
6. Write a program to implement Dijkstra's shortest path algorithm.
7. The Dijkstra's shortest path algorithm as given in this chapter, and implemented in Programming Exercise

6, only outputs the length of the shortest path. Redo Programming Exercise 6 so that the program also outputs the shortest path. (Also see Exercise 5 of Section 10.6 of this chapter.)

8. Write a program to implement the breadth-first topological ordering algorithm.
9. Let  $G$  be a directed graph with the vertex set  $V = \{v_1, v_2, \dots, v_n\}$ , where  $n \geq 0$ . Recall that a topological ordering of  $V$  is a linear ordering  $v_{i1}, v_{i2}, \dots, v_{in}$  of the vertices such that, if  $v_{ij}$  is a predecessor of  $v_{ik}$ , then  $j < k$ ,  $1 \leq j \leq n$ ,  $1 \leq k \leq n$ , then  $v_{ij}$  precedes  $v_{ik}$ ; that is,  $j < k$  in this linear ordering. Suppose that  $G$  has no cycles. The following algorithm, a depth-first topological algorithm, lists the nodes of the graph in a topological ordering.

In a depth-first topological algorithm, we start with finding a vertex that has no successor (such a vertex exists because the graph has no cycles), and place it last in the topological order. After we have placed all the successors of a vertex in the topological order, we place the vertex in the topological order before any of its successors. Clearly, in depth-first topological ordering, first we find the vertex to be placed in `topOrder[n]`, then `topOrder[n - 1]` and so on. Write a program to implement the depth-first topological ordering.

## Trees and Networks

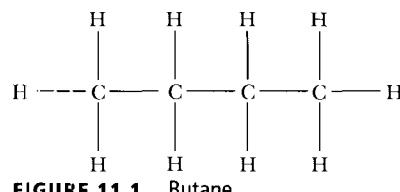
**The objectives of this chapter are to:**

- Learn the basic properties of trees
- Explore applications of trees
- Learn about networks

In Chapter 10, we discussed graphs and their basic properties in some detail. In this chapter, we study special types of graphs, called trees. Trees, like graphs, have numerous applications in computer science. We have already seen one of the applications of trees, in Chapter 9, to model the behavior of comparison-based algorithms using decision trees. In addition to trees, we also discuss networks in this chapter.

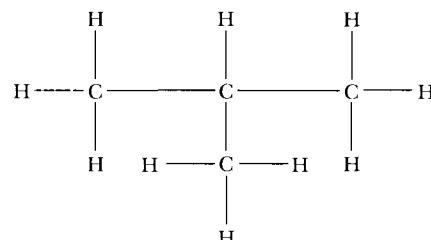
## 11.1 TREES

It is believed that the term *graph* derived from the phrase *graphic notation*, introduced in chemistry by E. Frankl and adopted in chemistry, in 1884, by A. Crum Brown. Each atom of a chemical compound is represented by a point in a plane, and atomic bonds are represented by lines, as shown in Figure 11.1 for the chemical compound with the formula  $C_4H_{10}$ .



**FIGURE 11.1** Butane

In chemistry, chemical compounds with formula  $C_kH_{2k+2}$  are known as parafins, which contain  $k$  carbon atoms and  $2k + 2$  hydrogen atoms. In the graphical representation, each of the carbon atoms corresponds to a vertex of degree 4 and each of the hydrogen atoms corresponds to a vertex of degree 1. For the same chemical formula  $C_4H_{10}$ , we also have the graph shown in Figure 11.2.



**FIGURE 11.2** Isobutane

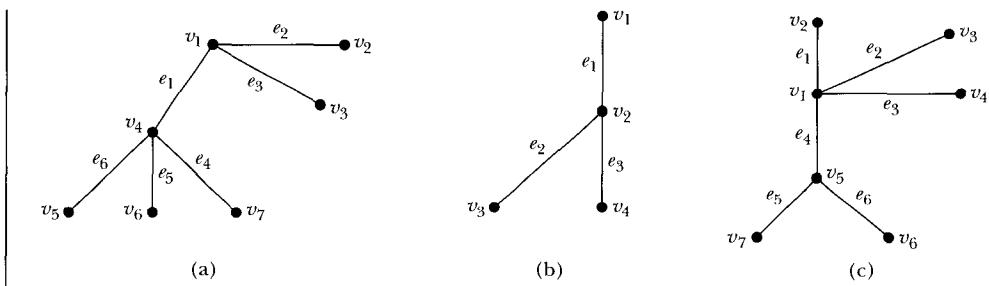
The graphs in Figures 11.1 and 11.2 have 4 vertices for carbon atoms and 10 vertices for hydrogen atoms. Therefore, each of these graphs contains 14 vertices and 13 edges. But if we look carefully, we see a difference between the two graphs. In the second graph, we see a vertex labeled C such that it has three adjacent vertices labeled by C, but this is not true in the first graph. Hence, these two graphs are not isomorphic. However, they correspond to the same chemical formula with different chemical properties. The name of the chemical compound corresponding to the first graph is butane and that of second graph is isobutane. Chemical compounds exhibiting this phenomenon are called isomers. Each of the above graphs is a connected graph, and neither of them contains any cycles. Cayley called them *trees*. Moreover, Cayley used the properties of trees to count the number of the isomers  $C_kH_{2k+2}$ .

**DEFINITION 11.1.1** ▶ A graph that is connected and has no cycles is called a **tree**.

Generally a graph that does not contain any cycles is called an **acyclic graph**. Using this notion, we can say that a connected acyclic graph is a tree.

### EXAMPLE 11.1.2

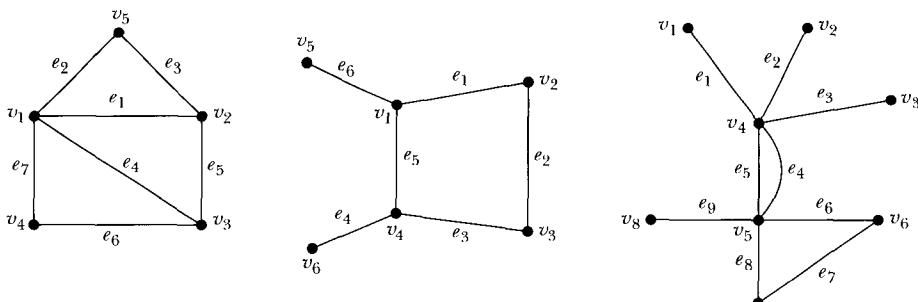
Consider the graphs shown in Figure 11.3. These graphs are connected and have no cycles. Hence, each of these graphs is a tree.



**FIGURE 11.3** Various trees

**EXAMPLE 11.1.3**

Consider the graphs shown in Figure 11.4. Each of these graphs is connected. However, each of these graphs has a cycle. Hence, none of these graphs is a tree.



**FIGURE 11.4** Graphs that are not trees

In the next few theorems, we prove some basic properties of trees. Consider the tree in Figure 11.3(a). In this tree, we find that there exists only one path  $(v_1, e_1, v_4, e_5, v_6)$  from  $v_1$  to  $v_6$ . This result is true for any two vertices in a tree and is proved in Theorem 11.1.5. First, however, let us make the following conventions about paths in a tree.

**REMARK 11.1.4** ▶ Let  $T$  be a tree. Then  $T$  is a simple connected graph, so  $T$  has no parallel edges and no loops. Let  $u$  and  $v$  be two vertices in  $T$ . It follows that there is at most one edge connecting  $u$  and  $v$ . Because  $G$  is connected, there is a path from  $u$  to  $v$ . Let  $P = (u, e_1, u_1, e_2, \dots, u_k, e_k, v)$ . If no confusion arises, then we write the path  $P$  as  $(u, u_1, \dots, u_k, v)$ ; i.e., when listing the vertices of the path, we will omit edges.

**Theorem 11.1.5:** Let  $u$  and  $v$  be two vertices of a tree  $T$ . Then there exists only one path from  $u$  to  $v$ .

**Proof:** If  $u = v$ , then the result is trivial.

Suppose that  $u \neq v$ . Because  $T$  is a connected graph, there is at least one path from  $u$  to  $v$ . Suppose there are distinct paths  $P_1 = (u, u_1, \dots, u_k, v)$  and  $P_2 = (u, v_1, \dots, v_t, v)$  from  $u$  to  $v$ . Because  $P_1$  and  $P_2$  are distinct, we have the following two cases.

**Case 1:**  $\{u_1, \dots, u_k\} \cap \{v_1, \dots, v_t\} = \emptyset$ . Then path  $P_1$  followed by path  $P_2$ , i.e.,

$$(u, u_1, \dots, u_k, v, v_t, \dots, v_1, u),$$

forms a cycle from  $u$  to  $u$ , which is a contradiction.

**Case 2:**  $\{u_1, \dots, u_k\} \cap \{v_1, \dots, v_t\} \neq \emptyset$ . Hence,  $u_i = v_j$  for some  $i$  and  $j$ .

Let  $w_1$  be the first common vertex, other than  $u$  and  $v$ , on paths  $P_1$  and  $P_2$ . Next, we follow path  $P_1$  until we come to the first vertex,  $w_s$ , which is again on both paths  $P_1$  and  $P_2$ . This vertex  $w_s$  is different from  $w_1$ . We must get such a vertex  $w_s$ , because  $P_1$  and  $P_2$  meet again at  $v$ . Let  $P_1^*$  be the portion of the path of  $P_1$  from  $w_1$  to  $w_s$  and  $P_2^*$  be the portion of the path of  $P_2$  from  $w_s$  to  $w_1$ . Then path  $P_1^*$  followed by path  $P_2^*$  forms a cycle from  $w_1$  to  $w_s$  in graph  $T$ . But this contradicts our assumption that  $T$  is a tree, so it has no cycles.

Hence,  $T$  does not contain two distinct paths between any two distinct vertices  $u$  and  $v$ . ■

**Theorem 11.1.6:** In a tree with more than one vertex, there are at least two vertices of degree 1.

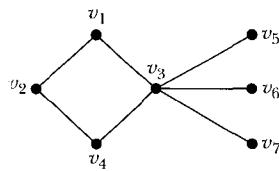
**Proof:** Let  $T$  be a tree with more than one vertex. Because  $T$  is a connected graph with at least two vertices, there is a path with at least two distinct vertices. Because the number of vertices and the number of edges is finite, the number of paths in  $T$  is also finite. Thus, we can find a path  $P$  of maximal length. Suppose path  $P$  is from vertex  $u$  to vertex  $v$ . We show that  $\deg(u) = \deg(v) = 1$ .

Suppose  $\deg(v) \neq 1$ . Let  $P$  be the path  $(u = v_1, e_1, v_2, e_2, v_3, \dots, v_{k-1}, e_{k-1}, v)$ . Because  $\deg(v) \neq 1$ , there exists an edge  $e_k$  with  $v$  as an end vertex such that  $e_k \neq e_{k-1}$ . Because  $G$  has no loops, the other end vertex of  $e_k$  cannot be  $v$ . Suppose the other end vertex is  $v_k$ . Suppose  $v_k = v_i$  for some  $i$  such that  $1 \leq i \leq k-1$ . Then  $(v, e_k, v_i, e_{i+1}, v_{i+1}, \dots, v_{k-1}, e_{k-1}, v)$  is a cycle from  $v$  to  $v$ , which contradicts the fact that  $T$  is a tree. If  $v_k \neq v_i$ ,  $1 \leq i \leq k-1$ , then we get the path  $(v_1, e_1, v_2, e_2, v_3, \dots, v_{k-1}, e_{k-1}, v, e_k, v_k)$  whose length is greater than that of  $P$ . This contradicts the fact that path  $P$  is of maximal length in  $T$ . It now follows that  $\deg(v) = 1$ . Similarly, we can show that  $\deg(u) = 1$ . ■

The converse of Theorem 11.1.6 is not true as shown by the following example.

### EXAMPLE 11.1.7

Consider the graph shown in Figure 11.5. This is a connected graph and it has at least two vertices of degree 1. But it contains a cycle. Hence, it is not a tree.



**FIGURE 11.5** A graph, with at least two vertices of degree 1, which is not a tree

We consider the tree of Figure 11.3(a) (Example 11.1.2). We find that the number of vertices of this tree is 7. Interestingly, we find that the number of edges of this tree is  $7 - 1$ . This is another basic property of trees and is proved in the next theorem.

**Theorem 11.1.8:** Let  $T$  be a tree with  $n$  vertices,  $n \geq 1$ . Then  $T$  has exactly  $n - 1$  edges.

**Proof:** We prove the result by induction on  $n$ .

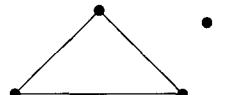
*Basis step:* Let  $n = 1$ . Because  $T$  is a simple graph, it does not contain any loop. Therefore, it follows that  $T$  has no edges. Thus, the number of edges is  $0 = 1 - 1$ . Hence, the theorem holds for  $n = 1$ .

*Inductive hypothesis:* Let  $k \geq 1$  be a positive integer. We assume that the theorem holds for any tree with  $k$  vertices.

*Inductive step:* Let  $T$  be a tree with  $k + 1$  vertices. Because  $k + 1 \geq 2$ , it follows from Theorem 11.1.6 that  $T$  has at least two vertices of degree 1. Let  $u$  be a vertex of degree 1 in  $T$ . We construct a new graph  $G$  by deleting  $u$  from  $T$  and also the edge  $e$ , which is incident on  $u$ . Now  $G$  is still a connected graph and does not contain any cycle. Hence,  $G$  is a tree with  $k$  vertices. By the inductive hypothesis, we find that  $G$  has exactly  $k - 1$  edges. This implies that  $T$  has  $k$  edges. Hence, by induction the theorem holds for any integer  $n$ . ■

---

**REMARK 11.1.9** ► The converse of Theorem 11.1.8 is not true. For example, consider the graph shown in Figure 11.6.



**FIGURE 11.6** A graph that is not a tree

The graph in Figure 11.6 has four vertices and three edges. However, it is not a tree because it has a cycle. Moreover, it is not connected.

**Theorem 11.1.10:** Let  $T$  be a graph with  $n$  vertices. Then the following conditions are equivalent.

- (i)  $T$  is a tree.
- (ii) If  $u$  and  $v$  are two vertices in  $T$ , then there exists only one path from  $u$  to  $v$ .
- (iii)  $T$  is a connected graph and has  $n - 1$  edges.
- (iv)  $T$  has no cycles and has  $n - 1$  edges.

**Proof:**

- (i)  $\Rightarrow$  (ii): It follows from Theorem 11.1.5.

(ii)  $\Rightarrow$  (iii): From assumption (ii) it follows that  $T$  is a connected graph and also acyclic. Hence,  $T$  is a tree. By Theorem 11.1.8, it follows that  $T$  has  $n - 1$  edges.

(iii)  $\Rightarrow$  (iv): Suppose  $T$  contains at least one cycle. Let  $C_1$  be a cycle in  $T$ . We construct a new graph  $T_1$  by removing an edge  $e_1$  from  $C_1$ . This new graph  $T_1 = (V, E - \{e_1\})$  contains all the vertices of  $T$  and also all the edges of  $T$  except the edge  $e_1$ . Because  $e_1$  is an edge of a cycle, it follows that  $T_1$  remains as a connected graph with  $n$  vertices. Suppose  $T_1$  contains a cycle  $C_2$ . Let  $e_2$  be an edge of  $C_2$ . Proceeding as above, we construct the graph  $T_2 = (V, E - \{e_1, e_2\})$  and find that  $T_2$  is a connected graph with  $n$  vertices. If  $T_2$  contains a cycle, we repeat the process, and because  $T$  contains a finite number of cycles, we eventually obtain a connected graph  $T_k = (V, E - \{e_1, e_2, \dots, e_k\})$ , which has no cycle. Hence,  $T_k$  is a tree with  $n$  vertices. Then by Theorem 11.1.8, it follows that  $T_k$  is a tree with  $n - 1$  edges. But from the above construction we find that the number of edges in  $T_k$  is  $n - k$ . Because  $T_k$  is obtained from  $T$  by deleting  $k$  edges, it follows that  $T$  has  $n - 1 + k$  edges. Now  $k > 1$  implies  $n - 1 + k > n - 1$ . This contradicts our assumption that  $T$  has  $n - 1$  edges. Hence,  $T$  has no cycles. Also  $T$  has  $n - 1$  edges.

(iv)  $\Rightarrow$  (i): Suppose  $T$  has no cycles and has  $n - 1$  edges. We show that  $T$  is a connected graph. Suppose  $T_1, T_2, \dots, T_k$  are the components of  $T$ . Because  $T$  has no cycles, we find that each of the connected subgraphs  $T_1, T_2, \dots, T_k$  has no cycles and hence each of  $T_1, T_2, \dots, T_k$  is a tree. Suppose each  $T_i$  has  $n_i$  vertices. Then from Theorem 11.1.8, we find that each  $T_i$  has  $n_i - 1$  edges. Then the total number of edges is

$$(n_1 - 1) + (n_2 - 1) + \dots + (n_k - 1) = (n_1 + n_2 + \dots + n_k) - k = n - k.$$

Thus,  $n - 1 = n - k$ , which implies that  $k = 1$ . This implies that  $T$  has only one component. Therefore,  $T$  is a connected graph. Also,  $T$  has no cycles. Hence,  $T$  is a tree. ■

## Isomorphism of Trees

Let  $T_1 = (V_1, E_1)$  and  $T_2 = (V_2, E_2)$  be two trees. Then  $T_1 = (V_1, E_1)$  and  $T_2 = (V_2, E_2)$  are simple graphs. Hence, according to the definition of isomorphism of simple graphs (Chapter 10),  $T_1$  is isomorphic to  $T_2$  if and only if there exists a one-to-one correspondence  $f : V_1 \rightarrow V_2$  such that two vertices  $u$  and  $v$  are adjacent in  $T_1$  if and only if  $f(u)$  and  $f(v)$  are adjacent in  $T_2$ .

### EXAMPLE 11.1.11

Consider trees  $T_1$  and  $T_2$  shown in Figure 11.7.

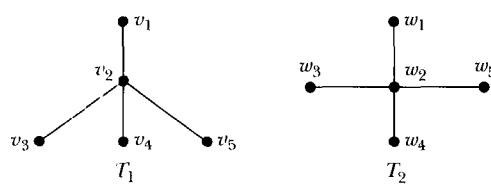


FIGURE 11.7 Trees

Define  $f : \{v_1, v_2, v_3, v_4, v_5\} \rightarrow \{w_1, w_2, w_3, w_4, w_5\}$  by

$$\begin{aligned} f : v_1 &\mapsto w_1 \\ v_2 &\mapsto w_2 \\ v_3 &\mapsto w_3 \\ v_4 &\mapsto w_4 \\ v_5 &\mapsto w_5 \end{aligned}$$

It follows that  $f$  is a one-to-one correspondence. We verify that  $v_i$  and  $v_j$  are adjacent in  $G_1$  if and only if  $w_i$  and  $w_j$  are adjacent in  $G_2$  by considering the adjacency matrices  $A_{G_1}$  and  $A_{G_2}$ . Notice that

$$A_{G_1} = \begin{bmatrix} v_1 & v_2 & v_3 & v_4 & v_5 \\ v_1 & 0 & 1 & 0 & 0 \\ v_2 & 1 & 0 & 1 & 1 \\ v_3 & 0 & 1 & 0 & 0 \\ v_4 & 0 & 1 & 0 & 0 \\ v_5 & 0 & 1 & 0 & 0 \end{bmatrix} \quad A_{G_2} = \begin{bmatrix} w_1 & w_2 & w_3 & w_4 & w_5 \\ w_1 & 0 & 1 & 0 & 0 \\ w_2 & 1 & 0 & 1 & 1 \\ w_3 & 0 & 1 & 0 & 0 \\ w_4 & 0 & 1 & 0 & 0 \\ w_5 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Because  $A_{G_1}$  and  $A_{G_2}$  are the same, these two trees are isomorphic.

**Theorem 11.1.12:** There are three nonisomorphic trees with five vertices.

**Proof:** Consider trees  $T_1$ ,  $T_2$ , and  $T_3$ , shown in Figure 11.8.

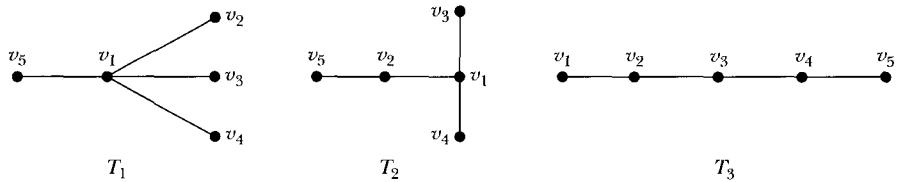


FIGURE 11.8 Trees  $T_1$ ,  $T_2$ , and  $T_3$

Each of the trees  $T_1$ ,  $T_2$ , and  $T_3$  consists of five vertices. We now show that any tree  $T$  with five vertices is isomorphic to one of these trees.

Notice that each vertex of  $T_3$  is of degree 1 or 2. Moreover,  $T_2$  contains a vertex of degree 3,  $T_1$  contains a vertex of degree 4, and  $T_2$  has no vertex of degree 4. Hence,  $T_1$ ,  $T_2$ , and  $T_3$  are nonisomorphic trees.

Now we show that any tree  $T$  with five vertices is isomorphic to one of  $T_1$ ,  $T_2$ ,  $T_3$ .

For this we consider a tree  $T$  with five vertices. By Theorem 11.1.8,  $T$  has four edges. Hence, the degree of each vertex of  $T$  is  $\leq 4$ .

**Case 1:** Suppose that  $T$  has a vertex of degree 4. In this case, all four edges are incident on this vertex. Hence,  $T$  is isomorphic to  $T_1$ .

**Case 2:** Suppose that  $T$  has no vertex of degree 4, but  $T$  has a vertex of degree 3. Suppose  $v_1, v_2, v_3, v_4$ , and  $v_5$  are the vertices of  $T$  and the degree of  $v_3 = 3$ .

Then there are three edges that are incident with  $v_3$ . This implies that  $v_3$  has only three adjacent vertices. Suppose  $v_2, v_4$ , and  $v_5$  are adjacent vertices of  $v_3$ . Because  $T$  is a connected graph, vertex  $v_1$  cannot be an isolated vertex. Therefore,  $v_1$  is adjacent to one of the vertices  $v_2, v_4, v_5$ . Furthermore, because the total number of edges in  $T$  is 4,  $v_1$  is adjacent to only one of the vertices  $v_2, v_4, v_5$ . It follows that in this case  $T$  is isomorphic to one of the trees shown in Figure 11.9.

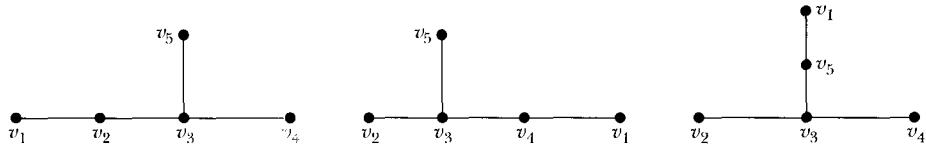


FIGURE 11.9 Trees

But each of these trees is isomorphic to  $T_2$ . Hence,  $T$  is isomorphic to  $T_2$ .

**Case 3:** The degree of each vertex of  $T$  is  $\leq 2$ . Because the sum of the degrees of the vertices is  $2 \cdot 4 = 8$ , it follows that the degree of each vertex cannot be 1. Therefore, there exists a vertex of degree 2. Suppose the vertices of  $T$  are  $v_1, v_2, v_3, v_4$ , and  $v_5$ . Let  $v_3$  be a vertex of degree 2. Then  $v_3$  has only two adjacent vertices, say  $v_2$  and  $v_4$  (see Figure 11.10(a)). Because vertex  $v_1$  of  $T_1$  is not an isolated vertex and  $v_1$  is not an adjacent vertex of  $v_3$ , it follows that  $v_1$  is an adjacent vertex of  $v_2, v_4$ , or  $v_5$ .

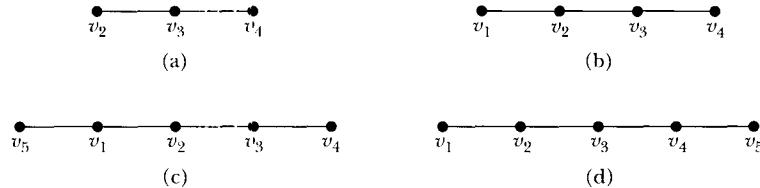


FIGURE 11.10 Trees

Suppose  $v_1$  is an adjacent vertex of  $v_2$ . Then  $v_1$  cannot be an adjacent vertex of  $v_4$  (see Figure 11.10(b)). Notice that the degree of  $v_2$  is now 2. Now consider vertex  $v_5$ . It follows that  $v_5$  is adjacent to either  $v_1$  or  $v_4$  but not to both (see Figures 11.10(c) and (d)). In either case,  $T$  is isomorphic to  $T_3$ .

The case that  $v_1$  is adjacent to  $v_4$  is similar to the case that  $v_1$  is adjacent to  $v_2$ . Hence, in this case also we can show that  $T$  is isomorphic to  $T_3$ .

Now consider the case that  $v_1$  is adjacent to  $v_5$ . Because  $T$  is connected, it follows that either  $v_1$  or  $v_5$  is adjacent to either  $v_2$  or  $v_4$ . If  $v_1$  is adjacent to  $v_2$  or  $v_4$ , then as before, we can conclude that  $T$  is isomorphic to  $T_3$ . If  $v_5$  is adjacent to  $v_2$  or  $v_4$ , then as in the case of  $v_1$  adjacent to  $v_2$  or  $v_4$ , we can conclude that  $T$  is isomorphic to  $T_3$ .

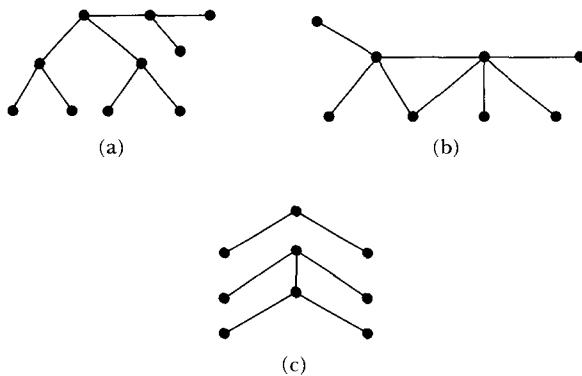
Combining all three cases, we find that there are only three nonisomorphic trees with five vertices. ■

## WORKED-OUT EXERCISES

**Exercise 1:** Which of the graphs in Figure 11.11 are trees and which are not? Give reasons.

**Solution:**

- (a) The given graph is a connected graph and does not contain any cycle. Hence, this graph is a tree.

**FIGURE 11.11** Various graphs

- (b) The given graph is a connected graph but contains a cycle. Hence, this graph is not a tree.  
 (c) The given graph does not contain any cycles but it is not a connected graph, so it is not a tree.

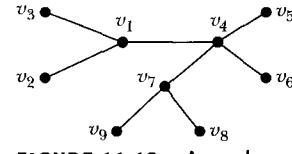
**Exercise 2:** How many vertices are there in a tree with 19 edges?

**Solution:** Let  $T$  be a tree with 19 edges. Suppose that  $T$  has  $n$  vertices. Then by Theorem 11.1.8,  $n - 1 = 19$ , i.e.,  $n = 20$ . Hence, there are 20 vertices in  $T$ .

**Exercise 3:** Does there exist a tree  $G$  with 10 vertices such that the total degree of  $G$  is 24? Justify your answer.

**Solution:** Suppose  $G$  is a tree with 10 vertices. Then by Theorem 11.1.10,  $G$  has 9 edges. This implies that the total degree of  $G$  is  $2 \cdot 9 = 18 \neq 24$ . Hence, there is no tree with 10 vertices and the total degree 24.

**Exercise 4:** Show that the graph in Figure 11.12, is bipartite. Furthermore, find the bipartition of this graph.

**FIGURE 11.12** A graph

**Solution:** Let  $G$  denote the graph in Figure 11.12. Notice that  $G$  is a tree. Now every tree with  $n \geq 2$  vertices is a bipartite graph (see Exercise 12, page 712 of this section). Hence,  $G$  is a bipartite graph and  $V_1 \cup V_2$  is a bipartition of  $G$ , where  $V_1 = \{v_2, v_3, v_4, v_8, v_9\}$  and  $V_2 = \{v_1, v_5, v_6, v_7\}$ .

## SECTION REVIEW

### Key Terms

tree

acyclic graph

### Key Definition

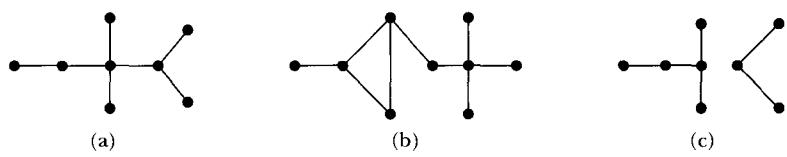
1. A graph that is connected and has no cycles is called a tree.

### Some Key Results

1. Let  $u$  and  $v$  be two vertices of a tree  $T$ . Then there exists only one path from  $u$  to  $v$ .
2. Let  $T$  be a tree with  $n$  vertices,  $n \geq 1$ . Then  $T$  has exactly  $n - 1$  edges.
3. There are three nonisomorphic trees with five vertices.

## EXERCISES

1. Which of the graphs in Figure 11.13 are trees and which are not? Give reasons.

**FIGURE 11.13** Various graphs

2. How many vertices are there in a tree with 19 edges?
3. How many edges are there in a tree with 16 vertices?
4. How many vertices are there in a tree with 20 edges?
5. Does there exist a tree  $T$  with 8 vertices such that the total degree of  $G$  is 18? Justify your answer.
6. Suppose there exists a cycle-free graph with 12 vertices that has 11 edges. Is it a connected graph? Justify your answer.
7. Suppose there exists a simple connected graph with 16 vertices that has 15 edges. Does it contain a vertex of degree 1? Justify your answer.
8. Draw a tree with two vertices of degree 3. Find the number of vertices of degree 1 in your tree.
9. Suppose  $T$  is a tree with two vertices of degree 3. Show that  $T$  has at least four vertices of degree 1.
10. Draw a graph having the given properties or explain why no such graph exists.
  - a. Tree; six vertices having degrees 1, 1, 1, 1, 3, 3
  - b. Tree; all vertices of degree 2
  - c. Tree; 10 vertices and 10 edges
  - d. Connected graph; 7 vertices, 7 edges
  - e. Tree; 5 vertices, total degree of the vertices is 10
11. Prove that a connected graph  $G$  with  $n \geq 2$  vertices is a tree if and only if the sum of the degrees of  $G$  is  $2(n - 1)$ .
12. Prove that a tree with  $n \geq 2$  vertices is a bipartite graph.
13. Consider a complete graph  $K_{m,n}$  where  $m \geq 1$ ,  $n \geq 2$ . Show that  $K_{m,n}$  is a tree if and only if  $m = 1$ .
14. Construct a graph  $G$  with 6 vertices such that  $G$  has exactly 5 edges but  $G$  is not a tree.
15. Find all trees with 4 vertices.
16. If a new edge is added to a tree, then show that the resulting new graph contains a cycle.

## 11.2 ROOTED TREE

In this section, we discuss special types of trees and describe algorithms to implement some of these trees. Recall that a graph  $G$  is an ordered triple  $(V, E, g)$  where  $V$  is a finite nonempty set of vertices,  $E$  is a set of edges, and  $g$  is a mapping that associates with each edge  $e$  an unordered pair of vertices  $\{u, v\}$ . In this section, for convenience of writing different algorithms we assume that both sets  $V$  and  $E$  may be empty. We call this graph the empty graph, or null graph.

We begin with the following definition.

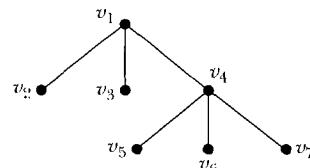
---

**DEFINITION 11.2.1** ► A **rooted tree** is a tree in which a particular vertex is designated as the root.

---

**REMARK 11.2.2** ► By an empty rooted tree we mean an empty graph.

In contrast to natural trees, which have their roots at the bottom, in graph theory rooted trees are typically drawn with the roots at the top. Figure 11.14 shows the way the tree in Figure 11.3(a) is drawn with  $v_1$  as the root.



**FIGURE 11.14** A tree

First we place the root,  $v_1$ , at the top. Below the root, and on the same level, we place the vertices,  $v_2$ ,  $v_3$ , and  $v_4$ , that can be reached from the root on a path of length 1. We continue in this way until the entire tree is drawn. Because the path from the root to any given vertex is unique, the level of each vertex is unique. We call the level of the root level 0. The vertices under the root are said to be on level 1, and so on. Thus, the **level** of a vertex  $v$  is the length of the path from the root to  $v$ .

**DEFINITION 11.2.3** ▶ Let  $T$  be a tree with  $v_0$  as a root and let  $(v_0, e_1, v_1, e_2, v_2, \dots, e_n, v_n)$  be a path in  $T$ . Then

- (i)  $v_k$  is called a **child** of  $v_{k-1}$  for  $k = 1, 2, \dots, n$ .
- (ii) If a vertex  $v$  of  $T$  has no children, then  $v$  is called a **terminal vertex**, or a **leaf**.
- (iii) If  $v$  is not a terminal vertex, then  $v$  is called an **internal vertex** of  $T$ .

**REMARK 11.2.4** ▶ The root of a tree is considered an internal vertex unless it is a trivial tree (i.e., a tree with a root as the only vertex), in which case it is considered as a leaf.

**DEFINITION 11.2.5** ▶ Let  $T$  be a rooted tree. Let  $u$  and  $v$  be distinct vertices in  $T$ . Then  $v$  is said to be a **descendant** of  $u$  if  $u$  and  $v$  are on the unique path from the root of  $T$  to  $v$  and  $u$  appears before  $v$  on this path.

#### EXAMPLE 11.2.6

Consider the tree  $T$  shown in Figure 11.15.

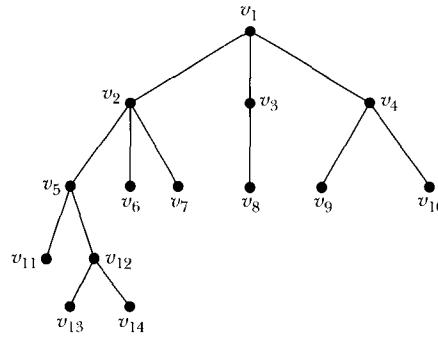


FIGURE 11.15 Rooted tree

In  $T$ , the level of each of vertices  $v_2$ ,  $v_3$ , and  $v_4$  is 1; the level of each of vertices  $v_5$ ,  $v_6$ ,  $v_7$ ,  $v_8$ ,  $v_9$ , and  $v_{10}$  is 2, and so on. The level of vertex  $v_{14}$  is 4. Also notice that vertices  $v_6$ ,  $v_7$ ,  $v_8$ ,  $v_9$ ,  $v_{10}$ ,  $v_{11}$ ,  $v_{13}$ , and  $v_{14}$  are leaves. All the other vertices are internal vertices. Moreover,  $v_5$ ,  $v_6$ ,  $v_7$ ,  $v_{11}$ ,  $v_{12}$ ,  $v_{13}$ , and  $v_{14}$  are descendants of  $v_2$ .

**DEFINITION 11.2.7** ▶ An **ordered rooted tree** is a rooted tree in which the children of each vertex are assigned a fixed ordering.

An ordered rooted tree is drawn in a plane such that at each level the left to right order of the vertices agrees with their prescribed order.

**DEFINITION 11.2.8** ▶ Let  $T$  be a rooted tree. The **height** of  $T$  is the number of vertices on a longest path from the root to a leaf.

The height of tree  $T$  of Figure 11.15 is 5.

**REMARK 11.2.9** ▶ Notice that we have defined the height of a rooted tree as the number of vertices on the longest path from the root to a leaf. There is no universally agreed-upon definition of the height of a rooted tree. Some authors define it as the length of the longest path, i.e., the number of edges, from the root to a leaf. According to

our definition, if a rooted tree has only one vertex, then its height is 1. However, according to the second definition, if a rooted tree has only one vertex, then its height is 0. Therefore, we recommend that readers read the definition of the height of a rooted tree when dealing with the concept of the height of a rooted tree.

## Binary Trees

Binary trees are of special interest in computer science. In this section, we briefly describe such trees and discuss how they are implemented in computer memory.

---

**DEFINITION 11.2.10** ► An ordered rooted tree  $T$  is called a **binary tree** if either it is an empty graph or each vertex has no children, one child, or two children. If a rooted tree has only one vertex and no edges, then the tree is called a **trivial tree**. If a vertex has one child, that child is designated as either a **left child** or a **right child** (but not both). If a vertex has two children, then the first child (according to the given order) is designated a left child and the other child is designated a right child.

---

**DEFINITION 11.2.11** ► Let  $T$  be a nonempty binary tree and  $v$  be a vertex in  $T$ . The **left subtree** of  $v$  is a binary tree with left child of  $v$  as the root and which contains all the descendants of this left child and all edges incident to these descendants. Similarly, the **right subtree** of  $v$  is a binary tree with right child of  $v$  as the root and which contains all the descendants of this right child and all edges incident to these descendants.

The vertex of the left subtree of  $v$  is denoted by  $L_v$  and the vertex of the right subtree of  $v$  is denoted by  $R_v$ . Now the left subtree,  $L_v$ , of  $v$  contains all the descendants of the left child of  $v$  and all edges incident to these descendants. Hence, if  $v_1, v_2, \dots, v_k$  are the only descendants of the left child of  $v$ , we write  $L_v = \{v_1, v_2, \dots, v_k\}$ . A similar convention applies for  $R_v$ .

---

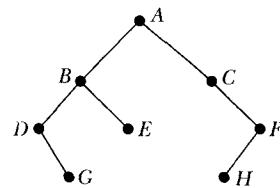
**REMARK 11.2.12** ► Let  $T$  be a nonempty binary tree and  $v$  be a vertex in  $T$ . If  $v$  has no left child, then its left subtree is empty. Similarly, if  $v$  has no right child, then its right subtree is empty.

If  $v$  is the root of the binary tree  $T$ , then sometimes we write  $L_T$  for  $L_v$  and  $R_T$  for  $R_v$ . Clearly, for any vertex  $v$  of  $T$ ,  $L_v \cap R_v = \emptyset$ . Also, we find that if  $v$  has no children, then  $L_v$  and  $R_v$  are both empty. If  $v$  has a left child but no right child, then  $R_v = \emptyset$ . Similarly, if  $v$  has a right child but no left child, then  $L_v = \emptyset$ .

**Convention:** Henceforth, by an empty binary tree, we mean the tree without any vertices and edges. Unless specified otherwise, by a binary tree, we mean a non-empty binary tree.

Next we consider various examples of binary trees.

The diagram in Figure 11.16 is an example of a binary tree.



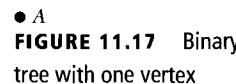
**FIGURE 11.16** Binary tree with one vertex

The root of this binary tree is  $A$ . Vertex  $B$  is the left child of  $A$  and vertex  $C$  is the right child of  $A$ . From the diagram, it follows that  $B$  is the root of the left subtree of  $A$ , i.e., the left subtree of the root. Similarly,  $C$  is the root of the right subtree of  $A$ , i.e., the right subtree of the root. Notice that  $L_A = \{B, D, E, G\}$  and  $R_A = \{C, F, H\}$ . Moreover, for vertex  $F$ , the left child is  $H$  and  $F$  has no right child.

**REMARK 11.2.13** ▶ Let  $T$  be a nonempty binary tree. Let  $r$  be the root of  $T$ . It follows that associated with  $r$ , there are two binary trees, which may be empty, called the left subtree and the right subtree of  $r$ . It also follows that every vertex in a binary tree is the root of some binary tree, which may be empty.

#### EXAMPLE 11.2.14

This example shows a binary tree with one vertex. See Figure 11.17.

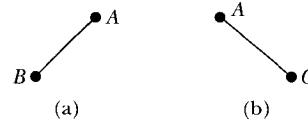


**FIGURE 11.17** Binary tree with one vertex

In Figure 11.17, the root of the binary tree =  $A$ ,  $L_A = \emptyset$ , and  $R_A = \emptyset$ .

#### EXAMPLE 11.2.15

Consider the binary trees shown in Figure 11.18.

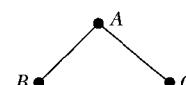


**FIGURE 11.18** Binary trees with two vertices

- In Figure 11.18(a), the root of the binary tree =  $A$ ,  $L_A = \{B\}$ , and  $R_A = \emptyset$ . The root of  $L_A = B$ ,  $L_B = \emptyset$ , and  $R_B = \emptyset$ .
- In Figure 11.18(b), the root of the binary tree =  $A$ ,  $L_A = \emptyset$ , and  $R_A = \{C\}$ . The root of  $R_A = C$ ,  $L_C = \emptyset$ , and  $R_C = \emptyset$ .

#### EXAMPLE 11.2.16

This example shows a binary tree with three vertices. See Figure 11.19.



**FIGURE 11.19**  
Binary tree with  
three vertices

In Figure 11.19, the root of the binary tree =  $A$ ,  $L_A = \{B\}$ , and  $R_A = \{C\}$ . The root of  $L_A = B$ ,  $L_B = \emptyset$ , and  $R_B = \emptyset$ . The root of  $R_A = C$ ,  $L_C = \emptyset$ , and  $R_C = \emptyset$ .

**DEFINITION 11.2.17** ▶ A **full binary tree** is a binary tree in which each vertex has either two children or no children.

A fundamental result about full binary trees is given in the next theorem.

**Theorem 11.2.18:** Let  $T$  be a full binary tree with  $i$  internal vertices. Then  $T$  has  $i + 1$  terminal vertices and  $2i + 1$  total vertices.

**Proof:** Because  $T$  is a full binary tree, each vertex of  $T$  has either two children or no children. There are  $i$  internal vertices. Thus, there are  $2i$  children. In a rooted tree, the root is the only vertex that is not a child of any vertex. Hence, there are  $2i + 1$  vertices in  $T$ , and consequently there are  $(2i + 1) - i = i + 1$  terminal vertices. ■

**Theorem 11.2.19:** Let  $T$  be a binary tree such that  $T$  has  $t$  terminal vertices and height  $h$ ,  $h \geq 1$ . Then  $t \leq 2^{h-1}$ .

**Proof:** We prove this theorem by induction on  $h$ .

*Basis step:* Suppose  $h = 1$ . Then  $T$  is a graph with only one vertex and no edge. Hence,  $t = 1$ . Then  $t = 1 = 2^{1-1}$ .

*Induction hypothesis:* Suppose  $k$  is a positive integer and assume that if  $T$  is a binary tree with  $t$  terminal vertices and height  $k$ , where  $1 \leq k < h$ , then  $t \leq 2^{k-1}$ .

*Inductive step:* Suppose  $T$  is a binary tree that has  $t$  terminal vertices and height  $h \geq 2$ . Let  $v$  be the root of  $T$ . Because  $T$  is a binary tree,  $v$  has one child,  $v_1$ , or  $v$  has two children,  $v_1$  and  $v_2$ .

**Case 1:**  $v$  has only one child,  $v_1$ . Then the subtree  $T_1 = T - \{v\}$  with  $v_1$  as a root is of height  $h - 1$  and has  $t$  terminal vertices. Now the terminal vertices of  $T_1$  are those of the tree  $T$ . Hence,  $T_1$  is a binary tree with height  $h - 1$  and  $t$  terminal vertices. By the induction hypothesis  $t \leq 2^{h-2} < 2^{h-1}$ .

**Case 2:**  $T$  has two children,  $v_1$  and  $v_2$ . Let  $T_i$  be the subtree rooted at  $v_i$ . Suppose  $T_i$  is of height  $h_i$  and has  $t_i$  terminal vertices,  $i = 1, 2$ . Clearly,  $h_i < h$ . Hence, by the inductive hypothesis  $t_i \leq 2^{h_i-1}$  for  $i = 1, 2$ . Now the terminal vertices of  $T$  consist of the terminal vertices of  $T_1$  and  $T_2$ . Moreover,  $T_1$  and  $T_2$  have no common terminal vertices. Hence,  $t = t_1 + t_2$ . Then

$$t = t_1 + t_2 \leq 2^{h_1-1} + 2^{h_2-1} \leq 2^{h-2} + 2^{h-2} = 2 \cdot 2^{h-2} = 2^{h-1}.$$

The result now follows by induction. ■

Next we describe an algorithm that can be used to find the height of a binary tree. If the binary tree is empty, then the height is 0. Suppose that the binary tree is nonempty. To find the height of the binary tree, we first find the height of the left subtree and the height of the right subtree. We then take the maximum of these two heights and add 1 to find the height of the binary tree. To find the height of the left (right) subtree, we apply the same procedure because the left (right) subtree is a binary tree. Therefore, the general algorithm to find the height of a binary tree is as follows.

**ALGORITHM 11.1:** Height of a binary tree.

*Input:* root—a reference to the root of the binary tree. If the binary tree is empty, root is null.

*Output:* Height of the binary tree

```

1. procedure height(root)
2. begin
3.   if root is null then
4.     return 0;
5.   else
6.     return 1 + maximum(height(left subtree of root),
7.                           height(right subtree of root));
8. end
```

Similarly, we can write algorithms to find the number of vertices and the number of terminal vertices in a binary tree.

## Binary Tree Traversal

Item insertion, deletion, and lookup operations require the binary tree to be traversed. Thus, the most common operation performed on a binary tree is to traverse the binary tree, or visit each vertex of the binary tree. As you can see from the diagram of a binary tree, the traversal must start at the root because we are typically given a reference to the root. For each vertex, we have two choices.

- Visit the vertex first.
- Visit the subtrees first.

These choices lead to three different traversals of a binary tree.

- Inorder traversal
- Preorder traversal
- Postorder traversal

**Inorder Traversal:** In an inorder traversal, the binary tree is traversed as follows.

1. Traverse the left subtree.
2. Visit the vertex.
3. Traverse the right subtree.

**Preorder Traversal:** In a preorder traversal, the binary tree is traversed as follows.

1. Visit the vertex.
2. Traverse the left subtree.
3. Traverse the right subtree.

**Postorder Traversal:** In a postorder traversal, the binary tree is traversed as follows.

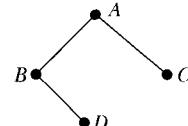
1. Traverse the left subtree.
2. Traverse the right subtree.
3. Visit the vertex.

Clearly, each of these traversal algorithms is recursive.

The listing of the vertices produced by the inorder traversal of a binary tree is called the **inorder sequence**. The listing of the vertices produced by the preorder traversal of a binary tree is called the **preorder sequence**. The listing of the vertices produced by the postorder traversal of a binary tree is called the **postorder sequence**.

By the term visiting a vertex, we mean to print the label, which is usually the data, of the vertex.

Next we illustrate inorder traversal for the binary tree in Figure 11.20.



**FIGURE 11.20**  
Binary tree for an  
inorder traversal

The root of the binary tree in Figure 11.20 is  $A$ . Therefore, we start the traversal at  $A$ .

1. Traverse the left subtree of  $A$ ; that is, traverse  $L_A = \{B, D\}$ .
2. Visit  $A$ .
3. Traverse the right subtree of  $A$ ; that is, traverse  $R_A = \{C\}$ .

We cannot do Step 2 until we have finished Step 1.

1. Traverse the left subtree of  $A$ ; that is, traverse  $L_A = \{B, D\}$ . Now  $L_A$  is a binary tree with root  $B$ . Because  $L_A$  is a binary tree, we apply the inorder traversal criteria to  $L_A$ .
  - 1.1 Traverse the left subtree of  $B$ ; that is, traverse  $L_B = \emptyset$ .
  - 1.2 Visit  $B$ .
  - 1.3 Traverse the right subtree of  $B$ ; that is, traverse  $R_B = \{D\}$ .

As before, first we complete Step 1.1 before going to Step 1.2.

- 1.1 Because the left subtree of  $B$  is empty, there is nothing to traverse. Step 1.1 is completed, so we proceed to Step 1.2.
- 1.2 Visit  $B$ ; that is, print  $B$ . Clearly, the first vertex printed is  $B$ . This completes Step 1.2, so we proceed to Step 1.3.
- 1.3 Traverse the right subtree of  $B$ ; that is, traverse  $R_B = \{D\}$ . Now  $R_B$  is a binary tree with root  $D$ . Because  $R_B$  is a binary tree, we apply the inorder traversal criteria to  $R_B$ .

- 1.3.1 Traverse the left subtree of  $D$ ; that is, traverse  $L_D = \emptyset$ .
- 1.3.2 Visit  $D$ .
- 1.3.3 Traverse the right subtree of  $D$ ; that is, traverse  $R_D = \emptyset$ .
- 1.3.1 Because the left subtree of  $D$  is empty, there is nothing to traverse. Step 1.3.1 is completed, so we proceed to Step 1.3.2.
- 1.3.2 Visit  $D$ ; that is, print  $D$ . This completes Step 1.3.2, so we proceed to Step 1.3.3.
- 1.3.3 Because the right subtree of  $D$  is empty, there is nothing to traverse. Step 1.3.3 is completed.

This completes Step 1.3. Because Steps 1.1, 1.2, and 1.3 are completed, Step 1 is completed, and so we go to Step 2.

2. Visit  $A$ ; that is, print  $A$ . This completes Step 2, so we proceed to Step 3.
3. Traverse the right subtree of  $A$ ; that is, traverse  $R_A = \{C\}$ . Now  $R_A$  is a binary tree with root  $C$ . Because  $R_A$  is a binary tree, we apply the inorder traversal criteria to  $R_A$ .

- 3.1 Traverse the left subtree of  $C$ ; that is, traverse  $L_C = \emptyset$ .
- 3.2 Visit  $C$ .
- 3.3 Traverse the right subtree of  $C$ ; that is, traverse  $R_C = \emptyset$ .
- 3.1 Because the left subtree of  $C$  is empty, there is nothing to traverse. Step 3.1 is completed.
- 3.2 Visit  $C$ ; that is, print  $C$ . This completes Step 3.2, so we proceed to Step 3.3.
- 3.3 Because the right subtree of  $C$  is empty, there is nothing to traverse. Step 3.3 is completed.

This completes Step 3, which in turn completes the traversal of the binary tree.

Clearly, the inorder traversal of the previous binary tree prints the vertices in the following order.

*Inorder sequence: BDAC*

Similarly, the preorder and postorder traversals print the vertices in the following order.

*Preorder sequence: ABDC*

*Postorder sequence: DBCA*

The following algorithm implements the inorder traversal algorithm on a binary tree.

### ALGORITHM 11.2: Inorder Traversal.

*Input:* `root`—a reference to the root of the binary tree. If the binary tree is empty, `root` is `null`.

*Output:* Vertices of the binary tree inorder sequence

1. **procedure** `inorder`(`root`)
2. **begin**

```

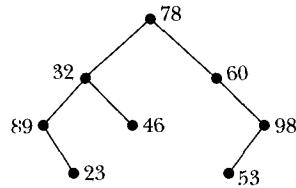
3. if root is not null then //if tree is not empty
4. begin
5.   p := left child of root;
6.   inorder(p);
7.   print label of root;
8.   p := right child of root;
9.   inorder(p);
10. end
11. end

```

In a similar way we can write algorithms to do preorder and postorder traversal on a binary tree.

## Binary Search Trees

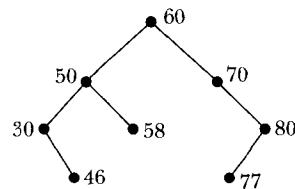
The basic operations performed on a binary tree are search, insertion, and deletion. However, both insertion and deletion require the binary tree to be traversed. Consider the binary tree in Figure 11.21.



**FIGURE 11.21** A binary tree

Suppose that we want to determine whether 50 is in the binary tree. To do so, we can use any of the previous traversal algorithms to visit each vertex and compare the search item with the data stored in the vertex. However, this could require us to traverse a large part of the binary tree, so the search would be slow. We need to visit each vertex in the binary tree until either the item is found or we have traversed the entire binary tree because no criteria exist to guide our search.

On the other hand, consider the binary tree in Figure 11.22.



**FIGURE 11.22** Binary search tree

In the binary tree in Figure 11.22, the value of each vertex is

- larger than the values of the vertices in its left subtree and
- smaller than the values of the vertices in its right subtree.

That is, the binary tree has some structure. Suppose that we want to determine whether 58 is in this binary tree. As before, we must start our search at the root. We compare 58 with the root; that is, we compare 58 with 60. Because  $58 < 60$ , it is guaranteed that 58 will not be in the right subtree of the root. Therefore, if 58 is in the binary tree, then it must be in the left subtree of the root. We go to the left child, which is the vertex with label 50. We now apply the same criteria at this vertex. Because  $58 > 50$ , we must go to the right child of this vertex, which is the vertex with value 58. At this vertex we find 58.

This example shows that every time we move down to a child, we eliminate one of the subtrees of the vertex from our search. If the binary tree is nicely constructed, then the search is very similar to the binary search on arrays.

The binary tree in Figure 11.22 is a special type of binary tree, called a *binary search tree*. (In the following definition, by the term key of the vertex we mean the key of the data item that uniquely identifies the item.)

---

**DEFINITION 11.2.20** ► A **binary search tree** is a binary tree such that if it is nonempty, then for each vertex  $v$ , the key of  $v$  is larger than the key of each vertex in its left subtree and smaller than the key of each vertex in its right subtree.

Next we describe the search and insert algorithm.

The function `search` searches the binary search tree for a given item. If the item is found in the binary search tree, it returns true; otherwise, it returns false. As usual, we begin our search at the root. Suppose `root` is a reference to the root of a binary search tree.

If the binary search tree is nonempty, we first compare the search item with the root. If they are the same, we stop the search and return true. Otherwise, if the search item is smaller than the root, we follow the left subtree; otherwise, we follow the right subtree. We repeat this process at the next vertex. If the search item is in the binary search tree, our search ends at the vertex containing the search item; otherwise, the search ends at an empty subtree. Thus, the general algorithm is as follows. (Suppose `root` is a reference to a binary search tree. If the binary search tree is empty, then `root` is null.)

#### ALGORITHM 11.3: Searching a binary search tree.

*Input:* `root`—A reference to a binary search tree  
`searchItem`—item to be searched for

*Output:* Returns true if `searchItem` is in the tree; otherwise, it returns false

```

1. function search(root, searchItem)
2. begin
3.   current := root;
4.   while current is not null do
5.     if label of current = searchItem then
6.       return true;
7.     else
8.       if label of current > searchItem then
9.         current := left child of current; //go to the left child
10.      else
```

```

11.         current := right child of current //go to the right child
12.     return false;
13. end

```

### Insert

After inserting an item in a binary search tree, the resulting binary tree must also be a binary search tree. To insert a new item, first we search the binary search tree and find the place where the new item is to be inserted. The search algorithm to find the place for the new item is similar to the search algorithm of the function search. Here we traverse the binary search tree with two references—a reference, say `current`, to check the `current` vertex and a reference, say `trailCurrent`, referring to the parent of `current`. Because duplicate items are not allowed, our search must end at an empty subtree. We can then use the reference `trailCurrent` to insert the new item at the proper place. The item to be inserted, `insertItem`, is passed as a parameter to the procedure `insert`.

#### **ALGORITHM 11.4:** Insertion into a binary search tree.

*Input:* `root`—a reference to the root of a binary search tree  
`insertItem`—item to be inserted into the binary search tree

*Output:* `root`—reference to the binary search tree after inserting `insertItem`

```

1. procedure insertBinSearchTree(root, insertItem)
2. begin
3.   Create a new vertex, newVertex, and copy insertItem into newVertex.
4.   if root is null then //the tree is empty
5.     root := newVertex;
6.   else
7.     begin
8.       current = root;
9.       while current is not null do //search the binary tree
10.      begin
11.        trailCurrent = current;
12.        if label of current = insertItem then
13.          begin
14.            print "Error: Cannot insert duplicate"
15.            return;
16.          end
17.        else
18.          if label of current > insertItem then
19.            current := left child cf current;
20.          else
21.            current := right child cf current;
22.        end

```

```

23.      //insert the new vertex in the binary search tree
24.      if label of trailCurrent > insertItem then
25.          left child of trailCurrent := newVertex;
26.      else
27.          right child of trailCurrent := newVertex;
28.      end
29. end

```

## Binary Search Tree Analysis

This section provides an analysis of the performance of binary search trees. Let  $T$  be a binary search tree with  $n$  vertices, where  $n > 0$ . Suppose that we want to determine whether an item,  $x$ , is in  $T$ . The performance of the search algorithm depends on the shape of  $T$ . Let us first consider the worst case. In the worst case,  $T$  is linear. That is,  $T$  is one of the forms shown in Figure 11.23.

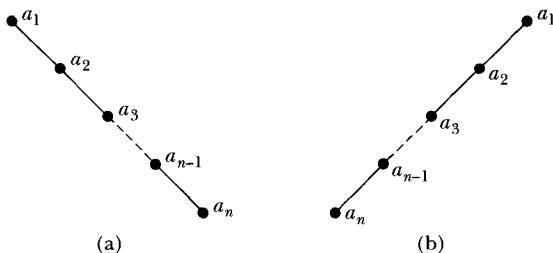


FIGURE 11.23 Linear binary trees

Because  $T$  is linear, the performance of the search algorithm on  $T$  is the same as its performance on a linear list. Therefore, in the successful case, on average, the search algorithm makes  $\frac{n+1}{2}$  key comparisons. In the unsuccessful case, it makes  $n$  comparisons.

We now consider the analysis of an arbitrary search tree. Let us consider the average-case behavior. In the successful case, the search would end at a vertex. Because there are  $n$  items, there are  $n!$  possible orderings of the keys. We assume that all  $n!$  orderings of the keys are possible. Let  $S(n)$  denote the number of comparisons in the average successful case and  $U(n)$  denote the number of comparisons in the average unsuccessful case.

The number of comparisons required to determine whether  $x$  is in  $T$  is one more than the number of comparisons required to insert  $x$  in  $T$ . Furthermore, the number of comparisons required to insert  $x$  in  $T$  is the same as the number of comparisons made in the unsuccessful search, reflecting that  $x$  is not in  $T$ . From this, it follows that

$$S(n) = 1 + \frac{U(0) + U(1) + \cdots + U(n-1)}{n}. \quad (11.1)$$

It is also known that

$$S(n) = \left(1 + \frac{1}{n}\right) U(n) - 3. \quad (11.2)$$

Solving Equations (11.1) and (11.2), it can be shown that

$$U(n) \approx 2.77 \lg n$$

and

$$S(n) \approx 2.77 \lg n.$$

We can now formulate the following result.

**Theorem 11.2.21:** Let  $T$  be a binary search tree with  $n$  vertices, where  $n > 0$ . The average number of vertices visited in a search of  $T$  is approximately  $1.39 \lg n$  and the number of key comparisons is approximately  $2.77 \lg n$ .

## Expression Trees

The usual notation for writing arithmetic expressions (the notation we learned in elementary school) is called **infix** notation, in which the operator is written between the operands. For example, in the expression  $a + b$ , the operator  $+$  is between the operands  $a$  and  $b$ . In infix notation, the operators have precedence. That is, we evaluate expressions from left to right, and multiplication and division have higher precedence than addition and subtraction. If we want to evaluate an expression in a different order, we must include parentheses. For example, in the expression  $x + y \cdot z$ , we first evaluate  $\cdot$  using the operands  $y$  and  $z$ , and then we evaluate  $+$  using the operand  $x$  and the result of  $y \cdot z$ .

In the early 1950s, the Polish mathematician Lukasiewicz discovered that if operators were written before the operands (**prefix**, or Polish notation; for example,  $+xy$ ) or after the operands (**suffix**, **postfix**, or reverse Polish notation; for example,  $xy+$ ), the parentheses could be omitted. For example, the expression

$$x + y \cdot z$$

in a postfix expression is

$$xyz \cdot +$$

The following example shows infix expressions and their equivalent postfix expressions.

### EXAMPLE 11.2.22

Table 11.1 shows various infix expressions and their equivalent postfix expressions.

**Table 11.1** Infix Expressions and their equivalent postfix expressions

| Infix Expression              | Equivalent Postfix Expression |
|-------------------------------|-------------------------------|
| $x + y$                       | $xy+$                         |
| $x + y \cdot z$               | $xyz \cdot +$                 |
| $x \cdot y + z$               | $xy \cdot z +$                |
| $(x + y) \cdot z$             | $xy + z \cdot$                |
| $(x - y) \cdot (z + w)$       | $xy - zw + \cdot$             |
| $(x + y) \cdot (z - w/t) + s$ | $xy + zw t / - \cdot s +$     |

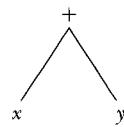
Shortly after Lukasiewicz's discovery, it was realized that postfix notation had important applications in computer science. In fact, many compilers now first

translate arithmetic expressions into some form of postfix notation and then translate this postfix expression into machine code.

Corresponding to an infix expression, we can construct a binary tree using the bottom-up approach. For example, consider the infix expression

$$x + y.$$

Corresponding to this expression, we can construct the binary tree called an **expression tree**, shown in Figure 11.24.

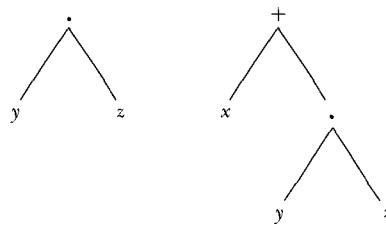


**FIGURE 11.24**

Now consider the expression

$$x + y \cdot z.$$

In this expression, the operator  $\cdot$  has a higher precedence. So first we construct the tree corresponding to the expression  $y \cdot z$  (see Figure 11.25(a)). Then we construct the binary tree corresponding to  $x + (y \cdot z)$  (see Figure 11.25(b)).



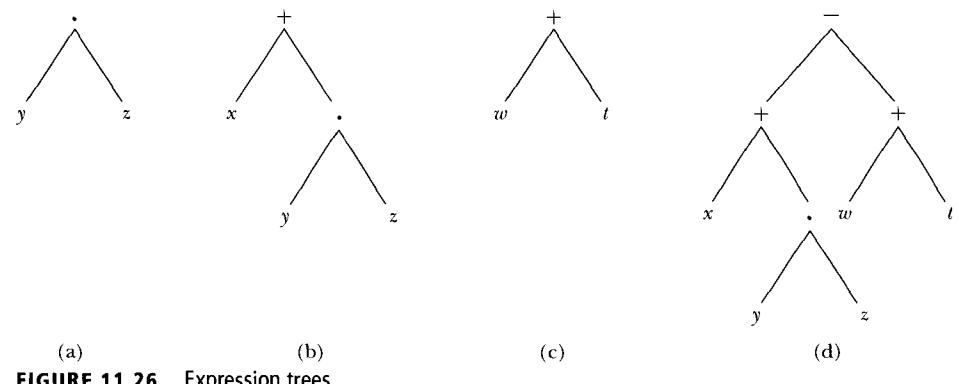
**FIGURE 11.25** Expression trees

### EXAMPLE 11.2.23

Consider the infix expression

$$(x + y \cdot z) - (w + t).$$

The expression corresponding to this infix expression is shown in Figure 11.26. Here we construct the expression corresponding to the expressions  $(x + y \cdot z)$  and  $(w + t)$  and then join those binary trees using the operator  $-$ .



**FIGURE 11.26** Expression trees

Let us consider the expression tree in Figure 11.24. If we do an inorder traversal of this binary tree, then the expression obtained is the infix expression  $x + y$ . However, if we do a preorder traversal of this binary tree, then the expression obtained is  $+xy$ , which is the prefix form of the expression  $x + y$ . Similarly, if we do the postorder traversal of this binary tree, then we get the expression  $xy+$ , which is the postfix form of the expression  $x + y$ .

**EXAMPLE 11.2.24**

Let us consider the binary tree in Figure 11.25(b). The inorder traversal of this binary tree produces the infix expression  $x + y \cdot z$ , the preorder traversal produces the expression  $+x \cdot yz$ , and the postorder traversal produces the expression  $xyz \cdot +$ .

Next we illustrate how to evaluate a postfix expression. Postfix expressions can be evaluated as follows: Scan the expression from left to right. When an operator is found, back up to get the required number of operands, perform the operation, and continue.

**EXAMPLE 11.2.25**

Consider the postfix expression:  $6\ 3 + 2\cdot$ . This expression is evaluated as follows:

$$\begin{array}{r} 6\ \underbrace{3+}_{} 2\cdot \\ 6+3=9 \\ 9\ \underbrace{2\cdot}_{} \\ 9\cdot 2=18 \\ 18 \end{array}$$

Thus,  $6\ 3 + 2\cdot = 18$ .

**EXAMPLE 11.2.26**

Consider the postfix expression:  $24\ 4\ 3\cdot - 16\ 4 / +$ . Let us evaluate this expression.

$$\begin{array}{r} 24\ 4\ 3\cdot - 16\ 4 / + \\ 24\ \underbrace{4\ 3\cdot}_{} - 16\ 4 / + \\ 4\cdot 3=12 \\ 24\ \underbrace{12-}_{} 16\ 4 / + \\ 24-12=12 \\ 12\ \underbrace{16\ 4 /}_{} + \\ 16/4=4 \\ 12\ \underbrace{4+}_{} \\ 12+4=16 \\ 16. \end{array}$$

**REMARK 11.2.27** ▶ From Examples 11.2.25 and 11.2.26, it is clear that in order to evaluate a postfix expression, we do not need to know the precedence of the operators.

**REMARK 11.2.28** ▶ Just as we can construct expression trees corresponding to an arithmetic expression, we can also construct an expression tree corresponding to a logical expression. For example, the expression tree corresponding to the logical expression  $p \vee (q \wedge r)$  is the binary tree shown in Figure 11.27.

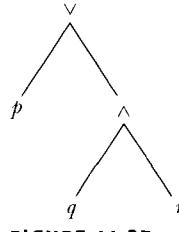


FIGURE 11.27

## Isomorphism of Binary Trees

In the preceding sections, we discussed binary trees in detail. Figure 11.18 (in Example 11.2.15) shows two binary trees with two vertices. It can be shown that these are the only two distinct binary trees with two vertices, in the sense that any binary tree with two vertices must be one of these forms. In other words, there are two nonisomorphic binary trees with two vertices. What about the number of nonisomorphic binary trees with 3 nodes or 4 nodes? In this section, first we formally define the notion of isomorphism of binary trees and then give a formula to determine the number of nonisomorphic binary trees with  $n$  vertices, where  $n \geq 0$ .

---

**DEFINITION 11.2.29** ▶ Let  $T_1 = (V_1, E_1)$  and  $T_2 = (V_2, E_2)$  be nonempty binary trees. Let  $r_1$  and  $r_2$  be the roots of  $T_1$  and  $T_2$ , respectively. We say that  $T_1$  is *isomorphic* to  $T_2$  if there exists a one-to-one correspondence  $f : V_1 \rightarrow V_2$  such that the following properties hold.

- (i)  $f(r_1) = r_2$ .
- (ii) For all  $u, v \in V_1$ ,  $u$  and  $v$  are adjacent if and only if  $f(u)$  and  $f(v)$  are adjacent.
- (iii) For all  $u, v \in V_1$ ,  $u$  is a left child of  $v$  if and only if  $f(u)$  is a left child of  $f(v)$ .
- (iv) For all  $u, v \in V_1$ ,  $u$  is a right child of  $v$  if and only if  $f(u)$  is a right child of  $f(v)$ .

The function  $f$  is called an *isomorphism* of  $T_1$  onto  $T_2$ . If such a function  $f$  does not exist, then we say that  $T_1$  and  $T_2$  are *nonisomorphic*.

---

**REMARK 11.2.30** ▶ If  $T_1$  and  $T_2$  are empty binary trees, they are considered isomorphic.

Let  $a_n$  denote the number of nonisomorphic binary trees with  $n$  vertices, where  $n \geq 1$ . Then it follows that  $a_1 = 1$ . From Example 11.2.15 it follows that  $a_2 = 2$ . It can be shown that  $a_3 = 5$ .

Next, we derive a recurrence relation for  $a_n$ . Let  $T$  be a binary tree with  $n$  vertices,  $n > 1$ . Let  $r$  be the root of  $T$ . Then there are  $n - 1$  vertices in the left and right subtree of  $r$ . If there are, say  $i$ , vertices in the left subtree of  $r$ , then there are  $n - 1 - i$  vertices in the right subtree of  $r$ . There are  $a_i$  binary trees with  $i$  vertices and  $a_{n-1-i}$  binary trees with  $n - 1 - i$  vertices. It follows that there are  $a_i$  ways to construct the left subtrees of  $r$  and  $a_{n-1-i}$  ways to construct the right subtrees of  $r$ . By the multiplication principle, it follows that there are  $a_i a_{n-1-i}$  ways to construct the binary tree  $T$  with  $i$  vertices in its left subtree and  $n - 1 - i$  vertices in its right subtree. Now  $i$  ranges from 0 to  $n - 1$ . Hence, the number of nonisomorphic binary trees with  $n$  vertices is

$$a_n = \sum_{i=0}^{n-1} a_i a_{n-1-i}. \quad (11.3)$$

Thus, we have the recurrence relation  $a_n = \sum_{i=0}^{n-1} a_i a_{n-1-i}$ ,  $n > 1$ , with the initial condition  $a_1 = 1$ . The elements of the sequence  $\{a_n\}$  are known as *Catalan's number*. It is known that the explicit formula for  $a_n$  is given by

$$a_n = \frac{C(2n, n)}{n+1}.$$

We have thus proved the following theorem.

**Theorem 11.2.31:** The number of nonisomorphic binary trees with  $n$  vertices  $n \geq 1$  is  $\frac{C(2n, n)}{n+1}$ .

## WORKED-OUT EXERCISES

**Exercise 1:** List the vertices of the binary tree in Figure 11.28 in inorder, preorder, and postorder sequence.

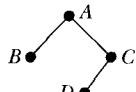


FIGURE 11.28  
A binary tree

**Solution:** For the inorder sequence, the first vertex listed is  $B$ . This completes the traversal of the left subtree of  $A$ . The next vertex visited, i.e., listed, is  $A$ . After this, we traverse the right subtree of  $A$ . The root of the right subtree of  $A$  is  $C$ . To do an inorder traversal of the binary tree with root  $C$ , first we visit the left subtree. So we go to the root of the left subtree of  $C$ , which is  $D$ . The left subtree of  $D$  is empty, so the left subtree is traversed. Next we list  $D$ . The left subtree of  $C$  is visited, so we visit  $C$ , i.e., list  $C$ . Because  $C$  has no right subtree, the traversal ends. Hence, the vertices in the inorder sequence are  $BADC$ .

Let us consider the preorder traversal. In preorder traversal, first visit the vertex, then traverse its left subtree and then traverse its right subtree. The traversal starts at  $A$ . So first we list  $A$ . Next we traverse the left subtree of  $A$ . The root of the left subtree of  $A$  is  $B$ . So we start traversal at  $B$ . First we list  $B$ , and then traverse the left followed by the traversal of the right subtree of  $B$ . Now both the left and right subtrees of  $B$  are empty. This shows that the left subtree of  $A$  is completely traversed. Next we traverse the right subtree of  $A$ . For this, we go to the root,  $C$ , of the right subtree of  $A$ . Next we start a preorder traversal at  $C$ . For this first we visit  $C$ , i.e., list  $C$ . Then we traverse the left subtree of  $C$  followed by the traversal of the right subtree of  $C$ . The root of the left subtree of  $C$  is  $D$ . So we do a preorder traversal of the binary tree with root  $D$ . First we list  $D$ , and then traverse the left subtree of  $D$  followed by the traversal of the right subtree of  $D$ . Both the left and right subtrees of  $D$  are empty. Thus, the

left subtree of  $C$  is traversed. Because the right subtree of  $C$  is empty, the preorder traversal of the binary tree ends. It follows that the preorder sequence is  $ABCD$ .

Now consider the postorder traversal. In postorder traversal, first we traverse the left subtree, then the right subtree, followed by a visit to the vertex. We start the postorder traversal at  $A$ . First we traverse the left subtree of  $A$ . Notice that the traversal of the left subtree of  $A$  results in the listing of the vertex  $B$ . Next we traverse the right subtree of  $A$ . The root of the right subtree of  $A$  is  $C$ . We do a postorder traversal of the binary tree with root  $C$ . For this, first we traverse the left subtree of  $C$  and then the right subtree of  $C$ . The traversal of the left subtree of  $C$  results in listing the vertex  $D$ . Next, the right subtree of  $C$  is empty. Thus, we have traversed the left and right subtree of  $C$ . Next, we visit the node  $C$ , i.e., list  $C$ . This results in a complete traversal of the right subtree of  $A$ . So next we list  $A$ . Hence, the listing of the vertices of the binary tree in postorder sequence is  $BDCA$ .

**Exercise 2:** Insert 10, 70, and 15, in this order, in the binary search tree of Figure 11.29. Draw the binary search tree after each insertion.

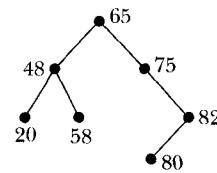


FIGURE 11.29  
Binary search tree

**Solution:** Figure 11.30 shows the binary search tree after inserting 10, 70, and 15, in this order.

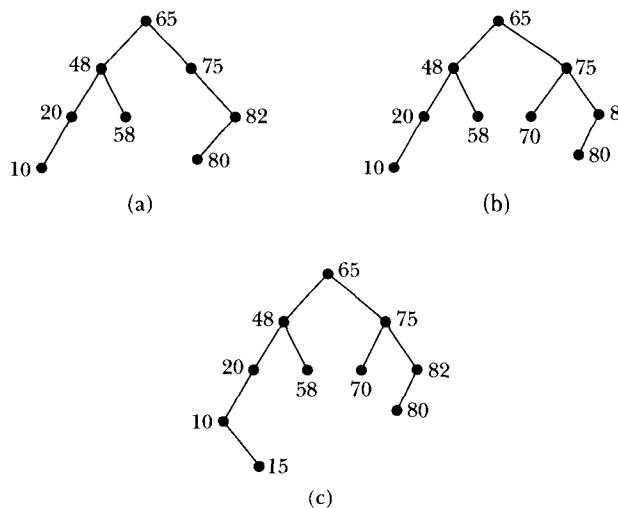
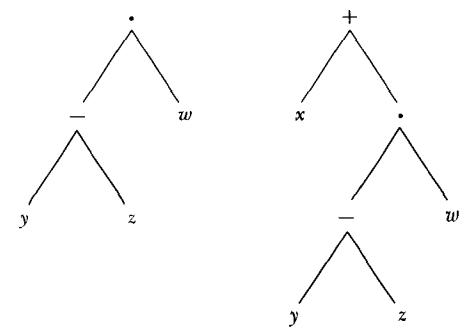


FIGURE 11.30 Binary search tree

**Exercise 3:** Draw the expression corresponding to the infix expression  $x + (y - z) \cdot w$ .

**Solution:** For the expression, first we construct the expression tree for the expressions  $y - z$  which is then joined with the operator  $\cdot$  and the operand  $w$ . Finally, we join this tree with the tree of the operand  $x$  and the operator  $+$ . (See Figure 11.31)

FIGURE 11.31 Expression tree for the expression  $x + (y - z) \cdot w$ 

**Exercise 4:** Evaluate the postfix expression  
 $24\ 4\ +\ 3\ \cdot\ 7\ / 26\ 4\ -\ +$

**Solution:** We have

$$\begin{aligned}
 & 24\ 4\ +\ 3\ \cdot\ 7\ / 26\ 4\ -\ + \\
 & \underbrace{24\ 4+}_{24+4=28}\ 3\ \cdot\ 7\ / 26\ 4\ -\ + \\
 & 28\ \cdot\ 7\ / 26\ 4\ -\ + \\
 & \underbrace{28\cdot}_{28\cdot3=84} 7\ / 26\ 4\ -\ + \\
 & 84\ \underbrace{7/}_{84/7=12}\ 26\ 4\ -\ + \\
 & 12\ \underbrace{26\ 4-}_{26-4=22}\ + \\
 & 12\ \underbrace{22+}_{12+22=34} \\
 & 34
 \end{aligned}$$

## SECTION REVIEW

### Key Terms

|                     |                    |                     |
|---------------------|--------------------|---------------------|
| rooted tree         | binary tree        | postorder traversal |
| level               | trivial tree       | inorder sequence    |
| child               | left child         | preorder sequence   |
| terminal vertex     | right child        | postorder sequence  |
| leaf                | left subtree       | binary search tree  |
| internal vertex     | right subtree      | infix               |
| descendant          | full binary tree   | prefix              |
| ordered rooted tree | inorder traversal  | postfix             |
| height              | preorder traversal | expression tree     |

### Some Key Definitions

1. A rooted tree is a tree in which a particular vertex is designated as the root.
2. Let  $T$  be a tree with  $v_0$  as a root and let  $(v_0, e_1, v_1, e_2, v_2, \dots, e_n, v_n)$  be a path in  $T$ . Then

- (i)  $v_k$  is called a child of  $v_{k-1}$  for  $k = 1, 2, \dots, n$ .
  - (ii) If a vertex  $v$  of  $T$  has no children, then  $v$  is called a terminal vertex, or a leaf.
  - (iii) If  $v$  is not a terminal vertex, then  $v$  is called an internal vertex of  $T$ .
3. An ordered rooted tree  $T$  is called a binary tree if either it is an empty graph or each vertex has no children, one child, or two children. If a rooted tree has only one vertex and no edges, then the tree is called a trivial tree. If a vertex has one child, that child is designated as either a left child or a right child (but not both). If a vertex has two children, then the first child (according to the given order) is designated a left child and the other child is designated a right child.
4. Let  $T$  be a nonempty binary tree and  $v$  be a vertex in  $T$ . The left subtree of  $v$  is a binary tree with the left child of  $v$  as the root which contains all the descendants of this left child and all edges incident to these descendants. Similarly, the right subtree of  $v$  is a binary tree with the right child of  $v$  as the root which contains all the descendants of this right child and all edges incident to these descendants. The vertex set of the left subtree of  $v$  is denoted by  $L_v$  and the vertex set of the right subtree of  $v$  is denoted by  $R_v$ . Now the left subtree,  $L_v$ , of  $v$  contains all the descendants of the left child of  $v$  and all edges incident to these descendants. Hence, if  $v_1, v_2, \dots, v_k$  are the only descendants of the left child of  $v$ , then we write  $L_v = \{v_1, v_2, \dots, v_k\}$ . A similar convention applies for  $R_v$ .

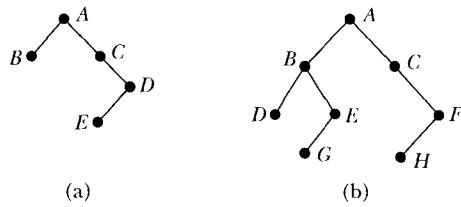
## Some Key Results

1. Let  $T$  be a full binary tree with  $i$  internal vertices. Then  $T$  has  $i + 1$  terminal vertices and  $2i + 1$  total vertices.
2. Let  $T$  be a binary tree such that  $T$  has  $t$  terminal vertices and height  $h$ . Then  $t \leq 2^{h-1}$ .
3. Let  $T$  be a binary search tree with  $n$  vertices, where  $n > 0$ . The average number of vertices visited in a search of  $T$  is approximately  $1.39 \lg n$  and the number of key comparisons is approximately  $2.77 \lg n$ .

## EXERCISES

1. Draw all binary trees with three vertices.
2. Draw all binary trees with four vertices.

Use the binary trees in Figure 11.32 for Exercises 3–5.



**FIGURE 11.32** Binary trees

3. List the vertices of the binary trees in Figure 11.32 in inorder sequence.

4. List the vertices of the binary trees in Figure 11.32 in preorder sequence.
5. List the vertices of the binary trees in Figure 11.32 in postorder sequence.
6. Write an algorithm to do a preorder traversal of a binary tree.
7. Write an algorithm to do a postorder traversal of a binary tree.
8. Write an algorithm to determine the number of vertices in a binary tree.
9. Write an algorithm to determine the number of terminal vertices, i.e., leaves, in a binary tree.
10. Write an algorithm to determine the number of vertices in a binary tree that have only one child.
11. Prove Theorem 11.2.21.

12. Let  $T$  be a binary tree with  $t$  terminal vertices. If  $h_T$  is the length of the longest path from the root to a leaf in  $T$ , then prove that  $t \leq 2^{h_T}$ .
13. The vertices of a binary tree in preorder sequence are  $ABCDEFGHIJKLM$  and in inorder sequence are  $CEDFBAHJKGML$ . Draw the binary tree.
14. Evaluate the postfix expression  $6 \ 4 + 3 \cdot 16 \ 4 / -$
15. Evaluate the postfix expression  

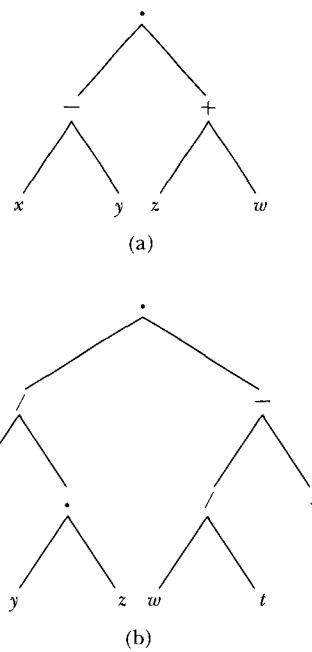
$$12 \ 25 \ 5 \ 1 / / \cdot 8 \ 7 + -$$
16. Evaluate the postfix expression  

$$70 \ 14 \ 4 \ 5 \ 15 \ 3 / \cdot - - / 6 +$$
17. Convert the infix expression  $(A + B) \cdot (C + D) - E$  to postfix notation.
18. Convert the infix expression  $A - (B + C) \cdot D + E / F$  to postfix notation.
19. Convert the infix expression  

$$((A + B) / (C - D) + E) \cdot F - G$$
 to postfix notation.
20. Convert the infix expression  

$$A + B \cdot (C + D) - E / F \cdot G + H$$
 to postfix notation.
21. Write the equivalent infix expression for the postfix expression  $a \ b \ c +$
22. Write the equivalent infix expression for the postfix expression  $ab + cd -$ .
23. Write the equivalent infix expression for the postfix expression  $ab - c - d$ .
24. Draw the expression trees for each of the infix expressions in Exercises 17–20.

Use the expression trees in Figure 11.33 for Exercises 25–27.

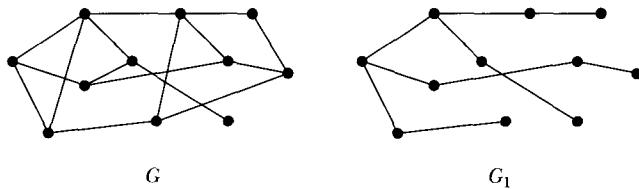


**FIGURE 11.33** Expression trees

25. Write the infix expressions corresponding to the expression trees in Figure 11.33.
26. Write the prefix expressions corresponding to the expression trees in Figure 11.33.
27. Write the postfix expressions corresponding to the expression trees in Figure 11.33.
28. Find the number of nonisomorphic binary trees with 5 vertices.
29. Find the number of nonisomorphic binary trees with 8 vertices.

## 11.3 SPANNING TREES

In the preceding sections, we discussed binary trees and expression trees. In this section, we study another type of trees. Consider graphs  $G$  and  $G_1$  shown in Figure 11.34.



**FIGURE 11.34** Graphs  $G$  and  $G_1$

Graph  $G_1$  is a subgraph of graph  $G$ . Notice that  $G_1$  contains all the vertices of  $G$  and it is also a tree. Such a subgraph is called a *spanning tree* of  $G$ . More formally, we have the following definition.

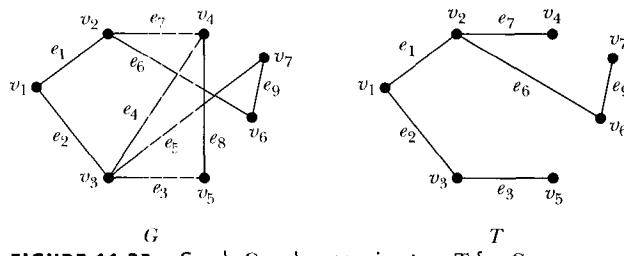
---

**DEFINITION 11.3.1** ► A tree  $T$  is called a **spanning tree** of a graph  $G$  if  $T$  is a subgraph of  $G$  and  $T$  contains all the vertices of  $G$ .

In this section, we discuss the problem of finding a spanning tree of a given graph.

**EXAMPLE 11.3.2**

Suppose seven universities in one city are involved in a joint research project funded by a major corporation. The research centers at each university frequently send a large amount of data to the other universities. To expedite the transmission of data and protect it from hackers, as well as to finish the project on time, the company wants to connect the seven universities using fiber-optics lines. However, to minimize the cost of construction, the company wants to lay as few lines as possible. Also, for reasons such as hard surfaces, it is not possible to lay a direct fiber-optics line between some of the universities. In fact, it is not necessary to have a direct line between any two universities. The engineers look at the areas surrounding the universities to determine the possible lines that can be laid to connect the universities (see graph  $G$  in Figure 11.35).



**FIGURE 11.35** Graph  $G$  and a spanning tree  $T$  for  $G$

To lay the fiber-optics lines so that the universities can be connected, but not necessarily directly connected, the engineers find a tree  $T$  that is a subgraph of  $G$  such that  $T$  contains all the vertices of  $G$ . One possible solution is shown by the tree  $T$  in Figure 11.35. As we can see,  $T$  is a spanning tree of  $G$ .

If we associate the cost factor of laying the fiber-optics lines, then we will try to find a spanning tree that has a minimum cost. We will discuss such issues in the next section.

---

**REMARK 11.3.3** ▶ Note that a spanning tree of a graph does not need to be unique.

In the beginning of this chapter, we remarked that Cayley was the first person to introduce (in 1857) the term *tree*, and in 1889, he proved the following theorem.

**Theorem 11.3.4:** The complete graph  $K_n$ ,  $n \geq 3$ , has  $n^{n-2}$  nonisomorphic spanning trees.

The following theorem gives a necessary and sufficient condition for a graph to have a spanning tree.

**Theorem 11.3.5:** A graph  $G$  has a spanning tree if and only if  $G$  is connected.

**Proof:** Suppose  $G$  has a spanning tree,  $G_1$ .  $G_1$  contains all the vertices of  $G$ . Then between any two vertices there exists a path in  $G_1$ , which is also a path of  $G$ . Hence,  $G$  is a connected graph.

Conversely, suppose  $G$  is a connected graph. If  $G$  has no cycles, then  $G$  is a tree. Assume that  $G$  has cycles. Let  $C_1$  be a cycle in  $G$  and  $e_1$  be an edge in  $C_1$ . Now construct the graph  $G_1 = G - \{e_1\}$ , which is obtained by deleting  $e_1$  from  $G$  but not removing any vertex from  $G$ . Clearly,  $G_1$  is a subgraph of  $G$  and it contains all the vertices of  $G$ . Because  $e_1$  is an edge of a cycle,  $G_1$  is still a connected graph. If  $G_1$  is acyclic, then  $G_1$  is a tree. If  $G_1$  contains a cycle  $C_2$ , then we delete an edge  $e_2$  from  $C_2$  and construct a connected subgraph  $G_2$  that contains all the vertices. If  $G_2$  contains cycles, then we continue this process. Because  $G$  has a finite number of edges, it contains only a finite number of cycles. Hence, continuing the process of deleting an edge from a cycle, we eventually obtain a connected subgraph  $G_k$  that contains all the vertices of  $G$  and is also acyclic. It follows that  $G_k$  is a spanning tree of  $G$ . ■

Consider graph  $G$  in Figure 11.36. In this graph, we find that  $(v_1, e_2, v_3, e_4, v_4, e_7, v_2, e_1, v_1)$  is a cycle. We remove  $e_4$  from this cycle and then obtain the graph  $G_1 = G - \{e_4\}$ . This is a connected subgraph and it contains all the vertices (see graph  $G_1$  in Figure 11.36).

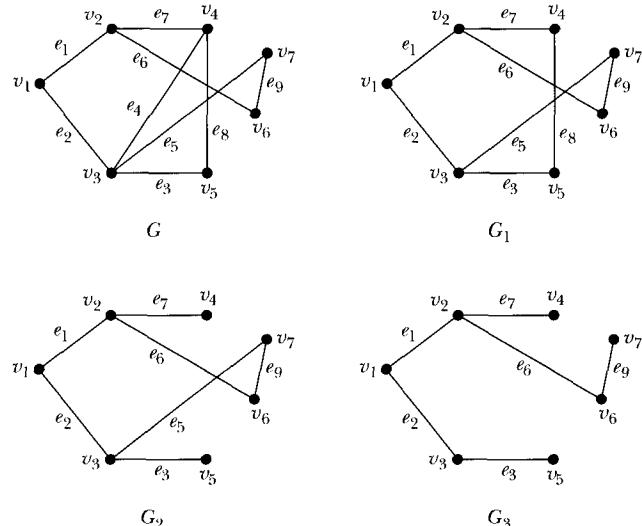


FIGURE 11.36 Graphs  $G$ ,  $G_1$ ,  $G_2$ , and  $G_3$

Now in graph  $G_1$ ,  $(v_1, e_2, v_3, e_3, v_5, e_8, v_4, e_7, v_2, e_1, v_1)$  is a cycle. We delete  $e_8$  from this cycle and obtain the subgraph  $G_2 = G - \{e_4, e_8\}$  (see graph  $G_2$  in Figure 11.36). This subgraph  $G_2$  is a connected subgraph that contains all the vertices of  $G$ . In  $G_2$ ,  $(v_1, e_2, v_3, e_5, v_7, e_9, v_6, e_8, v_2, e_1, v_1)$  is a cycle. We delete  $e_5$  from this cycle and obtain the subgraph  $G_3 = G - \{e_4, e_8, e_5\}$  (see graph  $G_3$  in Figure 11.36). Now  $G_3$  is a subgraph of  $G$  and it does not contain any cycle. Hence,  $G_3$  is a spanning tree of  $G$ . Notice that  $G_3$  is the same as  $T$  in Figure 11.35.

By Theorem 11.3.5, every connected graph has a spanning tree. Moreover, the proof of this theorem also shows how to construct a spanning tree. However, the process of finding a cycle and eliminating an edge may not be efficient. Next we give a better algorithm that can be used to find a spanning tree in a connected graph. This algorithm is known as the breadth-first search spanning tree algorithm.

The breadth-first search spanning tree algorithm is similar to the breadth-first topological ordering discussed in Chapter 10. Let  $G = (V, E)$  be a connected

graph. Let  $T = (V_1, E_1)$  be a spanning tree of  $G$ . We start with a vertex,  $v_1$ , designated as the root. Initially,  $T$  only consists of the root node  $v$ , i.e.,  $V_1 = \{v_1\}$  and  $E_1 = \emptyset$ . Next we look at the vertices that are adjacent to  $v_1$ . (This will produce all the vertices at the first level in the tree.) All of these vertices and one of their edges to  $v_1$  are added to  $T$ . We repeat this process for the vertices at level 1, and so on. Next we give the algorithm to implement the breadth-first search spanning tree algorithm.

**ALGORITHM 11.5:** Breadth-first search spanning tree algorithm.

*Input:*  $G$ —a connected graph  
 $n$ —the number of vertices in  $G$   
 $v$ —the root of the spanning tree  $T$

*Output:*  $V(T)$ —vertex set of a spanning tree  $T$  of  $G$   
 $E(T)$ —edge set of  $T$

```

1. procedure bfst ( $G, n, v, T$ )
2. begin
3.    $V(T) := \{v\};$ 
4.    $E(T) := \emptyset;$ 
5.   queue :=  $\{v\};$ 
6.   while queue is not empty do
7.     begin
8.        $u :=$  first element of queue
9.       Remove the first element of queue;
10.      for each  $w$  adjacent to  $u$  do
11.        if  $w \notin V(T)$  then
12.          begin
13.             $V(T) := V(T) \cup \{w\};$ 
14.             $E(T) := E(T) \cup \{(u, w)\};$ 
15.            queue := queue  $\cup \{w\};$ 
16.          end
17.        end
18.      end

```

**EXAMPLE 11.3.6**

Consider the graph in Figure 11.37(a). The graphs in Figures 11.37(b)–(f) show the execution of the breadth-first search spanning tree algorithm.

The dashed lines in Figures 11.37(b)–(f) show how the vertices and edges are added to the spanning tree. After the fifth iteration, the next three iterations will not add anything to the spanning tree. The dashed lines in Figure 11.37(f) represent the edges of the spanning tree constructed by the breadth-first search spanning tree algorithm.

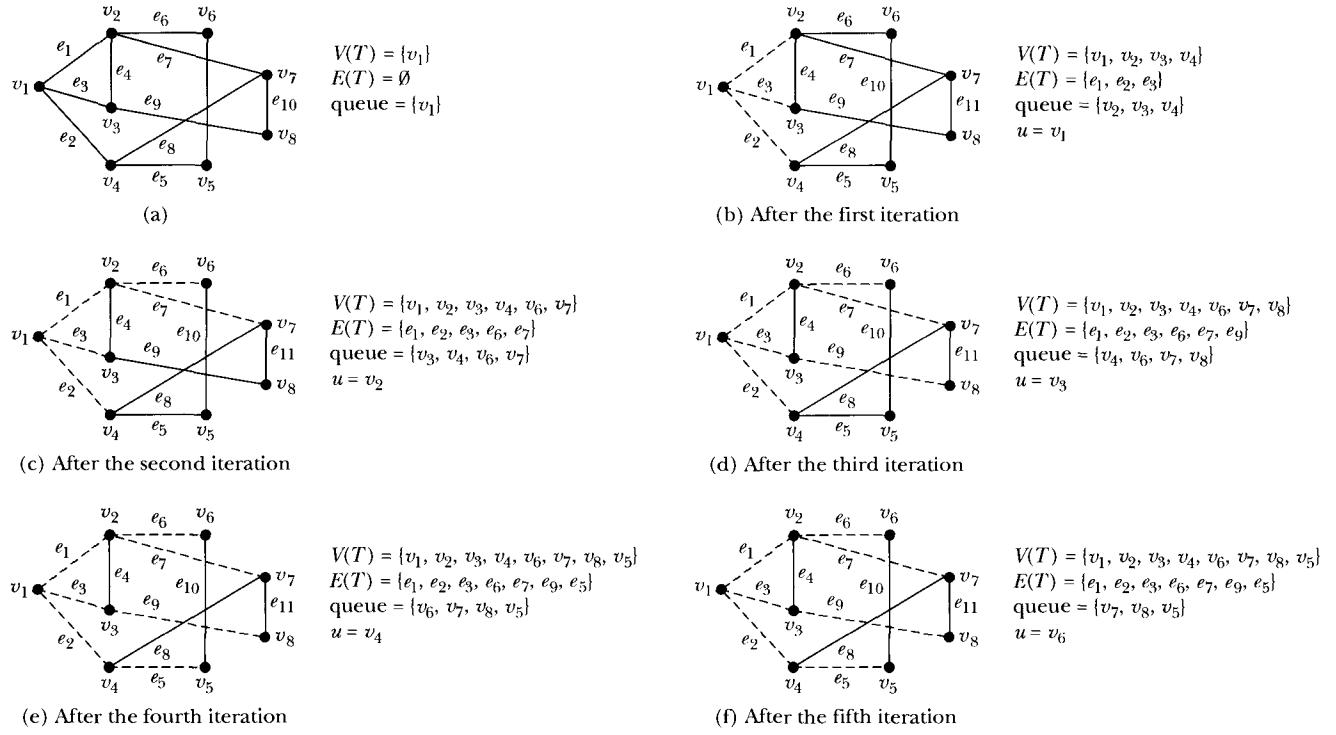


FIGURE 11.37 Breadth-first search spanning tree

## Minimal Spanning Tree

Consider the graph in Figure 11.38, which represents a company's airline connections among seven cities. The number on each edge represents some cost factor of maintaining the connection between the cities.

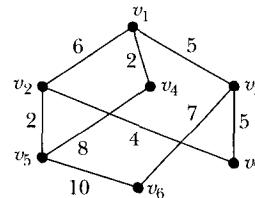
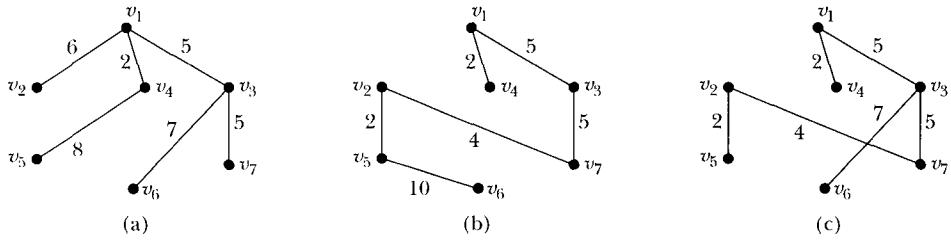


FIGURE 11.38 Airline connections among cities and the cost factors of maintaining the connections

Due to financial hardship, the company needs to shut down the maximum number of connections and still be able to fly from one city to another (maybe not directly). The graphs in Figure 11.39 show three different solutions.

The total cost factor of maintaining the remaining connections in Figure 11.39(a) is 33, in Figure 11.39(b) it is 28, and in Figure 11.39(c) it is 25. Out of these three solutions, the desired solution is of course the one shown in Figure 11.39(c), because it gives the lowest cost factor.

By a **weighted tree** we mean a tree such that each edge in the tree is assigned a nonnegative real number, called the *weight* of the edge.



**FIGURE 11.39** Possible solutions to the graph in Figure 11.38

**DEFINITION 11.3.7** ► Let  $T$  be a weighted tree. The **weight** of  $T$  is the sum of the weights of all the edges in  $T$ .

**DEFINITION 11.3.8** ► Let  $G$  be a weighted connected graph. A **minimal spanning tree** of  $G$  is a spanning tree with the minimum weight.

There are two well-known algorithms, Prim's algorithm and Kruskal's algorithm, for finding a minimal spanning tree of a graph. This section discusses Prim's algorithm.

**Prim's algorithm** builds the tree iteratively by adding edges until a minimal spanning tree is obtained. We start with a designated vertex, which we call the source vertex. At each iteration, a new edge that does not complete a cycle is added to the tree.

Let  $G$  be a weighted connected graph with  $n > 0$  vertices. Let  $v_1$  be the source vertex. Let  $T$  be the partially built tree. Initially,  $V(T)$  contains the source vertex and  $E(T)$  is empty. In the first iteration, we choose one of the edges, say  $e_1$ , of smallest weight in  $G$  such that  $e_1$  is incident with  $v_1$ . We place  $e_1$  in  $E(T)$  and the other end vertex, say  $v_2$ , of  $e_1$  in  $V(T)$ . Next we choose an edge  $e_2$  of smallest weight that is incident with either  $v_1$  or  $v_2$ , but the other end vertex, say  $v_3$ , of  $e_2$  is different from  $v_1, v_2$ . We add  $e_2$  in  $E(T)$  and  $v_3$  in  $V(T)$ . Thus, at each iteration, a new vertex that is not in  $V(T)$  is added to  $V(T)$ , such that an edge exists from a vertex in  $V(T)$  to the new vertex so that the corresponding edge has the smallest weight. The corresponding edge is added to  $E(T)$ . We repeat the process until we have  $n - 1$  edges in  $E(T)$ .

The general form of Prim's algorithm is as follows. (Let  $n$  be the number of vertices in  $G$ ).

**ALGORITHM 11.6: Prim's algorithm.**

*Input:*  $G$ —weighted connected graph  
 $n$ —number of vertices in  $G$   
 $W$ —weight matrix of  $G$   
 $v$ —root of a minimal spanning tree

*Output:*  $T$ —minimal spanning tree

- ```

1. procedure mstPrim(G,n,W,v,T)
2. begin
3.   V(T) := {v}; //initialize vertex set V(T) of T
4.   E(T) := Ø;    //initialize edge set E(T) of T
5.   for i := 1 to n - 1 do
6.     begin

```

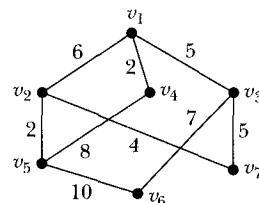
```

7.     minWeight :=  $\infty$ ;
8.     for  $j := 1$  to  $n$  do
9.         if  $v_j$  is in  $V(T)$  then
10.            for  $k := 1$  to  $n$  do
11.                if  $v_k$  is not in  $V(T)$  and  $W[v_j, v_k] < \text{minWeight}$  then
12.                    begin
13.                        endVertex :=  $v_k$ ;
14.                        e := ( $v_j, v_k$ );
15.                        minWeight :=  $W[v_j, v_k]$ ;
16.                    end
17.         $V(T) := V(T) \cup \{\text{endVertex}\}$ ;
18.         $E(T) := E(T) \cup \{e\}$ ;
19.    end
20. end

```

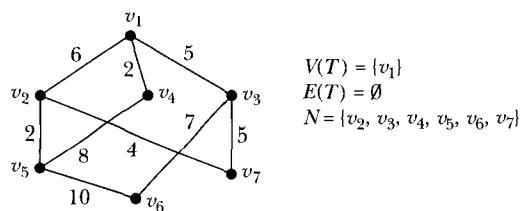
**REMARK 11.3.9** ► In Algorithm 11.6,  $W[v_j, v_k]$  denotes the weight of the edge from  $v_j$  to  $v_k$ .

Let us illustrate Prim's algorithm using graph  $G$  in Figure 11.40 (which is same as the graph in Figure 11.38).



**FIGURE 11.40**  
Weighted graph  $G$

Let  $N$  denote the set of vertices of  $G$  that are not in  $T$ . Suppose that the source vertex is  $v_1$ . After the statements in Lines 3 and 4 execute,  $V(T)$ ,  $E(T)$ , and the set  $N$  are as shown in Figure 11.41.



**FIGURE 11.41**  $G$ ,  $V(T)$ ,  $E(T)$ , and  $N$  after Steps 1 and 2 execute

The **for** loop in Line 8 checks the following edges:

Edge	$(v_1, v_2)$	$(v_1, v_3)$	$(v_1, v_4)$
Weight of the edge	6	5	2

Clearly, the edge  $(v_1, v_4)$  has the smallest weight. Therefore, the statements in Lines 17 and 18 add vertex  $v_4$  to  $V(T)$  and the edge  $(v_1, v_4)$  to  $E(T)$ . Figure 11.42 shows the resulting graph,  $V(T)$ ,  $E(T)$ , and  $N$ . (The dashed line shows the edge in  $T$ .)

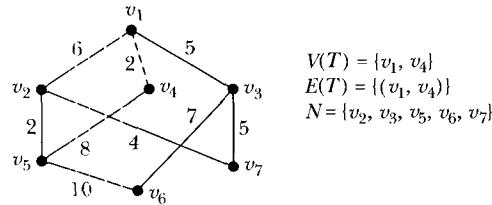


FIGURE 11.42  $G$ ,  $V(T)$ ,  $E(T)$ , and  $N$  after the first iteration of Step 3

Next the `for` loop in Line 8 checks the following edges:

Edge	$(v_1, v_2)$	$(v_1, v_3)$	$(v_4, v_5)$
Weight of the edge	6	5	8

Clearly, the edge  $(v_1, v_3)$  has the smallest weight. Therefore, the statements in Lines 17 and 18 add vertex  $v_3$  to  $V(T)$  and the edge  $(v_1, v_3)$  to  $E(T)$ . Figure 11.43 shows the resulting graph,  $V(T)$ ,  $E(T)$ , and  $N$ .

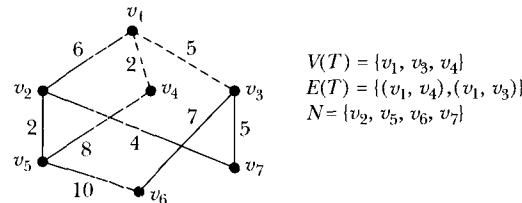


FIGURE 11.43  $G$ ,  $V(T)$ ,  $E(T)$ , and  $N$  after the second iteration of Step 3

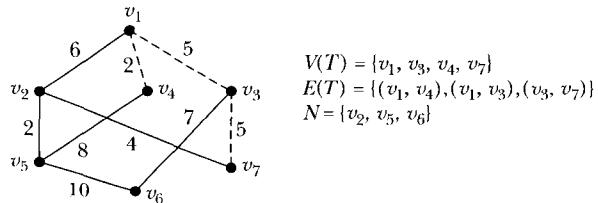
At the next iteration, the `for` loop in Line 8 checks the following edges:

Edge	$(v_1, v_2)$	$(v_3, v_6)$	$(v_3, v_7)$	$(v_4, v_5)$
Weight of the edge	6	7	5	8

Clearly, the edge  $(v_3, v_7)$  has the smallest weight. Therefore, the statements in Lines 17 and 18 add vertex  $v_7$  to  $V(T)$  and the edge  $(v_3, v_7)$  to  $E(T)$ . Figure 11.44 shows the resulting graph,  $V(T)$ ,  $E(T)$ , and  $N$ . (The dashed lines show the edges in  $T$ .)

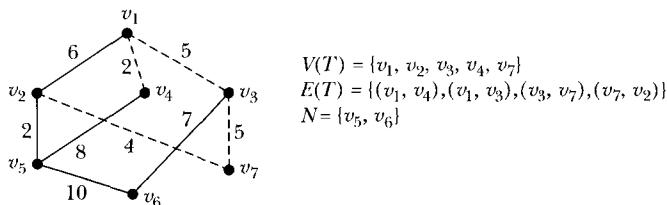
At the next iteration, the `for` loop in Line 8 checks the following edges:

Edge	$(v_1, v_2)$	$(v_3, v_6)$	$(v_4, v_5)$	$(v_7, v_2)$
Weight of the edge	6	7	8	4



**FIGURE 11.44**  $G$ ,  $V(T)$ ,  $E(T)$ , and  $N$  after the third iteration of Step 3

Clearly, the edge  $(v_7, v_2)$  has the smallest weight. Therefore, the statements in Lines 17 and 18 add vertex  $v_2$  to  $V(T)$  and the edge  $(v_7, v_2)$  to  $E(T)$ . Figure 11.45 shows the resulting graph,  $V(T)$ ,  $E(T)$ , and  $N$ . (The dashed lines show the edges in  $T$ .)

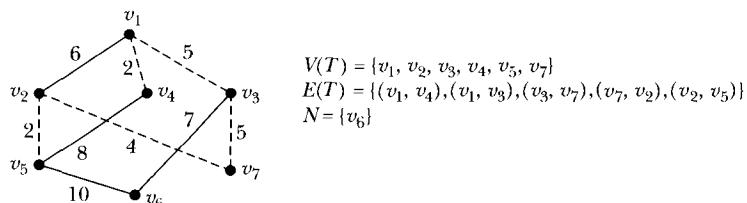


**FIGURE 11.45**  $G$ ,  $V(T)$ ,  $E(T)$ , and  $N$  after the fourth iteration of Step 3

At the next iteration, the `for` loop in Line 8 checks the following edges:

Edge	$(v_2, v_5)$	$(v_3, v_6)$	$(v_4, v_5)$
Weight of the edge	2	7	8

Clearly, the edge  $(v_2, v_5)$  has the smallest weight. Therefore, the statements in Lines 17 and 18 add vertex  $v_5$  to  $V(T)$  and the edge  $(v_2, v_5)$  to  $E(T)$ . Figure 11.46 shows the resulting graph,  $V(T)$ ,  $E(T)$ , and  $N$ . (The dashed lines show the edges in  $T$ .)

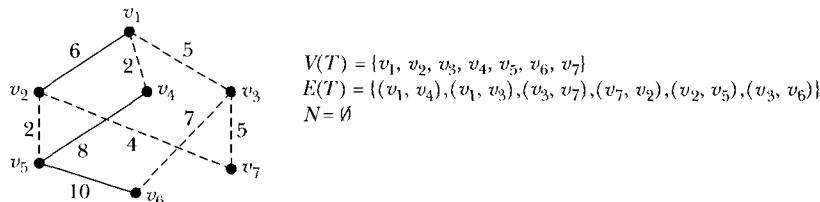


**FIGURE 11.46**  $G$ ,  $V(T)$ ,  $E(T)$ , and  $N$  after the fifth iteration of Step 3

At the next iteration, the `for` loop in Line 8 checks the following edges:

Edge	$(v_3, v_6)$	$(v_5, v_6)$
Weight of the edge	7	10

Clearly, the edge  $(v_3, v_6)$  has the smallest weight. Therefore, the statements in Lines 17 and 18 add vertex  $v_6$  to  $V(T)$  and the edge  $(v_3, v_6)$  to  $E(T)$ . Figure 11.47 shows the resulting graph,  $V(T)$ ,  $E(T)$ , and  $N$ . (The dashed lines show the edges in  $T$ .)

FIGURE 11.47  $G$ ,  $V(T)$ ,  $E(T)$ , and  $N$  after the sixth iteration of Step 3

The dashed lines show a minimal spanning tree of  $G$  of weight 25.

**REMARK 11.3.10** ► The definition of the function `minimalSpanning` contains three nested `for` loops. Therefore, in the worst case, Prim's algorithm given in this section is of the order  $O(n^3)$ .

We leave the proof of the following theorem as an exercise.

**Theorem 11.3.11:** Prim's algorithm correctly finds a minimal spanning tree.

**REMARK 11.3.12** ► As remarked earlier, there are two well-known algorithms for determining a minimal spanning tree of a connected graph. This section describes Prim's algorithm. The second algorithm, due to Kruskal's algorithm, is described in the programming exercises at the end of this section.

## WORKED-OUT EXERCISES

**Exercise 1:** Use the breadth-first search spanning tree algorithm to find a spanning tree of the graph in Figure 11.48 using  $v_1$  as the root node.

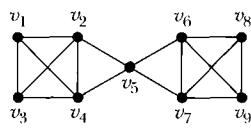
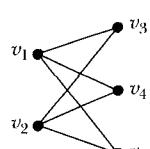
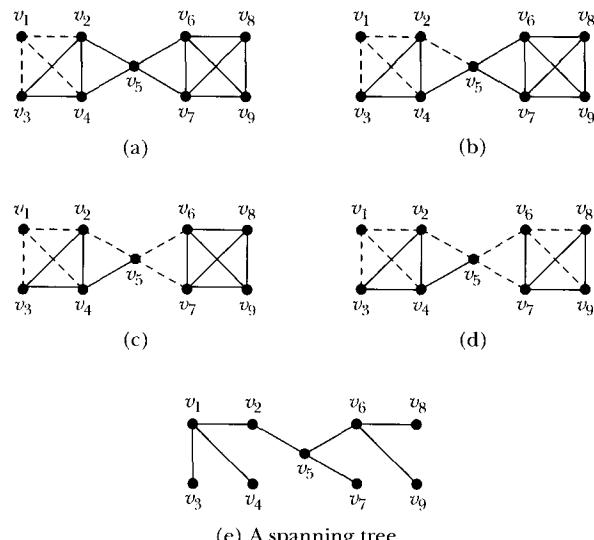


FIGURE 11.48 A tree

**Solution:** The dashed lines in the diagrams in Figure 11.49 show how the vertices and edges are added to the spanning tree.

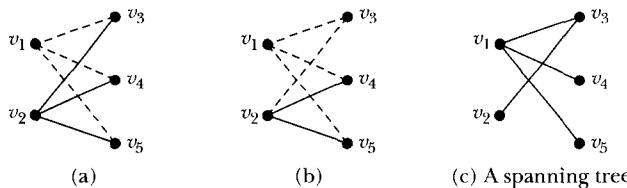
**Exercise 2:** Use the breadth-first search spanning tree algorithm to find a spanning tree of the graph  $K_{2,3}$  in Figure 11.50 using  $v_1$  as the root node.

FIGURE 11.50 Graph  $K_{2,3}$ 

(e) A spanning tree

FIGURE 11.49 Spanning tree

**Solution:** The dashed lines in the diagrams in Figure 11.51 show how the vertices and edges are added to the spanning tree.

FIGURE 11.51 Spanning tree for  $K_{2,3}$ 

**Exercise 3:** Use Prim's algorithm to find a minimal spanning tree of the graph in Figure 11.52 using  $v_1$  as the root node.

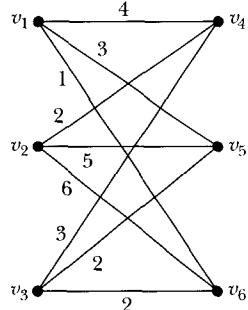


FIGURE 11.52 A graph

**Solution:** The dashed lines in the diagrams in Figure 11.53 show how the vertices and edges are added to a minimal spanning tree.

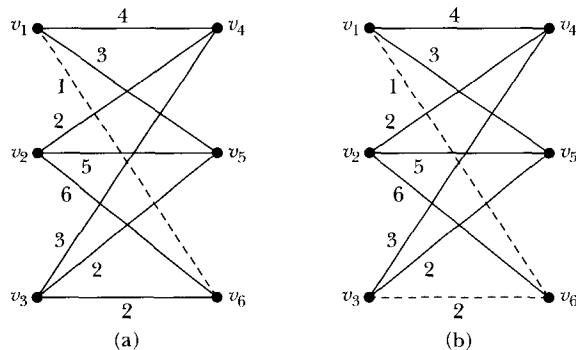


FIGURE 11.53 A minimal spanning tree for the graph in Figure 11.52

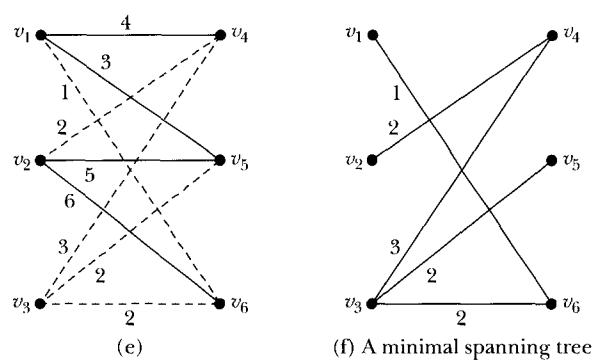
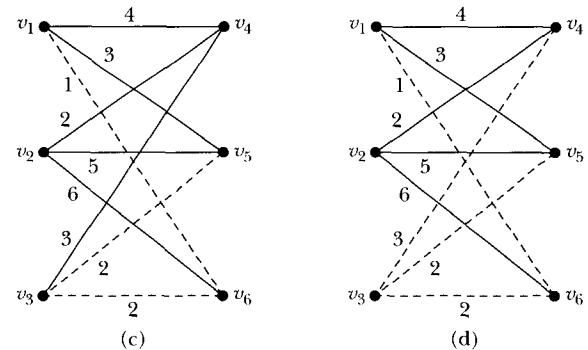


FIGURE 11.53 Continued.

The edges examined for Figure 11.52 are  $(v_1, v_4)$ ,  $(v_1, v_5)$ , and  $(v_1, v_6)$ . Among these edges, the edge  $(v_1, v_6)$  is of the smallest weight, so this edge with the vertex  $v_6$  is added to the minimal spanning tree (see Figure 11.53(a)). In the next iteration, the edges examined are  $(v_1, v_4)$ ,  $(v_1, v_5)$ ,  $(v_2, v_6)$ , and  $(v_3, v_6)$ . Among these edges, the edge  $(v_3, v_6)$  is of the smallest weight, so this edge with the vertex  $v_6$  is added to the minimal spanning tree (see Figure 11.53(b)). The edges and vertices added to the minimal spanning tree in the next three iterations are shown in Figures 11.53(c), (d), and (e). Figure 11.53(f) shows the minimal spanning tree created by Prim's algorithm.

## SECTION REVIEW

### Key Terms

spanning tree  
weighted tree  
weight

minimal spanning tree  
Prim's algorithm

## Some Key Definitions

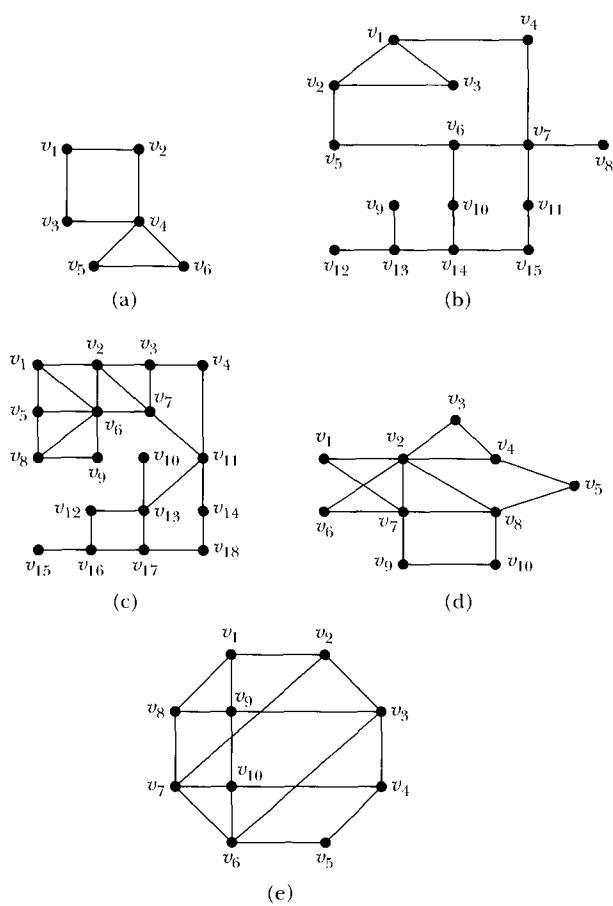
1. A tree  $T$  is called a spanning tree of a graph  $G$  if  $T$  is a subgraph of  $G$  and  $T$  contains all the vertices of  $G$ .
2. Let  $G$  be a weighted graph. A minimal spanning tree of  $G$  is a spanning tree with the minimum weight.

## Some Key Results

1. A graph  $G$  has a spanning tree if and only if  $G$  is connected.
2. Prim's algorithm correctly finds a minimal spanning tree.

## EXERCISES

Use the graphs in Figure 11.54 for Exercises 1–10.



**FIGURE 11.54** Various graphs

2. Use the breadth-first search spanning tree algorithm to find a spanning tree of the graph in Figure 11.54(b) using  $v_1$  as the root node.
3. Use the breadth-first search spanning tree algorithm to find a spanning tree of the graph in Figure 11.54(c) using  $v_1$  as the root node.
4. Use the breadth-first search spanning tree algorithm to find a spanning tree of the graph in Figure 11.54(d) using  $v_1$  as the root node.
5. Use the breadth-first search spanning tree algorithm to find a spanning tree of the graph in Figure 11.54(e) using  $v_1$  as the root node.
6. Use the breadth-first search spanning tree algorithm to find a spanning tree of the graph in Figure 11.54(a) using  $v_4$  as the root node.
7. Use the breadth-first search spanning tree algorithm to find a spanning tree of the graph in Figure 11.54(b) using  $v_7$  as the root node.
8. Use the breadth-first search spanning tree algorithm to find a spanning tree of the graph in Figure 11.54(c) using  $v_{11}$  as the root node.
9. Use the breadth-first search spanning tree algorithm to find a spanning tree of the graph in Figure 11.54(d) using  $v_3$  as the root node.
10. Use the breadth-first search spanning tree algorithm to find a spanning tree of the graph in Figure 11.54(e) using  $v_{10}$  as the root node.
11. Suppose a graph  $G$  is a cycle with  $n$  vertices, where  $n \geq 1$ . Find the number of different spanning trees of  $G$ .
12. Use the breadth-first search spanning tree algorithm to find a spanning tree of  $K_5$ .
13. Use the breadth-first search spanning tree algorithm to find a spanning tree of  $K_{3,4}$ .
14. Use the breadth-first search spanning tree algorithm to find a spanning tree of  $K_{m,n}$ ,  $m \geq 1$ ,  $n \geq 1$ .
15. Give an example of a graph  $G$  that has two spanning trees,  $T_1$  and  $T_2$ , such that  $T_1$  and  $T_2$  have no common edges.

Use the graphs in Figure 11.55 for Exercises 16–20.

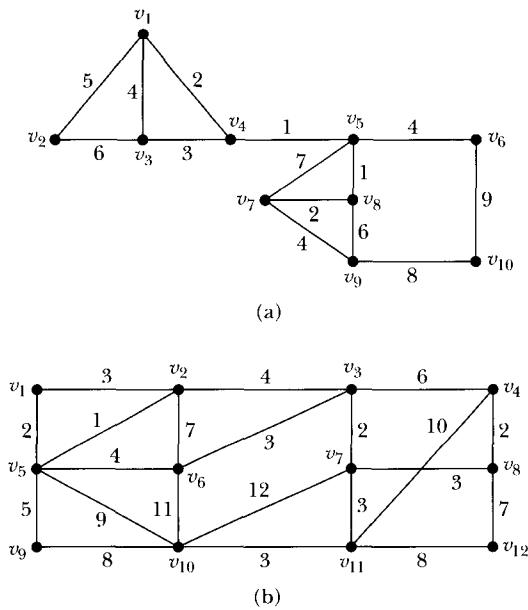


FIGURE 11.55 Graphs

16. Use Prim's algorithm to find a minimal spanning tree of the graph in Figure 11.55(a) with  $v_1$  as the root.
17. Use Prim's algorithm to find a minimal spanning tree of the graph in Figure 11.55(a) with  $v_5$  as the root.
18. Use Prim's algorithm to find a minimal spanning tree of the graph in Figure 11.55(a) with  $v_7$  as the root.
19. Use Prim's algorithm to find a minimal spanning tree of the graph in Figure 11.55(b) with  $v_1$  as the root.
20. Use Prim's algorithm to find a minimal spanning tree of the graph in Figure 11.55(b) with  $v_6$  as the root.
21. Prove Theorem 11.3.11.
22. Modify Prim's algorithm to find a maximal spanning tree, i.e., a spanning tree with a maximal weight.
23. Use the modified Prim's algorithm of Exercise 22 to find a maximal spanning tree of the graph in Figure 11.55(a) with  $v_1$  as the root.
24. Use the modified Prim's algorithm of Exercise 22 to find a maximal spanning tree of the graph in Figure 11.55(b) with  $v_1$  as the root.

## 11.4 NETWORKS

A company in the United States wants to export some of its products in packets to a department store in India. There are different routes, with some constraints, through which the packets can be sent. The digraph in Figure 11.56 shows various routes with vertex  $s$  as the company's main factory and vertex  $t$  as the department store in India. The other vertices in the digraph represent the intermediate places where the goods may be stored. Let  $e$  be an arc in the graph. We refer to the end of  $e$  with the arrow as the head, written  $head(e)$ , and the other end as the tail of  $e$ , written  $tail(e)$ . The number assigned to an arc  $e$  represents the maximum number of packets that can be sent to place  $B$  from place  $A$ , where  $head(e) = B$  and  $tail(e) = A$ . The company would like to know the route through which the maximum number of packets can be sent from the main factory to the department store in India.

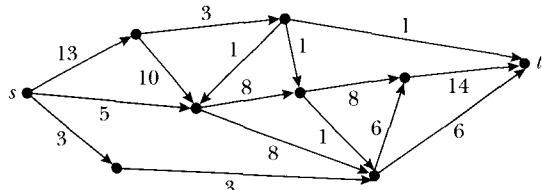


FIGURE 11.56 A graph

Consider a similar problem: An oil company wants to send oil from its oil field to its main refinery. For this purpose the company uses different pipelines. There is no direct, single pipeline from the oil field to the refinery. All the pipelines may have different capacities. (See Figure 11.57.) Here vertex  $s$  denotes the oil field and  $t$  denotes the main refinery. Each arc represents a pipeline from one store to another store. The number assigned to an arc  $e$  represents the maximum capacity of the flow of oil through the pipeline, which may not be the same for all

pipes. The arrow of the arc indicates the direction of the flow of oil through the pipeline packets, which can be sent to place  $B$  from place  $A$  where  $\text{head}(e) = B$  and  $\text{tail}(e) = A$ . The company wants to know the path through which the maximum quantity of oil can be sent from the oil field to the refinery.

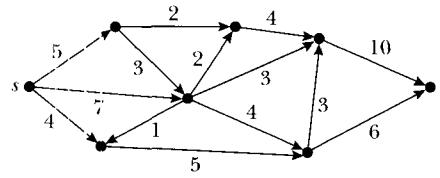


FIGURE 11.57 A graph

To solve these types of problems by graph theory, we introduce the concept of networks.

Let  $G = (V, E)$  be a digraph. Let  $e$  be an arc with end vertices  $v_i$  and  $v_j$ . A simple digraph does not contain loops or parallel edges. Hence, an arc  $e$  with  $\text{tail}(e) = u$  and  $\text{head}(e) = v$  can also be written  $uv$  or  $u - v$ .

---

**DEFINITION 11.4.1** ▶ A **single-source, single-sink network** is a simple digraph  $N = (V, E)$  such that the underlying graph is connected. It has a distinguished vertex  $s$ , called the **source**, and a distinguished vertex  $t$ , called the **target**, or **sink**, with the in-degree of  $s$  equal to 0 and the out-degree of  $t$  equal to 0.

A single-source, single-sink network with source  $s$  and sink  $t$  is also called an  **$s-t$  network**.

The diagrams in Figures 11.56 and 11.57 are  $s-t$  networks.

---

**DEFINITION 11.4.2** ▶ Let  $N = (V, E)$  be an  $s-t$  network such that each arc  $e \in E$  is assigned a nonnegative integer  $C(e)$ , called the **capacity** of  $e$ . Then  $N$  is called a **transport network**, or simply a **network**.

Let  $e$  be an arc with end vertices  $v_i$  and  $v_j$ . Then the capacity  $C(e)$  of  $e$  is also denoted by  $C(v_i v_j)$ ,  $C_{ij}$ , or  $C_{v_i v_j}$ .

Let  $N = (V, E)$  be an  $s-t$  network. If  $v \in V$ ,  $\text{out}(v)$  denotes the set of arcs going out of  $v$  and  $\text{in}(v)$  denotes the set of arcs going into  $v$ . It follows that

$$\text{out}(v) = \{e \in E \mid \text{tail}(e) = v\}$$

and

$$\text{in}(v) = \{e \in E \mid \text{head}(e) = v\}.$$

**EXAMPLE 11.4.3**

A transport network with seven vertices is shown in Figure 11.58.

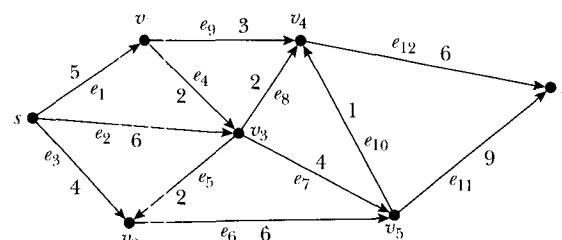


FIGURE 11.58 A transport network

In this transport network,  $\text{in}(v_2) = \{e_3, e_5\}$  and  $\text{out}(v_2) = \{e_6\}$ .

**DEFINITION 11.4.4** ▶ Let  $N = (V, E)$  be a transport network. Let  $X$  and  $Y$  be subsets of  $V$ . Then  $A(X, Y)$  denotes the subset of arcs,

$$A(X, Y) = \{e \in E \mid \text{tail}(e) \in X, \text{head}(e) \in Y\}$$

**EXAMPLE 11.4.5**

Consider the transport network in Example 11.4.3 (see Figure 11.58). In this network,

$$A(\{v_1, v_3\}, \{v_2, v_4\}) = \{e_3, e_5, e_8\}.$$

**DEFINITION 11.4.6** ▶ Let  $N = (V, E)$  be a transport network. A **flow**  $F$  in  $N$  is a function  $F : E \rightarrow \mathbb{N} \cup \{0\}$  that assigns a nonnegative integer  $F(e)$  to each arc  $e$  of  $N$  such that

- (i) **(capacity constraint)**  $F(e) \leq C(e)$ , for every arc  $e$  of  $N$ ,
- (ii) **(flow conservation)**

$$\sum_{e \in \text{in}(v)} F(e) = \sum_{e \in \text{out}(v)} F(e), \quad (11.4)$$

for every vertex  $v$  of  $N$  other than source  $s$  and sink  $t$ .

**Notation 11.4.7:** In the diagram of a transport network with a flow, each edge is labeled by a pair of integers  $a, b$ , where the first integer,  $a$ , denotes the capacity of the arc and the second integer,  $b$ , denotes the amount of flow through the arc.

Let  $N = (V, E)$  be a transport network with flow  $F$ . Let  $e$  be an arc with end vertices  $v_i$  and  $v_j$ . Then we denote  $F(e)$  by  $F(v_i v_j)$ ,  $F_{ij}$ , or  $F_{v_i v_j}$  and call it the **flow in edge**  $v_i v_j$ . If there is no arc from  $v_i$  to  $v_j$ , then we assume that  $F_{ij} = 0$ .

**EXAMPLE 11.4.8**

The network in Figure 11.59 is a transport network with a flow.

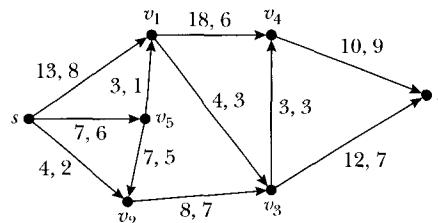


FIGURE 11.59 A transport network with a flow

Note that the function  $F$  is such that  $F(sv_1) = 8 < 13 = C(sv_1)$ ,  $F(v_1 v_4) = 6 < 18 = C(v_1 v_4)$ , and so on.

Let  $N = (V, E)$  be a transport network with flow  $F$ . Suppose that  $V = \{v_0, v_1, v_2, \dots, v_n\}$ , where  $v_0 = s$ ,  $v_n = t$ . Then for any vertex  $v_j$ ,

$$\sum_{v_i \in V} F_{ij}$$

denotes the sum  $F_{0j} + F_{1j} + F_{2j} + \dots + F_{nj}$ . The value of this sum is called the **flow into vertex  $v_j$** . Similarly,

$$\sum_{v_i \in V} F_{ji},$$

which denotes the sum  $F_{j0} + F_{j1} + F_{j2} + \dots + F_{jn}$ , is called the **flow out of  $v_j$** .

**EXAMPLE 11.4.9**

We consider the network in Example 11.4.8. In this network, we see that the flow into vertex  $v_2$  is 7 and the flow out of  $v_2$  is 7. Here  $s = v_0$  and  $t = v_6$ . Note that

$$\sum_{v_i \in V} F_{i2} = F_{02} + F_{12} + F_{22} + F_{32} + F_{42} + F_{52} + F_{62} = 2 + 0 + 0 + 0 + 0 + 5 + 0 = 7$$

and

$$\sum_{v_i \in V} F_{2i} = F_{20} + F_{21} + F_{22} + F_{23} + F_{24} + F_{25} + F_{26} = 0 + 0 + 0 + 7 + 0 + 0 + 0 = 7.$$

The property of flow given by (11.4) is called the **conservation of flow**.

**Theorem 11.4.10:** Let  $N = (V, E)$  be a transport network with a flow  $F$ .

If  $V = \{v_0, v_1, v_2, \dots, v_n\}$  with  $s = v_0$  and  $t = v_n$ , then the flow out of the source  $s$  equals the flow into the sink  $t$ , i.e.,

$$\sum_{v_i \in V} F_{0i} = \sum_{v_i \in V} F_{in}.$$

**Proof:** We have

$$\begin{aligned} \sum_{v_j \in V} \sum_{v_i \in V} F_{ij} &= \sum_{v_j \in V} F_{0j} + \sum_{v_j \in V} F_{1j} + \sum_{v_j \in V} F_{2j} + \dots + \sum_{v_j \in V} F_{nj} \\ &= (F_{00} + F_{01} + F_{02} + \dots + F_{0n}) + (F_{10} + F_{11} + F_{12} + \dots + F_{1n}) \\ &\quad + \dots + (F_{n0} + F_{n1} + F_{n2} + \dots + F_{nn}) \\ &= (F_{00} + F_{10} + F_{20} + \dots + F_{n0}) + (F_{01} + F_{11} + F_{21} + \dots + F_{n1}) \\ &\quad + \dots + (F_{0n} + F_{1n} + F_{2n} + \dots + F_{nn}) \\ &= \sum_{v_j \in V} F_{j0} + \sum_{v_i \in V} F_{ji} + \sum_{v_i \in V} F_{ji} + \dots + \sum_{v_j \in V} F_{jn} \\ &= \sum_{v_j \in V} \sum_{v_i \in V} F_{ji}. \end{aligned}$$

Thus,

$$\begin{aligned} 0 &= \sum_{v_j \in V} \sum_{v_i \in V} F_{ij} - \sum_{v_j \in V} \sum_{v_i \in V} F_{ji} \\ &= \left( \sum_{v_i \in V} F_{in} - \sum_{v_i \in V} F_{ni} \right) + \left( \sum_{v_i \in V} F_{i0} - \sum_{v_i \in V} F_{0i} \right) \end{aligned}$$

$$\begin{aligned}
& + \sum_{v_i \in V, j \neq 0, n} \left( \sum_{v_i \in V} F_{ij} - \sum_{v_i \in V} F_{ji} \right) \\
& = \sum_{v_i \in V} F_{in} - \sum_{v_i \in V} F_{0i}
\end{aligned}$$

because  $F_{ni} = 0 = F_{i0} \forall i \in V$ , and by Definition 11.4.6(ii),

$$\sum_{v_i \in V} F_{ij} - \sum_{v_i \in V} F_{ji} = 0 \quad \text{if } j \in V \setminus \{s, t\}. \quad \blacksquare$$

From Theorem 11.4.10, we find that the total flow into the sink  $t$  equals the total flow out of the source  $s$ .

---

**DEFINITION 11.4.11** ▶ Let  $N = (V, E)$  be a transport network with a flow  $F$ . The value

$$d = \sum_{e \in \text{out}(s)} F(e) = \sum_{e \in \text{In}(t)} F(e)$$

is called the **value of the flow  $F$** .

**EXAMPLE 11.4.12**

Consider the network in Example 11.4.8. In this network, the value of the flow is 16.

From Definition 11.4.11, for a transport network  $N = (V, E)$  with a flow  $F$  and  $V = \{v_0, v_1, \dots, v_n\}$  with  $s = v_0$  and  $t = v_n$ ,

$$d = \sum_{v_i \in V} F_{0i} = \sum_{v_i \in V} F_{in}.$$

---

**DEFINITION 11.4.13** ▶ Let  $N = (V, E)$  be a transport network. Suppose that  $\{V_s, V_t\}$  is a partition of  $V$  such that  $s \in V_s$  and  $t \in V_t$ . Then the set  $A(V_s, V_t)$ , the set of all arcs  $e \in E$  with  $\text{tail}(e) \in V_s$ ,  $\text{head}(e) \in V_t$ , is called an  **$s$ - $t$  cut of the network  $N$** . That is,

$$A(V_s, V_t) = \{e \in E \mid \text{tail}(e) \in V_s, \text{head}(e) \in V_t\}$$

is an  **$s$ - $t$  cut of  $N$** .

The following example illustrates Definition 11.4.13.

**EXAMPLE 11.4.14**

Consider the transport network  $N = (V, E)$  in Example 11.4.3, see Figure 11.58. Let  $V_s = \{s, v_1, v_2, v_3\}$  and  $V_t = \{t, v_4, v_5\}$ . Then

$$\begin{aligned}
A(V_s, V_t) &= \{e_1, e_2, e_3, e_4, e_5\}, \\
A(V_s, V_t) &= \{e_6, e_7, e_8, e_9\}, \\
A(V_t, V_s) &= \emptyset.
\end{aligned}$$

From the definition of  $s$ - $t$  cut of the network  $N$ , it follows that

$$A(\{s\}, V - \{s\}) = \text{out}(s) = \{e_1, e_2, e_3\}$$

and

$$A(V - \{t\}, \{t\}) = \text{in}(t) = \{e_{11}, e_{12}\}.$$

**DEFINITION 11.4.15** ▶ Let  $N = (V, E)$  be a transport network with a flow  $F$ . If  $X$  and  $Y$  are two nonempty subsets of  $V$ , then we define

- (i)  $C(X, Y) = \sum_{e \in A(X, Y)} C(e),$
- (ii)  $F(X, Y) = \sum_{e \in A(X, Y)} F(e).$

Consider the  $s$ - $t$  cut  $A(V_s, V_t) = \{e_6, e_7, e_8, e_9\}$  of the network in Example 11.4.3, see Figure 11.58. We find that the capacity of this cut is

$$C(e_6) + C(e_7) + C(e_8) + C(e_9) = 6 + 4 + 2 + 3 = 15.$$

**DEFINITION 11.4.16** ▶ Let  $N = (V, E)$  be a transport network with a flow  $F$ . Then  $C(V_s, V_t)$  is called the **capacity of the  $s$ - $t$  cut  $A(V_s, V_t)$** .

**Theorem 11.4.17:** Let  $N = (V, E)$  be a transport network with a flow  $F$ .

Let  $d$  be the value of the flow  $F$ . If  $A(V_s, V_t)$  is an  $s$ - $t$  cut of  $N$ , then

- (i)  $d = F(V_s, V_t) - F(V_t, V_s),$
- (ii)  $d \leq C(V_s, V_t).$

### Proof:

- (i)  $F(\{s\}, V) = \sum_{e \in A(\{s\}, V)} F(e) = \sum_{e \in \text{out}(s)} F(e) = d$  and  $F(V, \{s\}) = \sum_{e \in A(V, \{s\})} F(e) = 0$ . Let  $v \in V - \{s, t\}$ . Then

$$\begin{aligned} F(\{v\}, V) &= \sum_{e \in A(\{v\}, V)} F(e) \\ &= \sum_{e \in \text{out}(v)} F(e) \\ &= \sum_{e \in \text{in}(v)} F(e) \\ &= \sum_{e \in A(V, \{v\})} F(e) \\ &= F(V, \{v\}). \end{aligned}$$

Thus, for any vertex  $v \in V - \{s, t\}$ ,  $F(\{v\}, V) - F(V, \{v\}) = 0$ . Hence,

$$\sum_{v \in V_s} (F(\{v\}, V) - F(V, \{v\})) = F(\{s\}, V) - F(V, \{s\}) = d.$$

This implies that

$$F(V_s, V) - F(V, V_s) = d.$$

Now

$$F(V_s, V) = F(V_s, V_s \cup V_t) = F(V_s, V_s) + F(V_s, V_t)$$

and

$$F(V, V_s) = F(V_s \cup V_t, V_s) = F(V_s, V_s) + F(V_t, V_s).$$

Hence,  $F(V_s, V) - F(V, V_s) = F(V_s, V_t) - F(V_t, V_s)$ . This implies that  $d = F(V_s, V_t) - F(V_t, V_s)$ .

(ii) By part (i),  $d = F(V_s, V_t) - F(V_t, V_s)$ . This implies

$$\begin{aligned} d &\leq F(V_s, V_t) \quad \text{because } F(e) \geq 0 \\ &= \sum_{e \in A(V_s, V_t)} F(e) \\ &\leq \sum_{e \in A(V_s, V_t)} C(e) \quad \text{by capacity constraint} \\ &= C(V_s, V_t). \blacksquare \end{aligned}$$

From Theorem 11.4.17(ii), it follows that in a transport network with a flow  $F$ , the value of the flow  $F$  will not exceed the capacity of any cut  $A(V_s, V_t)$ .

---

**DEFINITION 11.4.18** ▶ Let  $N = (V, E)$  be a transport network with a flow  $F$ . Then a **minimal cut** in  $N$  is an  $s$ - $t$  cut  $A(V_s, V_t)$  with the minimum capacity.

From Theorem 11.4.17(ii), it follows that in a transport network with a flow  $F$ ,  $d \leq \min\{C(V_s, V_t) | A(V_s, V_t) \text{ is an } s\text{-}t \text{ cut}\}$ . Thus, we find that in a transport network the value of a flow is bounded above. We are interested in finding flows having values equal to this upper bound.

---

**DEFINITION 11.4.19** ▶ Let  $N = (V, E)$  be a transport network. A flow  $F$  in  $N$  is called a **maximal flow** if for every flow  $F'$  in  $N$ , the value of  $F'$  is less than or equal to the value of the flow  $F$ .

**Theorem 11.4.20:** Let  $N = (V, E)$  be a transport network with flow  $F$  such that the value of the flow  $F$  is

$$\min\{C(V_s, V_t) | A(V_s, V_t) \text{ is an } s\text{-}t \text{ cut in } N\}.$$

Then  $F$  is a maximal flow.

**Proof:** Let  $F_1$  be a flow in  $N$ . Then from Theorem 11.4.17(ii), the value of  $F_1 \leq \min\{C(V_s, V_t) | A(V_s, V_t) \text{ is an } s\text{-}t \text{ cut}\}$ . This implies that the value of the flow  $F_1 \leq$  the value of the flow  $F$ . Hence,  $F$  is a maximal flow in  $N$ . ■

Let  $N = (V, E)$  be a transport network and let  $A(V_s, V_t)$  be an  $s$ - $t$  cut in  $N$ . Suppose that there exists a flow  $F$  in  $N$  such that

$$F(e) = \begin{cases} C(e), & \text{if } e \in A(V_s, V_t), \\ 0, & \text{if } e \in A(V_t, V_s). \end{cases} \quad (11.5)$$

Then  $C(V_s, V_t) = \sum_{e \in A(V_s, V_t)} C(e)$  and the value of  $F$  is

$$\begin{aligned} F(V_s, V_t) - F(V_t, V_s) &= \sum_{e \in A(V_s, V_t)} F(e) - \sum_{e \in A(V_t, V_s)} F(e) \\ &= \sum_{e \in A(V_s, V_t)} C(e) \\ &= C(V_s, V_t) \\ &= \min\{C(V_s, V_t) | A(V_s, V_t) \text{ is any } s\text{-}t \text{ cut}\} \end{aligned}$$

This implies that the flow defined by (11.5) is a maximal flow. Also we find that the  $s$ - $t$  cut  $A(V_s, V_t)$  is a minimal cut for this flow.

Let  $H$  be a directed graph. A graph  $G$  is said to be an **underlying graph** of  $H$  if  $G$  is obtained from  $H$  by changing the arcs to the edges, i.e., ignoring the direction of the arcs. In other words, the vertex set of  $G$  is the same as the vertex set of  $H$  and edges of  $G$  are the arcs of  $H$ .

**DEFINITION 11.4.21** ▶ Let  $N = (V, E)$  be a transport network. A **quasipath** in  $N$  is an alternating sequence  $(v_0, e_1, v_1, e_2, v_2, \dots, v_{k-1}, e_{k-1}, v_k)$  of vertices and arcs such that  $(v_0, e_1, v_1, e_2, v_2, \dots, v_{k-1}, e_{k-1}, v_k)$  is a path in the underlying graph of  $N$ .

**DEFINITION 11.4.22** ▶ Let  $Q = (v_0, e_1, v_1, e_2, v_2, \dots, v_{l-1}, e_{l-1}, v_l)$  be a quasipath in a transport network  $N$  with a flow  $F$ . An arc  $e_i$  of  $Q$  is called a **forward arc** if it is directed from  $v_{i-1}$  to  $v_i$ , and an arc  $e_i$  of  $Q$  is called a **backward arc** if it is directed from  $v_i$  to  $v_{i-1}$ .

Figure 11.60 shows forward and backward arcs.

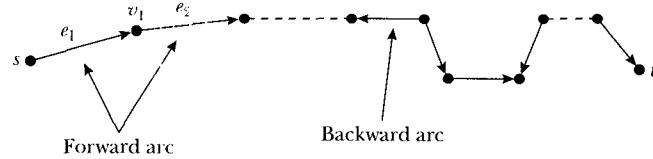


FIGURE 11.60 Forward and backward arcs

**DEFINITION 11.4.23** ▶ Let  $N$  be a transport network with a flow  $F$  and  $Q$  be a quasipath in  $N$ .

- (i) For each arc  $e$  in  $Q$ , we associate a nonnegative integer  $i(e)$ , defined by

$$i(e) = \begin{cases} C(e) - F(e), & \text{if } e \text{ is a forward arc,} \\ C(e), & \text{if } e \text{ is a backward arc.} \end{cases}$$

The number  $i(e)$  is called the **slack** on the arc  $e$ .

- (ii) To  $Q$  we associate a nonnegative integer  $i(Q)$ , called **slack** of  $Q$ , defined by

$$i(Q) = \min\{i(e) \mid e \text{ is an arc in } Q\}.$$

**DEFINITION 11.4.24** ▶ Let  $N$  be a network with a flow  $F$ . A quasipath  $Q$  in  $N$  is called  **$F$ -saturated** if  $i(Q) = 0$  and  **$F$ -unsaturated** if  $i(Q) > 0$ .

**EXAMPLE 11.4.25**

Consider the quasipaths  $Q$  and  $Q_1$  in Figure 11.61.

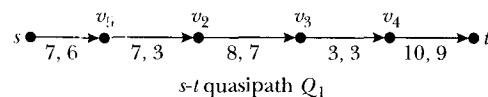
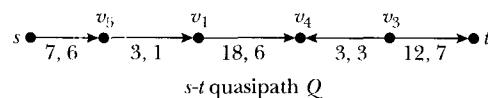


FIGURE 11.61 Quasipaths

Let us determine  $i(Q)$ . Now  $i(sv_5) = 7 - 6 = 1$ ,  $i(v_5v_1) = 3 - 1 = 2$ ,  $i(v_1v_4) = 18 - 6 = 12$ ,  $i(v_4v_3) = 3$  (because  $v_4v_3$  is a backward arc), and  $i(v_3t) = 5$ . Thus,  $i(Q) =$

$\min\{1, 2, 12, 3, 5\} = 1 > 0$ . Hence, the quasipath  $Q$  is  $F$ -unsaturated. We now determine  $i(Q_1)$ . Here  $i(sv_5) = 7 - 6 = 1$ ,  $i(v_5v_2) = 7 - 3 = 4$ ,  $i(v_2v_3) = 8 - 7 = 1$ ,  $i(v_3v_4) = 3 - 3 = 0$ ,  $i(v_4t) = 10 - 9 = 1$ . Thus,  $i(Q_1) = \min\{1, 4, 1, 0, 1\} = 0$ . Hence, the quasipath  $Q_1$  is  $F$ -saturated.

**REMARK 11.4.26** ► An  $F$ -unsaturated quasipath is also called a **flow-augmenting path**.

*(A flow-augmenting path is a quasipath that can be used to increase the total flow in a transport network.)*

**Lemma 11.4.27:** Let  $N = (V, E)$  be a transport network with a flow  $F$  and

$$Q = (s = v_0, e_1, v_1, e_2, v_2, \dots, v_{m-1}, e_{m-1}, v_m = t)$$

an  $F$ -unsaturated quasipath from  $s$  to  $t$  in  $N$  with  $i(Q) = \lambda$ . Let  $d$  be the value of the flow  $F$ . Define a function  $F^*$  from  $E$  to the set of nonnegative integers by

$$F^*(e) = \begin{cases} F(e), & \text{if } e \text{ is not in } Q, \\ F(e) + \lambda, & \text{if } e \text{ is a forward arc in } Q, \\ F(e) - \lambda, & \text{if } e \text{ is a backward arc in } Q. \end{cases}$$

Then  $F^*$  is a flow whose value is  $d + \lambda$ .

**Proof:** It follows from the definitions of  $i(Q)$  and  $F^*$  that  $0 \leq F^*(e) \leq C(e)$ . We now verify the flow conservation:  $\sum_{e \in \text{in}(v)} F^*(e) = \sum_{e \in \text{out}(v)} F^*(e)$ , for every vertex  $v$  of  $N$  other than source  $s$  and sink  $t$ . Consider a vertex  $v \in V - \{s, t\}$ . If  $v$  is not a vertex of  $Q$ , then  $F^*(e) = F(e)$ , for all  $e \in \text{in}(v)$  and also for all  $e \in \text{out}(v)$ . Hence, in this case,  $\sum_{e \in \text{in}(v)} F^*(e) = \sum_{e \in \text{out}(v)} F^*(e)$ .

Suppose now that  $v$  is a vertex on  $Q$ . We may have the following four possibilities.

Case 1:  $s = v_0, e_1, v_1, e_2, v_2, \dots, \rightarrow v \rightarrow, \dots, v_{m-1}, e_{m-1}, v_m = t$

Case 2:  $s = v, e_1, v_1, e_2, v_2, \dots, \rightarrow v \leftarrow, \dots, v_{m-1}, e_{m-1}, v_m = t$

Case 3:  $s = v_0, e_1, v_1, e_2, v_2, \dots, \leftarrow v \rightarrow, \dots, v_{m-1}, e_{m-1}, v_m = t$

Case 4:  $s = v_0, e_1, v_1, e_2, v_2, \dots, \leftarrow v \leftarrow, \dots, v_{m-1}, e_{m-1}, v_m = t$

In case (1) if the head of  $e_j$  is  $v$  and the tail of  $e_{j+1}$  is  $v$ , then  $F^*(e_j) = F(e_j) + \lambda$  and  $F^*(e_{j+1}) = F(e_{j+1}) + \lambda$ . Hence, in  $\sum_{e \in \text{in}(v)} F^*(e)$ ,

$$F^*(e_j) = F(e_j) + \lambda,$$

and in  $\sum_{e \in \text{out}(v)} F^*(e)$ ,

$$F^*(e_{j+1}) = F(e_{j+1}) + \lambda.$$

Hence, the net flow into  $v$  is the same as the total flow out from  $v$ . We can show this is also true in the remaining three cases. Hence, flow of conservation holds.

We now show that the value of flow has been increased by  $\lambda$ . For this we compute  $\sum_{e \in \text{out}(s)} F^*(e)$ . From the quasipath

$$Q = (s = v_0, e_1, v_1, e_2, v_2, \dots, v_{m-1}, e_{m-1}, v_m = t),$$

we find that  $e_1$  is the only arc from  $\text{out}(s)$  that is also in  $Q$ , and clearly  $e_1$  is a forward

arc. Hence,

$$\begin{aligned}
 \sum_{e \in \text{out}(s)} F^*(e) &= F^*(e_1) + \sum_{e \in \text{out}(s) - \{e_1\}} F^*(e) \\
 &= F(e) + \lambda + \sum_{e \in \text{out}(s) - \{e_1\}} F(e) \\
 &= \sum_{e \in \text{out}(s)} F(e) + \lambda \\
 &= \text{the value of } F + \lambda \\
 &= d + \lambda. \blacksquare
 \end{aligned}$$

From Lemma 11.4.27, we find that if there exists an  $F$ -unsaturated quasipath  $Q$  from  $s$  to  $t$  in  $N$  with  $i(Q) = \lambda$ , then we can increase the flow by  $\lambda$ . If there does not exist an  $F$ -unsaturated quasipath  $Q$  from  $s$  to  $t$  in  $N$ , then we will show that the existing flow is a maximal flow.

#### EXAMPLE 11.4.28

Consider the transport network in Figure 11.62(a). Let  $F$  be the flow in Figure 11.62(a).

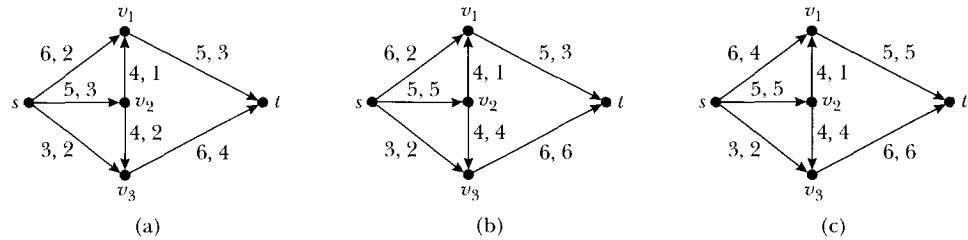


FIGURE 11.62 Transport networks with flows

Consider the quasipath  $Q : s \rightarrow v_2 \rightarrow v_3 \rightarrow t$  from  $s$  to  $t$ . For this quasipath  $i(Q) = \min\{2, 2, 2\} = 2$ . Hence, we can increase the flow  $F$  by 2 along this path. Let  $F_1$  be the new flow. Figure 11.62(b) shows the flow  $F_1$ .

In the network in Figure 11.62(b), the quasipath  $Q : s \rightarrow v_2 \rightarrow v_3 \rightarrow t$  is saturated, because  $i(Q) = 0$ . However, consider the quasipath  $Q_1 : s \rightarrow v_1 \rightarrow t$ . Now  $i(Q_1) = \min\{4, 2\} = 2$ . Thus,  $Q_1$  is unsaturated and we can increase the flow  $F_1$  along this path by 2. Let  $F_2$  denote this new flow (see Figure 11.62(c)).

Notice that in the network in Figure 11.62(c), there are no unsaturated paths from  $s$  to  $t$ . We can show that  $F_2$  is a maximal flow for the transport network of Figure 11.62(a).

**Theorem 11.4.29:** Let  $N = (V, E)$  be a transport network with a flow  $F$ .

Then  $F$  is a maximal flow if and only if there does not exist an  $F$ -unsaturated quasipath  $Q$  from  $s$  to  $t$  in  $N$ .

**Proof:** Suppose there exists an  $F$ -unsaturated quasipath  $Q$  from  $s$  to  $t$  in  $N$ . Then the given flow cannot be maximal, because from Lemma 11.4.27 we find that this flow can be increased. Hence, if  $F$  is a maximal flow, then there does not exist an  $F$ -unsaturated quasipath  $Q$  from  $s$  to  $t$  in  $N$ .

Conversely, assume that there does not exist an  $F$ -unsaturated quasipath  $Q$  from  $s$  to  $t$  in  $N$ . Let  $X$  be the set of vertices  $u$  in  $N$  such that either  $u = s$  or there

exists an  $F$ -unsaturated quasipath  $P = (s = v_0, f_1, u_1, f_2, u_2, \dots, u_{k-1}, f_{k-1}, u_k = u)$  from  $s$  to  $u$  in  $N$ . Because there is no  $F$ -unsaturated quasipath  $Q$  from  $s$  to  $t$  in  $N$  it follows that  $t \notin X$ . Let  $V_s = X$  and  $V_t = V - X$ . Then we obtain an  $s$ - $t$  cut  $A(V_s, V_t)$ . Let  $w$  be a vertex in  $N$  such that  $w \notin X$ . Suppose there exists an arc  $e$  from  $u$  to  $w$  and let  $F(e) < C(e)$ . Then the quasipath

$$P_1 : (s = v_0, f_1, u_1, f_2, u_2, \dots, u_{k-1}, f_{k-1}, u_k = u, e, w)$$

from  $s$  to  $w$  in  $N$  is an  $F$ -unsaturated quasipath, which implies that  $w \in X$ , a contradiction. Again, if there exists an arc  $e$  from  $w$  to  $u$  with  $F(e) > 0$ , then the quasipath

$$P_1 : (s = v_0, f_1, u_1, f_2, u_2, \dots, u_{k-1}, f_{k-1}, u_k = u, e, w)$$

from  $s$  to  $w$  in  $N$  is an  $F$ -unsaturated quasipath, which implies that  $w \in X$ , a contradiction. Thus, for any arc  $e = uw$  with  $u \in X, w \in V - X$  we must have  $F(e) = C(e)$ , and for any arc  $e = wu$  with  $u \in X, w \in V - X$  we must have  $F(e) = 0$ . Then the value of

$$\begin{aligned} F &= F(V_s, V_t) - F(V_t, V_s) \\ &= \sum_{e \in A(V_s, V_t)} F(e) - \sum_{e \in A(V_t, V_s)} F(e) \\ &= \sum_{e \in A(V_s, V_t)} C(e) \\ &= C(V_s, V_t) \\ &= \min\{C(V_s, V_t) \mid A(V_s, V_t) \text{ is any } s\text{-}t \text{ cut}\}. \end{aligned}$$

Hence,  $F$  is a maximal flow by Theorem 11.4.20. ■

**REMARK 11.4.30** ▶ From the above theorem we find that corresponding to the maximal flow  $F$  we obtain an  $s$ - $t$  cut  $A(V_s, V_t)$  such that  $C(V_s, V_t) = \min\{C(V_s, V_t) \mid A(V_s, V_t) \text{ is an } s\text{-}t \text{ cut}\}$ . Hence,  $A(V_s, V_t)$  is a minimal cut.

**Theorem 11.4.31: The Max-Flow, Min-Cut Theorem.** Let  $N = (V, E)$  be a transport network with a flow. Then there exists a maximal flow in  $N$ ; i.e., there exists a flow in  $N$  with value  $\min\{C(V_s, V_t) \mid A(V_s, V_t) \text{ is an } s\text{-}t \text{ cut}\}$ .

**Proof:** Let  $F$  be a flow in  $N$ . We know from Theorem 11.4.17 that  $d \leq \min\{C(V_s, V_t) \mid A(V_s, V_t) \text{ is an } s\text{-}t \text{ cut}\}$ , where  $d =$  the value of the flow  $F$ . If  $F$  is not a maximal flow, then by Theorem 11.4.29, we find that there exists an  $F$ -unsaturated quasipath  $Q$  from  $s$  to  $t$ . Then from Lemma 11.4.27, it follows that we can form a new flow,  $F_1$ , in  $N$  such that the value  $d_1$  of  $F_1$  is  $d_1 = d + i(Q)$ . If this flow  $F_1$  is not maximal, then there exists an  $F_1$ -unsaturated quasipath  $Q_1$  from  $s$  to  $t$ . Then we can form a new flow,  $F_2$ , in  $N$  such that the value  $d_2$  of  $F_2$  is  $d_2 = d_1 + i(Q_1)$ . Now  $d < d_1 < d_2 \leq \min\{C(V_s, V_t) \mid A(V_s, V_t) \text{ is an } s\text{-}t \text{ cut}\}$  is a strictly increasing sequence of positive integers  $d, d_1, d_2$  bounded above. Hence, repeating this process, we eventually get a flow  $F_k$  such that there does not exist an  $F_k$ -unsaturated quasipath from  $s$  to  $t$ . Then from Theorem 11.4.29, it follows that  $F_k$  is a maximal flow in  $N$ . ■

Next we give an algorithm for finding a maximal flow in a transport network. This algorithm is based on the algorithm designed by Ford and Fulkerson and uses

a modification suggested by Edmond and Karp. The algorithm uses a labeling technique to label the vertices. Let  $G = (V, E)$  be a transport network. Let  $u$  and  $v$  be vertices in  $G$  such that there is an arc  $e$  from  $u$  to  $v$ , in  $G$ , i.e.,  $\text{tail}(e) = u$  and  $\text{head}(e) = v$ . We call  $u$  a **parent**, or an **immediate predecessor**, of  $v$  and write  $p(v) = u$ . By assigning a label to a vertex  $v$ , we mean assigning the value  $p(v)$  and a nonnegative integer, denoted by  $\text{value}(v)$ . For example, if  $\text{label}(v) = (u, 5)$ , then  $p(v) = u$  and  $\text{value}(v) = 5$ . If  $\text{label}(v) = (\text{null}, 0)$ , then no parent is assigned to  $v$  and  $\text{value}(v) = 0$ . Now the source,  $s$ , has no parent, so to  $s$  we assign the label,  $\text{label}(s) = (s, \infty)$ .

**ALGORITHM 11.7:** Finding a maximal flow in a transport network: The flow-augmentation algorithm.

*Input:*  $G$ —network;  $\{v_0, v_1, \dots, v_n\}$  is the vertex set of  $G$ , with source  $s = v_0$  and sink  $t = v_n$ .  
 $C$ —capacity of each edge  
 $n$ —the number of vertices in  $G$

*Output:* A maximal flow  $F$

```

1. procedure maxFlow( $G, C, n$ )
2. begin
3.   for each edge  $e$  do
4.      $F(e) = 0$ ;
5.    $s := v_0$ ;
6.    $t := v_n$ ;
7.   while MaxFlowNotFound do
8.     begin
9.       for each vertex  $v$  do //remove the label of all vertices
10.       $\text{label}(v) := (\text{null}, 0)$ ;
11.       $\text{label}(s) := (s, \infty)$ ; //label source
12.       $U := \{s\}$ ;
13.      while  $t$  is not labeled do
14.        begin
15.          if  $U$  is empty then //flow is maximal
16.            return  $F$ ;
17.           $v :=$  next vertex in  $U$ ;
18.           $U := U - \{v\}$ ;
19.          for each edge  $(v, w)$  such that  $p(w)$  is null do
20.            if  $F((vw)) < C((vw))$  then
21.              begin
22.                 $p(w) := v$ ;
23.                 $\text{val}(w) := \min\{\text{value}(v), C(vw) - F(vw)\}$ ;
24.                 $U := U \cup \{w\}$ ;
25.              end
26.              for each edge  $(w, v)$  such that  $p(w)$  is null do

```

```

27.      if  $F(wv) > 0$  then
28.          begin
29.               $p(w) := v;$ 
30.               $\text{val}(w) := \min\{\text{value}(v), F(wv)\};$ 
31.               $U := U \cup \{w\};$ 
32.          end
33.      end
34.          //Augment the flow on a path P from s to t
35.       $v := t;$ 
36.       $\lambda := \text{value}(t);$ 
37.      while  $v \neq s$  do
38.          begin
39.               $u := p(v);$ 
40.              //either  $(u,v)$  is an arc or  $(v,u)$  is an arc
41.              if  $(u,v)$  is an arc then
42.                   $F(uv) := F(uv) + \lambda; // (u,v) is a forward arc$ 
43.              else
44.                   $F(vu) := F(vu) - \lambda; // (u,v) is a backward arc$ 
45.               $v := u;$ 
46.          end
47.      end
48. end

```

**REMARK 11.4.32** ▶ In Algorithm 11.7, the set  $U$  works like a queue.

### EXAMPLE 11.4.33

Consider the transport network  $G$  in Figure 11.63(a).

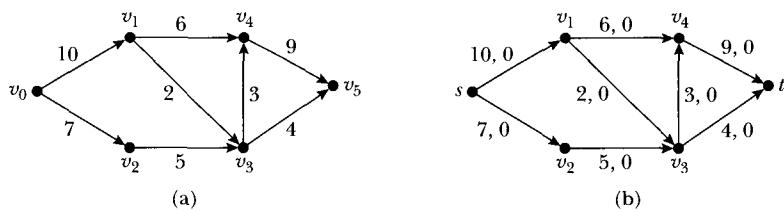


FIGURE 11.63 Transport network  $G$

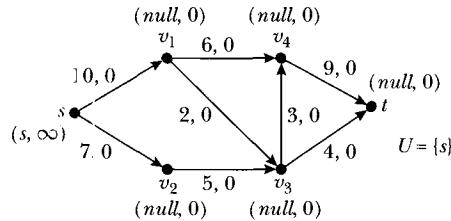
We use the flow-augmentation algorithm to find a maximum flow in  $G$ .

The **for** loop in Line 3 sets the flow in each edge to 0, the statement in Line 5 sets  $s$  to  $v_0$ , and the statement in Line 6 sets  $t$  to  $v_5$  (see Figure 11.63(b)).

The **while** loop in Line 7 continues to execute until a maximum flow is found. Note that after finding a maximal flow the next iteration of a **while** loop in Line 7 does take place. However, in this case, the set  $U$  will eventually become empty and the statement in Line 16 will return the maximal flow found by the algorithm.

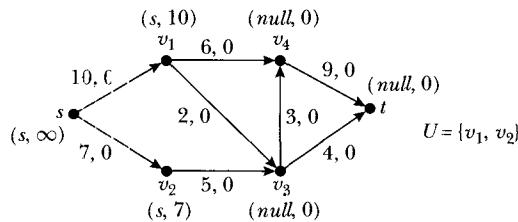
Iteration 1 of the `while` loop at Line 7:

The statement in Line 9 sets the label of each vertex  $v$  to  $(\text{null}, 0)$ , and the statement in Line 11 sets the label of  $s$  to  $(s, \infty)$ . The statement in Line 12 sets  $U$  to  $\{s\}$ . (See Figure 11.64.)



**FIGURE 11.64** Transport network before the  
`while` loop at Line 13 executes

Next, the `while` loop in Line 13 executes until  $t$  is labeled (or  $U$  is empty, in which case a maximum flow has been found, see the statement in Line 15). The statement in Line 17 sets  $v$  to  $s$ , and the statement in Line 18 sets  $U$  to  $\emptyset$ . Next, the `for` loops in Lines 19 and 26 update the label of each vertex  $w$  such that  $w$  is not scanned; i.e.,  $w$  has not been assigned a parent, and label the vertex  $w$  as follows: If  $(s, w)$  is an arc, i.e., it is a forward edge and  $F(sw) < C(sw)$ , then  $p(w)$  is set to  $s$ ; and value of  $w$  is set to the minimum of  $\text{value}(s)$  and the difference of the capacity,  $C(sw)$ , of the arc  $(s, w)$  and the current flow,  $F(sw)$ , in the arc  $(s, w)$ . If  $(w, s)$  is an arc, i.e.,  $(s, w)$  is a backward edge and  $F(sw) > 0$ , then  $p(w)$  is set to  $s$ , and value of  $w$  is set to the minimum of  $\text{value}(s)$  and the current flow,  $F(sw)$ , in the arc  $(s, w)$ . In both of these cases,  $w$  is added to  $U$ . Note that vertices  $v_1$  and  $v_2$  are labeled as follows:  $p(v_1) = s$ ,  $\text{value}(v_1) = \min\{\infty, C(sv_1) - F(sv_1)\} = \min\{\infty, 10\} = 10$ . Similarly,  $p(v_2) = s$  and  $\text{value}(v_2) = 7$ . If  $(s, w)$  is not an edge, then vertex  $w$  is left as is. After these `for` loops execute, the diagram in Figure 11.65 results.



**FIGURE 11.65** Transport network after the `for` loop  
at Lines 19 and 26 executes

Next, the control goes back to the `while` loop at Line 13. Because  $t$  is not labeled, the body of the `while` loop executes. Because  $U \neq \emptyset$ , the statement in Line 17 sets  $v$  to  $v_1$ , and the statement in Line 18 sets  $U$  to  $\{v_2\}$ . Next, the `for` loops in Lines 19 and 26 update the label of each vertex  $w$  such that  $w$  is not scanned; i.e.,  $w$  has not been assigned a parent, and label the vertex  $w$  as follows: If  $(v_1, w)$  is an arc, i.e., it is a forward edge and  $F(v_1w) < C(v_1w)$ , then  $p(w)$  is set to  $v_1$ ; and value of  $w$  is set to the minimum of  $\text{value}(v_1)$  and the difference of the capacity,  $C(v_1w)$ , of the arc  $(v_1, w)$  and the current flow,  $F(v_1w)$ , in the arc  $(v_1, w)$ . If  $(w, v_1)$  is an arc, i.e.,  $(v_1, w)$  is a backward edge and  $F(v_1w) > 0$ , then  $p(w)$  is set to  $v_1$ , and value of  $w$  is set to the minimum of  $\text{value}(v_1)$  and the current flow,  $F(v_1w)$ , in the arc  $(v_1, w)$ . In both of these cases,  $w$  is added to  $U$ . Note that the vertices  $v_3$  and  $v_4$  are labeled as follows:  $p(v_3) = v_1$ ,  $\text{value}(v_3) = \min\{10, C(v_1v_3) - F(v_1v_3)\} =$

$\min\{10, 6\} = 6$ . Similarly,  $p(v_4) = v_1$  and  $\text{value}(v_4) = 2$ . If  $(v_1, w)$  is not an edge, then vertex  $w$  is left as is. After these for loops execute, the diagram in Figure 11.66 results.

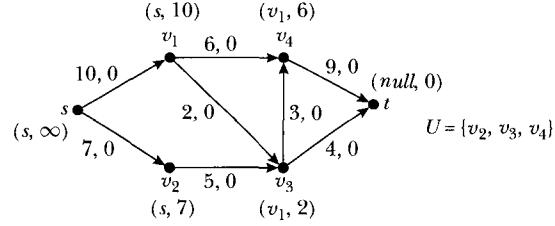


FIGURE 11.66 Transport network after the for loop at Lines 19 and 26 executes

In the next iteration of the while loop at Line 13, vertex  $v_2$  is removed from  $U$ . Now there is an edge from  $v_2$  to  $v_3$ . However, the vertex  $v_3$  is labeled. Therefore, after this iteration of the while loop at Line 13, the network is the same as shown in Figure 11.67.

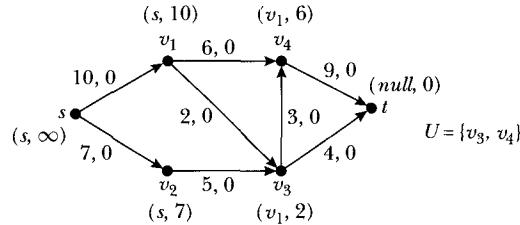


FIGURE 11.67 Transport network after the while loop at Line 13 executes

Next, because  $t$  is not labeled, the next iteration of the while loop at Line 13 takes place. Because  $U \neq \emptyset$ , the statement in Line 17 sets  $v$  to  $v_3$  and the statement in Line 18 sets  $U$  to  $\{v_4\}$ . Next, the for loops in Lines in 19 and 26 update the label of each vertex  $w$  such that  $w$  is not scanned, i.e.,  $w$  has not been assigned a parent, and label vertex  $w$  as follows: If  $(v_3, w)$  is an arc, i.e., it is a forward edge and  $F(v_3w) < C(v_3w)$ , then  $p(w)$  is set to  $v_3$ ; and value of  $w$  is set to the minimum of  $\text{value}(v_3)$  and the difference of the capacity,  $C(v_3w)$ , of the arc  $(v_3, w)$  and the current flow,  $F(v_3w)$ , in the arc  $(v_3, w)$ . If  $(w, v_3)$  is an arc, i.e.,  $(v_3, w)$  is a backward edge and  $F(v_3w) > 0$ , then  $p(w)$  is set to  $v_3$ , and value of  $w$  is set to the minimum of  $\text{value}(v_3)$  and the current flow,  $F(v_3w)$ , in the arc  $(v_3, w)$ . In both of these cases,  $w$  is added to  $U$ . Note that vertex  $t$  is labeled as follows:  $p(t) = v_3$ ,  $\text{value}(t) = \min\{2, C(v_3t) - F(v_3t)\} = \min\{2, 4\} = 2$ . If  $(v_3, w)$  is not an edge, then vertex  $w$  is left as is. After these for loops execute, the diagram in Figure 11.68 results.

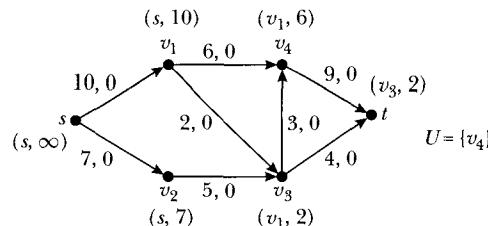
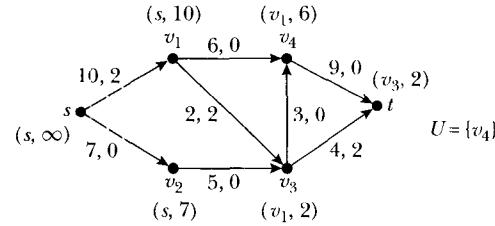


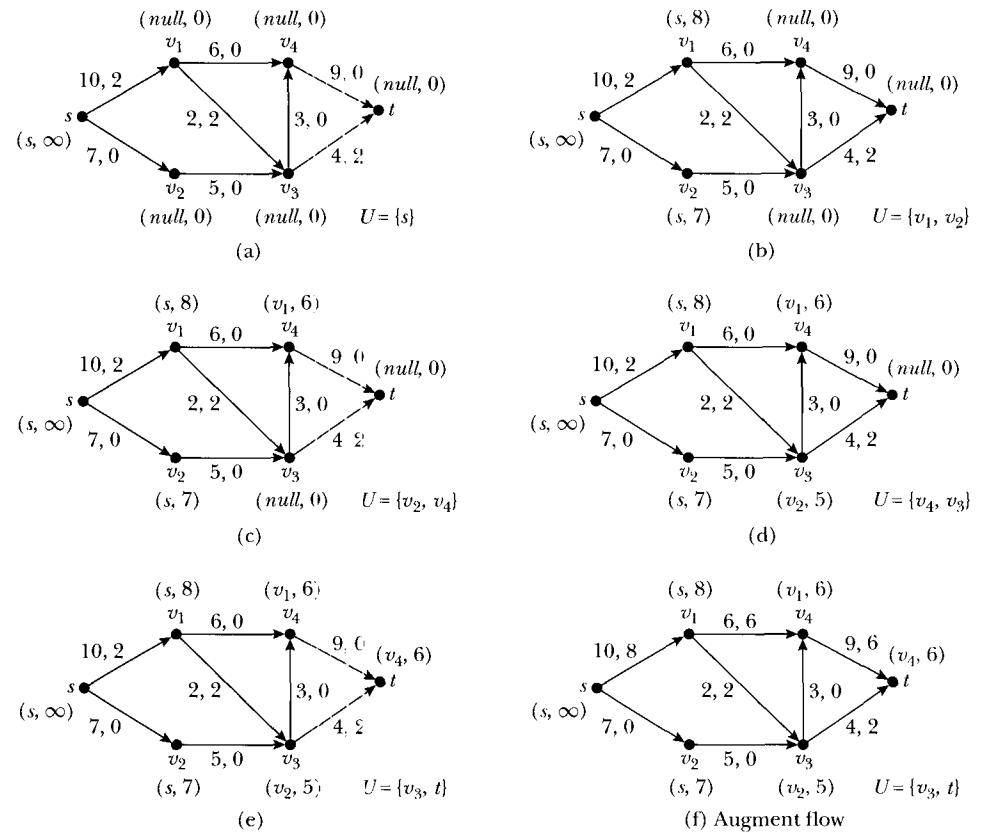
FIGURE 11.68 Transport network after the for loop at Lines 19 and 26 executes

Next, the statements in Lines 35 to 46 augment the flow in the path  $P = (s, v_1, v_3, t)$  as follows: Each of the arcs  $(s, v_1)$ ,  $(v_1, v_3)$ , and  $(v_3, t)$  is a forward arc and  $\text{value}(t) = 2$ . Therefore, the value of the flow in these arcs is increased by 2 (see Figure 11.69).



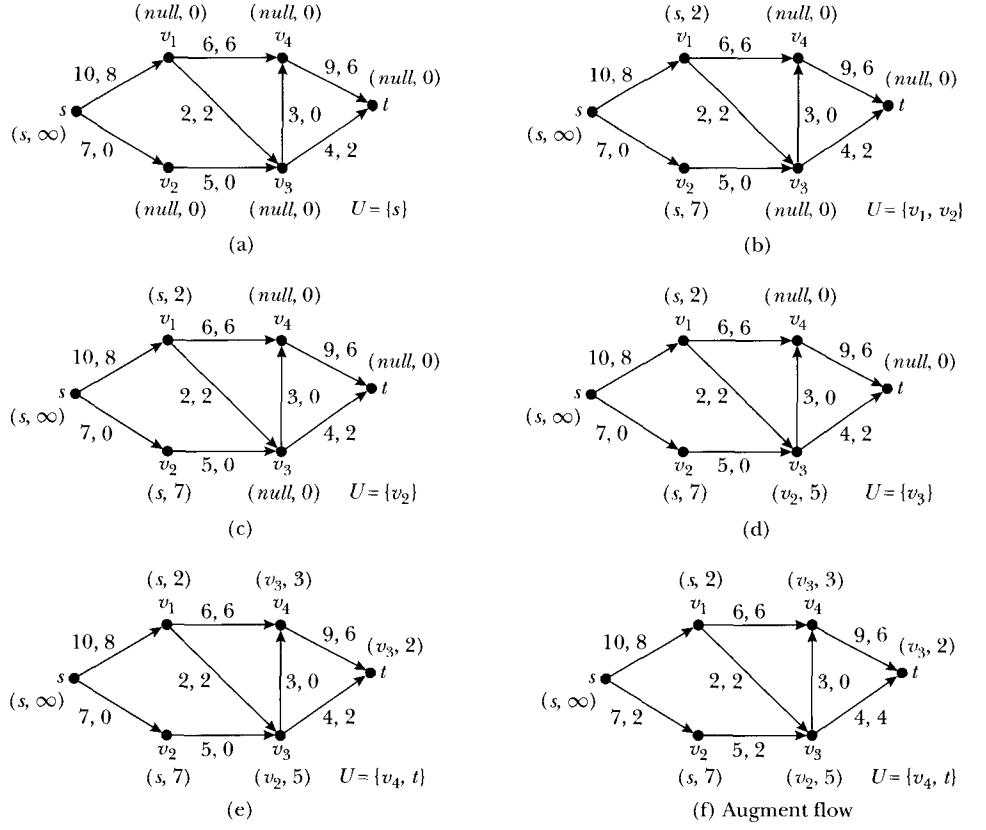
**FIGURE 11.69** Transport network after the statements in Lines 35 to 47 execute

After this the control goes back to the `while` loop in Line 7. The body of this `while` loop executes. The current labels are removed, the label of  $s$  is set to  $(s, \infty)$ , and  $U$  is set to  $\{s\}$ . The diagrams in Figure 11.70 show how the vertices are labeled and the flow is augmented.



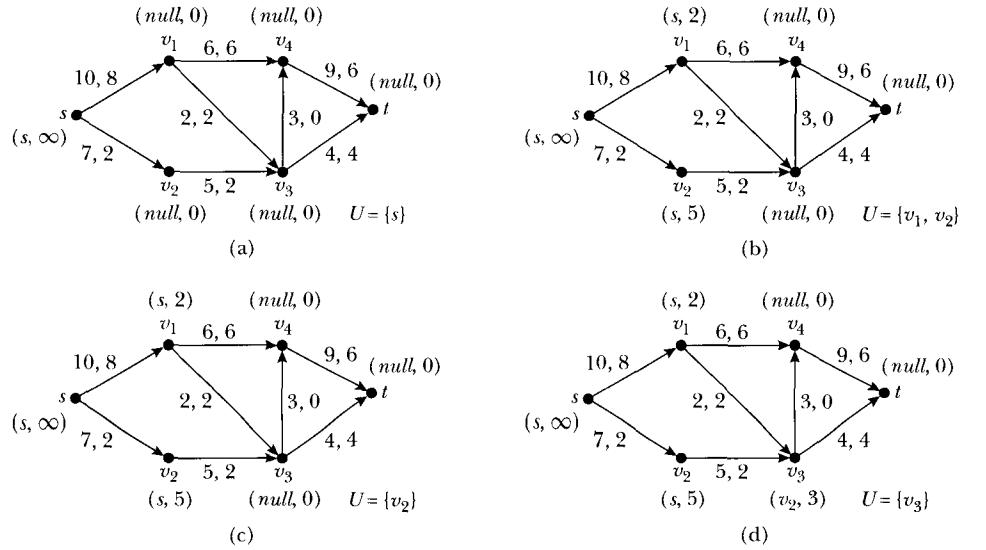
**FIGURE 11.70** Transport network after the second iteration of the `while` loop at Line 7

Once again, the control goes back to the `while` loop in Line 7. The body of this `while` loop executes. The current labels are removed, the label of  $s$  is set to  $(s, \infty)$ , and  $U$  is set to  $\{s\}$ . The diagrams in Figure 11.71 show how the vertices are labeled and the flow is augmented.



**FIGURE 11.71** Transport network after the third iteration of the `while` loop at Line 7

Once again, the control goes back to the `while` loop in Line 7. The body of this `while` loop executes. The current labels are removed, the label of  $s$  is set to  $(s, \infty)$ , and  $U$  is set to  $\{s\}$ . The diagrams in Figure 11.72 show how the vertices are labeled and the flow is augmented.



**FIGURE 11.72** Transport network after the fourth iteration of the `while` loop at Line 7

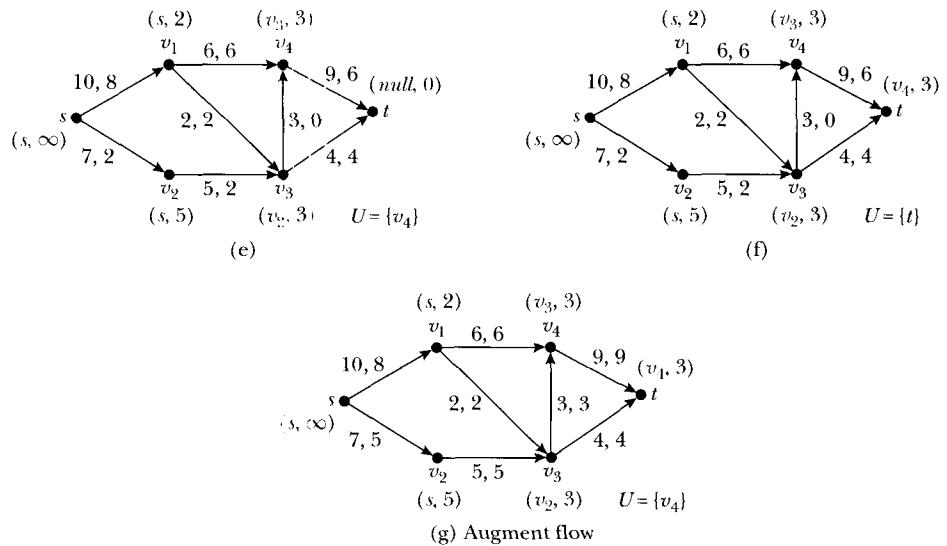


FIGURE 11.72 Continued

Once again, the control goes back to the `while` loop in Line 7. The current labels are removed, the label of  $s$  is set to  $(s, \infty)$ , and  $U$  is set to  $\{s\}$ . Notice that the max-flow has been found. However, the statements in Lines 9 to 32 execute. Eventually, the statement in Line 15 would evaluate to true and the statement in Line 16 returns the current flow. When the algorithm terminates, the resulting network is as shown in Figure 11.73.

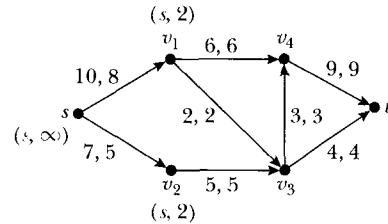


FIGURE 11.73 Transport network with a maximal flow

---

**REMARK 11.4.34** ▶ Notice that, when the max-flow algorithm terminates, some of the vertices are labeled and some are not.

We leave the proof of the following theorem as an exercise.

**Theorem 11.4.35:** Let  $N$  be a transport network.

- (i) Suppose during an iteration of the flow-augmentation algorithm sink  $t$  is unlabeled. Then the present flow in the network is a maximal flow.
- (ii) Suppose at the termination of the algorithm  $X$  is the set of labeled vertices and  $X' = V - X$ . Then  $A(X, X')$  is a minimal cut.

## Matching (Revisited)

In this section, we use the property of the  $s$ - $t$  network to solve some matching problems. Recall from Chapter 10 that a graph  $G = (V, E)$  is a bipartite graph if there exists a bipartition  $V = V_1 \cup V_2$  of  $V$  such that each edge  $e \in E$  joins a vertex  $v_1 \in V_1$  and a vertex  $v_2 \in V_2$ .

Recall that a set  $M$  of edges of a graph  $G$  is called a matching in  $G$  if no two edges in  $M$  have an end vertex in common. If  $e \in M$  with end vertices  $u$  and  $v$ , we say that  $M$  matches  $u$  with  $v$ . A maximal matching is a matching with the greatest number of edges.

Let  $G = (V, E)$  be a bipartite graph with bipartition  $V = V_1 \cup V_2$ . Then the problem of finding a maximum matching in  $G$  can be converted into a maximal-flow problem in an  $s$ - $t$  network  $N$  constructed from the graph  $G = (V, E)$  as follows.

1. The vertex set of  $N$  is  $V \cup \{s, t\}$ , where  $s$  and  $t$  are two new vertices that are the source and sink, respectively, of  $N$ .
2. Each edge  $e$  of  $G$  between a vertex  $u \in V_1$  and a vertex  $v \in V_2$  corresponds to an arc in network  $N$  directed from  $u$  to  $v$ .
3. For each vertex  $u \in V_1$ , an arc in network  $N$  is drawn from source  $s$  to vertex  $u$ .
4. For each vertex  $v \in V_2$ , an arc in network  $N$  is drawn from  $v$  to sink  $t$ .
5. Each arc  $e$  in network  $N$  is assigned a capacity  $C(e) = 1$ .

We call this  $s$ - $t$  network  $N$  the network associated with the bipartite graph  $G$  and denote it by  $N_G$ .

### EXAMPLE 11.4.36

Consider the bipartite graph  $G = (V, E)$  of Figure 11.74, where

$$V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$$

and  $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\}$ . Here  $V_1 = \{v_1, v_2, v_3\}$  and  $V_2 = \{v_4, v_5, v_6, v_7\}$ .

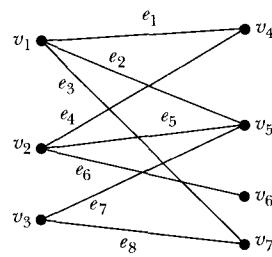
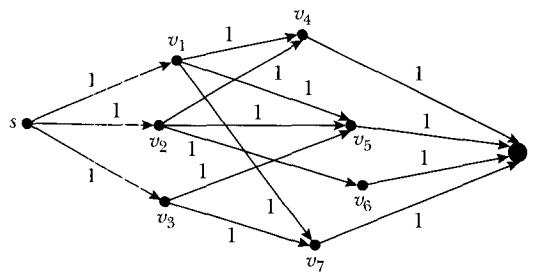


FIGURE 11.74 Bipartite graph

The  $s$ - $t$  network  $N_G$  associated with  $G$  is shown in Figure 11.75. Each arc in  $N_G$  is assigned a capacity 1. Hence,  $N_G$  is a transport network.

The following theorem shows that the maximal matching problem can be treated as a maximal-flow problem.

We leave the proof of the following theorem as an exercise.



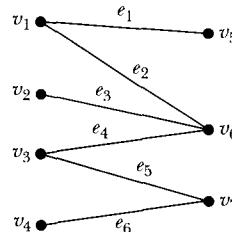
**FIGURE 11.75** An  $s$ - $t$  network  $N_G$  associated with graph  $G$

**Theorem 11.4.37:** Let  $G = (V, E)$  be a bipartite graph with a bipartition  $V = V_1 \cup V_2$  and let  $N_G$  be the transport network associated with  $G$ . Then

- (i) there exists a one-to-one correspondence between the integral flows, i.e., flows with integer values, of network  $N_G$  and the matching in graph  $G$ ; and
- (ii) a maximal flow in  $N_G$  corresponds to a maximal matching of  $G$ .

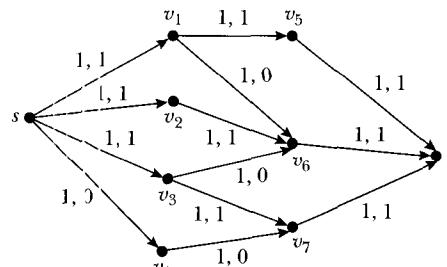
#### EXAMPLE 11.4.38

Consider the bipartite graph  $G = (V, E)$  (see Figure 11.76), where  $V_1 = \{v_1, v_2, v_3, v_4\}$  and  $V_2 = \{v_5, v_6, v_7\}$ .



**FIGURE 11.76**  
A bipartite graph

The digraph in Figure 11.77 shows the transport network  $N_G$ , with a maximal flow, associated with this graph  $G$ .



**FIGURE 11.77** Transport network  $N_G$ , with a maximal flow

Let  $M = \{e_1, e_3, e_5\}$ . Then  $M$  is a maximal matching for  $G$ .

## WORKED-OUT EXERCISES

**Exercise 1:** In the  $s-t$  network shown in Figure 11.78, find the value of the flow. Is the flow maximal? If not, find a maximal flow.

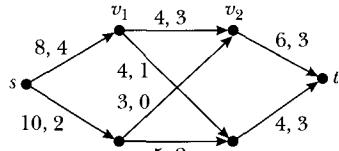
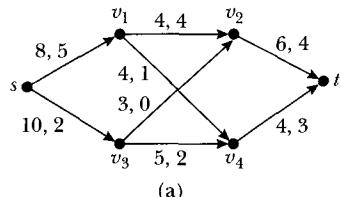


FIGURE 11.78 An  $s-t$  network

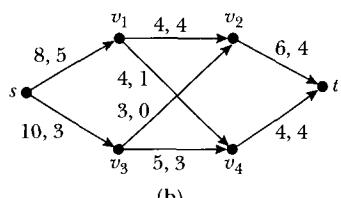
**Solution:** Let  $F$  be the given flow. The value of the given flow is 6. Consider the quasipath  $Q : s \rightarrow v_1 \rightarrow v_2 \rightarrow t$ . The slack  $i(Q) = \min\{4, 1, 3\} = 1 > 0$ . Hence,  $Q$  is an  $F$ -unsaturated quasipath from  $s$  to  $t$ . Therefore, this flow is not maximal. We increase the flow by 1 through this path. Then we obtain the network in Figure 11.79(a) with this new flow, say  $F_1$ .

In the network shown in Figure 11.79(a), we have a quasipath  $Q_1 : s \rightarrow v_3 \rightarrow v_4 \rightarrow t$ . The slack  $i(Q_1) = \min\{8, 3, 1\} = 1 > 0$ . Hence,  $Q_1$  is an  $F_1$ -unsaturated quasipath from  $s$  to  $t$ . Therefore, the flow  $F_1$  is not maximal. We increase the flow  $F_1$  by 1 through this path. Then we obtain the network shown in Figure 11.79(b) with this new flow, say  $F_2$ , and the value of  $F_2$  is 8.

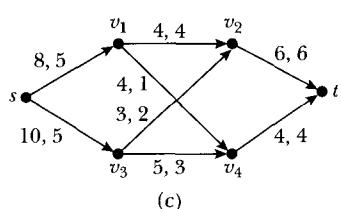
In the network shown in Figure 11.79(b), we have a quasipath  $Q_2 : s \rightarrow v_3 \rightarrow v_2 \rightarrow t$ . The slack  $i(Q_2) = \min\{7, 3, 2\} = 2 > 0$ . Hence,  $Q_2$  is an  $F_2$ -unsaturated quasipath from  $s$  to  $t$ . Therefore, the flow  $F_2$  is not maximal. We increase the flow  $F_2$  by 2 through this path. Then we obtain the network shown in Figure 11.79(c) with this new flow, say  $F_3$ , and the value of  $F_3$  is 10.



(a)



(b)



(c)

FIGURE 11.79  $s-t$  network with flows

In Figure 11.79(c), we have  $i(v_2t) = i(v_4t) = 0$ . It follows that the network shown in Figure 11.79(c) has no more unsaturated  $s-t$  paths. Hence, the flow  $F_3$  is a maximal flow. Let  $V_s = \{s, v_1, v_2, v_3, v_4\}$  and  $V_t = \{t\}$ . Then  $V_s, V_t$  is a partition of  $V$  and  $s \in V_s$  and  $t \in V_t$ . Hence, we have an  $s-t$  cut  $A(V_s, V_t)$  and the capacity of this cut is

$$C(v_2t) + C(v_4t) = 6 + 4 = 10.$$

**Exercise 2:** Find a maximum matching for the bipartite graph  $G$  in Figure 11.80.

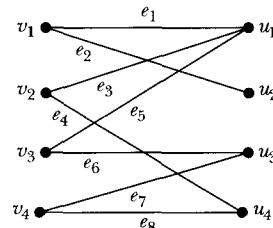


FIGURE 11.80 Bipartite graph

**Solution:** For the given bipartite graph, consider the associated  $s-t$  network  $N_G$ . The diagram of  $N_G$  with the capacities of the arcs is shown in Figure 11.81.

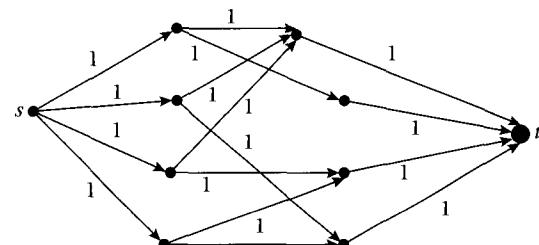


FIGURE 11.81  $s-t$  network of the graph in Figure 11.80

Now by the maximal-flow algorithm, we obtain a maximal flow, which is shown in Figure 11.82.

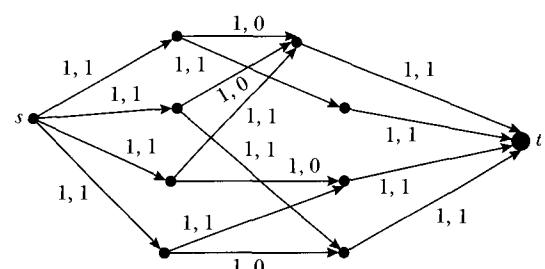


FIGURE 11.82  $s-t$  network with a maximal flow

Hence,  $M = \{e_2, e_4, e_5, e_7\}$  is a maximum matching. Note that the first iteration of the max-flow algorithm augments the  $s-t$  path  $s - v_1 - u_1 - t$ , the second iteration augments the  $s-t$  path  $s - v_2 - u_4 - t$ , the third iteration augments the  $s-t$  path  $s - v_3 - u_3 - t$ , and the fourth iteration augments the

*s-t* path  $s - v_4 - u_3 - v_3 - u_1 - v_1 - u_2 - t$ . Moreover, notice that in the path  $s - v_4 - u_3 - v_3 - u_1 - v_1 - u_2 - t$ , some

of the edges are backward. On a backward edge we reduce the flow by the slack of the *s-t* path.

## SECTION REVIEW

---

### Key Terms

single-source, single-sink network	flow conservation	quasipath
source	flow in edge	forward arc
target	flow into	backward arc
sink	flow out of	slack
<i>s-t</i> network	conservation of flow	<i>F</i> -saturated
capacity	value of a flow	<i>F</i> -unsaturated
transport network	<i>s-t</i> cut of a network	flow augmenting
network	capacity of an <i>s-t</i> cut	parent
flow	minimal cut	immediate predecessor
capacity constraint	maximal flow	

### Some Key Definitions

1. A single-source, single-sink network is a simple digraph  $N = (V, E)$  such that the underlying graph is connected. It has a distinguished vertex  $s$ , called the source, and a distinguished vertex  $t$ , called the target, or sink, with the in-degree of  $s$  equal to 0 and the out-degree of  $t$  equal to 0.
  2. Let  $N = (V, E)$  be an *s-t* network such that each arc  $e \in E$  is assigned a non-negative integer  $C(e)$ , called the capacity of  $e$ . Then  $N$  is called a transport network, or simply a network.
  3. Let  $N = (V, E)$  be a transport network. Suppose that  $X$  and  $Y$  are nonempty subsets of  $V$ . Then the set  $A(X, Y)$ , the set of all arcs  $e \in E$  with  $\text{tail}(e) \in X$ ,  $\text{head}(e) \in Y$ , i.e.,  $A(X, Y) = \{e \in E \mid \text{tail}(e) \in X, \text{head}(e) \in Y\}$ .
  4. Let  $N = (V, E)$  be a transport network. A flow  $F$  in  $N$  is a function  $F : E \rightarrow \mathbb{N} \cup \{0\}$  that assigns a nonnegative integer  $F(e)$  to each arc  $e$  of  $N$  such that
    - (Capacity constraint)  $F(e) \leq C(e)$ , for every arc  $e$  of  $N$ ,
    - (Flow conservation)  $\sum_{e \in \text{in}(v)} F(e) = \sum_{e \in \text{out}(v)} F(e)$ , for every vertex  $v$  of  $N$  other than source  $s$  and sink  $t$ .
  5. Let  $N = (V, E)$  be a transport network with a flow  $F$ . The value  $d = \sum_{e \in \text{out}(s)} F(e) = \sum_{e \in \text{in}(t)} F(e)$  is called the value of the flow  $F$ .
  6. Let  $N = (V, E)$  be a transport network. Suppose that  $V_s$  and  $V_t$  is a partition of  $V$  such that  $s \in V_s$  and  $t \in V_t$ . Then the set  $A(V_s, V_t)$ , the set of all arcs  $e \in E$  with  $\text{tail}(e) \in V_s$ ,  $\text{head}(e) \in V_t$ , is called an *s-t* cut of the network  $N$ . That is,
- $$A(V_s, V_t) = \{e \in E \mid \text{tail}(e) \in V_s, \text{head}(e) \in V_t\}.$$
7. Let  $N = (V, E)$  be a transport network with a flow  $F$ . If  $X$  and  $Y$  are two nonempty subsets of  $V$ , then we define

$$\begin{aligned} \text{(i)} \quad C(X, Y) &= \sum_{e \in A(X, Y)} C(e), \\ \text{(ii)} \quad F(X, Y) &= \sum_{e \in A(X, Y)} F(e). \end{aligned}$$

8. Let  $N = (V, E)$  be a transport  $t$  network with a flow  $F$ . Then  $C(V_s, V_t)$  is called the capacity of the  $s$ - $t$  cut  $A(V_s, V_t)$ .
9. Let  $N = (V, E)$  be a transport network with a flow  $F$ . Then a minimal cut in  $N$  is an  $s$ - $t$  cut  $A(V_s, V_t)$  with the minimum capacity.
10. Let  $N = (V, E)$  be a transport network. A flow  $F$  in  $N$  is called a maximal flow if for every flow  $F'$  in  $N$ , the value of  $F'$  is less than or equal to the value of the flow  $F$ .
11. Let  $N = (V, E)$  be a transport network. A quasipath in  $N$  is an alternating sequence  $(v_0, e_1, v_1, e_2, v_2, \dots, v_{k-1}, e_{k-1}, v_k)$  of vertices and arcs such that  $(v_0, e_1, v_1, e_2, v_2, \dots, v_{k-1}, e_{k-1}, v_k)$  is a path in the underlying graph of  $N$ .
12. Let  $Q = (v_0, e_1, v_1, e_2, v_2, \dots, v_{k-1}, e_{k-1}, v_k)$  be a quasipath in a transport network  $N$  with a flow  $F$ . An arc  $e_i$  of  $Q$  is called a forward arc if it is directed from  $v_{i-1}$  to  $v_i$ , and an arc  $e_i$  of  $Q$  is called a backward arc if it is directed from  $v_i$  to  $v_{i-1}$ .
13. Let  $N$  be a transport network with a flow  $Q$  be a quasipath in  $N$ .
  - (i) For each arc in  $Q$ , we associate a nonnegative integer  $i(e)$ , defined by

$$i(e) = \begin{cases} C(e) - F(e), & \text{if } e \text{ is a forward arc,} \\ C(e), & \text{if } e \text{ is a backward arc.} \end{cases}$$

The number  $i(e)$  is called the slack on the arc  $e$ .

- (ii) To  $Q$  we associate a nonnegative integer  $i(Q)$ , defined by

$$i(Q) = \min\{i(e) \mid e \text{ is an arc in } Q\}.$$

14. Let  $N$  be a transport network with a flow. A quasipath  $Q$  in  $N$  is called  $F$ -saturated if  $i(Q) = 0$  and  $F$ -unsaturated if  $i(Q) > 0$ .

## Some Key Results

1. Let  $N = (V, E)$  be a transport network with a flow  $F$ . If  $V = \{v_0, v_1, v_2, \dots, v_n\}$  with  $s = v_0$  and  $t = v_n$ , then the flow out of the source  $s$  equals the flow into the sink  $t$ , i.e.,  $\sum_{v_i \in V} F_{0i} = \sum_{v_i \in V} F_{in}$ .
2. Let  $N = (V, E)$  be a transport network with a flow  $F$ . Let  $d$  be the value of the flow  $F$ . If  $A(V_s, V_t)$  is an  $s$ - $t$  cut of  $N$ , then
  - (i)  $d = F(V_s, V_t) - F(V_t, V_s)$ ,
  - (ii)  $d \leq C(V_s, V_t)$ .
3. Let  $N = (V, E)$  be a transport network with a flow  $F$ . Then  $F$  is a maximal flow if and only if there does not exist an  $F$ -unsaturated quasipath  $Q$  from  $s$  to  $t$  in  $N$ .
4. Let  $N = (V, E)$  be a transport network with a flow. Then there exists a maximal flow in  $N$ ; i.e., there exists a flow in  $N$  with value  $\min\{C(V_s, V_t) \mid A(V_s, V_t)$  is an  $s$ - $t$  cut}.

## EXERCISES

1. Verify that the diagram shown in Figure 11.83 represents a transport network with a flow. Find the value of the flow.

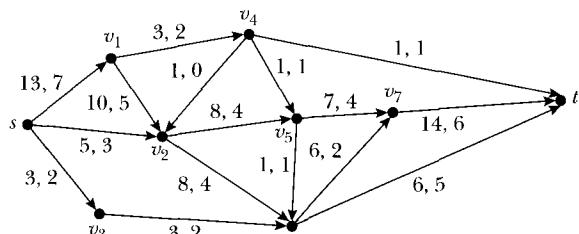


FIGURE 11.83 Transport network

2. In the transport network of Exercise 1
- verify the law of conservation,
  - find the capacity of the  $s-t$  cut  $A(\{s, v_1, v_2, v_3\}, \{t, v_4, v_5, v_6, v_7\})$ .
3. In the transport network shown in Figure 11.84, find the value of the flow. Is the flow maximal? If the flow is not maximal, find a maximal flow.

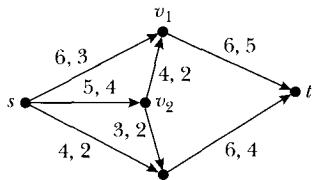


FIGURE 11.84 Transport network

4. Consider the transport network shown in Figure 11.85. Find an  $F$ -unsaturated  $s-t$  path  $Q$ . Also, find  $i(Q)$ .

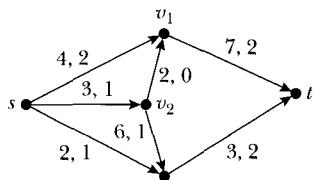


FIGURE 11.85 Transport network

5. Verify that flow of conservation of the transport network of Exercise 3. Find the value of the flow. If the flow is not maximal, find a maximal flow.
6. Verify that Figure 11.86 represents a transport network. Find the value of the flow. Find an  $F$ -unsaturated  $s-t$  path  $Q$ . Also find  $i(Q)$ .

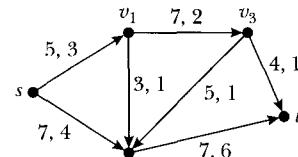


FIGURE 11.86 Transport network

7. Is the flow a maximal flow in the transport network in Exercise 6? If it is not a maximal flow, find a maximal flow.
8. In the  $s-t$  network shown in Figure 11.87, with a flow  $F$ , find an  $F$ -unsaturated  $s-t$  quasipath  $Q$ . Find  $i(Q)$ . If the flow is not maximal, then find a maximal flow.

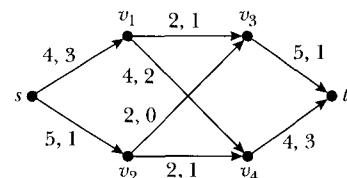


FIGURE 11.87  $s-t$  network

9. For the transport network shown in Figure 11.88, use flow-augmentation algorithm, Algorithm 11.7, to find a maximal flow.

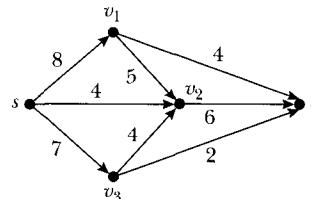


FIGURE 11.88 A transport network

10. Find a maximal matching for the bipartite graphs in Figure 11.89.

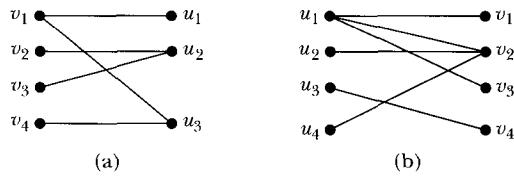


FIGURE 11.89 Bipartite graphs

## ► PROGRAMMING EXERCISES

- Write a program to determine if a graph is a tree.
- Write a program to implement various operations on binary search trees. The program must implement the

following operations (at least): the inorder, preorder, and postorder traversal; search; insert; determining the height, number of nodes, number of leaves, and

- number of single parents (vertices that have only one child).
3. Write a program to implement the breadth-first search spanning tree algorithm given in this chapter.
  4. Write a program to implement Prim's algorithm given in this chapter.
  5. **Kruskal's algorithm to find a minimal spanning tree of a connected graph.** Let  $G$  be a connected weighted graph. Kruskal's algorithm finds a minimal

spanning tree  $T$  of  $G$  as follows: Initially, the vertex set  $V(T)$  of  $T$  consists of all the vertices of  $G$ , the edge set  $E(T)$  of  $T$  is empty, and the set  $E$  consists of all the edges of  $G$ . At each step, an edge  $e \in E$  of smallest weight is selected. (Ties are broken arbitrarily, i.e., at any step, if there is more than one edge of smallest weight, select one of the edges.) If the addition of  $e$  to  $E(T)$  does not produce a cycle in  $T$ ,  $e$  is added to  $E(T)$ . When the number of edges in  $E(T)$  is  $n - 1$ , the algorithm terminates.

**ALGORITHM 11.8:** Kruskal's algorithm to find a minimal spanning tree of a connected graph.

*Input:*  $G$ —weighted connected graph  
 $n$ —number of vertices in  $G$   
 $W$ —weight matrix of  $G$

*Output:*  $T$ —minimal spanning tree

```

1. procedure mstKruskal( $G, n, W, T$ )
2. begin
3.    $V(T) := V(G);$  //initialize vertex set  $V(T)$  of  $T$ 
4.    $E(T) := \emptyset;$  //initialize edge set  $E(T)$  of  $T$ 
5.    $E := E(G);$  // $E(G)$  denotes the edge set of  $G$ .
6.    $i := 0;$ 
7.   while  $i < n - 1$  do
8.     begin
9.       Find an edge  $e \in E$  of smallest weight //break ties arbitrarily
10.       $E := E - \{e\};$ 
11.      if  $E(T) \cup \{e\}$  does not form a cycle in  $T$  then
12.        begin
13.           $E(T) := E(T) \cup \{e\};$ 
14.           $i := i + 1;$ 
15.        end
16.      end
17.    end

```

Write a program to implement Kruskal's algorithm to find a minimal spanning tree of a connected graph.

6. Write a program to implement the flow-augmentation algorithm given in this chapter.

## Boolean Algebra and Combinatorial Circuits

**The objectives of this chapter are to:**

- Learn about Boolean expressions
- Become aware of the basic properties of Boolean algebra
- Explore the application of Boolean algebra in the design of electronic circuits
- Learn the application of Boolean algebra in switching circuits

In Chapter 3, we introduced the notion of Boolean algebra. In 1854, George Boole in his book *The Laws of Thoughts* described the basic laws of logic. In 1938, C. E. Shannon showed how Boole's basic laws of logic can be used in the design of electronic circuits. In the first two sections of this chapter, we further study Boolean algebra and its basic properties. We then show how Boolean algebra is used in the design of electronic circuits.

## 12.1 TWO-ELEMENT BOOLEAN ALGEBRA

In Chapter 1, we considered the two-element set  $B = \{T, F\}$ . On this set we defined the operations  $\vee$  (or),  $\wedge$  (and), and  $\sim$  (not) as follows:

$$\begin{array}{llll} T \vee T = T, & T \vee F = T, & F \vee T = T, & F \vee F = F, \\ T \wedge T = T, & T \wedge F = F, & F \wedge T = F, & F \wedge F = F, \\ \sim T = F, & \sim F = T. \end{array}$$

These operations on  $B$  can be described as follows.

$\vee$	T	F	$\wedge$	T	F	$\sim$	T	F
T	T	T	T	T	F	T	F	
F	T	F	F	F	F	F	T	

Let  $S$  be a nonempty set. Now  $S$  and  $\emptyset$  are subsets of  $S$ . Let  $B = \{S, \emptyset\}$ . Then  $B$  is a two-element set. As in Chapter 1, we can define the operations  $\cup$  (union),  $\cap$  (intersection), and  $'$  (complement) on  $B$  as follows.

$\cup$	S	$\emptyset$	$\cap$	S	$\emptyset$	$'$	S	$\emptyset$
S	S	S	S	S	$\emptyset$	S	$\emptyset$	
$\emptyset$	S	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	S

Let us consider electric switches. The words associated with electric switches are “on” and “off.” Let  $B = \{\text{on}, \text{off}\}$ . Then  $B$  is a two-element set.

In view of these examples, in general, we can consider a two-element set  $B = \{1, 0\}$ , where 1 may stand for  $T$  or the whole set  $S$  or for the word “on” and 0 may stand for  $F$  or the empty set  $\emptyset$  or for the word “off.” We can define the three operations  $+$ ,  $\cdot$ , and  $'$  on  $B$  as follows.

$+$	1	0	$\cdot$	1	0	$'$	1	0
1	1	1	1	1	0	1	0	
0	1	0	0	0	0	0	1	

In other words,

$$\begin{array}{llll} 1 + 1 = 1, & 1 + 0 = 1, & 0 + 1 = 1, & 0 + 0 = 0, \\ 1 \cdot 1 = 1, & 1 \cdot 0 = 0, & 0 \cdot 1 = 0, & 0 \cdot 0 = 0, \\ 1' = 0, & 0' = 1. \end{array}$$



**George Boole**  
(1815–1864)

Born in the English industrial town of Lincoln, Boole obtained his early mathematics instruction from his father. Although he did not study for an academic degree, Boole excelled and became an assistant teacher at the age of 16. He opened his own school at the age of 20 and published his first paper in the *Cambridge Mathematical Journal*.

### Historical Notes

at the age of 24. Boole was appointed chair of the mathematics department at Queens College, Cork, in 1849 and had an outstanding teaching record there for the remainder of his career.

Boole’s greatest accomplishment revolved around an article published in 1854, “An Investigation into the Laws of Thought, on Which are Founded the Mathematical Theories of Logic and Probabilities.” This article dis-

cussed a fundamental connection between logic and mathematics, particularly using the concepts of “and,” “or,” and “not.” Boolean algebra was thus developed and defined. This connection has provided a foundation to mathematicians in the areas of abstract algebra and numerical analysis. In addition, it provided a foundation for the design of electrical circuits.

The operations  $+$ ,  $\cdot$ , and  $'$  on  $B = \{0, 1\}$  are called the **Boolean sum**, **Boolean product**, and **Boolean complementation**, respectively.

On the two-element set  $B = \{T, F\}$ , the Boolean sum corresponds to the logical operation  $\vee$  (or), the product corresponds to  $\wedge$  (and), and complementation corresponds to  $\sim$  (negation, or not).

**DEFINITION 12.1.1** ► The set  $B = \{0, 1\}$  together with Boolean sum,  $+$ , Boolean product,  $\cdot$ , and Boolean complementation,  $'$ , is called a **two-element Boolean algebra**. We denote this two-element Boolean algebra by  $B$ .

**DEFINITION 12.1.2** ► Any literal symbol such as  $x, y, z, x_1, x_2, \dots, x_n$  (with or without subscripts) used to represent an element of  $B = \{0, 1\}$  is called a **Boolean variable**.

**DEFINITION 12.1.3** ► Let  $x_1, x_2, \dots, x_n$  be Boolean variables. A **Boolean expression** over  $B$  is defined recursively as follows.

1. 0 and 1 are Boolean expressions.
2.  $x_1, x_2, \dots, x_n$  are Boolean expressions.
3. If  $\alpha$  is a Boolean expression, then  $(\alpha')$  is a Boolean expression.
4. If  $\alpha_1$  and  $\alpha_2$  are Boolean expressions, then  $(\alpha_1 \cdot \alpha_2)$  and  $(\alpha_1 + \alpha_2)$  are also Boolean expressions.
5. The only expressions that are Boolean expressions on  $x_1, x_2, \dots, x_n$  are those that are determined by rules 1–4.

**EXAMPLE 12.1.4**  $(x_1 + (x_2')), ((x_1 \cdot (x_1')) + x_3)$ , and  $((x_1 \cdot x_2) + (x_3 + x_2))'$  are Boolean expressions.

**Convention:** To avoid using so many parentheses in writing Boolean expressions we adopt the following convention: We omit the outer pair of parentheses in a Boolean expression. For example, the expressions  $(x_1')$  and  $((x_1 \cdot x_2) + (x_3 + x_2))'$  are written  $x_1'$  and  $(x_1 \cdot x_2) + (x_3 + x_2)'$ , respectively.

**DEFINITION 12.1.5** ► A Boolean expression that contains  $n$  distinct Boolean variables is usually referred to as a **Boolean expression in  $n$  variables**.



### Historical Notes

**Claude Elwood Shannon** (1916–2001)  
Shannon was a graduate of the University of Michigan and later

received a mathematics Ph.D. from MIT in 1940. It was his Master's thesis work on relay circuits at MIT that led to his contribution to computers and telecommunication. While working at

Bell Laboratories in 1948, Shannon's Theorem was introduced, giving the maximum rate at which error-free bits could be transmitted over a noisy channel. This helped to establish more reliable communication.

Shannon continued to work at Bell Laboratories until 1972. In 1952, he devised an experiment illustrating the capabilities of telephone relays. Later,

he worked in the area of artificial intelligence and in 1956 he devised a complex chess program using the MANIAC computer. He worked at MIT until his retirement in 1978.

We denote Boolean expressions on  $B$  by  $\alpha, \beta, \gamma$ , and so on. If  $\alpha$  is a Boolean expression in  $n$  distinct variables  $x_1, x_2, \dots, x_n$ , then we denote it by  $\alpha(x_1, x_2, \dots, x_n)$ .

Let  $B = \{0, 1\}$  be a two-element Boolean algebra with operations  $+$ ,  $\cdot$ , and  $'$ . Let  $\alpha(x_1, x_2, \dots, x_n)$  be a Boolean expression on  $B$ . If we assign 0 or 1 to each  $x_i$  in  $\alpha$ , then we find a value of  $\alpha$  in  $B$  corresponding to this assignment.

**EXAMPLE 12.1.6**

Let  $\alpha(x_1, x_2) = (x_1 \cdot x_2) + x'_2$ . Suppose  $x_1 = 0$  and  $x_2 = 1$ . Then  $\alpha(0, 1) = (0 \cdot 1) + 1' = 0 + 0 = 0$ . Hence, the value of  $\alpha(x_1, x_2)$  is 0 when  $x_1 = 0$  and  $x_2 = 1$ .

**REMARK 12.1.7** ▶ In a Boolean expression in which parentheses are not used to specify the order of operations, we assume that the product,  $\cdot$ , is evaluated before the sum,  $+$ . For example,  $x_1 \cdot x_2 + x_3 = (x_1 \cdot x_2) + x_3$  and  $x_1 + x_2 \cdot x_3 = x_1 + (x_2 \cdot x_3)$ .

In Chapter 1, we used truth tables to show the truth values of a statement formula for different assignments of truth values to the statement variables. We can do the same for Boolean expressions. Corresponding to a Boolean expression  $\alpha(x_1, x_2, \dots, x_n)$ , we can construct its truth table showing the values of  $\alpha(x_1, x_2, \dots, x_n)$  for all possible assignments 0 or 1 to  $x_i$ . The following example illustrates this.

**EXAMPLE 12.1.8**

In this example, we construct the truth table for the Boolean expression  $\alpha(x_1, x_2, x_3) = (x_1 + x_2) \cdot (x_1 + (x_2 \cdot x'_3))$  as follows. (Because  $\alpha$  is a Boolean expression in three variables,  $x_1, x_2$ , and  $x_3$ , and each of these variables can have the value 0 or 1, there are a total of eight different values that can be assigned to these variables.)

$x_1$	$x_2$	$x_3$	$x_1 + x_2$	$x'_3$	$x_2 \cdot x'_3$	$x_1 + (x_2 \cdot x'_3)$	$\alpha$
1	1	1	1	0	0	1	1
1	1	0	1	1	1	1	1
1	0	1	1	0	0	1	1
1	0	0	1	1	0	1	1
0	1	1	1	0	0	0	0
0	1	0	1	1	1	1	1
0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0

According to the definition of a Boolean expression, 1 and 0 are Boolean expressions. Also, corresponding to every Boolean expression  $\alpha$  we can construct its truth table. Therefore, we can also construct the truth tables for the Boolean expressions 1 and 0. The truth tables for 1 and 0 satisfy the following:

The truth value of 1 is always 1 for any assignment of values 0, 1 to the variables  $x_1, x_2, \dots, x_n$ .

The truth value of 0 is always 0 for any assignment of values 0, 1 to the variables  $x_1, x_2, \dots, x_n$ .

**DEFINITION 12.1.9** ▶ Two Boolean expressions  $\alpha(x_1, x_2, \dots, x_n)$  and  $\beta(x_1, x_2, \dots, x_n)$  are said to be **equal** if they assume the same value for every assignment of values 0, 1 to the variables  $x_1, x_2, \dots, x_n$ .

**EXAMPLE 12.1.10**

Consider the Boolean expressions  $\alpha(x_1, x_2) = (x_1 + x_2)'$  and  $\beta(x_1, x_2) = x'_1 \cdot x'_2$ . In this example, we show that  $\alpha$  and  $\beta$  are equal. To verify this we construct the following truth table of  $\alpha(x_1, x_2)$ , which is

$x_1$	$x_2$	$x_1 + x_2$	$(x_1 + x_2)'$
1	1	1	0
1	0	1	0
0	1	1	0
0	0	0	1

If we assign the value 1 to  $x_1$  and the value 0 to  $x_2$ , then,  $\alpha(1, 0) = (1 + 0)' = 1' = 0$ . Similarly,  $\alpha(1, 1) = 0$ ,  $\alpha(0, 1) = 0$ , and  $\alpha(0, 0) = 1$ . Also, there are only four possible assignments of values 1 and 0 to  $x_1$  and  $x_2$ .

Similarly, the truth table for  $\beta(x_1, x_2)$  is:

$x_1$	$x_2$	$x'_1$	$x'_2$	$x'_1 \cdot x'_2$
1	1	0	0	0
1	0	0	1	0
0	1	1	0	0
0	0	1	1	1

It follows that  $\alpha(1, 1) = \beta(1, 1)$ ,  $\alpha(1, 0) = \beta(1, 0)$ ,  $\alpha(0, 1) = \beta(0, 1)$ , and  $\alpha(0, 0) = \beta(0, 0)$ . Hence,  $\alpha$  and  $\beta$  are equal.

**REMARK 12.1.11** ► If two Boolean expressions  $\alpha$  and  $\beta$  are equal, then we write  $\alpha = \beta$ .

The following theorem lists various properties of Boolean variables.

**Theorem 12.1.12:** Let  $x$ ,  $x_1$ ,  $x_2$ , and  $x_3$  be Boolean variables. Then the following assertion holds:

- (i) *Commutative laws:*  $x_1 + x_2 = x_2 + x_1$  and  $x_1 \cdot x_2 = x_2 \cdot x_1$
- (ii) *Associative laws:*  $(x_1 + x_2) + x_3 = x_1 + (x_2 + x_3)$  and  $(x_1 \cdot x_2) \cdot x_3 = x_1 \cdot (x_2 \cdot x_3)$
- (iii) *Distributive laws:*  $x_1 \cdot (x_2 + x_3) = (x_1 \cdot x_2) + (x_1 \cdot x_3)$  and  $x_1 + (x_2 \cdot x_3) = (x_1 + x_2) \cdot (x_1 + x_3)$
- (iv) *Idempotent laws:*  $x + x = x$  and  $x \cdot x = x$
- (v) *Identity laws:*  $x + 0 = x$  and  $x \cdot 1 = x$
- (vi) *Inverse laws:*  $x + x' = 1$  and  $x \cdot x' = 0$
- (vii) *Dominance laws:*  $x + 1 = 1$  and  $x \cdot 0 = 0$
- (viii) *Absorption laws:*  $x_1 + x_1 \cdot x_2 = x_1$  and  $x_1 \cdot (x_1 + x_2) = x_1$
- (ix) *DeMorgan's laws:*  $(x_1 + x_2)' = x'_1 \cdot x'_2$  and  $(x_1 \cdot x_2)' = x'_1 + x'_2$
- (x) *Double complementation law:*  $(x')' = x$

**Proof:** We only prove (i), (ii), and (vii) and leave the others as exercises.

- (i) Let  $\alpha(x_1, x_2) = x_1 + x_2$  and  $\beta(x_1, x_2) = x_2 + x_1$ . We prove that for any assignment of values 0 or 1 to  $x_1$  and  $x_2$ , the value of  $\alpha$  is the same as that of  $\beta$ . We verify it by the following truth table:

$x_1$	$x_2$	$x_1 + x_2$	$x_2 + x_1$
1	1	1	1
1	0	1	1
0	1	1	1
0	0	0	0

From the table we find that  $\alpha(1, 1) = \beta(1, 1)$ ,  $\alpha(1, 0) = \beta(1, 0)$ ,  $\alpha(0, 1) = \beta(0, 1)$ , and  $\alpha(0, 0) = \beta(0, 0)$ . Hence,  $\alpha = \beta$ .

Similarly, we can prove  $x_1 \cdot x_2 = x_2 \cdot x_1$ .

- (ii) Let  $\alpha(x_1, x_2, x_3) = (x_1 \cdot x_2) \cdot x_3$  and  $\beta(x_1, x_2, x_3) = x_1 \cdot (x_2 \cdot x_3)$ . Consider the following truth table:

$x_1$	$x_2$	$x_3$	$x_2 \cdot x_3$	$x_1 \cdot x_2$	$(x_1 \cdot x_2) \cdot x_3$	$x_1 \cdot (x_2 \cdot x_3)$
1	1	1	1	1	1	1
1	1	0	0	1	0	0
1	0	1	0	0	0	0
1	0	0	0	0	0	0
0	1	1	1	0	0	0
0	1	0	0	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0

Hence,  $\alpha(b_1, b_2, b_3) = \beta(b_1, b_2, b_3)$  for any assignment of values  $b_1, b_2, b_3 \in \{0, 1\}$  to  $x_1, x_2$ , and  $x_3$ , respectively. Hence,  $\alpha = \beta$ . This proves that  $(x_1 \cdot x_2) \cdot x_3 = x_1 \cdot (x_2 \cdot x_3)$ .

- (vii) Let  $\alpha(x) = x + 1$  and  $\beta = 1$ . Consider the following truth table:

$x$	$\alpha = x + 1$	$\beta = 1$
1	1	1
0	1	1

This implies that  $\alpha = \beta$ . Similarly,  $x \cdot 0 = 0$ . ■

**Convention:** To avoid using so many parentheses in writing Boolean expressions we adopt the following convention: If a Boolean expression contains only one of · or + several times, sometimes we omit the parentheses. For example,  $(x_1 \cdot x_2) \cdot x_3$  is written  $x_1 \cdot x_2 \cdot x_3$  and  $(x_1 + x_2) + ((x_3 + x_4) + x_2)$  is written  $x_1 + x_2 + x_3 + x_4 + x_2$ .

**Theorem 12.1.13:** Let  $\alpha, \beta, \gamma$  be Boolean expressions in the Boolean variables  $x_1, x_2, \dots, x_n$ . Then,

- (i) *Commutative laws:*  $\alpha + \beta = \beta + \alpha$  and  $\alpha \cdot \beta = \beta \cdot \alpha$
- (ii) *Associative laws:*  $(\alpha + \beta) + \gamma = \alpha + (\beta + \gamma)$  and  $(\alpha \cdot \beta) \cdot \gamma = \alpha \cdot (\beta \cdot \gamma)$
- (iii) *Distributive laws:*  $\alpha \cdot (\beta + \gamma) = (\alpha \cdot \beta) + (\alpha \cdot \gamma)$  and  $\alpha + (\beta \cdot \gamma) = (\alpha + \beta) \cdot (\alpha + \gamma)$
- (iv) *Idempotent laws:*  $\alpha + \alpha = \alpha$  and  $\alpha \cdot \alpha = \alpha$
- (v) *Identity laws:*  $\alpha + 0 = \alpha$  and  $\alpha \cdot 1 = \alpha$
- (vi) *Inverse laws:*  $\alpha + \alpha' = 1$  and  $\alpha \cdot \alpha' = 0$
- (vii) *Dominance laws:*  $\alpha + 1 = 1$  and  $\alpha \cdot 0 = 0$

- (viii) *Absorption laws:*  $\alpha + (\alpha \cdot \beta) = \alpha$  and  $\alpha \cdot (\alpha + \beta) = \alpha$
- (ix) *DeMorgan's laws:*  $(\alpha + \beta)' = \alpha' \cdot \beta'$  and  $(\alpha \cdot \beta)' = \alpha' + \beta'$
- (x) *Double complementation law:*  $(\alpha')' = \alpha$

**Proof:** We only prove  $\alpha + \beta = \beta + \alpha$  and leave the others as exercises.

Now  $\alpha + \beta$  is a Boolean expression in the Boolean variables  $x_1, x_2, \dots, x_n$ . Because each  $x_i$  is a Boolean variable, we can assign the values 0, 1 to the variables  $x_1, x_2, \dots, x_n$ . Let  $b_1, b_2, \dots, b_n$  be an arbitrary assignment of the values to  $x_1, x_2, \dots, x_n$ , respectively, where  $b_i$  is 0 or 1. For this assignment, suppose the values of  $\alpha$  and  $\beta$  are  $a$  and  $b$ , respectively. Now  $a + b = b + a$  when  $a, b \in \{0, 1\}$ . Hence,  $(\alpha + \beta)(b_1, b_2, \dots, b_n) = (\beta + \alpha)(b_1, b_2, \dots, b_n)$ . This implies that  $\alpha + \beta = \beta + \alpha$ . ■

**DEFINITION 12.1.14** ▶ Let  $\alpha$  be a Boolean expression. A Boolean expression  $\beta$  is said to be a **dual** of  $\alpha$  if  $\beta$  is obtained from  $\alpha$  by replacing each occurrence of a Boolean sum by a Boolean product, replacing each occurrence of a Boolean product by a Boolean sum, replacing each occurrence of 1 by 0, and replacing each occurrence of 0 by 1. If  $\beta$  is a dual of  $\alpha$ , then we write  $\beta = \alpha^d$ .

#### EXAMPLE 12.1.15

Consider the Boolean expression  $\alpha = (x_1 + 1) \cdot x'_2 + x_3$ . Then  $\alpha^d = ((x_1 \cdot 0) + x'_2) \cdot x_3$ .

Let  $x, x_1, x_2$ , and  $x_3$  be Boolean variables. From Theorem 12.1.12, it follows that

- (i) if  $\alpha = x_1 + x_2$  and  $\beta = x_2 + x_1$ , then  $\alpha = \beta$  and also  $\alpha^d = \beta^d$ ;
- (ii) if  $\alpha = (x_1 + x_2) + x_3$  and  $\beta = x_1 + (x_2 + x_3)$ , then  $\alpha = \beta$  and also  $\alpha^d = \beta^d$ ;
- (iii) if  $\alpha = x_1 \cdot (x_2 + x_3)$  and  $\beta = x_1 \cdot x_2 + x_1 \cdot x_3$ , then  $\alpha = \beta$  and also  $\alpha^d = \beta^d$ ;
- (iv)  $x + x = x$  and  $(x + x)^d = x^d$ ;
- (v)  $x + 0 = x$  and  $(x + 0)^d = x^d$ ;
- (vi)  $x + 1 = 1$  and  $(x + 1)^d = 1^d$ ;
- (vii)  $x_1 + x_1 \cdot x_2 = x_1$  and  $(x_1 + x_1 \cdot x_2)^d = x_1^d$ ;
- (viii)  $(x_1 + x_2)' = x'_1 \cdot x'_2$  and  $((x_1 + x_2)')^d = (x'_1 \cdot x'_2)^d$ .

In general, we can prove that if  $\alpha = \beta$  for any two Boolean expressions  $\alpha$  and  $\beta$ , then  $\alpha^d = \beta^d$ . This result is known as the **duality principle**.

Next, we introduce the notion of Boolean functions and establish the relationship between Boolean expressions and Boolean functions.

Let  $n$  be a positive integer and  $B$  be a two-element Boolean algebra. Then  $B^n$  denotes the set of all  $n$ -tuples over  $B$ , i.e.,

$$B^n = \{(b_1, b_2, \dots, b_n) \mid b_i \in B\}.$$

A function  $f$  on  $B$  in  $n$  variables is a function  $f : B^n \rightarrow B$ . The  $n$  variables are emphasized by writing  $f(x_1, x_2, \dots, x_n)$ , where each  $x_i$ ,  $1 \leq i \leq n$  is a variable on  $B$ ; i.e., each  $x_i$  is a Boolean variable.

Let  $\alpha(x_1, x_2, x_3) = x_1 + x_2 \cdot x_3$  be a Boolean expression in three variables. We can define the function  $f : B^3 \rightarrow B$  corresponding to  $\alpha$  as follows: For all  $(b_1, b_2, b_3) \in B^3$  define

$$f(b_1, b_2, b_3) = b_1 + b_2 \cdot b_3.$$

For example,

$$\begin{aligned}f(1, 0, 1) &= 1 + 0 \cdot 1 = 1 + 0 = 1, \\f(0, 0, 1) &= 0 + 0 \cdot 1 = 0 + 0 = 0, \\f(1, 1, 0) &= 1 + 1 \cdot 0 = 1 + 0 = 1.\end{aligned}$$

**DEFINITION 12.1.16** ▶ Let  $B$  be a two-element Boolean algebra and let  $\alpha(x_1, x_2, \dots, x_n)$  be a Boolean expression. Then the corresponding function  $f_\alpha$  is a function  $f_\alpha : B^n \rightarrow B$  defined by  $f_\alpha(b_1, b_2, \dots, b_n) = \alpha(b_1, b_2, \dots, b_n)$  for each  $n$ -tuple  $(b_1, b_2, \dots, b_n) \in B^n$ .

**EXAMPLE 12.1.17**

Let  $\alpha(x_1, x_2, x_3) = (x_1 + x_2) \cdot x_3$  be a Boolean expression in three variables. Then the corresponding Boolean function  $f_\alpha : B^3 \rightarrow B$  is defined by

$$f_\alpha(b_1, b_2, b_3) = \alpha(b_1, b_2, b_3) = (b_1 + b_2) \cdot b_3$$

for all  $(b_1, b_2, b_3) \in B^3$ . We can describe the function  $f_\alpha(x_1, x_2, x_3)$  by the following table:

$x_1$	$x_2$	$x_3$	$x_1 + x_2$	$\alpha = (x_1 + x_2) \cdot x_3$	$f_\alpha(x_1, x_2, x_3)$
1	1	1	1	1	1
1	1	0	1	0	0
1	0	1	1	1	1
1	0	0	1	0	0
0	1	1	1	1	1
0	1	0	1	0	0
0	0	1	0	0	0
0	0	0	0	0	0

**DEFINITION 12.1.18** ▶ A function  $f : B^n \rightarrow B$  is called a **Boolean function** if there exists a Boolean expression  $\alpha$  such that  $f = f_\alpha$ .

We find that for every Boolean expression  $\alpha(x_1, x_2, \dots, x_n)$  there corresponds the function  $f_\alpha : B^n \rightarrow B$  defined by  $f_\alpha(b_1, b_2, \dots, b_n) = \alpha(b_1, b_2, \dots, b_n)$ . Now we consider the following question:

Is every function  $f : B^n \rightarrow B$  a Boolean function?

To answer this question we introduce some special types of Boolean expressions.

**DEFINITION 12.1.19** ▶ A Boolean expression  $\alpha(x_1, x_2, \dots, x_n)$  is said to be a **minterm** in the variables  $x_1, x_2, \dots, x_n$  if it is of the form

$$\tilde{x}_1 \cdot \tilde{x}_2 \cdot \tilde{x}_3 \cdots \tilde{x}_n,$$

where each  $\tilde{x}_i$  is either  $x_i$  or  $x'_i$ .

**EXAMPLE 12.1.20**

The Boolean expressions  $x_1 \cdot x_2 \cdot x_3$ ,  $x'_1 \cdot x'_2 \cdot x_3$ , and  $x_1 \cdot x_2 \cdot x_3$  are minterms in the variables  $x_1$ ,  $x_2$ , and  $x_3$ .

**DEFINITION 12.1.21** ► A Boolean expression  $\alpha(x_1, x_2, \dots, x_n)$  is said to be in **disjunctive normal form (DNF)**, or **sum-of-product form**, in the variables  $x_1, x_2, \dots, x_n$  if there are minterms  $\alpha_1, \alpha_2, \dots, \alpha_m$  in the variables  $x_1, x_2, \dots, x_n$  such that  $\alpha = \alpha_1 + \alpha_2 + \dots + \alpha_m$ .

Consider the Boolean expression  $\alpha = (x_1 + x'_2) \cdot x_3$  and its truth table.

Row	$x_1$	$x_2$	$x_3$	$x'_2$	$x_1 + x'_2$	$\alpha = (x_1 + x'_2) \cdot x_3$
1	1	1	1	0	1	1
2	1	1	0	0	1	0
3	1	0	1	1	1	1
4	1	0	0	1	1	0
5	0	1	1	0	0	0
6	0	1	0	0	0	0
7	0	0	1	1	1	1
8	0	0	0	1	1	0

Note that

$$\alpha(1, 1, 1) = 1 \quad (\text{row 1})$$

$$\alpha(1, 0, 1) = 1 \quad (\text{row 3})$$

$$\alpha(0, 0, 1) = 1 \quad (\text{row 7})$$

Corresponding to row 1, let  $\alpha_1(x_1, x_2, x_3) = x_1 \cdot x_2 \cdot x_3$ . Then  $\alpha_1(1, 1, 1) = 1$  and  $\alpha_1(b_1, b_2, b_3) = 0$  for any assignment  $(b_1, b_2, b_3) \neq (1, 1, 1)$ .

Corresponding to row 3, let  $\alpha_2(x_1, x_2, x_3) = x_1 \cdot x'_2 \cdot x_3$ . Then  $\alpha_2(1, 0, 1) = 1 \cdot 0' \cdot 1 = 1 \cdot 1 \cdot 1 = 1$  and  $\alpha_2(b_1, b_2, b_3) = 0$  for any assignment  $(b_1, b_2, b_3) \neq (1, 0, 1)$ .

Next, in row 7,  $\alpha(0, 0, 1) = 1$ . Let  $\alpha_3(x_1, x_2, x_3) = x'_1 \cdot x'_2 \cdot x_3$ , then we find that  $\alpha_3(0, 0, 1) = 0' \cdot 0' \cdot 1 = 1 \cdot 1 \cdot 1 = 1$  and  $\alpha_3(b_1, b_2, b_3) = 0$  for any assignment  $(b_1, b_2, b_3) \neq (0, 0, 1)$ .

Each of  $\alpha_1$ ,  $\alpha_2$ , and  $\alpha_3$  is a minterm in the variables  $x_1$ ,  $x_2$ , and  $x_3$ . Let  $\beta = \alpha_1 + \alpha_2 + \alpha_3$ . Then  $\beta$  is a Boolean expression in the disjunctive normal form.

Let us verify that  $\alpha = \beta$ . For this we consider the truth table:

Row	$x_1$	$x_2$	$x_3$	$\alpha$	$\alpha_1$	$\alpha_2$	$\alpha_3$	$\beta$
1	1	1	1	1	1	0	0	1
2	1	1	0	0	0	0	0	0
3	1	0	1	1	0	1	0	1
4	1	0	0	0	0	0	0	0
5	0	1	1	0	0	0	0	0
6	0	1	0	0	0	0	0	0
7	0	0	1	1	0	0	1	1
8	0	0	0	0	0	0	0	0

From this table it follows that  $\alpha = \beta = \alpha_1 + \alpha_2 + \alpha_3$ .

This shows that  $\alpha$  can be expressed in DNF.

**Theorem 12.1.22:** Let  $\alpha$  be a Boolean expression in the variables  $x_1, x_2, \dots, x_n$ . Suppose that  $\alpha_1, \alpha_2, \dots, \alpha_m$  are minterms in the variables  $x_1, x_2, \dots, x_n$  such that  $\alpha = \alpha_1 + \alpha_2 + \dots + \alpha_m$ . Then

- (i) for any assignment  $x_1 = b_1, x_2 = b_2, \dots, x_j = b_j, \dots, x_n = b_n$  of values  $b_1, b_2, \dots, b_n \in \{0, 1\}$ ,  $\alpha_i(b_1, b_2, \dots, b_j, \dots, b_n) = 0$  if and only if  $\tilde{x}_j(b_j) = 0$  for some  $j$ ,  $1 \leq j \leq n$ ; where  $\alpha_i = \tilde{x}_1 \cdot \tilde{x}_2 \cdot \tilde{x}_3 \cdots \tilde{x}_n$  is a minterm in the variables  $x_1, x_2, \dots, x_n$ . (Note that  $\tilde{x}_j(b_j)$  denotes the value of  $\tilde{x}_j$  when  $x_j$  is replaced by  $b_j$ .)
- (ii) for any assignment  $x_1 = b_1, x_2 = b_2, \dots, x_j = b_j, \dots, x_n = b_n$  of values  $b_1, b_2, \dots, b_n \in \{0, 1\}$ ,  $\alpha(b_1, b_2, \dots, b_n) = 1$  if and only if  $\alpha_i(b_1, b_2, \dots, b_n) = 1$  for some  $i$ , where  $1 \leq i \leq m$ .

**Proof:**

- (i) Because  $1 \cdot 0 = 0 \cdot 1 = 0 \cdot 0 = 0$ , part (i) follows.
- (ii) Because  $1 + 0 = 0 + 1 = 1 + 1 = 1$ , part (ii) follows. ■

We leave the proof of the following theorem as an exercise.

**Theorem 12.1.23:** Let  $\alpha$  be a Boolean expression in the variables  $x_1, x_2, \dots, x_n$  such that  $\alpha \neq 0$ . Then there exist minterms  $\alpha_1, \alpha_2, \dots, \alpha_m$  in the variables  $x_1, x_2, \dots, x_n$  such that  $\alpha = \alpha_1 + \alpha_2 + \dots + \alpha_m$ .

The following example further illustrates how to construct the minterms  $\alpha_1, \alpha_2, \dots, \alpha_m$  corresponding to  $\alpha$ .

**EXAMPLE 12.1.24**

Let  $\alpha(x_1, x_2, x_3) = x_1 \cdot x'_2 + x_2 \cdot (x'_1 + x_3)$ . Let us construct the truth table for all possible assignments of values 1 or 0 to  $x_1, x_2$ , and  $x_3$ . There are eight possible assignments of values 1 or 0 to  $x_1, x_2$ , and  $x_3$  in  $\alpha$ . Hence,

Row	$x_1$	$x_2$	$x_3$	$x'_2$	$x_1 \cdot x'_2$	$x'_1$	$x'_1 + x_3$	$x_2 \cdot (x'_1 + x_3)$	$\alpha(x_1, x_2, x_3)$
1	1	1	1	0	0	0	1	1	1
2	1	1	0	0	0	0	0	0	0
3	1	0	1	1	1	0	1	0	1
4	1	0	0	1	1	0	0	0	1
5	0	1	1	0	0	1	1	1	1
6	0	1	0	0	0	1	1	1	1
7	0	0	1	1	0	1	1	0	0
8	0	0	0	1	0	1	1	0	0

Next we consider only those rows in which the value of  $\alpha$  is 1. Here these rows are 1, 3, 4, 5, and 6. For each of these rows we construct the minterm

$$\tilde{x}_1 \cdot \tilde{x}_2 \cdot \tilde{x}_3 \cdots \tilde{x}_n$$

satisfying the condition

$$\tilde{x}_i = \begin{cases} x_i, & \text{if } x_i = 1, \\ x'_i, & \text{if } x_i = 0. \end{cases}$$

For row 1, the minterm  $\alpha_1 = x_1 \cdot x_2 \cdot x_3$ .

For row 3, the minterm  $\alpha_3 = x_1 \cdot x'_2 \cdot x_3$ .

For row 4, the minterm  $\alpha_4 = x_1 \cdot x'_2 \cdot x'_3$ .

For row 5, the minterm  $\alpha_5 = x'_1 \cdot x_2 \cdot x_3$ .

For row 6, the minterm  $\alpha_6 = x'_1 \cdot x_2 \cdot x'_3$ .

$$\text{Then } \alpha = \alpha_1 + \alpha_3 + \alpha_4 + \alpha_5 + \alpha_6 = x_1 \cdot x_2 \cdot x_3 + x_1 \cdot x'_2 \cdot x_3 + x_1 \cdot x'_2 \cdot x'_3 + x'_1 \cdot x_2 \cdot x_3 + x'_1 \cdot x_2 \cdot x'_3.$$

In the next example, we illustrate how to show that a function  $f : B^n \rightarrow B$  is a Boolean function.

### EXAMPLE 12.1.25

Let  $f : B^3 \rightarrow B$  be a function. We know that  $f$  is completely known if we know the image of every triple  $(b_1, b_2, b_3) \in B^3$  under  $f$ . Suppose that  $f(x_1, x_2, x_3)$  is given by the following table:

Row	$x_1$	$x_2$	$x_3$	$f(x_1, x_2, x_3)$
1	1	1	1	0
2	1	1	0	1
3	1	0	1	1
4	1	0	0	0
5	0	1	1	0
6	0	1	0	1
7	0	0	1	1
8	0	0	0	0

Note that the function  $f(x_1, x_2, x_3)$  takes the value 1 for assignments 1 or 0 to  $x_1, x_2, x_3$  in the second, third, sixth and seventh rows of the table. For each of these rows we construct the minterm  $\tilde{x}_1 \cdot \tilde{x}_2 \cdot \tilde{x}_3$  satisfying the following conditions:

$$\tilde{x}_i = \begin{cases} x_i, & \text{if } x_i = 1, \\ x'_i, & \text{if } x_i = 0. \end{cases}$$

For the second row, the minterm is  $\alpha_2 = x_1 \cdot x_2 \cdot x'_3$ .

For the third row, the minterm is  $\alpha_3 = x_1 \cdot x'_2 \cdot x_3$ .

For the sixth row, the minterm is  $\alpha_6 = x'_1 \cdot x_2 \cdot x'_3$ .

For the seventh row, the minterm is  $\alpha_7 = x'_1 \cdot x'_2 \cdot x_3$ .

Let  $\alpha = \alpha_2 + \alpha_3 + \alpha_6 + \alpha_7$ .

We note that  $\alpha_2(b_1, b_2, b_3) = 1$  if and only if  $b_1 = 1, b_2 = 1$ , and  $b_3 = 0$ . In other words,  $\alpha_2(1, 1, 0) = 1$  and  $\alpha_2(b_1, b_2, b_3) = 0$  if  $(b_1, b_2, b_3) \neq (1, 1, 0)$ .

Similarly,  $\alpha_3(b_1, b_2, b_3) = 1$  if and only if  $(b_1, b_2, b_3) = (1, 0, 1)$ ,  $\alpha_6(b_1, b_2, b_3) = 1$  if and only if  $(b_1, b_2, b_3) = (0, 1, 0)$ , and  $\alpha_7(b_1, b_2, b_3) = 1$  if and only if  $(b_1, b_2, b_3) = (0, 0, 1)$ .

We now define  $f_\alpha : B^3 \rightarrow B$  by

$$\begin{aligned} f_\alpha(b_1, b_2, b_3) &= \alpha(b_1, b_2, b_3) \\ &= \alpha_2(b_1, b_2, b_3) + \alpha_3(b_1, b_2, b_3) + \alpha_6(b_1, b_2, b_3) + \alpha_7(b_1, b_2, b_3). \end{aligned}$$

Now  $f_\alpha(1, 1, 1) = \alpha(1, 1, 1) = 0$ , because

$$\alpha_2(1, 1, 1) = \alpha_3(1, 1, 1) = \alpha_6(1, 1, 1) = \alpha_7(1, 1, 1) = 0.$$

Similarly,

$$\begin{aligned} f_\alpha(1, 1, 0) &= \alpha(1, 1, 0) = 1 \\ f_\alpha(1, 0, 1) &= 1 \\ f_\alpha(1, 0, 0) &= 0 \\ f_\alpha(0, 1, 1) &= 0 \\ f_\alpha(0, 1, 0) &= 1 \\ f_\alpha(0, 0, 1) &= 1 \\ f_\alpha(0, 0, 0) &= 0 \end{aligned}$$

It follows that  $f = f_\alpha$ . Hence,  $f$  is a Boolean function.

Proceeding as in Example 12.1.25, we can establish the following theorem.

**Theorem 12.1.26:** Let  $f : B^n \rightarrow B$  be a function such that  $f \neq 0$ .

Then there exists a Boolean expression  $\alpha$  in DNF such that  $f(b_1, b_2, \dots, b_n) = \alpha(b_1, b_2, \dots, b_n)$  for all  $(b_1, b_2, \dots, b_n) \in B^n$ .

**Proof:** Because  $f \neq 0$ , there exists an  $n$ -tuple  $(b_1, b_2, \dots, b_n) \in B^n$  such that  $f(b_1, b_2, \dots, b_n) = 1$ . Let

$$T = \{(b_1, b_2, \dots, b_n) \in B^n \mid f(b_1, b_2, \dots, b_n) = 1\}.$$

Now  $T \neq \emptyset$ . Suppose that the number of elements in  $T$  is  $k$ , where  $1 \leq k \leq 2^n$ . Let  $\gamma_1, \gamma_2, \dots, \gamma_k$  be the elements of  $T$ . For each  $\gamma_i = (b_{i1}, b_{i2}, \dots, b_{in}) \in T$ , define a Boolean expression  $\alpha_i = \tilde{x}_1 \cdot \tilde{x}_2 \cdot \tilde{x}_3 \cdots \tilde{x}_n$ , where

$$\tilde{x}_j = \begin{cases} x_j, & \text{if } b_{ij} = 1, \\ x'_j, & \text{if } b_{ij} = 0. \end{cases}$$

Now for each  $\gamma_i$  we get a minterm  $\alpha_i$ . From the definition of  $\gamma_i$ , it follows that

$$\alpha_i(\gamma_i) = \alpha_i(b_{i1}, b_{i2}, \dots, b_{in}) = 1$$

and

$$\alpha_i(b_1, b_2, \dots, b_n) = 0 \quad \text{if } (b_1, b_2, \dots, b_n) \neq \gamma_i.$$

Now  $\alpha = \alpha_1 + \alpha_2 + \cdots + \alpha_k$  is a Boolean expression in DNF in the variables  $x_1, x_2, \dots, x_n$ . We now show that for any  $(b_1, b_2, \dots, b_n) \in B^n$ ,

$$f(b_1, b_2, \dots, b_n) = \alpha(b_1, b_2, \dots, b_n).$$

Let  $(b_1, b_2, \dots, b_n) \in B^n$ . Suppose that  $f(b_1, b_2, \dots, b_n) = 1$ . Then  $(b_1, b_2, \dots, b_n) = \gamma_i$  for some  $\gamma_i \in T$ . Therefore,  $\alpha_i(\gamma_i) = 1$ . Thus,  $\alpha(\gamma_i) = 1$ . This implies that  $f(\gamma_i) = \alpha(\gamma_i)$ .

On the other hand, suppose that  $f(b_1, b_2, \dots, b_n) = 0$ . Then  $(b_1, b_2, \dots, b_n) \notin T$ . This implies that  $\alpha_i(b_1, b_2, \dots, b_n) = 0$  for each  $\alpha_1, \alpha_2, \dots, \alpha_k$ . Therefore,  $\alpha(b_1, b_2, \dots, b_n) = 0$ . Thus,  $f(b_1, b_2, \dots, b_n) = \alpha(b_1, b_2, \dots, b_n)$ .

Hence, it follows that for all  $(b_1, b_2, \dots, b_n) \in B^n$ ,  $f(b_1, b_2, \dots, b_n) = \alpha(b_1, b_2, \dots, b_n)$ . ■

**Corollary 12.1.27:** Every function  $f : B^n \rightarrow B$  is a Boolean function.

**Proof:** If  $f = 0$ , then  $f$  is defined by the Boolean expression zero. If  $f \neq 0$ , then the result follows from Theorem 12.1.26. ■

**EXAMPLE 12.1.28**

Let  $f : B^2 \rightarrow B$  be defined by

$x_1$	$x_2$	$f(x_1, x_2)$
1	1	1
1	0	0
0	1	0
0	0	1

Note that for  $\gamma_1 = (1, 1)$  and  $\gamma_2 = (0, 0)$ ,  $f(\gamma_1) = 1$  and  $f(\gamma_2) = 1$ .

Let  $\alpha_1$  and  $\alpha_2$  be minterms such that  $\alpha_1 = x_1 \cdot x_2$  and  $\alpha_2 = x'_1 \cdot x'_2$ . Let  $\alpha = \alpha_1 + \alpha_2$ . Now

$$\begin{aligned}\alpha(1, 1) &= \alpha_1(1, 1) + \alpha_2(1, 1) = 1 \cdot 1 + 1' \cdot 1' = 1 + 0 \cdot 0 = 1 + 0 = 1 = f(1, 1), \\ \alpha(1, 0) &= \alpha_1(1, 0) + \alpha_2(1, 0) = 1 \cdot 0 + 1' \cdot 0' = 0 + 0 \cdot 1 = 0 + 0 = 0 = f(1, 0), \\ \alpha(0, 1) &= \alpha_1(0, 1) + \alpha_2(0, 1) = 0 \cdot 1 + 0' \cdot 1' = 0 + 1 \cdot 0 = 0 + 0 = 0 = f(0, 1), \\ \alpha(0, 0) &= \alpha_1(0, 0) + \alpha_2(0, 0) = 0 \cdot 0 + 0' \cdot 0' = 0 + 1 \cdot 1 = 0 + 1 = 1 = f(0, 0).\end{aligned}$$

Thus,  $f(b_1, b_2) = \alpha(b_1, b_2)$  for all  $(b_1, b_2) \in B^2$ . Hence,  $f = f_\alpha$ , where  $\alpha = \alpha_1 + \alpha_2 = x_1 \cdot x_2 + x'_1 \cdot x'_2$ .

Just as we can form minterms, we can also form maxterms.

**DEFINITION 12.1.29** ► A Boolean expression  $\alpha$  in the variables  $x_1, x_2, \dots, x_n$  is called a **maxterm** if

$$\alpha = \tilde{x}_1 + \tilde{x}_2 + \tilde{x}_3 + \cdots + \tilde{x}_n,$$

where each  $\tilde{x}_i$  denotes either  $x_i$  or  $x'_i$ .

**EXAMPLE 12.1.30**

The expressions  $x_1 + x'_2 + x'_3$ ,  $x'_1 + x_2 + x_3$ , and  $x'_1 + x'_2 + x'_3$  are examples of the maxterms in the variables  $x_1, x_2$ , and  $x_3$ .

**DEFINITION 12.1.31** ► A Boolean expression  $\alpha(x_1, x_2, \dots, x_n)$  is said to be in **conjunctive normal form (CNF)** if there exist maxterms  $\alpha_1, \alpha_2, \dots, \alpha_m$  in the variables  $x_1, x_2, \dots, x_n$  such that

$$\alpha = \alpha_1 \cdot \alpha_2 \cdots \alpha_m.$$

**EXAMPLE 12.1.32**

Let  $\alpha = (x_1 + x_2) \cdot (x'_1 + x_2) \cdot (x_1 + x'_2)$ . Then  $\alpha$  is a Boolean expression in CNF in the variables  $x_1$  and  $x_2$ .

We leave the proof of the following theorem as an exercise.

**Theorem 12.1.33:** Let  $\alpha$  be a Boolean expression in the variables  $x_1, x_2, \dots, x_n$  such that  $\alpha \neq 1$ . Then there exist maxterms  $\alpha_1, \alpha_2, \dots, \alpha_m$  in the variables  $x_1, x_2, \dots, x_n$  such that  $\alpha = \alpha_1 \cdot \alpha_2 \cdots \alpha_m$ .

## WORKED-OUT EXERCISES

**Exercise 1:** Construct the truth tables for the Boolean expressions.

(a)  $x \cdot (y + x')$       (b)  $x \cdot y' + y \cdot (x' + z)$

**Solution:**

(a) The truth table for the Boolean expression  $x \cdot (y + x')$  is given by the following table:

x	y	$x'$	$y + x'$	$x \cdot (y + x')$
1	1	0	1	1
1	0	0	0	0
0	1	1	1	0
0	0	1	1	0

(b) The truth table for  $x \cdot y' + y \cdot (x' + z)$  is given by the following table:

x	y	z	$x'$	$y'$	$x \cdot y'$	$x' + z$	$y \cdot (x' + z)$	$x \cdot y' + y \cdot (x' + z)$
1	1	0	0	0	0	1	1	1
1	1	0	0	0	0	0	0	0
1	0	1	0	1	0	1	0	1
1	0	0	0	1	0	0	0	1
0	1	1	0	0	0	1	1	1
0	1	0	1	0	0	1	1	1
0	0	1	1	1	0	0	0	0
0	0	0	1	0	1	0	0	0

**Exercise 2:** Consider the Boolean function  $f(x, y, z)$  given by the following table:

Row	x	y	z	$f(x, y, z)$
1	1	1	1	1
2	1	1	0	0
3	1	0	1	0
4	1	0	0	1
5	0	1	1	1
6	0	1	0	0
7	0	0	1	0
8	0	0	0	0

Find the Boolean expression  $\alpha$  in DNF such that  $f = f_\alpha$ .

**Solution:** The function  $f$  takes the value 1 for the assignments in the first, fourth, and fifth rows of the given table. The minterms corresponding to first, fourth, and fifth rows are, respectively,  $x \cdot y \cdot z$ ,  $x \cdot y' \cdot z'$  and  $x' \cdot y \cdot z$ .

Let  $\alpha = x \cdot y \cdot z + x \cdot y' \cdot z' + x' \cdot y \cdot z$ . Then  $\alpha$  is a Boolean expression in DNF in three variables. Note that  $f = f_\alpha$ .

**Exercise 3:** Consider the Boolean function  $f(x, y, z)$  given by the following table:

Row	x	y	z	$f(x, y, z)$
1	1	1	1	0
2	1	1	0	1
3	1	0	1	0
4	1	0	0	1
5	0	1	1	0
6	0	1	0	0
7	0	0	1	0
8	0	0	0	1

Find the Boolean expression in DNF that represents  $f$ .

**Solution:** The function  $f$  takes the value 1 for the assignments in the second, fourth, and eighth rows of the given table. The minterms corresponding to these rows are, respectively,  $x \cdot y \cdot z'$ ,  $x \cdot y' \cdot z'$ , and  $x' \cdot y' \cdot z'$ . Hence, the Boolean expression in DNF representing the function  $f$  is  $x \cdot y \cdot z' + x \cdot y' \cdot z' + x' \cdot y' \cdot z'$ .

**Exercise 4:** Consider the Boolean function  $f(x, y, z)$  given by the following table:

Row	x	y	z	$f(x, y, z)$
1	1	1	1	1
2	1	1	0	1
3	1	0	1	1
4	1	0	0	0
5	0	1	1	1
6	0	1	0	1
7	0	0	1	1
8	0	0	0	0

Find the Boolean expression in CNF representing the Boolean function  $f$ .

**Solution:** The function  $f$  takes the value 0 for the assignments in the fourth and eighth rows of the given table. The maxterms corresponding to these rows are  $x' + y + z$  and  $x + y + z$ , respectively. Hence, the Boolean expression in CNF representing the function  $f$  is  $(x' + y + z) \cdot (x + y + z)$ .

**Exercise 5:** Consider the Boolean function  $f(x, y, z)$  given by the following table:

Row	x	y	z	$f(x, y, z)$
1	1	1	1	1
2	1	1	0	1
3	1	0	1	1
4	1	0	0	0
5	0	1	1	1
6	0	1	0	1
7	0	0	1	0
8	0	0	0	0

Find the Boolean expression in CNF for the Boolean function  $f$ .

**Solution:** The function  $f$  takes the value 0 for the assignments in the fourth, seventh, and eighth rows of the given table. The maxterms corresponding to these rows are, respectively,  $(x' + y + z)$ ,  $(x + y + z')$ , and  $(x + y + z)$ . Hence, the Boolean expression in CNF representing the function  $f$  is  $(x' + y + z) \cdot (x + y + z') \cdot (x + y + z)$ .

## SECTION REVIEW

### Key Terms

Boolean sum	Boolean expression in $n$ variables	disjunctive normal form (DNF)
Boolean product	equal Boolean expressions	sum-of-product form
Boolean complementation	dual	maxterm
two-element Boolean algebra	duality principle	conjunctive normal form (CNF)
Boolean variable	Boolean function	
Boolean expression	minterm	

### Some Key Definitions

1. The set  $B = \{0, 1\}$  together with Boolean sum,  $+$ , Boolean product,  $\cdot$ , and Boolean complementation,  $'$ , is called a two-element Boolean algebra. We denote this two-element Boolean algebra by  $B$ .
2. Any literal symbol such as  $x, y, z, x_1, x_2, \dots, x_n$  (with or without subscripts) used to represent an element of  $B = \{0, 1\}$  is called a Boolean variable.
3. Two Boolean expressions  $\alpha(x_1, x_2, \dots, x_n)$  and  $\beta(x_1, x_2, \dots, x_n)$  are said to be equal if they assume the same value for every assignment of values 0, 1 to the variables  $x_1, x_2, \dots, x_n$ .
4. A function  $f : B^n \rightarrow B$  is called a Boolean function if there exists a Boolean expression  $\alpha$  such that  $f = f_\alpha$ .
5. A Boolean expression  $\alpha(x_1, x_2, \dots, x_n)$  is said to be a minterm in the variables  $x_1, x_2, \dots, x_n$  if it is of the form  $\tilde{x}_1 \cdot \tilde{x}_2 \cdot \tilde{x}_3 \cdots \tilde{x}_n$ , where each  $\tilde{x}_i$  is either  $x_i$  or  $x'_i$ .
6. A Boolean expression  $\alpha(x_1, x_2, \dots, x_n)$  is said to be in disjunctive normal form (DNF), or sum-of-product form, in the variables  $x_1, x_2, \dots, x_n$  if there are minterms  $\alpha_1, \alpha_2, \dots, \alpha_m$  in the variables  $x_1, x_2, \dots, x_n$  such that  $\alpha = \alpha_1 + \alpha_2 + \cdots + \alpha_m$ .
7. A Boolean expression  $\alpha$  in the variables  $x_1, x_2, \dots, x_n$  is called a maxterm if  $\alpha = \tilde{x}_1 + \tilde{x}_2 + \tilde{x}_3 + \cdots + \tilde{x}_n$ , where each  $\tilde{x}_i$  denotes either  $x_i$  or  $x'_i$ .
8. A Boolean expression  $\alpha(x_1, x_2, \dots, x_n)$  is said to be in conjunctive normal form (CNF) if there exist maxterms  $\alpha_1, \alpha_2, \dots, \alpha_m$  in the variables  $x_1, x_2, \dots, x_n$  such that  $\alpha = \alpha_1 \cdot \alpha_2 \cdots \alpha_m$ .

## Key Result

- Let  $\alpha$  be a Boolean expression in the variables  $x_1, x_2, \dots, x_n$  such that  $\alpha \neq 0$ . Then there exist minterms  $\alpha_1, \alpha_2, \dots, \alpha_m$  in the variables  $x_1, x_2, \dots, x_n$  such that  $\alpha = \alpha_1 + \alpha_2 + \dots + \alpha_m$ .

## EXERCISES

In the following exercises +, ·, and' denote the Boolean sum, Boolean product, and Boolean complement, respectively, on  $B = \{0, 1\}$ .

- Find the values of  $\alpha(1, 0, 1)$ ,  $\alpha(0, 0, 1)$ , and  $\alpha(1, 1, 0)$ , where  $\alpha$  is the Boolean expression.
  - $\alpha(x_1, x_2, x_3) = x_1 \cdot x_2 + x'_1 + x_3$
  - $\alpha(x_1, x_2, x_3) = x_1 + x'_2 + x_3 + 1$
  - $\alpha(x_1, x_2, x_3) = (x_1 \cdot x_2 + x_2 \cdot x'_3 + x_3)'$
  - $\alpha(x_1, x_2, x_3) = x_1 \cdot x'_2 + x_2 \cdot x'_3 + x_3 \cdot x'_1$
- Find the value of the Boolean expressions for  $x_1 = 0$ ,  $x_2 = 1$ , and  $x_3 = 1$ .
  - $(x_1 \cdot x'_2 \cdot x_3) \cdot x_1 + x'_2 + x_3$
  - $(x_1 \cdot x'_2) \cdot x_1 + (x'_2 + x_3) \cdot (x'_2 + x_3)'$
- Construct a truth table for each of the Boolean expressions.
  - $x_1 \cdot x_2 + x'_1 \cdot x'_2$
  - $x_1 \cdot (x_1 + x'_2)$
  - $x_1 \cdot x_2 + x_1 \cdot x'_2 + x'_1 \cdot x_2$
  - $x_1 \cdot x_2 + x_1 \cdot x'_2 \cdot x'_3$
- For each of the Boolean expressions in Exercise 3, find the dual and then construct the truth table for each of the dual expressions.
- Show that the following Boolean expressions  $\alpha$  and  $\beta$  are equal.
  - $\alpha = x_1 + x_2 \cdot x'_2 + x_3$ ,  $\beta = x_1 + x_3$
  - $\alpha = (x_1 + x_2) \cdot (x_1 + x'_2) \cdot (x'_1 + x_3)$ ,  
 $\beta = x_1 \cdot x'_2 \cdot x_3 + x_1 \cdot x_2 \cdot x_3$
  - $\alpha = (x_1 + x_3) \cdot (x'_1 \cdot x'_2)', \beta = x_1 + x_2 \cdot x_3$
- Express the following Boolean expressions in DNF in the variables  $x_1$  and  $x_2$ .
  - $\alpha = x_1 + x_2 \cdot x_1$
  - $\alpha = (x_1 + x'_2) \cdot (x_1 + x_2)$
  - $\alpha = x_1 + 1$
  - $\alpha = x'_1 + x'_2 \cdot x'_1$
  - $\alpha = (x_1 + x_2) \cdot (x'_1 + x_2) \cdot (x_1 + x'_2)$
- Express the following Boolean expressions in CNF in the variables  $x_1$  and  $x_2$ .
  - $\alpha = x_1 + x_1 \cdot x_2$
  - $\alpha = x_1 \cdot x_2 + x_1 \cdot x'_2 + x_2 \cdot x_1$
  - $\alpha = x_1 \cdot x'_2 + x_2 \cdot x'_3 + 1$
  - $\alpha = (x_1 + x'_2)' \cdot x_3$
  - $\alpha = x_1 \cdot (x_2 + x'_3) + x_1$

- Express the following Boolean expressions in DNF in the variables  $x_1$ ,  $x_2$ , and  $x_3$ .

- $\alpha = (x_1 + x_2) \cdot (x_1 + x'_2) \cdot (x'_1 + x_3)$
- $\alpha = (x_1 + x_2 + x_3) \cdot (x_1 \cdot x_2 + x'_1 \cdot x_3)$
- $\alpha = (x_1 \cdot x'_2 + x_1 \cdot x_3)' + x'_1$
- $\alpha = x_1 \cdot x_2 + (x_1 + x_2) \cdot (x_1 + x_3)$
- $\alpha = x_1 \cdot x_2 + x_3 \cdot (x'_1 + x_3)$

- Express the following Boolean expressions in CNF in the variables  $x_1$ ,  $x_2$ , and  $x_3$ .

- $\alpha = x'_1 \cdot x_2 + x_3$
- $\alpha = x'_1 + (x_1 \cdot x'_2 + x_2 \cdot x_3)'$
- $\alpha = (x_1 + x_2) \cdot (x'_1 + x_3)$
- $\alpha = (x_1 + x_2) \cdot x'_3 + (x_1 + x_3) \cdot x'_2$
- $\alpha = (x_1 + x'_2)' + (x'_1 + x'_3)'$

- For each of Boolean expressions  $\alpha$ , define the corresponding functions  $f_\alpha : B^3 \rightarrow B$ .

- $\alpha = x_1 \cdot x_3 + x'_1 \cdot x_2$
- $\alpha = x_1 \cdot x_2 \cdot x_3 + x_1 \cdot x'_2 \cdot x_3$
- $\alpha = x_1 + x'_2 + x_3$
- $\alpha = x_1 \cdot x'_2 + x_2 \cdot x'_3 + 1$
- $\alpha = (x_1 + x'_2)' \cdot x_3$
- $\alpha = x_1 \cdot (x_2 + x'_3) + x_1$

- Find the Boolean expression in DNF for each of the Boolean functions given by the following tables.

a.	$x_1$	$x_2$	$f(x, y)$	b.	$x$	$y$	$z$	$f(x, y, z)$
	1	1	1		1	1	1	1
	1	0	1		1	1	0	1
	0	1	1		1	0	1	1
	0	0	0		1	0	0	1
					0	1	1	0
					0	1	0	0
					0	0	1	1
					0	0	0	1

c.	$x$	$y$	$z$	$f(x, y, z)$
	1	1	1	1
	1	1	0	1
	1	0	1	1
	1	0	0	0
	0	1	1	1
	0	1	0	0
	0	0	1	0
	0	0	0	0

12. Find the Boolean expression in CNF for each of the Boolean functions given by the following tables.

a.	$x_1$	$x_2$	$f(x_1, x_2)$
1	1	1	1
1	0	0	0
0	1	0	0
0	0	1	1

b.	$x$	$y$	$z$	$f(x, y, z)$
1	1	1	1	1
1	1	0	1	1
1	0	1	1	1
1	0	0	0	0
0	1	1	1	1
0	1	0	0	0
0	0	1	0	0
0	0	0	1	1

c.	$x$	$y$	$z$	$f(x, y, z)$
1	1	1	1	0
1	1	1	0	0
1	0	1	1	0
1	0	0	0	1
0	1	1	1	0
0	1	0	0	1
0	0	1	0	0
0	0	0	1	0
0	0	0	0	1

13. Prove the remaining parts of Theorem 12.1.12.

14. Show that there are  $2^{2^3}$  different Boolean functions in three variables on  $B$ .

## 12.2 BOOLEAN ALGEBRA

In the preceding section, we discussed two-element Boolean algebra and its basic properties in detail. As we remarked at the beginning of this chapter, Boolean algebra has numerous applications in the design of electronic circuits. In this section, we discuss Boolean algebra in general and describe its basic properties.

There are several definitions of a Boolean algebra available in the literature. Throughout this chapter, we shall adopt the definition given by Huntington in 1904.

**DEFINITION 12.2.1** ▶ A **Boolean algebra** is a nonempty set  $B$  together with two binary operations,  $+$ , and  $\cdot$ , on  $B$  (known as addition and multiplication, respectively) and a unary operation,  $'$ , on  $B$  (called the complementation) satisfying the following axioms.

(i) For all  $a$  and  $b$  in  $B$ ,

$$a + b = b + a \quad \text{commutative law for addition}$$

$$a \cdot b = b \cdot a \quad \text{commutative law for multiplication}$$

(ii) For all  $a, b, c \in B$ ,

$$a + (b \cdot c) = (a + b) \cdot (a + c) \quad \text{distributive law of } + \text{ over } \cdot$$

$$a \cdot (b + c) = (a \cdot b) + (a \cdot c) \quad \text{distributive law of } \cdot \text{ over } +$$

(iii)  $B$  contains distinct elements 0 and 1 (known as the **zero element** and the **unit element**) such that for all  $a \in B$ ,

$$a + 0 = a \quad \text{existence of additive identity}$$

$$a \cdot 1 = a \quad \text{existence of multiplicative identity}$$

(iv) For each  $a \in B$ , there exists an element  $a'$  in  $B$ , called the **complement**, or **negation**, of  $a$  in  $B$  such that

$$a + a' = 1, \quad a \cdot a' = 0 \quad \text{existence of complement}$$

Axioms (i), (ii), (iii), and (iv) of Definition 12.2.1 are called **Huntington's postulates for Boolean algebra**.

**REMARK 12.2.2** ▶

(i)  $(a')$ ' will be denoted by  $a''$  and so on. We usually write  $a \cdot b$  as  $ab$ .

(ii) The binary operations in the definition need not be written as  $+$  and  $\cdot$ . Instead of these symbols, we may use other symbols such as  $\cup$ ,  $\cap$  (union and intersection) or  $\vee$ ,  $\wedge$  (join and meet) to denote these operations.

- (iii) A Boolean algebra is usually denoted by a 6-tuple  $(B, +, \cdot, ', 0, 1)$ , or by  $(B, +, \cdot, ')$ , or simply by  $B$ .

**EXAMPLE 12.2.3**

Let  $S$  be any nonempty set and  $\mathcal{P}(S)$  be the set of all subsets of a set  $S$ . For any  $A, B \in \mathcal{P}(S)$ , let  $A + B = A \cup B$ ,  $A \cdot B = A \cap B$ , and  $A' = S - A$ . Then it can be shown that  $\mathcal{P}(S)$  becomes a Boolean algebra, where the empty set  $\emptyset$  and the set  $S$  are the additive identity element and the multiplicative identity element, respectively.

**EXAMPLE 12.2.4**

Let  $D_{30}$  be the set of all positive divisors of 30, i.e.,  $D_{30} = \{1, 2, 3, 5, 6, 10, 15, 30\}$ . For any  $a, b \in D_{30}$ , let  $a + b := \text{lcm}[a, b]$ ;  $a \cdot b = \gcd(a, b)$  and  $a' = \frac{30}{a}$ . Using the properties of lcm and gcd, it can be verified easily that  $(D_{30}, +, \cdot, ', 1, 30)$  is a Boolean algebra. Here 1 is the zero element and 30 is the unit element. Note that 30 is square-free; i.e., it is not divisible by the square of any integer greater than 1.

We can generalize the result of Example 12.2.4 as follows.

**Theorem 12.2.5:** Let  $n > 1$  be an integer and  $D_n$  be the set of positive integers that are divisors of  $n$ . For  $a, b \in D_n$ , define  $a + b = \text{lcm}[a, b]$ ;  $a \cdot b = \gcd(a, b)$  and  $a' = \frac{n}{a}$ . Then  $(D_n, +, \cdot, ', 1, n)$  is a Boolean algebra if and only if  $n$  is square-free; i.e.,  $n$  is not divisible by the square of any integer greater than 1.

**Proof:** Using the properties of integers and of lcm and gcd, we can easily show that axioms (1)–(3) of Definition 12.2.1 are satisfied.

Suppose that  $D_n$  is a Boolean algebra and  $n$  is divisible by the square of an integer greater than 1. Then there exists a prime integer  $p$  such that  $p^2$  divides  $n$ . This implies that  $n = p^2q$  for some integer  $q$ . Because  $p$  is a divisor of  $n$ , it follows that  $p \in D_n$ . Then the complement  $p'$  of  $p$  is in  $D_n$ . Also,  $\gcd(p, p') = 1$  and  $\text{lcm}[p, p'] = n$ . Thus,  $p' = \frac{n}{p} = pq$ . This implies that  $\gcd(p, p') = p > 1$ , a contradiction. Hence, if  $D_n$  is a Boolean algebra, then  $n$  is not divisible by the square of any integer greater than 1.

Conversely, suppose that  $n$  is an integer greater than 1 such that  $n$  is not divisible by the square of any integer greater than 1. Then the set  $D_n$  of all positive divisors of  $n$  forms a distributive lattice under the operations  $a + b = \text{lcm}[a, b]$ ,  $a \cdot b = \gcd(a, b)$  for all  $a, b \in D_n$ . Because  $1, n \in D_n$ , it follows that  $n$  is the multiplicative identity and 1 is the additive identity. Because  $n$  is square-free, it follows that for any  $a \in D_n$ ,  $a$  and  $\frac{n}{a}$  have no common factor other than 1. Then  $\gcd(a, \frac{n}{a}) = 1$  and  $\text{lcm}[a, \frac{n}{a}] = n$ . Hence,  $D_n$  is a Boolean algebra. ■

**REMARK 12.2.6** ▶ Let  $n = 50$ . Then  $n$  not square-free as  $5^2 = 25$  divides 50. Hence, by Theorem 12.2.5,  $(D_{50}, +, \cdot, ', 1, 50)$  is not a Boolean algebra.

**EXAMPLE 12.2.7**

**Two-Element Boolean Algebra.** Let  $B = \{0, 1\}$ . Then  $(B, +, \cdot, ', 0, 1)$  is a Boolean algebra, where  $+$ ,  $\cdot$ , and  $'$  are, respectively, the Boolean sum, Boolean product, and Boolean complementation. This two-element Boolean algebra is very useful in the theory of combinatorial circuits, as we will see in the next section.

Let us now describe some basic properties of a Boolean algebra. First, however, we introduce the following definitions.

**DEFINITION 12.2.8** ▶ By a **proposition** in a Boolean algebra we mean either a statement or an algebraic identity in the Boolean algebra.

For example, the statement “In a Boolean algebra 0 is unique” and the algebraic identities given in Definition 12.2.1 are all propositions.

**DEFINITION 12.2.9** ▶ By the **dual** of a proposition  $A$  in a Boolean algebra we mean the proposition obtained from  $A$  by interchanging  $+$  and  $\cdot$  and exchanging 0 and 1.

For example, the dual of the proposition

$$x \cdot (y + z) = (x \cdot y) + (x \cdot z)$$

is the proposition

$$x + (y \cdot z) = (x + y) \cdot (x + z)$$

and vice versa. The dual of the proposition “0 is unique in a Boolean algebra” is the proposition “1 is unique in a Boolean algebra.” Moreover, if a proposition  $B$  is the dual of a proposition  $A$ , then  $A$  is the dual of  $B$ .

**Theorem 12.2.10: Duality Principle.** If a proposition  $A$  is derivable from the axioms of a Boolean algebra, then the dual of  $A$  is also derivable from those axioms.

**Proof:** In the definition of a Boolean algebra, each axiom is a dual pair of propositions. Therefore, if in a proof of a proposition  $A$  we replace every proposition by its dual, the result is again a proof, because axioms are replaced by axioms. But this new proof is a proof of the dual of  $A$ . ■

The following theorem lists some basic properties of a Boolean algebra.

**Theorem 12.2.11:** In a Boolean algebra  $(B, +, \cdot, ')$  the following properties hold:

- (i) The elements 0 and 1 are unique.
- (ii) Each  $a \in B$  has a unique complement  $a' \in B$ .
- (iii) For each  $a \in B$ ,  $(a')' = a$ .
- (iv)  $0' = 1$  and  $1' = 0$ .
- (v) (Idempotent property):  $a + a = a$  and  $a \cdot a = a$  for every  $a \in B$ .
- (vi)  $a + 1 = 1$  and  $a \cdot 0 = 0$  for every  $a \in B$ .
- (vii) (Absorption property):  $a \cdot (a + b) = a$  and  $a + (a \cdot b) = a$  for all  $a, b \in B$ .

**Proof:**

- (i) If possible, let  $0_1$  and  $0_2$  be two zero elements of  $B$ . Then,  $a + 0_1 = a$  and  $a + 0_2 = a$  for all  $a \in B$ . Hence,

$$\begin{aligned} 0_2 + 0_1 &= 0_1 + 0_2 && \text{commutative law for } + \\ &= 0_2 && \text{because } 0_1 \text{ is an additive identity} \end{aligned}$$

Also,  $0_2 + 0_1 = 0_1$ , because  $0_2$  is an additive identity. Therefore,  $0_1 = 0_2$ ; i.e., the zero element 0 in  $B$  is unique. By the duality principle, the unit element 1 in  $B$  is unique.

- (ii) Let  $a'_1, a'_2$  be both complements of  $a$ . Then,

$$a + a'_1 = 1, \quad a \cdot a'_1 = 0, \quad a + a'_2 = 1, \quad a \cdot a'_2 = 0.$$

Now,

$$\begin{aligned} a'_1 &= a'_1 \cdot 1 && \text{because 1 is the multiplicative identity} \\ &= a'_1 \cdot (a + a'_2) \\ &= (a'_1 \cdot a) + (a'_1 \cdot a'_2) && \text{by the distributive law} \\ &= 0 + (a'_1 \cdot a'_2) && \text{because } a'_1 \cdot a = a \cdot a'_1 = 0 \\ &= a'_1 \cdot a'_2 && \text{because 0 is the additive identity} \end{aligned}$$

Similarly, we can show that  $a'_2 = a'_2 \cdot a'_1$ . Hence,

$$\begin{aligned} a'_1 &= a'_1 \cdot a'_2 \\ &= a'_2 \cdot a'_1 && \text{by the commutative law} \\ &= a'_2 \end{aligned}$$

- (iii) For each  $a \in B$ , there exists a unique element  $a' \in B$  such that  $a \cdot a' = 0$  and  $a + a' = 1$ . Hence,

$$\begin{aligned} a' + a &= 1 && \text{by the commutative law} \\ a' \cdot a &= 0 && \text{by the commutative law} \end{aligned}$$

These results imply that  $a$  is the complement of  $a'$ , i.e.,  $(a')' = a$ .

- (iv) For every  $a \in B$ , we have  $a \cdot 1 = a$  and  $0 + a = a$ . Replacing  $a$  by 0 and 1, respectively, we get  $0 \cdot 1 = 0$  and  $0 + 1 = 1$ . This shows that 1 is the complement of 0 in  $B$ . Hence,  $0' = 1$ . Again,  $0 = 0 \cdot 1 = 1 \cdot 0$  and  $1 = 0 + 1 = 1 + 0$ . Thus, 0 is the complement of 1. Hence,  $1' = 0$ .

- (v) We have

$$\begin{aligned} a + a &= (a + a) \cdot 1 && \text{because 1 is the identity for } \cdot \\ &= (a + a) \cdot (a + a') && \text{complementation law} \\ &= a + (a \cdot a') && \text{distributive law of } + \text{ over } \cdot \\ &= a + 0 && \text{complementation law} \\ &= a && 0 \text{ is the identity for } + \end{aligned}$$

The second part can be proved in a similar manner.

- (vi)

$$\begin{aligned} a + 1 &= (a + 1) \cdot 1 && 1 \text{ is the identity for } \cdot \\ &= 1 \cdot (a + 1) && \text{by the commutative law for } \cdot \\ &= (a + a') \cdot (a + 1) && \text{by the complementation law} \\ &= a + (a' \cdot 1) && \text{distributive law of } + \text{ over } \cdot \\ &= a + a' && 1 \text{ is the identity for } \cdot \\ &= 1 && \text{by the complementation law} \end{aligned}$$

The second part can be proved in a similar manner.

(vii)

$$\begin{aligned}
 a \cdot (a + b) &= (a + 0) \cdot (a + b) && 0 \text{ is the identity for } + \\
 &= a + (0 \cdot b) && \text{distributive law of } + \text{ over } \cdot \\
 &= a + 0 && \text{by part (vi)} \\
 &= a && 0 \text{ is the identity for } +
 \end{aligned}$$

The second part is the dual of the first part. ■

**Theorem 12.2.12:** In a Boolean algebra  $(B, +, \cdot, ')$  the following properties hold: For all  $a, b, c \in B$ ,

- (i) If  $b + a = c + a$  and  $b + a' = c + a'$ , then  $b = c$ . Also, if  $b \cdot a = c \cdot a$  and  $b \cdot a' = c \cdot a'$ , then  $b = c$ ;
- (ii) (Associative laws):  $a + (b + c) = (a + b) + c$  and  $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ ;
- (iii) (DeMorgan's laws):  $(a + b)' = a' \cdot b'$  and  $(a \cdot b)' = a' + b'$ ;
- (iv)  $a + b = (a' \cdot b')'$  and  $a \cdot b = (a' + b')'$ ;
- (v)  $a + b' = 1$  if and only if  $a + b = a$ ; also  $a \cdot b' = 0$  if and only if  $a \cdot b = a$ ;
- (vi)  $a + (a' \cdot b) = a + b$  and  $a \cdot (a' + b) = a \cdot b$ .

### Proof:

- (i) Assume that  $b + a = c + a$  and  $b + a' = c + a'$ . Then,

$$\begin{aligned}
 b &= b + 0 && 0 \text{ is the identity for } + \\
 &= b + (a \cdot a') && \text{by the complementation law} \\
 &= (b + a) \cdot (b + a') && \text{by the distributive law of } + \text{ over } \cdot \\
 &= (c + a) \cdot (c + a') && \text{by the given assumptions} \\
 &= c + (a \cdot a') && \text{by the distributive law of } + \text{ over } \cdot \\
 &= c + 0 && \text{by the complementation law} \\
 &= c && 0 \text{ is the identity for } +
 \end{aligned}$$

The second part is the dual of the first part.

- (ii) To prove the first part of (ii) we shall use the second part of (i), replacing  $b$  by  $a + (b + c)$  and  $c$  by  $(a + b) + c$ . To apply (i) we show

- (I)  $(a + (b + c)) \cdot a = ((a + b) + c) \cdot a$ , and
- (II)  $(a + (b + c)) \cdot a' = ((a + b) + c) \cdot a'$ .

- (I) We have

$$(a + (b + c)) \cdot a = a \cdot (a + (b + c)) = a.$$

Also,

$$\begin{aligned}
 ((a + b) + c) \cdot a &= a \cdot ((a + b) + c) && \text{commutative law for multiplication} \\
 &= a \cdot (a + b) + (a \cdot c) && \text{distributive law} \\
 &= a + (a \cdot c) && \text{by the absorption law} \\
 &= a && \text{by absorption law}
 \end{aligned}$$

(II) We have

$$\begin{aligned}
 (a + (b + c)) \cdot a' &= a' \cdot (a + (b + c)) \\
 &= (a' \cdot a) + a' \cdot (b + c) \\
 &= 0 + a' \cdot (b + c) \\
 &= a' \cdot (b + c).
 \end{aligned}$$

Also

$$\begin{aligned}
 ((a + b) + c) \cdot a' &= a' \cdot ((a + b) + c) \\
 &= a' \cdot (a + b) + a' \cdot c \\
 &= (a' \cdot a + a' \cdot b) + a' \cdot c \\
 &= 0 + (a' \cdot b) + a' \cdot c \\
 &= a' \cdot b + a' \cdot c \\
 &= a' \cdot (b + c).
 \end{aligned}$$

Thus,

$$(a + (b + c)) \cdot a' = ((a + b) + c) \cdot a'.$$

Hence, by part (i) above,

$$a + (b + c) = (a + b) + c.$$

The second part is the dual of the first part.

(iii) We have

$$\begin{aligned}
 (a + b) \cdot (a' \cdot b') &= ((a + b) + a') \cdot ((a + b) + b') \\
 &= (a' + (a + b)) \cdot (a + (b + b')) \\
 &= ((a' + a) + b) \cdot (a + 1) \\
 &= (1 + b) \cdot (a + 1) \\
 &= 1 \cdot 1 \\
 &= 1.
 \end{aligned}$$

Hence,

$$(a + b) \cdot (a' \cdot b') = 1. \quad (12.1)$$

Again,

$$\begin{aligned}
 (a + b) \cdot (a' \cdot b') &= ((a + b) \cdot a') \cdot b' \\
 &= (a' \cdot (a + b)) \cdot b' \\
 &= ((a' \cdot a) + (a' \cdot b)) \cdot b' \\
 &= (0 + a' \cdot b) \cdot b' \\
 &= (a' \cdot b) \cdot b' \\
 &= a' \cdot (b \cdot b') \\
 &= a' \cdot 0 \\
 &= 0.
 \end{aligned}$$

Thus,

$$(a + b) \cdot (a' \cdot b') = 0 \quad (12.2)$$

From (12.1) and (12.2) it follows that  $(a + b)' = a' \cdot b'$ .  
The second part is the dual of the first part.

- (iv) We have  $(a + b)' = a' \cdot b'$ . Hence  $(a + b)'' = (a' \cdot b')'$ , i.e.,  $a + b = (a' \cdot b')'$ .  
The second part is the dual of the first part.

We leave the proof of parts (v) and (vi) as exercises. ■

---

**REMARK 12.2.13** ► From now on we shall write both  $a \cdot (b \cdot c)$  and  $(a \cdot b) \cdot c$  as  $abc$  and, similarly,  $(a + b) + c$  and  $a + (b + c)$  as  $a + b + c$ .

In the first section of this chapter, we introduced two-element Boolean algebra. In this section, we discussed Boolean algebra in general, which may be either finite or infinite. Now we pose the following problem: Does there exist a Boolean algebra with three elements?

Recall from Chapter 3 that a lattice  $(L, \leq)$  is called a distributive lattice if it satisfies  $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$  for all  $a, b, c \in L$ . A distributive lattice  $(L, \leq)$  with the greatest element 1 and the least element 0 is called a Boolean algebra if for every element  $a \in L$  there exists an element  $a' \in L$ , called the complement of  $a$ , such that  $a \vee a' = 1$  and  $a \wedge a' = 0$ .

If we consider a Boolean algebra  $(B, +', 0, 1)$  defined by Huntington's postulates, then we can show that (see Exercise 16, page 794),  $(B, \leq)$  is a complemented distributive lattice, where  $a \leq b$  if and only if  $a + b = b$ ,  $a \vee b = a + b$ , and  $a \wedge b = ab$  for all  $a, b \in B$ . In the remainder of the chapter, by the symbol  $\leq$  in a Boolean algebra  $B$ , we mean the partial order relation as defined above.

---

**DEFINITION 12.2.14** ► Let  $B$  be a Boolean algebra. An element  $a \in B$  is called an **atom** in  $B$  if  $0 < a$  and there is no element  $b \in B$  such that  $0 < b < a$ . (In  $B$ ,  $x < y$  means that  $x \leq y$  and  $x \neq y$ .)

In a two-element Boolean algebra  $B = \{0, 1\}$ , clearly 1 is an atom. Consider the Boolean algebra  $(B, +', 0, 1)$ , where  $B$  is the set of all subsets of the set  $\{a, b, c\}$ ,  $A + B = A \cup B$ ,  $AB = A \cap B$ ,  $1 = S$ ,  $0 = \emptyset$ ,  $A' = S - A$ . This is a Boolean algebra with eight elements. In this Boolean algebra the subsets  $\{a\}$ ,  $\{b\}$ , and  $\{c\}$  are the atoms and they are the only atoms.

Clearly, every finite Boolean algebra contains atoms and the number of atoms is finite.

**Theorem 12.2.15:** Let  $B$  be a finite Boolean algebra. If  $b \in B$  is such that  $b \neq 0$ , then  $b$  can be expressed uniquely as a sum of atoms.

**Proof:** Let  $b \neq 0$  be an element of  $B$ . Then there exists an atom  $a$  such that  $a \leq b$ . Suppose  $a_1, a_2, \dots, a_n$  are all the atoms of  $B$  such that  $a_i \leq b$ . It follows that  $a_1 + a_2 + \dots + a_n \leq b$ . Let  $c = a_1 + a_2 + \dots + a_n$ . If  $bc' \neq 0$ , then there exists an atom, say  $d$ , such that  $d \leq bc'$ . Because  $bc' \leq b$ , it follows that  $d \leq b$ . Now  $a_1, a_2, \dots, a_n$  are all the atoms of  $B$  such that  $a_i \leq b$ . Then  $d = a_i$  for some  $i$ , where  $1 \leq i \leq n$ . This implies that  $d \leq c$  and  $d \leq bc'$  implies that  $d \leq c'$ . Then  $d \leq cc' = 0$ . This contradicts our assumption that  $d$  is an atom. Hence,  $bc' = 0$ . This implies (see Worked-Out Exercise 1, at the end of this section) that  $b + c = c$ . Then it follows that  $b \leq c$ . Hence,  $b = c = a_1 + a_2 + \dots + a_n$ . The uniqueness part is left as an Exercise. ■

From Theorem 12.2.15 we find that in a finite Boolean algebra every nonzero element can be expressed uniquely as a sum of atoms. It can be shown that this representation is also unique.

Let us consider the Boolean algebra  $B$  of all subsets of the set  $\{a, b, c\}$ , where  $\{a\}$ ,  $\{b\}$ , and  $\{c\}$  are the only atoms. In this Boolean algebra,  $\{a, b\}$ ,  $\{b, c\}$  are elements different from 0. We find that these elements can be expressed uniquely as  $\{a, b\} = \{a\} \cup \{b\} = \{a\} + \{b\}$ ,  $\{b, c\} = \{b\} \cup \{c\} = \{b\} + \{c\}$ .

Thus, all of the nonzero elements of this Boolean algebra will be obtained from the atoms  $\{a\}$ ,  $\{b\}$ , and  $\{c\}$ . Let  $T = \{\{a\}, \{b\}, \{c\}\}$ . From any nonempty subset of  $T$  we will obtain a unique element of the above Boolean algebra; for example, the subset  $\{\{a\}, \{b\}\}$  gives the unique element  $\{a\} + \{b\} = \{a\} \cup \{b\} = \{a, b\}$ .

Hence, we have the following theorem.

**Theorem 12.2.16:** Let  $B$  be a finite Boolean algebra with  $n$  atoms. Then the number of elements of  $B$  is the number of subsets of a set with  $n$  elements.

In Chapter 2, we proved that the number of subsets of a set with  $n$  elements is  $2^n$ . Hence, from Theorem 12.2.16, the number of elements of a finite Boolean algebra is some positive power of 2. Thus, we get the answer of the problem that there does not exist any Boolean algebra with three elements. Because 3 is not a positive power of 2, there is no Boolean algebra with three elements.

## WORKED-OUT EXERCISES

**Exercise 1:** Prove that the following properties are equivalent in a Boolean algebra  $B$ . For all  $a, c \in B$ :

- (a)  $ac = a$ ,      (b)  $ac' = 0$ ,      (c)  $a + c = c$ .

**Solution:**

(a)  $\Rightarrow$  (b):  $ac = a$  implies that  $(ac)' = ac'$ . Then  $a(ac') = ac'$ . But  $cc' = 0$  and  $a0 = 0$ . Hence  $0 = ac'$ .

(b)  $\Rightarrow$  (c): We have  $ac' = 0$ . Therefore,

$$a = a1 = a(c + c') = ac + ac' = ac + 0 = ac.$$

Hence,

$$a + c = (ac) + c = c + (ca) = c$$

(c)  $\Rightarrow$  (a): Given  $a + c = c$ . Now,  $ac = a(a + c) = a$  (absorption law).

Thus, (a)  $\Rightarrow$  (b), (b)  $\Rightarrow$  (c), and (c)  $\Rightarrow$  (a). Hence, the given three properties are equivalent.

**Exercise 2:** If  $a_1$  and  $a_2$  are elements of a Boolean algebra, then show that  $(a_1 + a_2)(a'_1 + a'_2) = a_2a'_1 + a_1a'_2$ .

**Solution:**

$$\begin{aligned} & (a_1 + a_2)(a'_1 + a'_2) \\ &= ((a_1 + a_2)a'_1) + ((a_1 + a_2)a'_2) \quad \text{by the distributive law} \\ &= ((a_1a'_1) + (a_2a'_1)) + ((a_1a'_2) + (a_2a'_2)) \quad \text{by the distributive law} \\ &= (0 + (a_2a'_1)) + ((a_1a'_2) + 0) \quad a'_i \text{ is the complement of } a_i \text{ for } i = 1, 2 \\ &= a_2a'_1 + a_1a'_2 \quad 0 \text{ is the identity for } + \end{aligned}$$

**Exercise 3:** If  $a_1, a_2$ , and  $a_3$  are elements of a Boolean algebra, then show that

$$\begin{aligned} a'_1 + a'_2(a_1 + a'_3) &= a'_1a_2a_3 + a'_1a_2a'_3 + a'_1a'_2a_3 \\ &\quad + a'_1a'_2a'_3 + a_1a'_2a_3 + a_1a'_2a'_3. \end{aligned}$$

**Solution:** We have

$$\begin{aligned} & a'_1 + a'_2(a_1 + a'_3) \\ &= a'_1 + a'_2a_1 + a'_2a'_3 \quad \text{by the distributive law} \\ &= a'_1(a_2 + a'_2)(a_3 + a'_3) + a'_2a_1(a_3 + a'_3) \\ &\quad + a'_2a'_3(a_1 + a'_1) \quad \text{because } a_i + a'_i = 1 \\ &= (a'_1a_2 + a'_1a'_2)(a_3 + a'_3) + a'_2a_1a_3 \\ &\quad + a'_2a_1a'_3 + a'_2a'_3a_1 \quad \text{by the distributive law} \\ &= a'_1a_2a_3 + a'_1a_2a'_3 + a'_1a'_2a_3 \\ &\quad + (a'_1a'_2a'_3 + a'_1a'_2a'_3) \\ &\quad + a_1a'_2a_3 + (a_1a'_2a'_3 + a_1a'_2a'_3) \quad \text{by the commutative laws} \\ &= a'_1a_2a_3 + a'_1a_2a'_3 + a'_1a'_2a_3 \\ &\quad + a'_1a'_2a'_3 + a_1a'_2a_3 + a_1a'_2a'_3. \end{aligned}$$

**Exercise 4:** If  $a_1, a_2$ , and  $a_3$  are elements of a Boolean algebra, then show that

$$a_1 + a'_2(a_1 + a'_3) = (a_1 + a'_2 + a_3)(a_1 + a'_2 + a'_3)(a_1 + a_2 + a'_3)$$

**Solution:** We have

$$\begin{aligned}
 & a_1 + a'_2(a_1 + a'_3) \\
 = & a_1 + a'_2a_1 + a'_2a'_3 && \text{by the distributive law} \\
 = & a_1 + a'_2a'_3 && \text{by the absorption law} \\
 = & (a_1 + a'_2)(a_1 + a'_3) && \text{by the distributive law} \\
 = & (a_1 + a'_2 + 0)(a_1 + a'_3 + 0) && 0 \text{ is the additive identity} \\
 = & (a_1 + a'_2 + a_3a'_3)(a_1 + a'_3 + a_2a'_2) && \text{because } a_3a'_3 = 0 \\
 & \quad \text{and } a_2a'_2 = 0 \\
 = & ((a_1 + a'_2) + a_3a'_3) \\
 & \quad \cdot ((a_1 + a'_3) + a_2a'_2) \\
 = & ((a_1 + a'_2) + a_3)((a_1 + a'_2) + a'_3) && \text{by the distributive law} \\
 & \quad \cdot ((a_1 + a'_3) + a_2)((a_1 + a'_3) + a'_2) \\
 = & (a_1 + a'_2 + a_3)(a_1 + a'_2 + a'_3) \\
 & \quad \cdot (a_1 + a_2 + a'_3)(a_1 + a'_2 + a'_3) && \text{by the associative} \\
 & \quad \text{and commutative laws} \\
 = & (a_1 + a'_2 + a_3)(a_1 + a'_2 + a'_3) && \text{by the commutative law} \\
 & \quad \cdot (a_1 + a'_2 + a'_3)(a_1 + a_2 + a'_3) \\
 = & (a_1 + a'_2 + a_3)(a_1 + a'_2 + a'_3) && \text{by the idempotent} \\
 & \quad \cdot (a_1 + a_2 + a'_3) && \text{property}
 \end{aligned}$$

**Exercise 5:** For any Boolean algebra  $B$ , prove that

$$(a + b)(b + c)(c + a) = ab + bc + ca.$$

**Solution:**

$$\begin{aligned}
 & (a + b)(b + c)(c + a) \\
 = & (a + b)(bc + ba + c + ca) && \text{by the distributive law} \\
 & \quad \text{and } cc = c \\
 = & abc + aba + ac +aca \\
 & \quad + bba + bba + bc + bca && \text{by the distributive law} \\
 = & abc + baa + ac + aac \\
 & \quad + bc + ba + bc + abc && \text{by the commutative laws} \\
 = & (abc + abc) + (ac + ac) \\
 & \quad + (bc + bc) + (ba + ba) && \text{by the idempotent law} \\
 = & abc + ac + bc + ab && \text{by the idempotent law} \\
 = & ac(b + 1) + bc + ab \\
 = & ac \cdot 1 + bc + ab && \text{because } b + 1 = 1 \\
 = & ab + bc + ca
 \end{aligned}$$

## SECTION REVIEW

### Key Terms

Boolean algebra	complement	dual
zero element	negation	atom
unit element	proposition	

### Some Key Definitions

1. A Boolean algebra is a nonempty set  $B$  together with two binary operations,  $+$ ,  $\cdot$ , on  $B$  (known as addition and multiplication, respectively) and a unary operation,  $'$ , on  $B$  (called the complementation) satisfying the following axioms:
  - (i) For all  $a$  and  $b$  in  $B$ ,  $a + b = b + a$  and  $a \cdot b = b \cdot a$ .
  - (ii) For all  $a, b, c \in B$ ,  $a + (b \cdot c) = (a + b) \cdot (a + c)$  and  $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$ .
  - (iii)  $B$  contains distinct elements 0 and 1 (known as the zero element and the unit element) such that for all  $a \in B$ ,  $a + 0 = a$  and  $a \cdot 1 = a$ .
  - (iv) For each  $a \in B$ , there exists an element  $a'$  in  $B$ , called the complement or negation of  $a$  in  $B$ , such that  $a + a' = 1$  and  $a \cdot a' = 0$ .
2. By a proposition in a Boolean algebra we mean either a statement or an algebraic identity in the Boolean algebra.
3. Let  $B$  be a Boolean algebra. An element  $a \in B$  is called an atom in  $B$  if  $0 < a$  and there is no element  $b \in B$  such that  $0 < b < a$ .

## Some Key Results

1. In a Boolean algebra  $(B, +, \cdot, ')$  the following properties hold:
  - (i) The elements 0 and 1 are unique.
  - (ii) Each  $a \in B$  has a unique complement  $a' \in B$ .
  - (iii) For each  $a \in B$ ,  $(a')' = a$ .
  - (iv)  $0' = 1$  and  $1' = 0$ .
  - (v)  $a + a = a$  and  $a \cdot a = a$  for every  $a \in B$ .
  - (vi)  $a + 1 = 1$  and  $a \cdot 0 = 0$  for every  $a \in B$ .
  - (vii)  $a \cdot (a + b) = a$  and  $a + (a \cdot b) = a$  for all  $a, b \in B$ .
  
2. In a Boolean algebra  $(B, +, \cdot, ')$  the following hold: For all  $a, b, c \in B$ ,
  - (i) If  $b + a = c + a$  and  $b + a' = c + a'$ , then  $b = c$ . Also, if  $b \cdot a = c \cdot a$  and  $b \cdot a' = c \cdot a'$ , then  $b = c$ ;
  - (ii)  $a + (b + c) = (a + b) + c$  and  $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ ;
  - (iii)  $(a + b)' = a' \cdot b'$  and  $(a \cdot b)' = a' + b'$ ;
  - (iv)  $a + b = (a' \cdot b')'$  and  $a \cdot b = (a' + b')'$ ;
  - (v)  $a + b' = 1$  if and only if  $a + b = a$ ; also  $a \cdot b' = 0$  if and only if  $a \cdot b = a$ ;
  - (vi)  $a + (a' \cdot b) = a + b$  and  $a \cdot (a' + b) = a \cdot b$ .

## EXERCISES

---

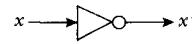
1. In a Boolean algebra  $B$ , for any  $a, b$  and  $c$ , prove the following.
  - a.  $(a + b') + ac = a + b'$
  - b.  $a + (b(ab'))' = a + b$
  - c.  $ab + ab' = a$
  - d.  $a + a'b = a + b$
2. In a Boolean algebra  $B$ , for any  $a$  and  $b$ , prove the following and write the dual of each statement.
  - a.  $(a' + b')' = ab$
  - b.  $ab' = 0$  if and only if  $ab = a$
  - c.  $a(b + 1) + a = a$
3. In a Boolean algebra  $B$ , for any  $a$  and  $b$ , prove that  $(ab + ab') + (a'b + a'b') = 1$ .
4. In a Boolean algebra  $B$ , for any  $a, b$ , and  $c$ , prove that  $((a + b)c)' = a'b' + c'$ .
5. In a Boolean algebra  $B$ , for any  $a, b$ , and  $c$ , prove the following.
  - a.  $a' + (b' + c) = (ab)' + c$
  - b.  $(a + b')(a' + b')(a + b)(a' + b) = 0$
- c.  $abc + bc = bc$
6. In a Boolean algebra  $B$ , for any  $a, b$ , and  $c$ , prove the following:  $(a + a'b)(b + a'c')(b + abc') = abc + a'bc + abc' + a'b'c'$ .
7. In a Boolean algebra  $B$ , prove that  $a + b = 0$  if and only if  $a = 0$  and  $b = 0$ .
8. Write the dual statement of Exercise 7 and prove it.
9. In a Boolean algebra  $B$ , for any  $a$  and  $b$ , prove that  $ab' + a'b = 0$  if and only if  $a = b$ .
10. Write the dual statement of Exercise 9 and prove it.
11. Write the dual statement of Exercise 6 and prove it.
12. Let  $B$  be the set of all positive divisors of 42. For any  $a, b \in B$ , let  $a + b = \text{lcm}[a, b]$ ;  $a \cdot b = \text{gcd}(a, b)$  and  $a' = \frac{42}{a}$ . Verify that  $(B, +, \cdot, ', 1, 42)$  is a Boolean algebra. Find the atoms of this Boolean algebra.
13. Does there exist a Boolean algebra with five elements? Justify your answer.
14. Prove parts (v) and (vi) of Theorem 12.2.12.
15. Prove the uniqueness part of Theorem 12.2.15.
16. Let  $B$  be a Boolean algebra as defined in this chapter. Prove that  $B$  is a complemented distributive lattice.

## 12.3 LOGICAL GATES AND COMBINATORIAL CIRCUITS

---

In the preceding two sections, we discussed Boolean algebra from a theoretical point of view. In this section, we show the application of Boolean algebra in the design of electronic circuits. The input to these circuits is a set of 0's and 1's. First we discuss circuits that have a single output, 0 or 1.

Consider the Boolean expression  $\alpha(x) = x'$ ; i.e.,  $\alpha(x)$  is the complement of  $x$ . Now  $\alpha(0) = 1$  and  $\alpha(1) = 0$ . This Boolean operation, i.e., complement, can be implemented using a device called the **NOT gate**, or the **inverter**. The symbol used for a NOT gate is shown in Figure 12.1.



**FIGURE 12.1**  
NOT (inverter) gate

As we can see, in Figure 12.1, the NOT gate has one input, shown by an arrow to its left and labeled by the variable  $x$ , and its output is shown by the arrow to its right and labeled by  $x'$ .

Now consider the Boolean expression  $\alpha(x, y) = xy$ ; i.e.,  $\alpha$  is the Boolean product of  $x$  and  $y$ . We know that  $\alpha(1, 1) = 1$ ,  $\alpha(1, 0) = 0$ ,  $\alpha(0, 1) = 0$ , and  $\alpha(0, 0) = 0$ . This Boolean operation, i.e., Boolean product, can be implemented using a device called an **AND gate**. The symbol used for an AND gate is shown in Figure 12.2.



**FIGURE 12.2** AND gate

As we can see, in Figure 12.2, there are two inputs, shown by two arrows to the left and labeled by  $x$  and  $y$ , and its output is shown by an arrow to its right and labeled by the Boolean expression  $xy$ .

Next consider the Boolean expression  $\alpha(x, y) = x + y$ ; i.e.,  $\alpha$  is the Boolean sum of  $x$  and  $y$ . We know that  $\alpha(1, 1) = 1$ ,  $\alpha(1, 0) = 1$ ,  $\alpha(0, 1) = 1$ , and  $\alpha(0, 0) = 0$ . This Boolean operation, i.e., Boolean sum, can be implemented using the device called an **OR gate**. The symbol used for an OR gate is shown in Figure 12.3.



**FIGURE 12.3** OR gate

As we can see, in Figure 12.3, there are two inputs, shown by two arrows to the left and labeled by  $x$  and  $y$ , and its output is shown by an arrow to its right and labeled by the Boolean expression  $x + y$ .

In circuitry theory, NOT, AND, and OR gates are the basic gates. As we will see, we can design any circuit using these gates. The circuits that we will design depend only on the inputs, not on the output. In other words, these circuits have no memory. Also these circuits are called **combinatorial circuits**.

---

**REMARK 12.3.1** ► The symbols NOT gate, AND gate, and OR gate are also considered as basic circuit symbols, which are used to build general circuits. Hence, we also use the word circuit instead of symbol.

Associated with each of the gates NOT, AND, and OR is a table called an **input-output table**, which is described next.

Consider the NOT gate that represents the Boolean expression  $x'$ . We say that the function of this basic circuit is the following: When it receives the input 1, it changes it to 0 as output, and when it receives the input 0, the output is 1. We can describe this in the following table.

NOT gate	Input $x$	Output $x'$
	1	0
	0	1

This table is called the input-output table for the NOT gate.

The AND gate is used to represent the Boolean expression  $xy$ . The input-output table for the AND gate is the following.

AND gate	Input $x$	Input $y$	Output $xy$
	1	1	1
	1	0	0
	0	1	0
	0	0	0

This table shows that when the AND gate receives the inputs 1 for  $x$  and 1 for  $y$ , then the output is 1. Similarly, when the AND gate receives the inputs 1 for  $x$  and 0 for  $y$ , then the output is 0, and so on.

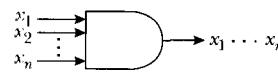
Similarly, the input-output table corresponding to the OR gate is as follows:

OR gate	Input $x$	Input $y$	Output $x + y$
	1	1	1
	1	0	1
	0	1	1
	0	0	0

---

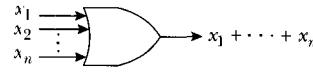
**REMARK 12.3.2** ► The symbols used for the different gates are the standard symbols established by the Institute of Electrical and Electronics Engineers.

Consider the Boolean expression  $x_1 x_2 \dots x_n$ ,  $n \geq 2$ . In the AND gate that represents this Boolean expression, we are allowed to use  $n$  arrows on the left of the AND gate. Therefore, the symbol used to represent the Boolean expression  $x_1 x_2 \dots x_n$  is as shown in Figure 12.4.



**FIGURE 12.4** AND gate  
with  $n$  inputs

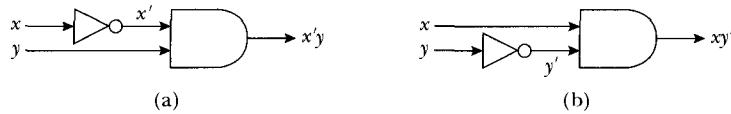
Similarly, for the Boolean expression  $x_1 + x_2 + \dots + x_n$  we use the symbol shown in Figure 12.5.



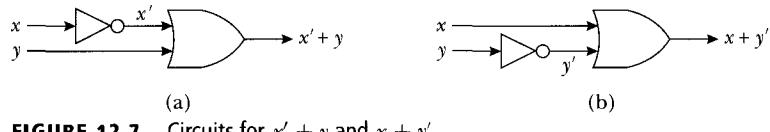
**FIGURE 12.5** OR gate with  
 $n$  inputs

We now extend these symbolic representations of Boolean expressions from basic Boolean expressions to arbitrary Boolean expressions. For this, we first consider the Boolean expressions  $x'y$ ,  $xy'$ ,  $x' + y$ ,  $x + y'$ ,  $x'y'$ , and  $x' + y'$ .

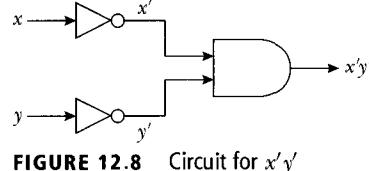
The circuits for the expressions  $x'y$  and  $xy'$  are shown in Figures 12.6(a) and (b), respectively.

**FIGURE 12.6** Circuits for  $x'y$  and  $xy'$ 

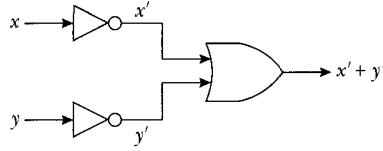
The circuits for the expressions  $x' + y$  and  $x + y'$  are shown in Figures 12.7(a) and (b), respectively.

**FIGURE 12.7** Circuits for  $x' + y$  and  $x + y'$ 

The circuit for the expression  $x'y'$  is shown in Figure 12.8.

**FIGURE 12.8** Circuit for  $x'y'$ 

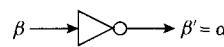
The circuit for  $x' + y'$  is shown in Figure 12.9.

**FIGURE 12.9** Circuit for  $x' + y'$ 

It follows that we can gradually build and represent any Boolean expression by gradually building a combinatorial circuit or circuit.

Let  $\alpha$  be a Boolean expression. Then its circuit diagram is defined as follows:

1. If  $\alpha$  is  $x'$ , then its circuit diagram is given by a NOT gate.
2. If  $\alpha$  is  $x + y$ , then its circuit diagram is given by an OR gate.
3. If  $\alpha$  is  $xy$ , then its circuit diagram is given by an AND gate.
4. If  $\alpha$  is  $\beta'$ , then its circuit diagram is as shown in Figure 12.10.

**FIGURE 12.10**Circuit for  $\beta' = \alpha$ 

In this circuit, there is a line on the left for the expression  $\beta$  and a line on the right representing  $\beta' = \alpha$ .

5. If  $\alpha$  is  $\beta x$ , then its circuit diagram is as shown in Figure 12.11.

In this circuit, there are two lines on the left, one for the expression  $\beta$  and one for the variable  $x$ , and there is a line on the right for the expression  $\beta x$ .

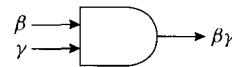
**FIGURE 12.11**Circuit for  $\beta x$ 

6. If  $\alpha$  is  $\beta + x$ , then its circuit diagram is as shown in Figure 12.12.

**FIGURE 12.12** Circuit for  $\beta + x$ 

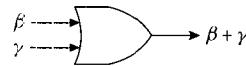
In this circuit, there are two lines on the left, one for the expression  $\beta$  and one for the variable  $x$ , and there is a line on the right for the expression  $\beta + x$ .

7. If  $\alpha$  is  $\beta\gamma$ , then its circuit diagram is as shown in Figure 12.13.

**FIGURE 12.13**Circuit for  $\beta\gamma$ 

In this circuit, there are two lines on the left for the expressions  $\beta$  and  $\gamma$ , and there is a line on the right for the expression  $\beta\gamma$ .

8. If  $\alpha$  is  $\beta + \gamma$ , then its circuit diagram is as shown in Figure 12.14.

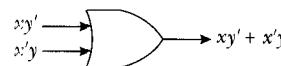
**FIGURE 12.14** Circuit for  $\beta + \gamma$ 

In this circuit, there are two lines on the left for the expressions  $\beta$  and  $\gamma$ , and there is a line on the right for the expression  $\beta + \gamma$ .

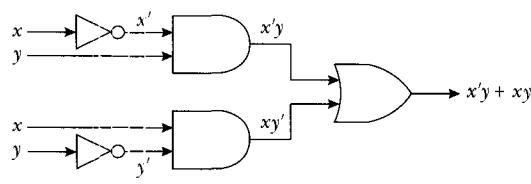
**EXAMPLE 12.3.3**

In this example, we show how to draw the circuit for the Boolean expression  $\alpha = xy' + x'y$ .

Let  $\beta = xy'$  and  $\gamma = x'y$ . The circuit diagram for  $\beta + \gamma$  is shown in Figure 12.15.

**FIGURE 12.15** Circuit for $\beta + \gamma$ 

We now work backward and draw the circuits for  $\beta$  and  $\gamma$ , which are then combined with the circuit in Figure 12.15. The circuit for  $\alpha = xy' + x'y$  is shown in Figure 12.16.

**FIGURE 12.16** Circuit for  $xy' + x'y$

**REMARK 12.3.4** ► In order to simplify drawing a circuit, a single input line can be split halfway and used as the input line for two or more separate gates. For example, the circuit in Figure 12.16 can be drawn as shown in Figure 12.17.

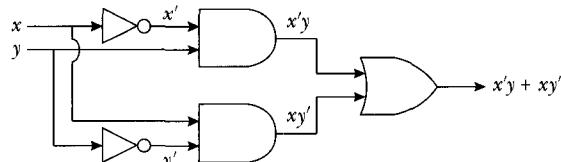


FIGURE 12.17 Circuit for  $xy' + x'y$

#### EXAMPLE 12.3.5

In this example, we draw the circuit diagram for  $\delta = (xy' + x'y)z$ . Suppose  $\alpha = xy' + x'y$ . We first draw the circuit for  $\alpha z$ . Then we continue to draw the circuit diagram for  $\alpha = xy' + x'y$  and obtain the circuit diagram for  $\delta = (xy' + x'y)z$  (see Figure 12.18).

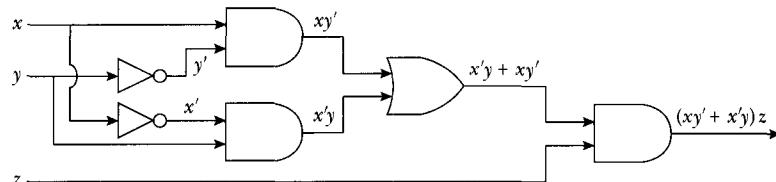


FIGURE 12.18 Circuit for  $(xy' + x'y)z$

#### REMARK 12.3.6

► Using the input-output tables of a NOT gate, AND gate, and OR gate, we can construct the input-output table for any circuits that are built from these basic gates. The input-output table obtained is the same as the truth table for the Boolean expression that represents the circuit. Hence, for every circuit there exists an input-output table, and conversely, for every input-output table we can draw a circuit whose input-output table is the same as the given table.

#### EXAMPLE 12.3.7

In this example, we find a Boolean expression that represents the circuit in Figure 12.19.

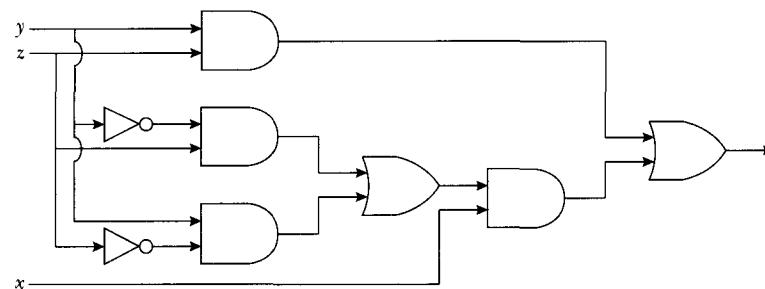


FIGURE 12.19 A circuit

First we write down the outputs through each gate. The required Boolean expression is the output through the last gate of the circuit. The diagram in Figure 12.20 illustrates this.

The Boolean expression that represents the circuit in Figure 12.20 is  $yz + x(yz' + y'z)$ .

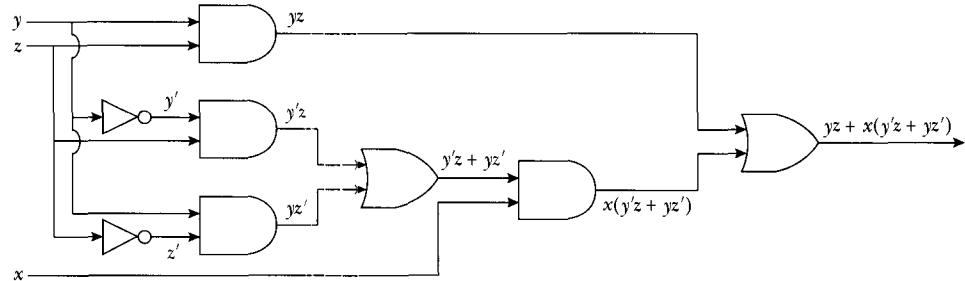


FIGURE 12.20 Circuit for  $yz + x(yz' + y'z)$

### EXAMPLE 12.3.8

In this example, we construct a circuit that represents the following input-output table.

$x$	$y$	Output
1	1	1
1	0	0
0	1	0
0	0	1

It can be shown that the Boolean expression  $xy + x'y'$  represents this input-output table. Hence, the circuit corresponding to this input-output table is as shown in Figure 12.21.

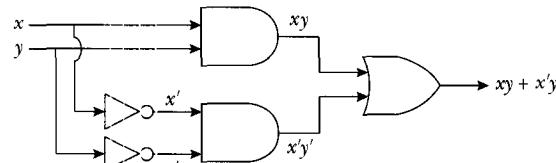


FIGURE 12.21 Circuit for  $xy + x'y'$

### EXAMPLE 12.3.9

Consider the Boolean expressions  $\alpha = xy$  and  $\beta = x(y + x')$ . Let  $C_1$  denote the circuit for  $\alpha$  and  $C_2$  denote the circuit for  $\beta$ . These circuits are shown in Figure 12.22.

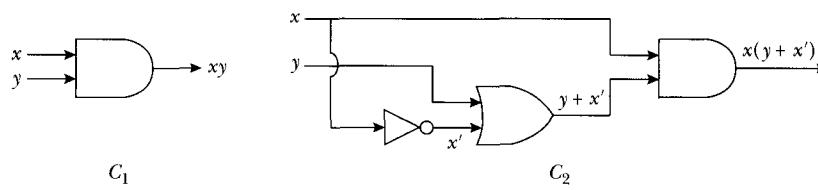


FIGURE 12.22 Circuits for  $xy$  and  $x(y + x')$

Note that  $\beta = x(y + x') = xy + xx' = xy + 0 = xy$ . Hence, the Boolean expressions  $\alpha$  and  $\beta$  are equal, i.e.,  $\alpha = \beta$ . However, note that circuit  $C_1$  requires only one gate while circuit  $C_2$  requires three gates. It follows that it is cheaper to construct  $C_1$  than  $C_2$ . Moreover, we say that circuit  $C_1$  is simpler than circuit  $C_2$ .

**DEFINITION 12.3.10** ▶ Two combinatorial circuits,  $C_1$  and  $C_2$ , having inputs  $x_1, x_2, \dots, x_n$  and a single output are said to be **equivalent** if whenever the circuits receive the same inputs, they give the same output.

**EXAMPLE 12.3.11**

Consider circuits  $C_1$  and  $C_2$  having inputs  $x$  and  $y$  and a single output, as shown in Figure 12.23.

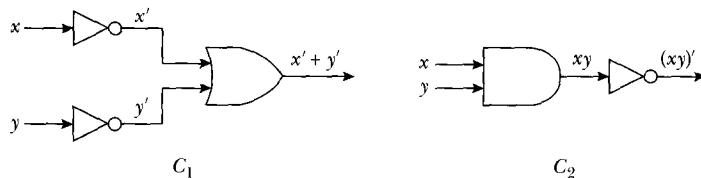


FIGURE 12.23 Circuits  $C_1$  and  $C_2$

The input-output table for circuits  $C_1$  and  $C_2$  is:

Input $x$	Input $y$	Output
1	1	0
1	0	1
0	1	1
0	0	1

Hence, these two circuits are equivalent.

Also, note that the Boolean expression for circuit  $C_1$  is  $x' + y'$  and the Boolean expression for circuit  $C_2$  is  $(xy)'$ . From Boolean algebra we know that  $(xy)' = x' + y'$ .

We leave the proof of the following theorem as an exercise.

**Theorem 12.3.12:** Two combinatorial circuits,  $C_1$  and  $C_2$ , having inputs  $x_1, x_2, \dots, x_n$  and a single output are equivalent if and only if the Boolean expression representing  $C_1$  is equal to the Boolean expression representing  $C_2$ .

**EXAMPLE 12.3.13**

Consider circuits  $C_1$  and  $C_2$  shown in Figure 12.24.

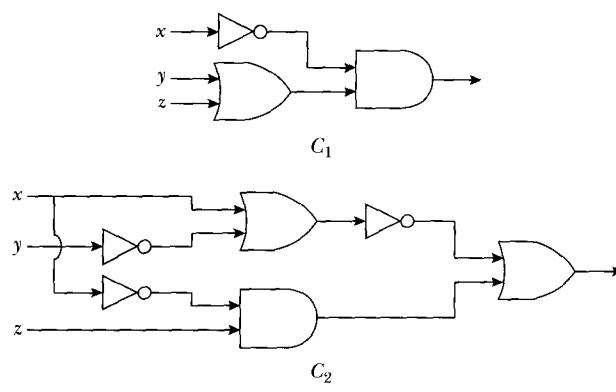


FIGURE 12.24 Circuits  $C_1$  and  $C_2$

The Boolean expression representing circuit  $C_1$  is  $x'(y + z)$ , and the Boolean expression representing circuit  $C_2$  is  $(x + y')' + x'z$ .

Now

$$\begin{aligned}(x + y')' + x'z &= x'y + x'z \quad \text{because } (x + y')' = x'(y')' \text{ and } (y')' = y \\ &= x'(y + z) \quad \text{by the distributive law}\end{aligned}$$

Hence, the Boolean expression representing  $C_1$  is equal to the Boolean expression representing  $C_2$ . Therefore, these two circuits are equivalent.

**DEFINITION 12.3.14** ► One circuit is said to be **simpler** than another circuit if the first circuit contains fewer gates than the second circuit.

In Figure 12.24, Example 12.3.13, we see that  $C_1$  contains three gates, whereas  $C_2$  contains six gates. Hence, the first circuit is simpler than the second circuit. Here we find that the effect of both circuits is the same but the first circuit is simpler than the second circuit.

Typically, we use the properties of Boolean algebra to simplify complex circuits used in digital computers and many other electronic devices.

**EXAMPLE 12.3.15**

A committee of three approves a bill by majority vote. Each member can vote for the proposal. To approve a bill two or three yes votes are necessary. We want to design a circuit that takes the votes of the three members of the committee as inputs and yields the decision whether the bill passes or not as output. We denote a yes vote by 1 and a no vote by 0, and output 1 for the approval of the bill. To accomplish this, we require a circuit that satisfies the following input-output table.

$x$	$y$	$z$	Pass?
1	1	1	1
1	1	0	1
1	0	1	1
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	0
0	0	0	0

The Boolean expression corresponding this input-output table is  $xyz + xyz' + xy'z + x'y'z$ . Next, let us simplify this expression. Now

$$\begin{aligned}xyz + xyz' + xy'z + x'y'z &= xy(z + z') + xy'z + x'y'z \\ &= xy + xy'z + x'y'z \quad \text{because } z + z' = 1 \\ &= xy + xyz + xy'z + x'y'z \quad \text{because } xy + xyz = xy(1 + z) = xy \cdot 1 = xy \\ &= xy + xz(y + y') + x'y'z \\ &= xy + xz + x'y'z \quad \text{because } y + y' = 1 \\ &= xy + xz + xyz + x'y'z \quad \text{because } xz + xyz = xz(1 + y) = xz \cdot 1 = xz \\ &= xy + xz + (x + x')yz \\ &= xy + xz + yz \quad \text{because } x + x' = 1 \\ &= x(y + z) + yz \quad \text{by the distributive law}\end{aligned}$$

Hence, the circuit corresponding to the expression  $xyz + xyz' + xy'z + x'y'z$  is equivalent to the circuit given by the expression  $x(y + z) + yz$ . Thus, the required circuit is as shown in Figure 12.25.

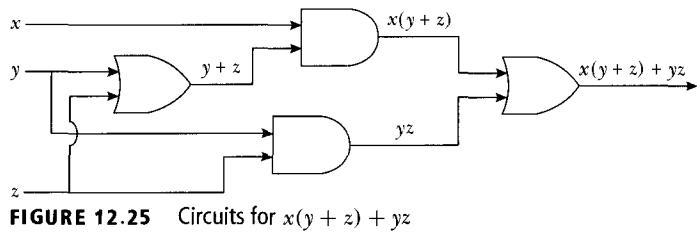


FIGURE 12.25 Circuits for  $x(y + z) + yz$

**REMARK 12.3.16** ▶ When designing a circuit engineers must take into consideration the cost of production. Because circuits with a smaller number of gates cost less, engineers try to design a circuit with as few gates as possible. In Example 12.3.15, if we draw a circuit corresponding to the Boolean expression  $xyz + xyz' + xy'z + x'y'z$ , then we need eight gates. However, the number of gates needed to construct the circuit corresponding to the expression  $x(y + z) + yz$  is four. Hence, in circuit theory, it is important to find an equivalent circuit that uses as few gates as possible. This can be accomplished by first finding the Boolean expression  $\alpha$  corresponding to the given circuit and then simplifying  $\alpha$ , using the properties of Boolean algebra, to find an equivalent Boolean expression  $\beta$  simpler than  $\alpha$ .

The problem of finding a simpler circuit is called the **minimization problem**. This problem depends on the set of available gates. The Quine-McCluskey method and Karnaugh maps are used to resolve minimization problems. In the next subsection, we describe a Karnaugh map to minimize a DNF Boolean expression.

In this section, we presented various examples illustrating the design of circuits using the gates NOT, AND, and OR. We also remarked that in circuitry theory these are the basic gates. However, there are other gates available that can be used to simplify the construction of a combinatorial circuit. Next, we describe two such gates called NAND and NOR.

**DEFINITION 12.3.17** ▶ The symbol used for the Boolean expression  $(xy)'$  is called a **NAND gate**.

The diagram in Figure 12.26 represents the NAND gate.

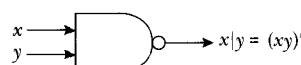


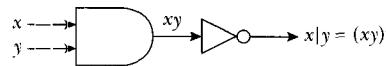
FIGURE 12.26 NAND gate

The input-output table for a NAND gate is as follows.

Input $x$	Input $y$	Output for NAND gate
1	1	0
1	0	1
0	1	1
0	0	1

In Boolean algebra, the expression  $(xy)'$  is also denoted as  $x|y$ . The binary operation symbol  $|$  used in  $x|y$  is called the **Scheffer stroke**.

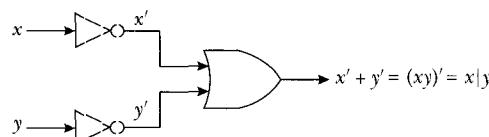
If we draw the circuit for the Boolean expression  $(xy)'$  using NOT and AND gates, we obtain the circuits shown in Figure 12.27.



**FIGURE 12.27** Circuit for  $(xy)'$  using NOT and AND gates

As we can see, the circuit in Figure 12.27 requires two gates. However, we can draw the same circuit using only one NAND gate.

Let us again consider the Boolean expression  $(xy)'$ . Now  $(xy)' = x' + y'$ . The circuit diagram for  $x' + y'$  is shown in Figure 12.28.



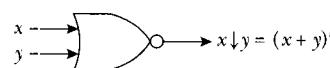
**FIGURE 12.28** Circuit for  $x' + y'$

Notice that this circuit contains three gates. We can replace this circuit by one NAND gate. Thus, we see that in a minimization problem the NAND gate plays a very useful role.

Next we describe the NOR gate.

**DEFINITION 12.3.18** ► The symbol that is used for the Boolean expression  $(x + y)'$  is called a **NOR gate**.

The diagram in Figure 12.29 represents the NOR gate.



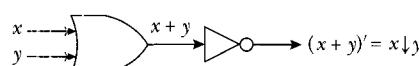
**FIGURE 12.29** NOR gate

The input-output table for the NOR gate is as follows.

Input x	Input y	Output for NOR gate
1	1	0
1	0	0
0	1	0
0	0	1

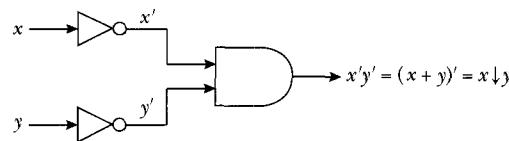
In Boolean algebra, the expression  $(x + y)'$  is also denoted by  $x \downarrow y$ . The binary operation symbol  $\downarrow$  used in  $x \downarrow y$  is called the **Peirce arrow**.

If we draw the circuit for the Boolean expression  $(x + y)'$  using NOT and OR gates, we obtain the circuit shown in Figure 12.30.



**FIGURE 12.30** Circuits for  $(x + y)'$  using NOT and OR gates

As we can see, the circuit in Figure 12.30 requires two gates. However, we can draw the same circuit using only one NOR gate. Let us again consider the Boolean expression  $(x + y)'$ . Now  $(x + y)' = x'y'$ . The circuit diagram for  $x'y'$  is shown in Figure 12.31.

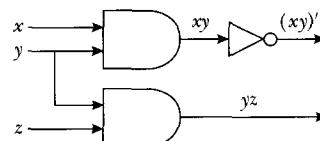


**FIGURE 12.31** Circuits for  $(x + y)' = x'y' = x \downarrow y$

Notice that this circuit contains three gates. We can replace this circuit by only one NOR gate. Thus, in this minimization problem the NOR gate also plays a useful role.

We now consider combinatorial circuits with more than one output and show how binary numbers are added.

The diagram in Figure 12.32 represents a circuit with more than one output.



**FIGURE 12.32** Circuit with two outputs

In Chapter 2, we described the addition of binary numbers. Recall that

$$\begin{array}{r} 1 \\ + 1 \\ \hline 10 \end{array}, \quad \begin{array}{r} 1 \\ + 0 \\ \hline 01 \end{array}, \quad \begin{array}{r} 0 \\ + 1 \\ \hline 01 \end{array}, \quad \begin{array}{r} 0 \\ + 0 \\ \hline 00 \end{array}$$

When we add the binary numbers 1 and 1, the sum is 0 and the carry is 1. Next we design a circuit that accomplishes this.

---

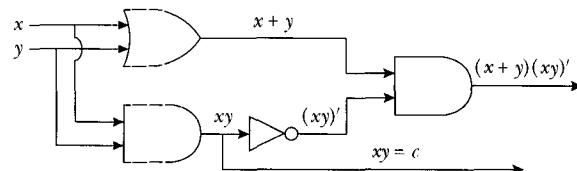
**DEFINITION 12.3.19** ▶ A **half adder** is a circuit that accepts as input two binary digits,  $x$  and  $y$ , and produces as output the binary sum  $cs$  of  $x$  and  $y$ . Here  $cs$  is a two-bit binary number, where  $s$  is called the sum bit and  $c$  is called the carry bit.

To design a half adder circuit, we require a circuit that satisfies the following input-output table.

Input $x$	Input $y$	Output $c$	Output $s$
1	1	1	0
1	0	0	1
0	1	0	1
0	0	0	0

This input table has two outputs,  $c$  and  $s$ . The diagram in Figure 12.33 represents the half adder circuit.

From the circuit diagram in Figure 12.33, it follows that  $s = (x + y)(xy)'$  and  $c = xy$ . For these two Boolean expressions we have the following input-output tables for  $(x + y)(xy)'$  and  $xy$ .



**FIGURE 12.33** Half adder

$x$	$y$	$x + y$	$xy = c$	$(xy)'$	$(x + y)(xy)' = s$
1	1	1	1	0	0
1	0	1	0	1	1
0	1	1	0	1	1
0	0	0	0	1	0

Next, we consider the problem of designing a circuit to add two binary numbers such that both binary numbers have more than one digit.

Now the addition of two binary digits may produce a carry for the column to its left. Hence, it is possible that we have to add three bits at certain points. For example, let us add the binary numbers  $x = 11010_2$  and  $y = 10110_2$ . To do so we add the corresponding digits from right to left. Whenever the corresponding digits are 1, the sum is 0 and the carry is 1, which is to be added with the bits left to these bits. Let us write these numbers as follows and demonstrate the additions.

$$\begin{array}{r}
 & 1 & 1 & 1 & 1 & 0 & \leftarrow \text{carry row} \\
 & 1 & 1 & 0 & 1 & 0 & = x \\
 + & & 1 & 0 & 1 & 1 & 0 & = y \\
 \hline
 & 1 & 1 & 0 & 0 & 0 & 0
 \end{array}$$

Thus,  $11010_2 + 10110_2 = 110000_2$ . To compute this sum we added the bits, from right to left, as follows.

$$\begin{array}{rcl}
 & 0 & 0 & 1 \\
 & \text{Step (i)} + 0, & \text{Step (ii)} + 1, & \text{Step (iii)} + 1, \\
 & \underline{0} & \underline{10} & \underline{10} \\
 & & & \\
 & 1 & 1 & 1 \\
 & \text{Step (iv)} + 0, & \text{Step (v)} + 1, & \text{and Step (vi)} \\
 & \underline{10} & \underline{11} & \underline{1}
 \end{array}$$

Notice that in Steps (ii)–(v) we added three digits. For example, consider Step (v). To perform  $1 + 1 + 1$ , we first perform  $1 + 1$ , i.e.,

$$\begin{array}{r} & 1 \\ + & 1 \\ \hline 10 \end{array}$$

Then we perform

$$\begin{array}{r} 10 \\ + 1 \\ \hline 11 \end{array}$$

It follows that the process of adding three bits can be divided into steps where we can use half adders.

In general, suppose we are adding three binary digits, say  $a, b, d$ , i.e.,

$$\begin{array}{r} a \\ b \\ + \quad d \\ \hline ? \end{array}$$

**Step 1.** First, using a half adder, we add  $a$  and  $b$  and obtain

$$\begin{array}{r} a \\ + \quad b \\ \hline c_1 s_1 \end{array}$$

Next we add  $c_1 s_1$  and  $d$ , i.e.,

$$\begin{array}{r} c_1 s_1 \\ + \quad d \\ \hline ? \end{array}$$

This is accomplished as follows:

**Step 2.** Using a half adder, we add  $s_1$  and  $d$  and obtain

$$\begin{array}{r} s_1 \\ + \quad d \\ \hline c_2 s \end{array}$$

This implies that  $s$  is the rightmost digit of the sum  $a + b + d$ .

Now the sum of  $a + b + d$  is a two-digit binary number,  $cs$ .  $s$  is already obtained and  $c$  is 1 if and only if either  $c_1$  or  $c_2$  is 1. Hence,  $c$  is obtained by using an OR gate with inputs  $c_1$  and  $c_2$ .

Next, we discuss how to design a circuit that adds three bits.

**DEFINITION 12.3.20** ► A **full adder** is a circuit that accepts as input three bits,  $a, b$ , and  $d$ , and produces as output the binary sum  $cs$  of  $a, b$ , and  $d$ .

We consider three one-digit binary numbers,  $a, b$ , and  $d$ . Suppose their binary sum is  $cs$ . We list all possible cases in the following table.

$a$	$b$	$d$	$c$	$s$
1	1	1	1	1
1	1	0	1	0
1	0	1	1	0
1	0	0	0	1
0	1	1	1	0
0	1	0	0	1
0	0	1	0	1
0	0	0	0	0

The first row of the table shows that  $1 + 1 + 1 = 11 = cs$ , the second row shows  $1 + 1 + 0 = 10 = cs$ , and so on.

In this table, we consider the sum of  $a$  and  $b$ . Let  $a + b = c_1 s_1$ . Now

$a$	$b$	Carry $c_1$	$s_1$
1	1	1	0
1	1	1	0
1	0	0	1
1	0	0	1
0	1	0	1
0	1	0	1
0	0	0	0
0	0	0	0

Next we compute the binary sum  $s_1 + d$ . Let  $s_1 + d = c_2 s_2$ . Now

$s_1$	$d$	Carry $c_2$	$s_2$
0	1	0	1
0	0	0	0
1	1	1	0
1	0	0	1
1	1	1	0
1	0	0	1
0	1	0	1
0	0	0	0

Next we add the carries,  $c_1$  and  $c_2$ .

$c_1$	$c_2$	$c$
1	0	1
1	0	1
0	1	1
0	0	0
0	1	1
0	0	0
0	0	0
0	0	0

From these tables, it follows that  $a$  and  $b$  are inputs to a first half adder with outputs  $s_1$  and  $c_1$ . The output  $s_1$  together with  $d$  forms inputs to a second half adder, whose outputs are  $s_2$  and  $c_2$ . Here  $s_2 = s$  is the rightmost bit of the final sum and from the OR gate with inputs  $c_1$  and  $c_2$ , the carry  $c$  of the binary sum  $a + b + d$  is obtained.

It follows that we can use two half adders to add  $a$ ,  $b$ , and  $d$  (see Figure 12.34).

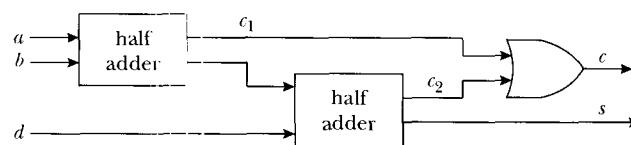


FIGURE 12.34 Diagram of full adder using half adders

**EXAMPLE 12.3.21**

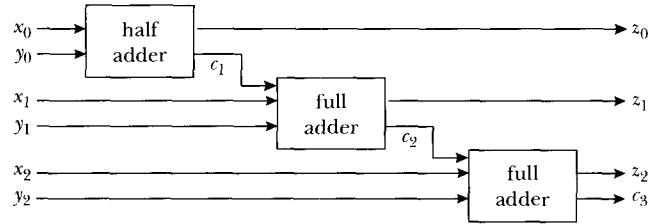
In this example, we use half adder and full adder circuits to add two three-bit binary numbers, say  $x = x_2x_1x_0$  and  $y = y_2y_1y_0$ .

To add  $x$  and  $y$ , we first add the bits  $x_0$  and  $y_0$ . This is an addition of two one-bit numbers, which can be accomplished by using a half adder. Suppose  $c_1$  is the carry produced by adding  $x_0$  and  $y_0$  and  $x_0 + y_0 = c_1z_0$ . (Notice that if  $x_0 = 1$  and  $y_0 = 1$ , then  $c_1 = 1$  and  $z_0 = 0$ . Similarly, if  $x_0 = 0$  and  $y_0 = 1$ , then  $c_1 = 0$  and  $z_0 = 1$ , and so on.)

Next, we add the bits  $x_1$  and  $y_1$  and the carry  $c_1$ ; i.e., we compute the sum  $x_1 + y_1 + c_1$ . This is a sum of three one-bit numbers and it can be accomplished using a full adder. Suppose  $x_1 + y_1 + c_1 = c_2z_1$ . Then  $c_2$  is the carry produced by the sum  $x_1 + y_1 + c_1$ .

Finally, we add the bits  $x_2$  and  $y_2$  and the carry  $c_2$ ; i.e., we compute the sum  $x_2 + y_2 + c_2$ . Again, this is a sum of three one-bit numbers and it can be accomplished using a full adder. Suppose  $x_2 + y_2 + c_2 = c_3z_2$ . Then  $c_3$  is the carry produced by the sum  $x_2 + y_2 + c_2$ .

Figure 12.35 shows the circuit to add  $x$  and  $y$ .



**FIGURE 12.35** Circuit to add three-bit binary numbers

It follows that  $x + y = c_3z_2z_1z_0$ . If  $c_3 = 0$ , then  $x + y = z_2z_1z_0$ . If  $c_3 = 1$ , then  $x + y = 1z_2z_1z_0$ . Therefore, if only three bits are used to add three-bit numbers and  $c_3 = 1$ , then  $c_3$  is considered overflow and will be dropped from the sum.

In this section, we introduced the logic gates NOT, AND, OR, NAND, and NOR. Before we leave the discussion of circuits, we make the following observations.

Let  $x$  and  $y$  be Boolean variables. Notice that

$$xy = (x')(y')' = (x' + y)'$$

Now, in a circuit,  $xy$  is implemented using an AND gate. The circuit corresponding to the expression  $(x' + y)'$  consists of NOT and OR gates. Because  $xy = (x' + y)'$ , it follows that in a circuit an AND gate can be replaced by NOT and OR gates. From this, we can conclude that any circuit which can be designed using NOT, AND, OR gates can also be designed using NOT and OR gates.

Next, let us now note the following:

$$x + y = (x')' + (y')' = (x'y')'$$

This implies that in a circuit an OR gate can be replaced by NOT and AND gates. That is, any circuit which is designed using NOT, AND, and OR gates can also be designed using only NOT and AND gates.

Let us now consider the NAND gate. We have

$$x' = (xx)' = x \mid x$$

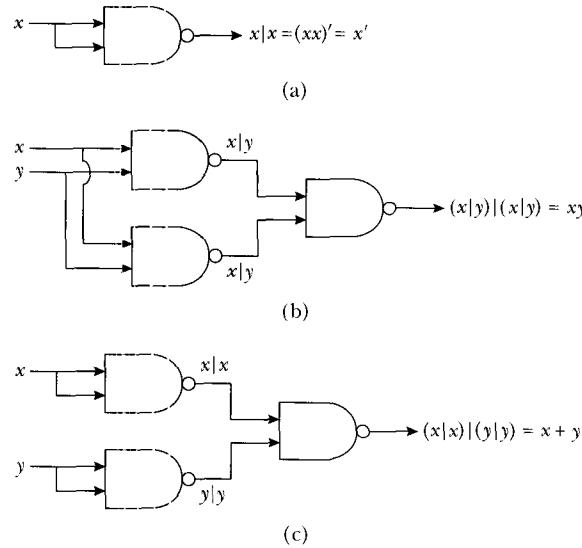
This implies that a NOT gate can be implemented using a NAND gate (see Figure 12.36(a)). Next,

$$xy = ((xy)')' = ((xy)'(xy)')' = ((x|y)(x|y))' = (x|y)|(x|y).$$

This implies that an AND gate can be implemented using NAND gates (see Figure 12.36(b)). Also,

$$x + y = (x')' + (y')' = (x'y')' = x' | y' = (x|x) | (y|y).$$

This implies that an OR gate can be implemented using NAND gates (see Figure 12.36(c)).



**FIGURE 12.36** Implementation of NOT, AND, and OR gates using NAND gates

It now follows that any circuit which is designed by using NOT, AND, and OR gates can also be designed using only NAND gates.

In a similar manner, we can show that any circuit which is designed by using NOT, AND, and OR gates can also be designed using only NOR gates (see Exercise 13, page 823).

## Karnaugh Maps and Minimization of Boolean Expressions

In the preceding section, we remarked that before a circuit is designed to implement a Boolean expression, the Boolean expression is minimized. To minimize a sum-of-product Boolean expression, i.e., a Boolean expression in DNF, we can use the properties of Boolean algebra. However, it can sometimes be quite complicated to do this. For example, let us consider the following sum-of-product:

$$xyz + xy'z + x'y'z + x'yz + x'yz' + xy'z'$$

Now

$$\begin{aligned} & xyz + xy'z + x'y'z + x'yz + x'yz' + xy'z' \\ &= xyz + (xy'z + xy'z) + x'y'z + x'yz + x'yz' + xy'z' \quad \text{because } a + a = a \\ &= xyz + xy'z + x'y'z + x'yz + x'yz' + (xy'z' + xy'z) \end{aligned}$$

$$\begin{aligned}
 &= xyz + xy'z + x'y'z + x'yz + x'y'z' + xy'(z' + z) \\
 &= xyz + xy'z + x'y'z + x'yz + x'y'z' + xy' \quad \text{because } z' + z = 1 \\
 &= xyz + xy'z + x'y'z + (x'yz + x'yz) + x'yz' + xy' \quad \text{because } a + a = a \\
 &= xyz + xy'z + x'y'z + x'yz + (x'yz + x'yz') + xy' \\
 &= xyz + xy'z + x'y'z + x'yz + x'y(z + z') + xy' \\
 &= xyz + xy'z + x'y'z + x'yz + x'y + xy' \quad \text{because } z' + z = 1 \\
 &= (xyz + xy'z) + (x'y'z + x'yz) + x'y + xy' \\
 &= xz(y + y') + x'z(y' + y) + x'y + xy' \\
 &= xz + x'z + x'y + xy' \\
 &= z(x + x') + x'y + xy' \\
 &= z + x'y + xy'.
 \end{aligned}$$

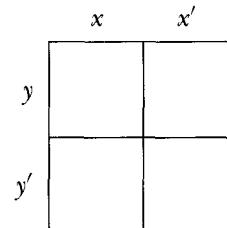
Hence,

$$xyz + xy'z + x'y'z + x'yz + x'y'z' + xy'z' = z + x'y + xy'.$$

Clearly, the expression  $z + x'y + xy$  requires fewer logic gates than the expression  $xyz + xy'z + x'y'z + x'yz + x'y'z' + xy'z'$ . As we can see, directly applying the properties of Boolean algebra to minimize a sum-of-product Boolean expression is very tedious.

Let us next describe the **Karnaugh map**, or **K-map** for short, to minimize a sum-of-product Boolean expression. We begin with sum-of-product forms involving two variables, say  $x$  and  $y$ .

Let  $\alpha(x, y)$  be a sum-of-product Boolean expression. To minimize  $\alpha$ , we use a rectangular array of two rows and two columns in which rows and columns are labeled as follows.



### Historical Notes

#### Maurice Karnaugh (b. 1924)

Maurice Karnaugh invented the Karnaugh map while studying for his Ph.D. in physics at Yale University in 1950. By 1952, Karnaugh had attained his degree and begun working at Bell Laboratories where he remained until 1966. His 1953 publication, "The Map Method for Synthesis of Combi-

national Logic Circuits," was a major contribution to computer science. This paper describing the Karnaugh map is essentially a development to the understanding of Boolean functions through two-dimensional diagrams.

Karnaugh's contribution to the understanding of Boolean logic gained him a strong reputation in digital

techniques and telecommunications. Between 1966 and 1993, he was a researcher for IBM, and between 1980 and 1999 he was a professor of data processing at New York University.

Each square, called a **cell**, corresponds to a minterm as follows:

	$x$	$x'$
$y$	$xy$	$x'y$
$y'$	$xy'$	$x'y'$

If a minterm  $xy$ ,  $xy'$ ,  $x'y$ , or  $x'y'$  is present in  $\alpha(x, y)$ , then we place a 1 in the cell corresponding to the minterm. If a minterm is not present, then the cell is left empty. The resulting array is called the *K-map* corresponding to the expression  $\alpha(x, y)$ .

---

**REMARK 12.3.22** ▶ Note that the minterms in the adjacent cells differ in one of the variables  $x$  or  $y$ . For example, the minterms  $xy$  and  $x'y$  in the adjacent cells (first row) differ in the variable  $x$ . Similarly, the minterms  $xy$  and  $xy'$  in the adjacent cells (first column) differ in the variable  $y$ .

We say that in a *K-map* two cells are **adjacent** if their minterms differ in only one variable.

### EXAMPLE 12.3.23

- (i) The *K-map* corresponding to the sum-of-product  $\alpha(x, y) = xy + x'y + x'y'$  is

	$x$	$x'$
$y$	1	1
$y'$		1

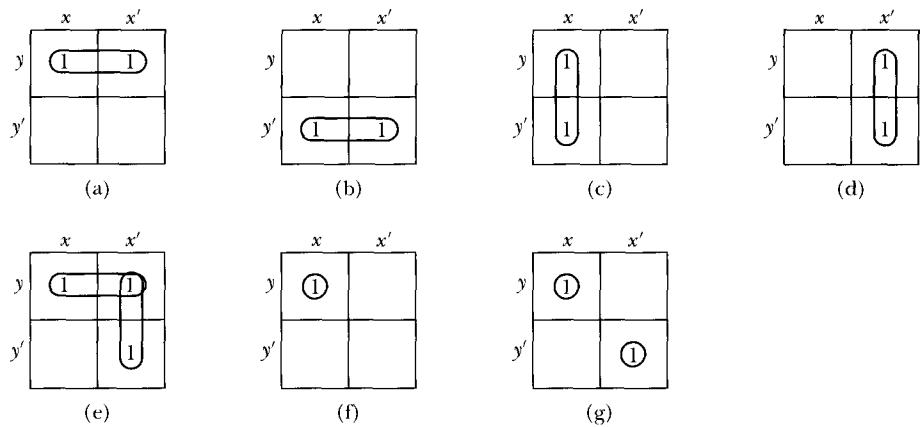
- (ii) The *K-map* corresponding to the sum-of-product  $\alpha(x, y) = x'y + xy'$  is

	$x$	$x'$
$y$		1
$y'$	1	

So the next obvious question is, how do we use the *K-map* to simplify the expression? To simplify a sum-of-product Boolean expression in the corresponding *K-map*, if possible, we group the 1's in the adjacent cells in a  $1 \times 2$ ,  $2 \times 1$ , or  $2 \times 2$  block. Here, by a  $1 \times 2$  block, we mean two adjacent cells in a row. Similarly, by adjacent cells in a  $2 \times 1$  block, we mean two adjacent cells in a column, and by adjacent cells in a  $2 \times 2$  block, we mean four adjacent cells in two rows and two columns.

As we will see, when writing the minimized sum-of-product Boolean expression in a  $1 \times 2$  or  $2 \times 1$  block, we can eliminate one of the variables, and in a  $2 \times 2$  block, we can eliminate both of the variables. Moreover, a 1 in a cell can be paired in more

than one block, and this is due to the fact that in a Boolean algebra  $a + a = a$ . If a 1 cannot be paired, we put a circle around it. The diagrams in Figure 12.37 illustrate *K*-maps.



**FIGURE 12.37** *K*-maps with two variables

Next, we discuss how to write the minimized form of the sum-of-product Boolean expression involving two variables using a *K*-map.

Consider the *K*-map in Figure 12.37(a). In this figure, the 1's are in the horizontal adjacent cells. The columns are labeled  $x$  and  $x'$ . Because  $x + x' = 1$ , we can eliminate both  $x$  and  $x'$ . The expression corresponding to this *K*-map is  $y$ . Note that the original Boolean expression corresponding to this *K*-map is  $xy + x'y$  and  $xy + x'y = (x + x')y = 1 \cdot y = y$ .

As in Figure 12.37(a), the minimized Boolean expression corresponding to the *K*-map in Figure 12.37(b) is  $y'$ .

In the *K*-map in Figure 12.37(c), the 1's are paired vertically. This would allow us to eliminate  $y$  and  $y'$ . Therefore, the Boolean expression corresponding to this *K*-map is  $x$ . Note that the original Boolean expression corresponding to this *K*-map is  $xy + xy'$  and  $xy + xy' = x(y + y') = x \cdot 1 = x$ .

As in Figure 12.37(c), the minimized Boolean expression corresponding to the *K*-map in Figure 12.37(d) is  $x'$ .

Consider the *K*-map in Figure 12.37(e). Note that the 1 corresponding to the cell  $x'y$  is paired with the 1's of the cells corresponding to  $xy$  and  $x'y'$ . This is allowed because in a Boolean algebra  $a + a = a$ .

Now the expression corresponding to the row of 1's is  $y$  and the expression corresponding to the column of 1's is  $x'$ . Hence, the minimized Boolean expression corresponding to the *K*-map in Figure 12.37(e) is  $x' + y$ .

The Boolean expression corresponding to the *K*-map in Figure 12.37(f) cannot be further simplified, so the Boolean expression is  $xy$ .

In the *K*-map in Figure 12.37(g), both of the 1's are circled. They cannot be combined. The minimized Boolean expression corresponding to this *K*-map is  $xy + x'y'$ .

## K-maps and Minimization of Boolean Expressions Involving Three Variables

Let  $\alpha(x, y, z)$  be a sum-of-product Boolean expression such that each minterm consists of three variables  $x, y, z$ . To define the *K*-map for  $\alpha(x, y, z)$ , we consider the following table.

	$xy$	$xy'$	$x'y'$	$x'y$
$z$	$xyz$	$xy'z$	$x'y'z$	$x'yz$
$z'$	$xyz'$	$xy'z'$	$x'y'z'$	$x'yz'$

Note that the columns are labeled  $xy$ ,  $xy'$ ,  $x'y'$ , and  $x'y$  so that the adjacent columns differ in only variable  $x$  or variable  $y$ . For example, the adjacent columns  $xy$  and  $xy'$  differ in  $y$ , and the adjacent columns  $x'y'$  and  $xy'$  differ in  $x$ . Moreover, notice that the first column,  $xy$ , and the last column,  $x'y$ , also differ in one variable,  $x$ .

**REMARK 12.3.24** ► In the  $K$ -map of three variables,  $x$ ,  $y$ , and  $z$ , we label the columns as  $xy$ ,  $xy'$ ,  $x'y'$ , and  $x'y$ . However, we can also start the labeling with, say  $x'y'$ . The important thing is that we label the columns so that the adjacent columns differ in one variable and the first and last columns also differ in one variable.

**REMARK 12.3.25** ► Note that the minterms in the adjacent cells differ in one of the variables  $x$ ,  $y$ , or  $z$ . For example, the minterms  $xy'z$  and  $x'y'z$  in the adjacent cells (first row and second and third column) differ in the variable  $x$ . Similarly, the minterms  $x'y'z'$  and  $x'y'z$  in the adjacent cells (third column) differ in the variable  $z$ . It follows that the cells containing the minterms  $xyz$  and  $xy'z$  are adjacent.

As before, we say that in a  $K$ -map two cells are adjacent if their minterms differ in only one variable.

As in the case of  $K$ -maps of two variables, in the  $K$ -map of a sum-of-product Boolean expression of three variables, if a minterm is present in the expression we place a 1 in the corresponding cell. For example, consider the sum-of-product Boolean expression

$$\alpha(x, y, z) = xyz + xy'z + xy'z' + x'y'z + xyz'.$$

Its  $K$ -map is

	$xy$	$xy'$	$x'y'$	$x'y$
$z$	1	1	1	
$z'$	1	1		

### EXAMPLE 12.3.26

The  $K$ -map corresponding to  $xyz + xy'z + x'y'z + x'yz + xyz' + x'y'z'$  is

	$xy$	$xy'$	$x'y'$	$x'y$
$z$	1	1	1	1
$z'$	1			1

To simplify a sum-of-product Boolean expression in three variables, in the corresponding  $K$ -map we group 1's in  $1 \times 1$ ,  $1 \times 2$ ,  $1 \times 4$ ,  $2 \times 1$ ,  $2 \times 2$ , and  $2 \times 4$

blocks. If there is a 1 in a  $1 \times 1$  block, it means it cannot be paired with any other cell. In blocks of  $1 \times 2$  and  $2 \times 1$ , we can eliminate one variable. In blocks of  $2 \times 2$  and  $1 \times 4$  we can eliminate two variables, and in a block of  $2 \times 4$  we can eliminate all the variables. The following example illustrates these situations.

**EXAMPLE 12.3.27**

Consider the K-maps in Figure 12.38.

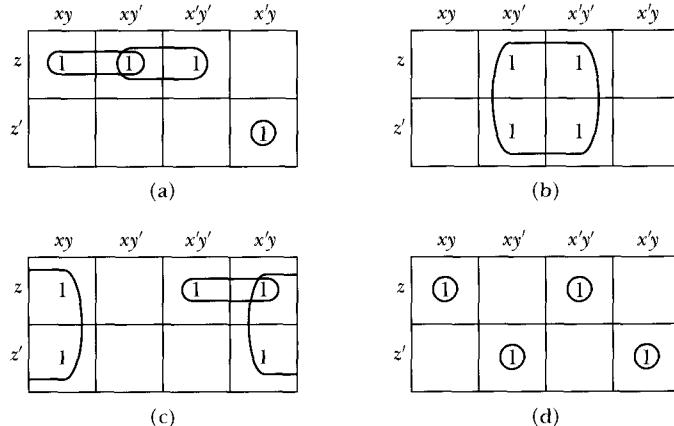


FIGURE 12.38 K-maps with three variables

- (i) Consider the K-map in Figure 12.38(a). Let us first consider the first  $1 \times 2$  block (first row and first two columns). For this block the columns of the adjacent cells are labeled  $xy$  and  $xy'$ , so the variable they differ in is  $y$ . Thus, we can eliminate  $y$ . Now corresponding to this block, the row is labeled  $z$ . Thus, the expression corresponding to this  $1 \times 2$  block is  $xz$ . Note that the expression corresponding to these adjacent cells is  $xyz + xy'z$  and  $xyz + xy'z = xz(y + y') = xz1 = xz$ .

Consider the second  $1 \times 2$  block (first row and second and third columns). For this block the columns of the adjacent cells are labeled  $x'y'$  and  $x'y$ , so the variable they differ in is  $x$ . Thus, we can eliminate  $x$ . Now corresponding to this block, the row is labeled  $z$ . Thus, the expression corresponding to this  $1 \times 2$  block is  $y'z$ . Note that  $xy'z + x'y'z = (x + x')y'z = 1y'z = y'z$ .

The Boolean expression corresponding to the 1 that has a circle around it is  $x'y'z'$ . Hence, the minimized Boolean expression corresponding to the K-map of Figure 12.38(a) is  $xz + y'z + x'y'z'$ .

- (ii) Consider the K-map in Figure 12.38(b). There is a  $2 \times 2$  block (second and third columns). Here the adjacent cells are labeled  $xy'$  and  $x'y'$ , so the variable they differ in is  $x$ . So we can eliminate  $x$ . Now the adjacent rows for this block are labeled  $z$  and  $z'$ , so we can also eliminate  $z$ . Hence, the Boolean expression corresponding to this  $2 \times 2$  block is  $y'$ . Note that  $xy'z + x'y'z + xy'z' + x'y'z' = xy'(z + z') + x'y'(z + z') = xy' + x'y' = (x + x')y' = y'$ .
- (iii) Consider the K-map in Figure 12.38(c). First let us consider the  $2 \times 2$  block (first and last columns). Here the adjacent cells are labeled  $xy$  and  $x'y$ , so the variable they differ in is  $x$ . So we can eliminate  $x$ . Now the adjacent rows for this block are labeled  $z$  and  $z'$ , so we can also eliminate  $z$ . Hence, the Boolean expression corresponding to this  $2 \times 2$  block is  $y$ .

As in part (i), the Boolean expression corresponding to the  $1 \times 2$  block is  $x'y'z$ . Hence, the minimized Boolean expression corresponding to the K-map in Figure 12.38(c) is  $y + x'y'z$ .

- (iv) In the *K*-map in Figure 12.38(d), there are four  $1 \times 1$  blocks. We cannot simplify the expressions corresponding to these cells. Hence, the minimized Boolean expression is  $xyz + x'y'z + xy'z' + x'y'z'$ .

## K-maps and Minimization of Boolean Expressions Involving Four Variables

Let  $\alpha(x, y, z, w)$  be a sum-of-product Boolean expression such that each minterm consists of four variables,  $x, y, z$ , and  $w$ . To define the *K*-map for  $\alpha(x, y, z, w)$  we consider the following table.

	$xy$	$xy'$	$x'y'$	$x'y$
$zw$	$xyzw$	$xy'zw$	$x'y'zw$	$xy'zw$
$zw'$	$xyzw'$	$xy'zw'$	$x'y'zw'$	$xy'zw'$
$z'w'$	$xyz'w'$	$xy'z'w'$	$x'y'z'w'$	$xy'z'w'$
$z'w$	$xyz'w$	$xy'z'w$	$x'y'z'w$	$xy'z'w$

Note that the rows are labeled  $zw, zw', z'w'$ , and  $z'w$  so that the adjacent rows differ in only one of the variables  $z$  or  $w$ . Similarly, as before, columns are labeled  $xy, xy'$ ,  $x'y'$ , and  $x'y$  so that the adjacent columns differ in only one of the variables  $x$  or  $y$ .

As before, we say that in a *K*-map two cells are adjacent if their minterms differ in only one variable.

### EXAMPLE 12.3.28

The *K*-map corresponding to the Boolean expression  $xyzw + xy'z'w + x'y'zw + xyzw' + x'y'zw' + x'y'z'w' + xy'zw' + x'y'zw$  is

	$xy$	$xy'$	$x'y'$	$x'y$
$zw$	1			1
$zw'$	1	1	1	
$z'w'$				1
$z'w$		1		1

To simplify a sum-of-product Boolean expression in four variables, in the corresponding *K*-map we group 1's in  $1 \times 1$ ,  $1 \times 2$ ,  $1 \times 4$ ,  $2 \times 1$ ,  $2 \times 2$ ,  $2 \times 4$ ,  $4 \times 1$ ,  $4 \times 2$ , and  $4 \times 4$  blocks. If there is a 1 in a  $1 \times 1$  block, it means it cannot be paired with any other cell. In  $1 \times 2$  and  $2 \times 1$  blocks, we can eliminate one variable. In  $2 \times 2$ ,  $1 \times 4$ , and  $4 \times 1$  blocks we can eliminate two variables. In  $2 \times 4$  and  $4 \times 2$

blocks we can eliminate three variables. In a  $4 \times 4$  block we can eliminate all the variables. The following example illustrates these situations.

**EXAMPLE 12.3.29**

Consider the K-map in Figure 12.39.

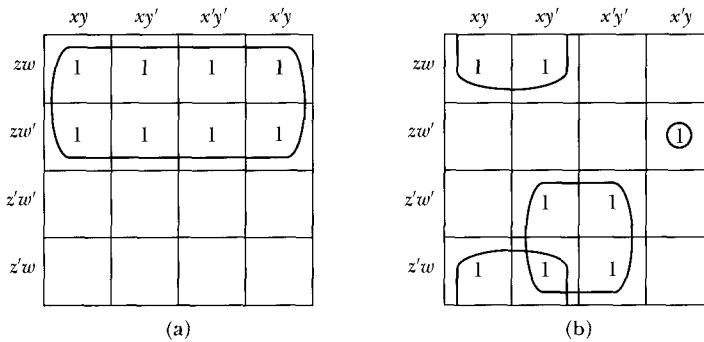


FIGURE 12.39 K-maps with four variables

- (i) Consider the K-map in Figure 12.39(a). This K-map has one  $2 \times 4$  block. From the columns, we can eliminate both  $x$  and  $y$ . The rows are labeled  $zw$  and  $zw'$ , so they differ in  $w$ . Thus, from rows, we can eliminate  $w$ . Hence, the minimized Boolean expression corresponding to this K-map is  $z$ .
- (ii) In the K-map in Figure 12.39(b), there are two  $2 \times 2$  blocks and one  $1 \times 1$  block.

Corresponding to the  $1 \times 1$  block, the Boolean expression is  $x'y'zw'$ .

Consider the first  $2 \times 2$  block (first and last rows and first and second columns). From the columns we can eliminate  $y$ , and from the rows we can eliminate  $z$ . Hence, the minimized Boolean expression corresponding to this block is  $xw$ .

Consider the second  $2 \times 2$  block (third and fourth rows and second and fourth columns). Here, we can eliminate  $x$  and  $w$ . The minimized Boolean expression corresponding to this block is  $y'z'$ .

It now follows that the minimized sum-of-product Boolean expression corresponding to this K-map is  $x'y'zw' + xw + y'z'$ .

The preceding examples show how to use K-maps to obtain a minimized sum-of-product Boolean expression. However, we have not yet outlined how to begin grouping squares that contain 1's. Before outlining the steps, let us consider the K-map in Figure 12.40(a).

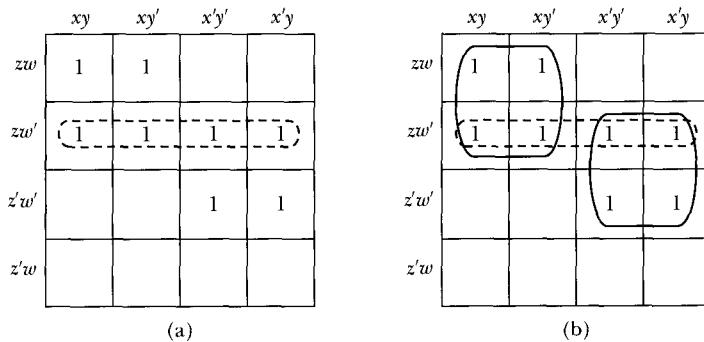


FIGURE 12.40 K-maps

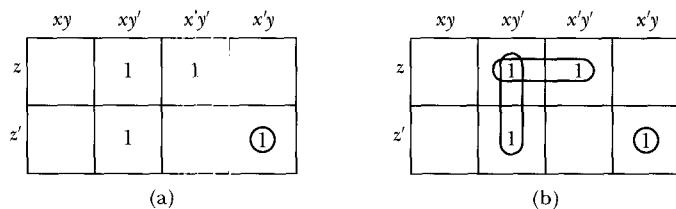
Suppose we start grouping the 1's in the second row (see Figure 12.40(a)). Then, when we group the 1's in the first and the third rows, we see that the 1's in the second row are again used (see Figure 12.40(b)). In fact, if we have grouped the 1's in the first row followed by grouping the 1's in the third row, then the 1's in the second row would have been consumed by these groupings.

To avoid these types of redundancies, we group the squares containing 1's as described in the list below. (Remember, the 1's must be grouped in  $1 \times 1$ ,  $1 \times 2$ ,  $1 \times 4$ ,  $2 \times 1$ ,  $2 \times 2$ ,  $2 \times 4$ ,  $4 \times 1$ ,  $4 \times 2$ , or  $4 \times 4$  blocks. Of course, these blocks must exist. For example, blocks of size  $2 \times 4$  or  $4 \times 2$  do not exist for sum-of-product Boolean expressions involving only two variables.)

1. First mark the 1's that cannot be paired with any other 1. Put a circle around them.
2. Next, from the remaining 1's, find the 1's that can be combined into two square blocks, i.e.,  $1 \times 2$  or  $2 \times 1$  blocks, and in only one way.
3. Next, from the remaining 1's, find the 1's that can be combined into four square blocks, i.e.,  $2 \times 2$ ,  $1 \times 4$ , or  $4 \times 1$  blocks, and in only one way.
4. Next, from the remaining 1's, find the 1's that can be combined into eight square blocks, i.e.,  $2 \times 4$  or  $4 \times 2$  blocks, and in only one way.
5. Next, from the remaining 1's, find the 1's that can be combined into 16 square blocks, i.e., a  $4 \times 4$  block. (Note that this could happen only for Boolean expressions involving four variables.)
6. Finally, look at the remaining 1's, i.e., the 1's that have not been grouped with any other 1. Find the largest blocks that include them.

### EXAMPLE 12.3.30

Consider the K-map in Figure 12.41.



**FIGURE 12.41** K-maps

In Figure 12.41(a), we identify the 1 that cannot be paired with any other 1 and put a circle around it. Among the remaining 1's we identify the 1's that can be grouped in  $1 \times 2$  or  $2 \times 1$  blocks and in only one way. For this, we first look at the 1 in the first row and third column and group it with the 1 in the first row and second column. Similarly, the 1 in the second row and second column is grouped as shown. Notice that the 1 in the first row and second column has two choices for grouping. That is why we did not consider this 1 in the second step. After making these groupings, no more 1's are left. The minimized sum-of-product Boolean expression is  $x'yz' + y'z + xy'$ .



## WORKED-OUT EXERCISES

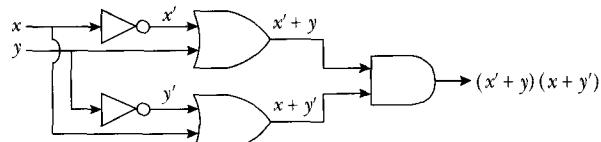
**Exercise 1:** Construct the circuit by using NOT, OR, and AND gates corresponding to each of the given Boolean expressions.

(a)  $(x' + y)(x + y')$

(b)  $x(y' + z) + y$

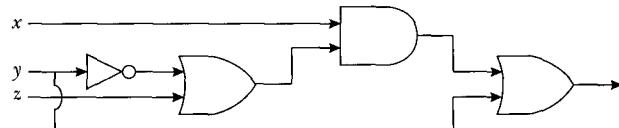
**Solution:**

(a) The circuit for  $(x' + y)(x + y')$  is shown in Figure 12.42.



**FIGURE 12.42** Circuit diagram

(b) The circuit for  $x(y' + z) + y$  is shown in Figure 12.43.



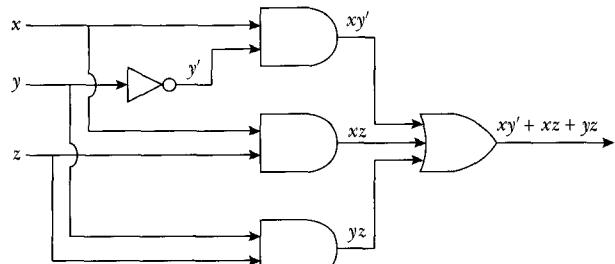
**FIGURE 12.43** Circuit diagram of  $x(y' + z) + y$

**Exercise 2:** Draw the circuit diagram for  $\delta = (xy' + x'y')'z$ .

**Solution:** Suppose  $\alpha = xy' + x'y$ ,  $\beta = (xy' + x'y')'$ . We first draw the circuit for  $\beta z$ . Next we draw the circuit diagram  $\beta$ . Now we draw the circuit diagram for  $\alpha = xy' + x'y$ . From these diagrams we obtain the circuit diagram for  $\delta = (xy' + x'y')'z$  (see Figure 12.44 below).

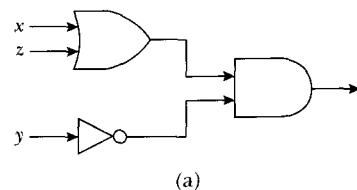
**Exercise 3:** Draw the circuit diagram for  $\delta = xy' + xz + yz$ .

**Solution:** The circuit diagram for  $\delta = xy' + xz + yz$  is shown in Figure 12.45.

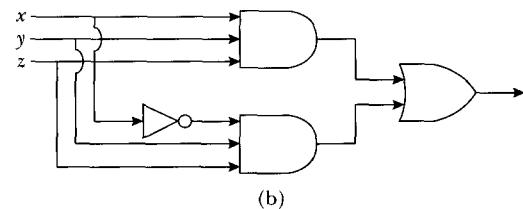


**FIGURE 12.45** Circuit diagrams of  $xy' + xz + yz$

**Exercise 4:** Write the Boolean expressions that represent the circuits in Figure 12.46.



(a)

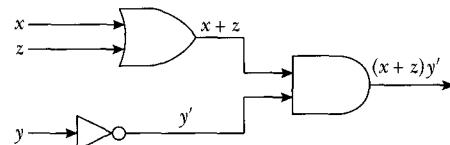


(b)

**FIGURE 12.46** Circuits

**Solution:**

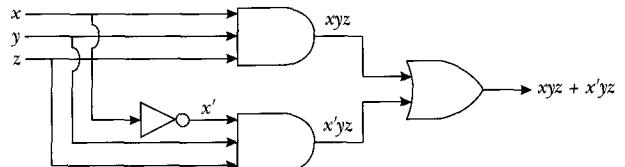
(a) First we write down the outputs through each gate (see Figure 12.47).



**FIGURE 12.47** Circuit diagrams of  $(x + z)y'$

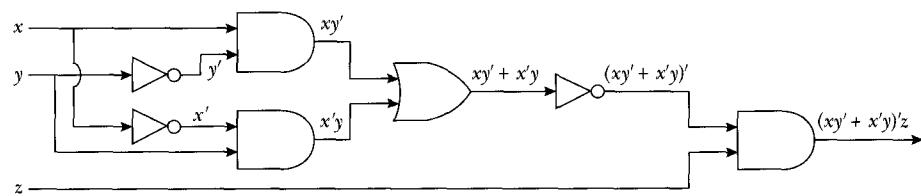
The required Boolean expression would be the output through the last gate of the circuit. Hence the Boolean expression that represents the given circuit is  $(x + z)y'$ .

(b) First we write down the outputs through each gate (see Figure 12.48).



**FIGURE 12.48** Circuit diagrams of  $xyz + x'yz$

The required Boolean expression would be the output through the last gate of the circuit. Hence the Boolean expression that represents the given circuit is  $xyz + x'yz$ .



**FIGURE 12.44** Circuit diagram of  $(xy' + x'y')'z$

**Exercise 5:** Construct the input-output table for the circuit in Figure 12.49.

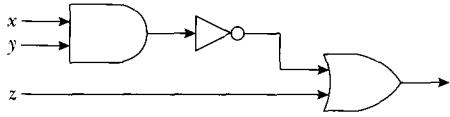


FIGURE 12.49 Circuit

**Solution:** The required Boolean expression would be the output through the last gate of the circuit. Hence the Boolean expression that represents the given circuit is  $(xy)' + z$ .

The input-output table for this circuit is the following.

x	y	z	$(xy)'$	$(xy)' + z$
1	1	1	0	1
1	1	0	0	0
1	0	1	1	1
1	0	0	1	1
0	1	1	1	1
0	1	0	1	1
0	0	1	1	1
0	0	0	1	1

**Exercise 6:** Find a circuit that represents the following input-output table.

x	y	Output
1	1	1
1	0	0
0	1	0
0	0	1

**Solution:** The Boolean expression that represents this table is  $xy + x'y'$ . Hence the circuit that represents this Boolean expression is shown in Figure 12.50.

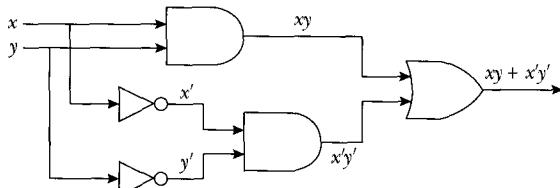


FIGURE 12.50 Circuit

**Exercise 7:** Show that the circuits in Figure 12.51 are equivalent.

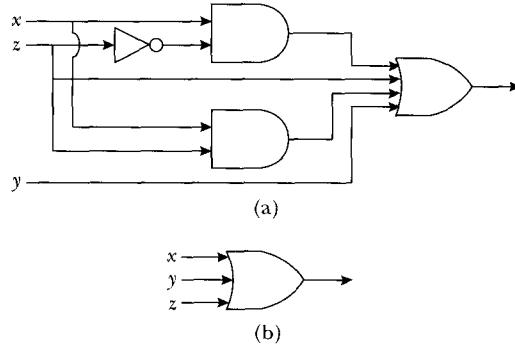


FIGURE 12.51 Circuits

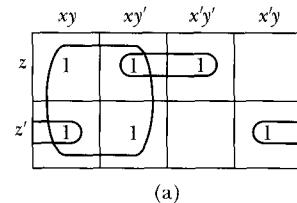
**Solution:** The Boolean expression for the first circuit is  $xz' + z + xz + y$ , and the Boolean expression for the second circuit is  $x + y + z$ .

Now

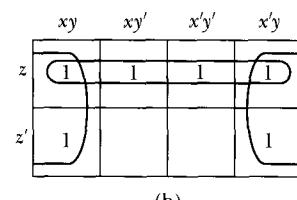
$$\begin{aligned} xz' + z + xz + y &= x(z' + z) + y + z \\ &= x + y + z \quad \text{because } z' + z = 1 \end{aligned}$$

Hence, the given two circuits are equivalent.

**Exercise 8:** Find the minimized sum-of-product Boolean expression corresponding to the K-maps in Figure 12.52.



(a)



(b)

FIGURE 12.52 K-map

**Solution:**

- Consider the K-map in Figure 12.52(a). In this K-map, there are two  $1 \times 2$  blocks and one  $2 \times 2$  block.

First consider the  $1 \times 2$  block in the first row and the second and third columns. In this block, the columns of the adjacent cells are labeled  $xy$  and  $x'y'$ , so they differ in  $x$ . We can eliminate  $x$ . The Boolean expression corresponding to this block is  $y'z$ .

Consider the  $1 \times 2$  block in the second row and the first and last columns. Here the columns are labeled  $xy$  and  $x'y$ , so they differ in  $x$ . We can eliminate  $x$ , and the Boolean expression corresponding to this  $1 \times 2$  block is  $yz'$ .

Now consider the  $2 \times 2$  block. The rows are labeled  $z$  and  $z'$ , so we can eliminate  $z$ , because  $z + z' = 1$ . Similarly, the columns are labeled  $xy$  and

$xy'$ , so we can eliminate  $y$ . Thus, the Boolean expression corresponding to this block is  $x$ .

It follows that the Boolean expression corresponding to the  $K$ -map in Figure 12.52(a) is  $x + y'z + yz'$ .

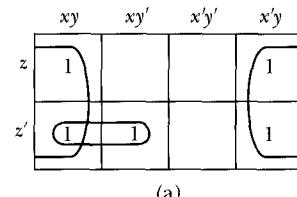
- (ii) As in part (i), we can show that the Boolean expression corresponding to the  $2 \times 2$  block is  $y$ .

Consider the  $1 \times 4$  block. Here we can eliminate both  $x$  and  $y$  to get the corresponding Boolean expression,  $z$ . Hence, the minimized sum-of-product Boolean expression corresponding to Figure 12.52(b) is  $y + z$ .

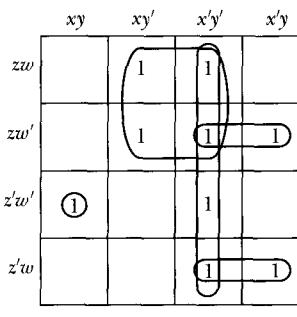
**Exercise 9:** Use a  $K$ -map to find the minimized sum-of-product Boolean expressions of the Boolean expressions.

- $xyz + xyz' + xy'z' + x'yz + x'y'z'$
- $xyz'w' + xy'zw + x'y'zw + xy'zw' + x'y'zw' + x'y'zw' + x'y'z'w' + x'y'z'w + x'y'z'w + x'y'z'w$

**Solution:** The  $K$ -maps of these Boolean expressions are shown in Figures 12.53(a) and (b), respectively.



(a)



(b)

**FIGURE 12.53**  $K$ -maps

- Figure 12.53(a) shows the grouping of 1's. The minimized sum-of-product Boolean expression is  $xz' + y$ .
- Figure 12.53(b) shows the grouping of 1's. The minimized sum-of-product Boolean expression is  $xyz'w' + x'zw' + x'z'w + y'z + x'y$ .

## SECTION REVIEW

### Key Terms

NOT gate	equivalent circuits	Peirce arrow
inverter	simpler circuit	half adder
AND gate	minimization problem	full adder
OR gate	NAND gate	Karnaugh map ( $K$ -map)
combinatorial circuits	Scheffer stroke	adjacent cells
input-output table	NOR gate	

### Some Key Definitions

- Two combinatorial circuits,  $C_1$  and  $C_2$ , having inputs  $x_1, x_2, \dots, x_n$  and a single output are said to be equivalent if whenever the circuits receive the same inputs, they give the same output.
- One circuit is said to be simpler than another circuit if the first circuit contains fewer gates than the second circuit.
- A half adder is a circuit that accepts as input two binary digits,  $x$  and  $y$ , and produces as output the binary sum  $cs$  of  $x$  and  $y$ . Here  $cs$  is a two-bit binary number, where  $s$  is called the sum bit and  $c$  is called the carry bit.
- A full adder is a circuit that accepts as input three bits,  $a$ ,  $b$ , and  $d$ , and produces as output the binary sum  $cs$  of  $a$ ,  $b$ , and  $d$ .

## Key Result

- Two combinatorial circuits,  $C_1$  and  $C_2$ , having inputs  $x_1, x_2, \dots, x_n$  and a single output are equivalent if and only if the Boolean expression representing  $C_1$  is equal to the Boolean expression representing  $C_2$ .

## EXERCISES

- Construct the circuit by using NOT, OR, and AND gates corresponding to each of the given Boolean expressions.
  - $xy + x'y$
  - $xy + (x + y)'y$
  - $xy' + (x' + y)y$
- For the circuits in Figures 12.54(a)–(e), write the associated Boolean expression.

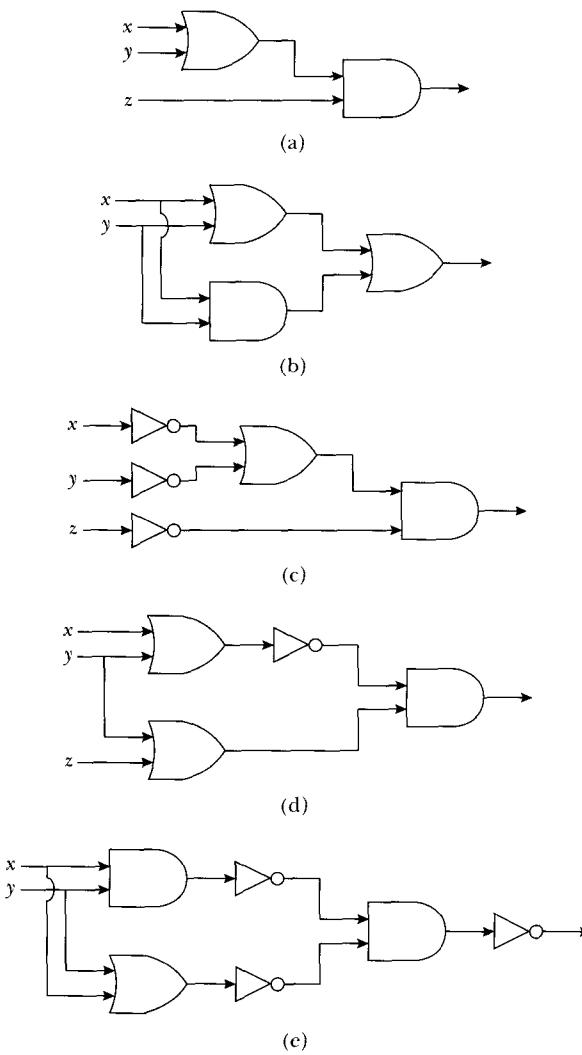


FIGURE 12.54 Various circuits

Draw a circuit by using NOT, OR, and AND gates corresponding to each of the given Boolean expressions in Exercises 3–6.

- $xy' + x'y$
- $x' + (xy + yz)$
- $x'y + (x(yz))'$
- $((xy)z + (xy' + x'z'))'$

For each of the Boolean expressions in Exercises 7–10, find the output for the given input.

- $(x + y)(x' + yz)$  for  $x = 0, y = 1$ , and  $z = 1$
- $xyz + x'y'z + yz$  for  $x = 0, y = 0$ , and  $z = 1$
- $(x + y + z)'$  for  $x = 1, y = 0$ , and  $z = 1$
- $xyz' + x'y'z + xz$  for  $x = 0, y = 1$ , and  $z = 0$
- With the help of an input-output table, determine which pairs of circuits in Figure 12.55 are equivalent.

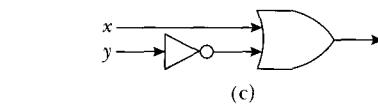
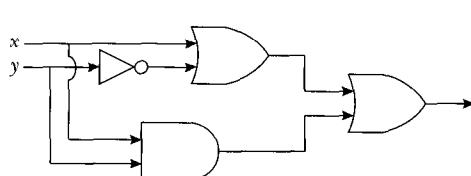
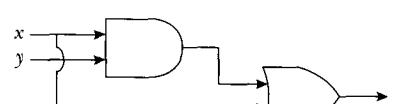
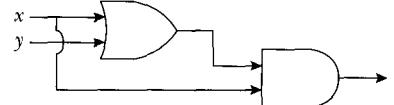
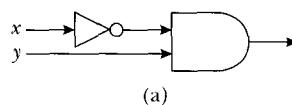
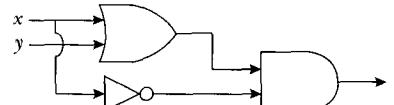


FIGURE 12.55 Circuits

12. With the help of Boolean algebra, determine which pairs of circuits in Figure 12.56 are equivalent.

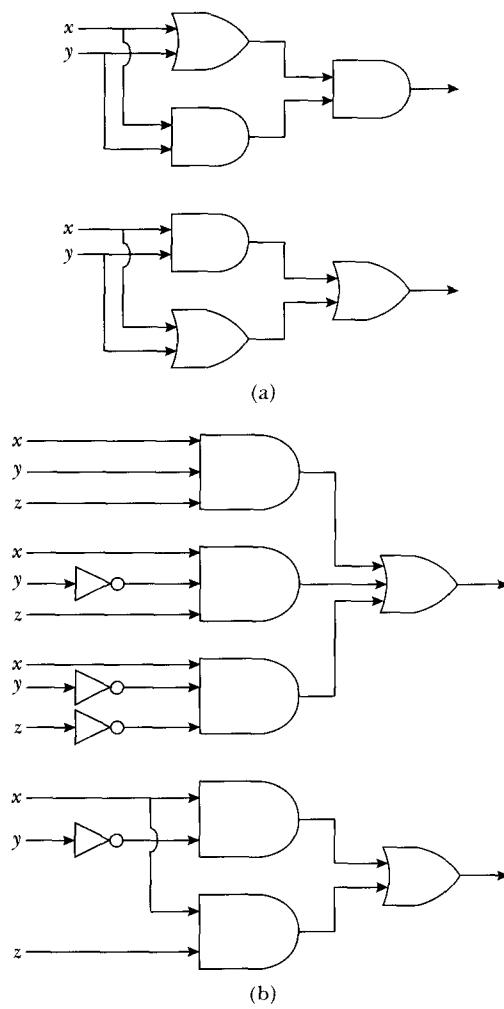


FIGURE 12.56 Circuits

13. Show that any circuit which is designed by using NOT, AND, and OR gates can also be designed using only NOR gates by showing that NOT, AND, and OR gates can be implemented using NOR gates.

Use K-maps to minimize the Boolean expressions in Exercises 14–22.

14.  $xy' + x'y + x'y'$
15.  $xy + xy' + x'y'$
16.  $xyz + xy'z + xyz' + xy'z' + x'y'z'$
17.  $xyz + xy'z + x'y'z' + x'yz'$
18.  $x'y'z + x'yz + xyz' + x'yz'$
19.  $xy'z + x'y'z + x'yz$
20.  $xyzw + xy'zw + x'y'zw + x'yz'w + xyz'w + xy'z'w + x'y'z'w + x'yzw + x'y'zw'$
21.  $xyz'w + x'y'zw + x'y'zw' + x'y'z'w + x'y'z'w + x'yzw + x'y'zw'$
22.  $xyzw + xyzw' + xyz'w' + xy'zw + xy'zw' + x'yzw + x'yzw' + x'y'z'w' + x'y'z'w$

## ► PROGRAMMING EXERCISES

1. Write a program that takes as input the values of two Boolean variables, say  $x$  and  $y$ , and a Boolean operator,  $+$ ,  $\cdot$ , or  $'$ . The program outputs the value of the corresponding expression. For example, if  $x = 0$ ,  $y = 1$ , and the Boolean operator is  $+$ , then the program outputs 1.
2. Write a program that takes as input the values of a Boolean function of up to four variables. The program outputs the DNF, i.e., the sum-of-product form, of the Boolean function.
3. Write a program that takes as input the values of a Boolean function of up to four variables. The program outputs the CNF of the Boolean function.
4. Write a program that takes a Boolean expression of up to four variables in the sum-of-product form. The program then outputs the corresponding K-map.
5. Write a program that takes as input the values of a Boolean function of up to four variables. The program then outputs the corresponding K-map.

## Finite Automata and Languages

**The objectives of this chapter are to:**

- Learn about various types of automata and their basic properties
- Learn about regular grammars and their relationship with various types of automata
- Become familiar with finite state machines
- Learn about various types of grammars

In a programming course, we learn how to write programs using the syntax rules of the programming language. Once the program is written, it is usually compiled to verify that the program follows the syntax rules of the programming language. The syntax rules of the programming language are also known as the grammar rules. In this chapter, we discuss special types of grammars and the languages they generate. We also introduce a mathematical model of a machine that can be used to recognize a special type of grammar. In addition, in the second section of this chapter, we describe how these machines can be used to model certain electronic devices and certain things that we use in daily life, such as vending machines.

## **13.1 FINITE AUTOMATA AND REGULAR LANGUAGES**

---

In computer science, we study different types of computer languages, such as Basic, Pascal, and C++. Hence, at the outset it is important to know what is meant by a language. In this section, we discuss a type of a language that can be recognized by special types of machines, which are also introduced in this section. First, however, let us review certain concepts related to languages, some of which were introduced in Chapter 5.

Recall from Chapter 5 that a string or a word over a nonempty finite set  $X$  is a finite sequence of elements of  $X$ . Words are the basic objects used in the definition of languages. The finite set  $X$  on which strings or words are built is called the *alphabet* of the language. We use the symbol  $\Sigma$  to denote an alphabet and the symbol  $\Sigma^+$  to denote the set of all nonempty strings on  $\Sigma$ . We use a new symbol,  $\lambda \notin \Sigma^+$ , which denotes the empty string and define  $\Sigma^* = \Sigma^+ \cup \{\lambda\}$ .

Let  $w = a_1 a_2 \cdots a_k \in \Sigma^*$ , where  $a_1, a_2, \dots, a_k \in \Sigma$ . Then  $k$  is called the **length** of the string  $w = a_1 a_2 \cdots a_k$ , and we express it by writing  $|w| = k$  or  $l(w) = k$ . We assume that  $|\lambda| = 0$ .

---

**DEFINITION 13.1.1** ▶ Let  $\Sigma$  be an alphabet. Any subset of  $\Sigma^*$  is called a **language** on  $\Sigma$ .

---

**REMARK 13.1.2** ▶ Let  $\Sigma$  be an alphabet. When we say that  $X$  is a language on  $\Sigma$ , we mean  $X \subseteq \Sigma^*$ .

From Definition 13.1.1, it follows that any language on  $\Sigma$  is a collection of strings from  $\Sigma^*$ . Therefore, we next look at various properties of strings, some of which were introduced in Chapter 5. Recall that the concatenation operation is used to join strings by placing them end to end to form a new string. More formally, we have the following definition.

---

**DEFINITION 13.1.3** ▶ Let  $\Sigma$  be an alphabet and let  $u, v \in \Sigma^*$ . Then the **concatenation** of  $u$  and  $v$ , written  $uv$ , is a binary operation on  $\Sigma^*$  defined as follows:

- (i) If  $u = \lambda$ , then  $uv = vu = v$ .
- (ii) If  $u = a_1 a_2 \cdots a_k, v = b_1 b_2 \cdots b_r$ , then  $uv$  is the string  $a_1 a_2 \cdots a_k b_1 b_2 \cdots b_r$ .

Using Definition 13.1.3 we can show that  $\Sigma^*$  is a semigroup with identity under the binary operation of concatenation. Moreover,  $|uv| = |u| + |v|$ .

Utilizing the concatenation of two strings, we define the concatenation of two languages of  $\Sigma^*$  as follows.

---

**DEFINITION 13.1.4** ▶ Let  $\Sigma$  be an alphabet and let  $X$  and  $Y$  be two languages on  $\Sigma$ . The **concatenation** of  $X$  and  $Y$ , written  $XY$ , is the set

$$XY = \{uv \mid u \in X, v \in Y\}.$$

It follows that if  $X$  and  $Y$  are languages on  $\Sigma$ , then  $XY$  is a language on  $\Sigma$ .

Let  $X$  be a language on  $\Sigma$ . The concatenation of  $X$  with itself is defined as follows:

$$\begin{aligned} X^0 &= \{\lambda\} \\ X^n &= XX^{n-1}, \quad n \geq 1. \end{aligned}$$

It follows that for  $n > 1$ ,  $X^n$  is the concatenation of  $X$  with itself  $n$  times.

**EXAMPLE 13.1.5**

Let  $\Sigma = \{a, b, c\}$  and let  $X = \{aa, ab\}$ ,  $Y = \{a, ca, \lambda\}$  be languages on  $\Sigma$ . Then

$$\begin{aligned} XY &= \{aaa, aaca, aa, aba, abca, ab\}, \\ Y^0 &= \{\lambda\}, \\ Y^2 &= \{aa, aca, a, caa, caca, ca, \lambda\}. \end{aligned}$$

Because languages over an alphabet  $\Sigma$  are subsets of  $\Sigma^*$ , we can define the union and intersection of languages  $X$  and  $Y$  as the intersection and union of subsets  $X$  and  $Y$  of  $\Sigma^*$ .

---

**DEFINITION 13.1.6** ▶ Let  $X$  be a language on  $\Sigma$ . Then the **Kleene star** of  $X$ , written  $X^*$ , is the language  $X^* = \bigcup_{i=0}^{\infty} X^i$ .

**EXAMPLE 13.1.7**

Let  $\Sigma = \{a, b\}$ . Consider the language  $L = \{bb\}$  consisting of a single string  $bb$  on  $\Sigma$ . Then

$$\begin{aligned} L^* &= \bigcup_{i=0}^{\infty} L^i \\ &= L^0 \cup L^1 \cup L^2 \cup L^3 \cup \dots \\ &= \{\lambda\} \cup \{bb\} \cup \{bbbb\} \cup \{bbbbbb\} \cup \dots \\ &= \{w \in \Sigma^* \mid w \text{ contains only } b's \text{ and the length of } w \text{ is even}\}. \end{aligned}$$

---

**DEFINITION 13.1.8** ▶ Let  $w = a_1a_2 \cdots a_{n-1}a_n$  be a string in  $\Sigma^*$ . The **reversal string**, or the **reversal** of  $w$ , written  $w^R$ , is the string  $w^R = a_n a_{n-1} \cdots a_2 a_1$ . If  $w$  is a string such that  $w^R = w$ , then  $w$  is called a **palindrome**.

**EXAMPLE 13.1.9**

The string  $abcacba$  is a palindrome but the string  $cacabc$  is not a palindrome.

## Deterministic Finite Automata

In this section, we are interested in the study of languages that are accepted by some type of machine. A deterministic finite automaton (pl. automata) is a mathematical model of a machine that accepts certain languages of some alphabet  $\Sigma$ .

---

**DEFINITION 13.1.10** ▶ A **deterministic finite automaton**, or **deterministic finite acceptor (DFA)**, is a quintuple  $M = (Q, \Sigma, q_0, \delta, F)$ , where

- (i)  $Q$  is a finite nonempty set of **states**,
- (ii)  $\Sigma$  is the **input alphabet** (a finite nonempty set of symbols),
- (iii)  $q_0$  is the **initial** (or **start**) **state**, a particular element of  $Q$ ,
- (iv)  $\delta$  is the **state transition function**,  $\delta : Q \times \Sigma \rightarrow Q$ , and
- (v)  $F$  is the set of **final** (or **accepting**) **states**, a (possibly empty) subset of  $Q$ .

**EXAMPLE 13.1.11**

Let  $M = (\{q_0, q_1, q_2\}, \{0, 1\}, q_0, \delta, \{q_2\})$ , where  $\delta$  is defined as follows:

$$\begin{aligned}\delta(q_0, 0) &= q_1, & \delta(q_1, 1) &= q_2, \\ \delta(q_0, 1) &= q_0, & \delta(q_2, 0) &= q_0, \\ \delta(q_1, 0) &= q_2, & \delta(q_2, 1) &= q_1.\end{aligned}$$

Then  $M$  is a DFA. Note that for  $M$ ,  $Q = \{q_0, q_1, q_2\}$ ,  $\Sigma = \{0, 1\}$ ,  $F = \{q_2\}$ , and  $q_0$  is the initial state. Moreover,  $\delta(q_0, 0) = q_1$  means that if  $M$  is in the state  $q_0$  and  $M$  receives the input symbol 0, then  $M$  goes to the state  $q_1$ .

A DFA operates in the following way: Initially, we assume that the DFA, say  $M$ , is in the initial state  $q_0$ . It starts operation when it receives an input string, say  $a_1 a_2 \dots a_n$ . When it reads the leftmost input symbol,  $a_1$ , then the transition function  $\delta : Q \times \Sigma \rightarrow Q$  of the DFA changes the state of  $M$ . For example, if  $\delta(q_0, a_1) = q$ , then  $M$  changes its state from  $q_0$  to  $q$ . After processing  $a_1$ ,  $M$  reads the next input symbol, which is  $a_2$ . Now  $M$  is in state  $q$  and the input symbol is  $a_2$ . Suppose  $\delta(q, a_2) = q'$ . Then  $M$  changes its state from  $q$  to  $q'$ , and so on.

**EXAMPLE 13.1.12**

Suppose we have the DFA  $M$  of Example 13.1.11 and suppose the input string is 101.

The first input symbol is 1. Initially,  $M$  is in state  $q_0$ . Now  $\delta(q_0, 1) = q_0$ , so after processing 1,  $M$  is in state  $q_0$ . The next input symbol is 0. Currently,  $M$  is in state  $q_0$ , it reads the next input symbol, 0, and  $\delta(q_0, 0) = q_1$ . Therefore, after processing the second input symbol, 0,  $M$  is in state  $q_1$ . The next input symbol is 1. Currently,  $M$  is in state  $q_1$ , it reads the next input symbol, 1, and  $\delta(q_1, 1) = q_2$ . Therefore, after processing the third input symbol, 1,  $M$  is in state  $q_2$ . It follows that after processing the input string,  $M$  is in state  $q_2$ .

In a similar manner, starting with the initial state  $q_0$ , if the input string is 01101, then after processing this string,  $M$  is in state  $q_1$ .

Notice that, when the DFA receives an input string, to process the string it always starts at the initial state.

The state transition function of a DFA is often described by means of a table, called a **transition table**, constructed as follows: The top row lists the input symbols and the leftmost column lists the states of DFA. For any two states  $q_i, q_j$  and for any input  $a$ ,  $\delta(q_i, a) = q_j$  if and only if the entry at the intersection of the  $q_i$ th row and the  $a$ th column is  $q_j$ .

**EXAMPLE 13.1.13**

The transition table of the DFA in Example 13.1.11, is given by

$\delta$	0	1
$q_0$	$q_1$	$q_0$
$q_1$	$q_2$	$q_2$
$q_2$	$q_0$	$q_1$

Sometimes we describe a DFA with the help of a directed graph, illustrated next.

Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA. We can associate a directed graph  $G = (V, E)$  with  $M$  in the following way: Let

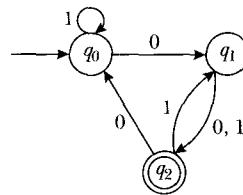
$V = Q$ , i.e., the vertices of  $G$  are the states of  $M$ , and

$E = \{(q_i, q_j) \in V \times V \mid \delta(q_i, a) = q_j \text{ for some } a \in \Sigma\}$ .

However, this description of the directed graph does not fully describe the behavior of  $M$ . So to describe  $M$  completely, we add additional information in the diagram of the associated directed graph as follows:

1. Each state of  $M$ , i.e., each element of  $Q$ , is represented by a small circle and the circle is labeled with the state.
2. The initial state  $q_0$  is identified by an incoming unlabeled arrow not originating from any vertex.
3. The vertices corresponding to the final states are drawn with a double circle.
4. If  $q_i$  and  $q_j$  are two states such that  $\delta(q_i, a) = q_j$  for some  $a \in \Sigma$ , then the arc from  $q_i$  to  $q_j$  is labeled  $a$ .
5. If  $\delta(q_i, a) = q_j$  and  $\delta(q_i, b) = q_j$  for  $a, b \in \Sigma$ , then the arc from  $q_i$  to  $q_j$  is labeled  $a, b$ .

The diagram of the directed graph corresponding to the DFA  $M$  drawn this way is called the **state diagram**, or the **transition diagram**, of  $M$ . Usually, we denote this diagram by  $G_M$ . As an example, the transition diagram of Example 13.1.11 is shown in Figure 13.1.



**FIGURE 13.1**  
Transition diagram

From Figure 13.1, it follows that the transition diagram  $G_M$  gives us a clear and intuitive picture of the DFA  $M$ .

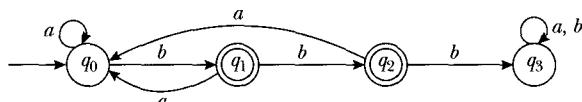
Let us consider another example of a DFA.

#### EXAMPLE 13.1.14

Let  $M = (\{q_0, q_1, q_2, q_3\}, \{a, b\}, q_0, \delta, \{q_1, q_2\})$ , where  $\delta$  is given by the table

$\delta$	$a$	$b$
$q_0$	$q_0$	$q_1$
$q_1$	$q_0$	$q_2$
$q_2$	$q_0$	$q_3$
$q_3$	$q_3$	$q_3$

The transition diagram of this DFA is shown in Figure 13.2.



**FIGURE 13.2** Transition diagram

Notice that  $Q = \{q_0, q_1, q_2, q_3\}$ ,  $\Sigma = \{a, b\}$ , and  $F = \{q_1, q_2\}$ .

Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA and  $w \in \Sigma^*$ . Earlier, starting with the initial state  $q_0$ , we described how to process  $w$ . However, in general, to describe the behavior of  $M$  on  $w$  we extend the transition function  $\delta$  to apply to a state and

a string rather than a state and an input symbol. More formally, we have the following definition.

**DEFINITION 13.1.15** ▶ Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA. The **extended transition function** for  $M$ , written  $\delta^*$ , is the function

$$\delta^* : Q \times \Sigma^* \rightarrow Q$$

defined recursively as follows:

- (i)  $\delta^*(q, \lambda) = q$  for all  $q \in Q$  ( $\lambda$  denotes the empty string in  $\Sigma^*$ ),
- (ii)  $\delta^*(q, wa) = \delta(\delta^*(q, w), a)$  for all  $w \in \Sigma^*$ ,  $a \in \Sigma$ , and  $q \in Q$ .

**REMARK 13.1.16** ▶ Note that for any  $a \in \Sigma$ ,

$$\delta^*(q, a) = \delta^*(q, \lambda a) = \delta(\delta^*(q, \lambda), a) = \delta(q, a).$$

Thus,  $\delta^*$  extends  $\delta$  from single letters to strings.

**EXAMPLE 13.1.17**

Consider the DFA of Example 13.1.14 (see Figure 13.2). Let  $w = abba \in \Sigma^*$ . Then

$$\begin{aligned} \delta^*(q_1, abba) &= \delta(\delta^*(q_1, abb), a) \\ \delta^*(q_1, abb) &= \delta(\delta^*(q_1, ab), b) \\ \delta^*(q_1, ab) &= \delta(\delta^*(q_1, a), b) \\ \delta^*(q_1, a) &= \delta(q_1, a) \\ &= q_0. \end{aligned}$$

Substitute backward to get

$$\begin{aligned} \delta^*(q_1, ab) &= \delta(\delta^*(q_1, a), b) \\ &= \delta(q_0, b) \\ &= q_1, \\ \delta^*(q_1, abb) &= \delta(\delta^*(q_1, ab), b) \\ &= \delta(q_1, b) \\ &= q_2, \\ \delta^*(q_1, abba) &= \delta(\delta^*(q_1, abb), a) \\ &= \delta(q_2, a) \\ &= q_0. \end{aligned}$$

Hence,  $\delta^*(q_1, abba) = q_0$ .

**Lemma 13.1.18:** Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA. Then for any  $u, v \in \Sigma^*$  and for any  $q \in Q$ ,

$$\delta^*(q, uv) = \delta^*(\delta^*(q, u), v).$$

**Proof:** We prove the result by induction on  $n = |v|$ , the length of  $v$ .

*Basis step:* Let  $n = 0$ . Then  $v = \lambda$  and

$$\delta^*(q, uv) = \delta^*(q, u\lambda) = \delta^*(q, u) = \delta^*(\delta^*(q, u), \lambda).$$

Thus, if  $|v| = 0$ , then  $\delta^*(q, uv) = \delta^*(\delta^*(q, u), v)$ .

*Inductive hypothesis:* Let  $n$  be a nonnegative integer and  $\delta^*(q, uv) = \delta^*(\delta^*(q, u), v)$  for all  $v \in \Sigma^*$  such that  $|v| = n$  and for all  $u \in \Sigma^*$ .

*Inductive step:* Let  $w$  be a string on  $\Sigma$  such that  $|w| = n + 1$  and  $u$  be any string on  $\Sigma$ . There exists  $v \in \Sigma^*$  and  $a \in \Sigma$  such that  $w = va$  and  $|v| = n$ . Let us write  $\delta^*(q, u) = q_1$ . Now

$$\begin{aligned}\delta^*(q, uw) &= \delta^*(q, uva) \\&= \delta(\delta^*(q, uv), a) && \text{by the definition of } \delta^* \\&= \delta(\delta^*(\delta^*(q, u), v), a) && \text{by the inductive hypothesis} \\&= \delta(\delta^*(q_1, v), a) \\&= \delta^*(q_1, va) && \text{by the definition of } \delta^* \\&= \delta^*(q_1, w) \\&= \delta^*(\delta^*(q, u), w).\end{aligned}$$

Thus, the result is true for a string of length  $n + 1$ . The result now follows by induction. ■

**DEFINITION 13.1.19** ▶ Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA and  $w \in \Sigma^*$ . Then  $w$  is said to be **accepted** by  $M = (Q, \Sigma, q_0, \delta, F)$  if  $\delta^*(q_0, w) \in F$ .

**EXAMPLE 13.1.20**

Consider the DFA of Example 13.1.14. Consider the string  $w = abb \in \Sigma^*$ . Now  $\delta^*(q_0, abb) = q_2 \in F$ . Hence,  $w = abb$  is accepted by  $M$ .

**DEFINITION 13.1.21** ▶ Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA. The **language accepted by  $M$** , written  $L(M)$ , is the set

$$L(M) = \{w \in \Sigma^* \mid \delta^*(q_0, w) \in F\}.$$

**DEFINITION 13.1.22** ▶ A language  $A$  on the alphabet  $\Sigma$  is said to be a **regular language** if there exists a DFA  $M$  such that  $L(M) = A$ .

We now consider an example of a regular language.

**EXAMPLE 13.1.23**

Let

$$L = \{w \mid w \in \{0, 1\}^* \text{ and } w \text{ does not contain three consecutive 1's}\}.$$

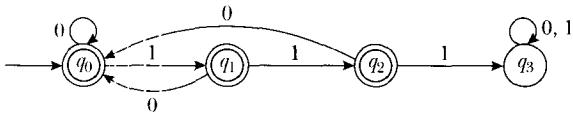
We show that  $L$  is a regular language. To show this we find a DFA  $M = (Q, \Sigma, q_0, \delta, F)$  such that  $L(M) = A$ .

Let us consider the DFA  $M = (Q, \Sigma, q_0, \delta, F)$ , where  $Q = \{q_0, q_1, q_2, q_3\}$ ,  $\Sigma = \{0, 1\}$ ,  $F = \{q_0, q_1, q_2\}$ , and  $\delta$  is given by

$\delta$	0	1
$q_0$	$q_0$	$q_1$
$q_1$	$q_0$	$q_2$
$q_2$	$q_0$	$q_3$
$q_3$	$q_3$	$q_3$

The transition diagram of  $M$  is given in Figure 13.3.

Let us verify that  $M$  does indeed accept the specified language. Note that as long as three consecutive 1's have not been read,  $M$  is in one of the states  $q_0$ ,  $q_1$ ,

FIGURE 13.3 Transition diagram of  $M$ 

or  $q_2$ . Moreover, whenever a 0 is read and  $M$  is in state  $q_0$ ,  $q_1$ , or  $q_2$ ,  $M$  returns to its initial state  $q_0$ . Because states  $q_0$ ,  $q_1$ , and  $q_2$  are also the final states, any input string not containing three consecutive 1's is accepted by  $M$ . However, if a string contains three consecutive 1's, then these consecutive 1's will take  $M$  to state  $q_3$ . Also,  $q_3$  is not a final state. Once  $M$  is in state  $q_3$ , it remains in this state regardless of the symbols in the rest of the input string. Hence,  $L(M) = L$ .

Let  $L \subseteq \Sigma^*$  and  $\Gamma$  be an alphabet such that  $\Sigma \subseteq \Gamma$ . Suppose  $L$  is a regular language. Then there exists a DFA  $M = (Q, \Sigma, q_0, \delta, F)$  such that  $L(M) = L$ . Choose a state  $q$  such that  $q \notin Q$ . Let  $M' = (Q', \Gamma, q_0, \delta', F)$ , where  $Q' = Q \cup \{q\}$  and  $\delta'$  is defined as follows:

$$\begin{aligned}\delta'(p, a) &= \delta(p, a) \quad \text{for all } p \in Q \text{ and } a \in \Sigma, \\ \delta'(p, a) &= q \quad \text{for all } p \in Q \text{ and } a \in \Gamma \setminus \Sigma, \\ \delta'(q, a) &= q \quad \text{for all } a \in \Gamma.\end{aligned}$$

We can show that  $L(M) = L(M')$ .

Now suppose that  $M_1 = (Q_1, \Gamma, q_0, \delta', F)$  is a DFA such that  $L(M_1) = L$ . Let  $M = (Q, \Sigma, q_0, \delta, F)$ , where  $Q = Q_1$  and  $\delta = \delta'|_{Q \times \Sigma}$ . That is,  $\delta$  is the restriction of  $\delta'$  to  $Q \times \Sigma$ . Then we can show that  $L(M) = L$ .

From these observations, it follows that the notion of regular language does not depend on the particular alphabet.

We now give an example of a language that is not regular.

#### EXAMPLE 13.1.24

The set  $L = \{a^i b^i \mid i \geq 0\}$  is not a regular language.

Suppose that  $L$  is a regular language. Then there exists a DFA  $M = (Q, \Sigma, q_0, \delta, F)$  such that  $L(M) = L$ . The number of states of  $M$  is finite. Suppose that the number of states of  $M$  is  $k$ , i.e.,  $|Q| = k$ . Let  $\delta^*(q_0, a^i) = q_{0i}$  for  $i \geq 0$ . Now  $q_{0i} \in Q$ , for  $i = 0, 1, 2, \dots, k+1$ . Because  $Q$  contains only  $k$  states, by the pigeonhole principle (see Chapter 7), there exist integers  $i$  and  $j$  with  $i > j \geq 0$  such that  $q_{0i} = q_{0j}$ . Now

$$\delta^*(q_0, a^i b^i) = \delta^*(\delta^*(q_0, a^i), b^i) = \delta^*(\delta^*(q_0, a^j), b^i) = \delta^*(q_0, a^j b^i).$$

This implies  $a^j b^i \in L$ . This is a contradiction because  $i > j$ . Therefore,  $L$  is not a regular language.

Let us now consider the DFA  $M$  (see Figure 13.4) given by the following transition diagram  $G_M$ .

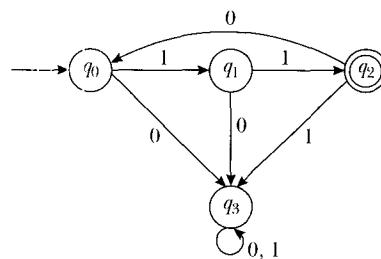


FIGURE 13.4 Transition diagram

In this  $G_M$ , we have a directed walk

$$P : q_1 \xrightarrow{1} q_2 \xrightarrow{0} q_0 \xrightarrow{1} q_1 \xrightarrow{0} q_3$$

from  $q_1$  to  $q_3$ . Here the string  $w = 1010$  is called the label of the walk  $P$ . Now

$$\begin{aligned}\delta^*(q_1, 1010) &= \delta(\delta^*(q_1, 101), 0), \\ \delta^*(q_1, 101) &= \delta(\delta^*(q_1, 10), 1), \\ \delta^*(q_1, 10) &= \delta(\delta^*(q_1, 1), 0), \\ \delta^*(q_1, 1) &= \delta(q_1, 1) \\ &= q_2.\end{aligned}$$

Thus,

$$\begin{aligned}\delta^*(q_1, 10) &= \delta(q_2, 0) = q_0, \\ \delta^*(q_1, 101) &= \delta(q_0, 1) = q_1, \\ \delta^*(q_1, 1010) &= \delta(q_1, 0) = q_3.\end{aligned}$$

This shows that when  $P$  is a directed walk from  $q_1$  to  $q_3$  with the label of the string  $w$ , we also see that  $\delta^*(q_1, w) = q_3$ .

Next consider the string  $w = 1101$ . For this string  $\delta^*(q_0, 1101) = q_1$ , and in the transition diagram we find a directed walk

$$P : q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_2 \xrightarrow{0} q_0 \xrightarrow{1} q_1$$

with the label 1101.

**DEFINITION 13.1.25** ▶ Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA and let  $G_M$  be its transition diagram. If

$$P : q_{i_0} \xrightarrow{a_1} q_{i_1} \xrightarrow{a_2} q_{i_2} \xrightarrow{a_3} \cdots \xrightarrow{a_{n-1}} q_{i_{n-1}} \xrightarrow{a_n} q_{i_n}$$

is a directed walk, where  $\delta(q_{i_{k-1}}, a_k) = q_{i_k}$ , then the string  $a_1 a_2 \cdots a_{n-1} a_n$  is called the **label of the directed walk  $P$** .

**Theorem 13.1.26:** Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA and let  $G_M$  be its transition diagram. Then for any two states  $q_i$  and  $q_j$ , and a string  $w \in \Sigma^*$ ,  $\delta^*(q_i, w) = q_j$  if and only if there is a directed walk  $P$  in  $G_M$  with the label  $w$  from  $q_i$  to  $q_j$ .

**Proof:** Suppose  $\delta^*(q_i, w) = q_j$ . We prove by induction, on the length of  $w$ , that there is a directed walk  $P$  in  $G_M$  with the label  $w$  from  $q_i$  to  $q_j$ .

*Basis step:* Let  $|w| = 1$ . Then  $w = a \in \Sigma$  and  $q_j = \delta^*(q_i, w) = \delta(q_i, a)$ . Hence, there is an arc from  $q_i$  to  $q_j$  with the label  $a$ .

*Inductive hypothesis:* Suppose that for any two states  $q_i, q_j$  and a string  $u \in \Sigma^*$  such that  $|u| \leq n$ , if  $\delta^*(q_i, u) = q_j$ , then there is a directed walk  $P$  in  $G_M$  with the label  $u$  from  $q_i$  to  $q_j$ .

*Inductive step:* Let  $q_i, q_j$  be two states and assume  $w \in \Sigma^*$  is of length  $n + 1$ . Then there exist  $u \in \Sigma^*$  and  $a \in \Sigma$  such that the length of  $u$  is  $n$  and  $w = ua$ . Suppose that  $\delta^*(q_i, w) = q_j$ . Then  $\delta(\delta^*(q_i, u), a) = q_j$ . Let  $\delta^*(q_i, u) = q_l$ . Because the length of  $u$  is  $n$ , by the inductive hypothesis, there exists a directed walk  $P_1$  from  $q_i$  to  $q_l$  with the label  $u$ . Now  $\delta(q_l, a) = q_j$  shows that there is an

arc with the label  $a$  from  $q_i$  to  $q_j$ . Now the walk  $P_1$  followed by the arc  $(q_i, q_j)$  gives a directed walk from  $q_i$  to  $q_j$  with the label  $w$ . Thus, the result is true for strings of length  $n + 1$ . The result now follows by induction.

Conversely, assume that for any two states  $q_i, q_j$  there is a directed walk  $P$  in  $G_M$  with the label  $w$  from  $q_i$  to  $q_j$ . We leave it as an exercise to prove by induction on length of  $w$  that  $\delta^*(q_i, w) = q_j$ . ■

Theorem 13.1.26 is helpful in describing the elements accepted by a DFA.

Consider the DFA given by the transition diagram in Figure 13.4. In this transition diagram, we find a directed walk

$$P : q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_2 \xrightarrow{0} q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_2$$

with the label 11011. Then from Theorem 13.1.26, it follows that  $\delta^*(q_0, 11011) = q_2$ . Because  $q_2$  is a final state of this DFA, this DFA accepts 11011. Consider now the string  $w = 0010$ . For this string

$$q_0 \xrightarrow{0} q_3 \xrightarrow{0} q_3 \xrightarrow{1} q_3 \xrightarrow{0} q_3$$

is the only directed walk from  $q_0$  with the label 0010. Now  $q_3$  is not a final state. Hence, 0010 is not accepted by this DFA  $M$ .

We now prove the following theorem, which is known as the **pumping lemma for regular languages**.

**Theorem 13.1.27:** Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA with  $n$  states. Let  $w \in L(M)$  be such that  $|w| \geq n$ . Then there exist strings  $x, y, z \in \Sigma^*$  such that  $w = xyz$ ,  $|xy| \leq n$ ,  $|y| \geq 1$  and  $xy^kz \in L(M)$  for all integers  $k \geq 0$ .

**Proof:** Let  $w = a_1 a_2 \cdots a_m$ , where  $a_i \in \Sigma$ . Then  $|w| = m$ . From the assumption,  $m \geq n$ . Because  $w \in L(M)$ , there exists a directed walk

$$P : q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \cdots \xrightarrow{a_j} q_j \xrightarrow{a_{j+1}} q_{j+1} \rightarrow \cdots \rightarrow q_{m-1} \xrightarrow{a_m} q_m$$

with the label  $a_1 a_2 \cdots a_m$ , where  $q_m$  is a final state. Let

$$\delta^*(q_0, a_1 a_2 \cdots a_i) = q_i, \quad i = 1, 2, \dots, m.$$

Then  $q_0, q_1, \dots, q_m \in Q$ . Now  $m \geq n$ , so  $q_0, q_1, \dots, q_n \in Q$ . Because  $m \geq n$  and  $|Q| = n$ , by the pigeonhole principle, not all  $q_0, q_1, \dots, q_m$  are distinct. There exist integers  $j$  and  $t$  such that  $0 \leq j < t \leq n$  and  $q_j = q_t$ , where  $\delta^*(q_0, a_1 a_2 \cdots a_j) = q_j$ . If  $j > 0$ ,  $\delta^*(q_0, a_1 a_2 \cdots a_t) = q_t$ , and  $\delta^*(q_0, \lambda) = q_0$ .

If  $j = 0$ , take  $x = \lambda$ ,  $y = a_1 a_2 \cdots a_t$ , and  $z = a_{t+1} a_{t+2} \cdots a_m$  so that  $w = xyz$ .

If  $j > 0$ , take  $x = a_1 a_2 \cdots a_j$ ,  $y = a_{j+1} a_{j+2} \cdots a_t$ , and  $z = a_{t+1} a_{t+2} \cdots a_m$  so that  $w = xyz$ .

Note that  $|y| = t - j > 0$ ,  $|xy| = t \leq n$ , and

$$\delta^*(q_0, xy) = \delta^*(q_0, x). \quad (13.1)$$

We show, by induction on  $k$ , that  $\delta^*(q_0, x) = \delta^*(q_0, xy^k)$  for each integer  $k \geq 0$ .

*Basis step:* For  $k = 0$ ,  $\delta^*(q_0, xy^0) = \delta^*(q_0, xy^0) = \delta^*(q_0, x\lambda) = \delta^*(q_0, x)$ . Thus, the result is true for  $k = 0$ .

*Inductive hypothesis:* Suppose  $\delta^*(q_0, x) = \delta^*(q_0, xy^k)$  for some nonnegative integer  $k$ .

*Inductive step:* Consider  $\delta^*(q_0, xy^{k+1})$ . Now

$$\begin{aligned} & \delta^*(q_0, xy^{k+1}) \\ &= \delta^*(\delta^*(q_0, xy^k), y) \\ &= \delta^*(\delta^*(q_0, x), y) \quad \text{by the induction hypothesis } \delta^*(q_0, x) = \delta^*(q_0, xy^k) \\ &= \delta^*(q_0, xy) \\ &= \delta^*(q_0, x) \quad \text{by 13.1} \end{aligned}$$

Therefore, the result is true for  $k + 1$ . Thus, by induction,  $\delta^*(q_0, x) = \delta^*(q_0, xy^k)$  for each integer  $k \geq 0$ .

Now

$$\begin{aligned} \delta^*(q_0, w) &= \delta^*(q_0, xyz) \\ &= \delta^*(\delta^*(q_0, xy), z) \\ &= \delta^*(\delta^*(q_0, x), z) \\ &= \delta^*(\delta^*(q_0, xy^k), z) \\ &= \delta^*(q_0, xy^k z) \end{aligned}$$

for all integers  $k \geq 0$ . Hence, if  $w \in L(M)$ , then  $xy^k z \in L(M)$  for all integers  $k \geq 0$ . ■

The pumping lemma is a powerful tool for showing that a language is not regular. The following example illustrates how to use the pumping lemma.

#### EXAMPLE 13.1.28

In Example 13.1.24 we proved that the language  $L = \{a^m b^m \mid m \geq 0\}$  over  $\Sigma = \{a, b\}$  is not a regular language. Here we use the pumping lemma, Theorem 13.1.27, to prove the same result.

Suppose  $L$  is regular. Then there exists a DFA  $M = (Q, \Sigma, q_0, \delta, F)$  such that  $L(M) = L$ . Let  $|Q| = n$ . Now  $w = a^n b^n \in L = L(M)$  and  $|w| = 2n > n$ . Hence, by the pumping lemma, there exist  $x, y, z \in \Sigma^*$  such that  $w = xyz$ ,  $|xy| \leq n$ ,  $|y| \geq 1$ , and  $xy^k z \in L(M)$  for all  $k = 0, 1, 2, \dots$ . Because  $|xy| \leq n$ ,  $xy$  only contains  $a$ 's. Also,  $|y| \geq 1$ . Hence,  $y = a^i$  for some  $i \geq 1$ . Now  $xy^2 z \in L(M)$  and  $|xy^2 z| = |xyz| + |y| = 2n + i = n + n + i$ . Hence, in  $xy^2 z$  the number of  $a$ 's is  $n + i$ , but the number of  $b$ 's is  $n$ . As a result,  $xy^2 z \notin L = L(M)$ , a contradiction. Hence,  $L$  is not regular.

#### EXAMPLE 13.1.29

Consider the language  $L = \{0^{i^2} \mid i \text{ is a positive integer}\}$  over the alphabet  $\{0, 1\}$ . We show that  $L$  is not a regular language.

The language  $L$  contains all strings of the form  $0, 0000, 00000000$ , i.e., if  $w \in L$ , then  $w$  consists of only 0's and the length of  $w$  is the square of an integer. Assume that  $L$  is regular. Then there exists a DFA  $M = (Q, \Sigma, q_0, \delta, F)$  such that  $L(M) = L$ . Let  $|Q| = n$ . Now  $w = 0^{n^2} \in L$  and  $|w| = n^2 \geq n$ . Hence, by the pumping lemma, there exist  $x, y, z \in \Sigma^*$  such that  $w = xyz$ ,  $|xy| \leq n$ ,  $|y| \geq 1$ , and  $xy^k z \in L(M)$  for all  $k = 0, 1, 2, \dots$ . In particular,  $xy^2 z \in L(M)$ . Now  $n^2 < |xy^2 z| \leq n^2 + n < (n + 1)^2$ . That is, the length of  $xy^2 z$  lies strictly between  $n^2$  and  $(n + 1)^2$ , so  $|xy^2 z|$  is not the square of an integer. Thus,  $xy^2 z \notin L = L(M)$ , a contradiction. Hence,  $L$  is not regular.

For additional examples of nonregular languages see the Worked-Out Exercises at the end of this section.

## Applications of the Pumping Lemma

In this section, we show some interesting applications of the pumping lemma.

**Theorem 13.1.30:** Let  $M$  be a DFA with  $n$  states. If  $L(M) \neq \emptyset$ , then there is a string  $w \in L(M)$  such that  $|w| < n$ .

**Proof:** Let  $w$  be a string in  $L(M)$  of the shortest possible length. Suppose  $|w| \geq n$ . By the pumping lemma, there exist  $x, y, z \in \Sigma^*$  such that  $w = xyz$ ,  $|xy| \leq n$ ,  $|y| \geq 1$ , and  $xy^kz \in L(M)$  for all  $k = 0, 1, 2, \dots$ . Hence, in particular,  $xz \in L(M)$ . Now

$$\begin{aligned} |xz| &= |x| + |z| \\ &< |x| + |y| + |z| \quad \text{because } |y| \geq 1 \\ &= |xyz| \\ &= |w|. \end{aligned}$$

This contradicts the choice of  $w$ . Hence,  $|w| < n$ . ■

Theorem 13.1.30 furnishes an algorithm to determine whether the language accepted by a DFA is empty.

Consider the DFA  $M = (Q, \Sigma, q_0, \delta, F)$ . Let  $|Q| = n$ . Let  $T = \{w \in \Sigma^* \mid |w| < n\}$ . Because  $\Sigma$  is finite,  $T$  is a finite subset of  $\Sigma^*$ . We now determine whether the strings of  $T$  are accepted by  $M$ . If none of the strings of  $T$  is accepted by  $M$ , then by Theorem 13.1.30,  $L(M) = \emptyset$ .

**Theorem 13.1.31:** Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA with  $n$  states.

Then  $L(M)$  is infinite if and only if  $L(M)$  contains a string  $x$  such that  $n \leq |x| < 2n$ .

**Proof:** Let  $w \in L(M)$  be such that  $n \leq |w| < 2n$ . By the pumping lemma, there exist strings  $x, y, z \in \Sigma^*$  such that  $w = xyz$ ,  $|xy| \leq n$ ,  $|y| \geq 1$ , and  $xy^kz \in L(M)$  for all  $k = 0, 1, 2, \dots$ . Because  $xy^kz$  are all distinct for all  $k = 0, 1, 2, \dots$ , it follows that  $L(M)$  is infinite.

Conversely, if  $L(M)$  is infinite, then there exists a string  $x$  in  $L(M)$  such that  $|x| \geq n$ . If  $|x| < 2n$ , the result is true. Suppose  $|x| \geq 2n$ . Let  $T = \{w \in \Sigma^* \mid |w| \geq 2n\}$ . Because  $x \in T$ ,  $T \neq \emptyset$ . There exists a string  $w \in T$  such that  $|w| \leq |v|$  for all  $v \in T$ . Because  $|w| \geq 2n$ , by the pumping lemma, there exist  $u, v, w \in \Sigma^*$  such that  $w = uvz$ ,  $|uv| \leq n$ ,  $|v| \geq 1$ , and  $uv^kz \in L(M)$  for all  $k = 0, 1, 2, \dots$ . In particular,  $uz \in L(M)$ . Because  $|uz| < |w|$ ,  $uz \notin T$ . Hence,  $|uz| < 2n$ .

We now show that  $|uz| \geq n$ . Suppose  $|uz| < n$ . Now  $|uv| \leq n$  implies that  $|v| \leq n$ . Thus,

$$|w| = |uvz| = |uz| + |v| < n + n = 2n.$$

This contradicts the fact that  $w \in T$ . Thus,  $|uz| \geq n$ . Hence,  $uz \in L(M)$  and  $n \leq |uz| < 2n$ . ■

Theorem 13.1.31 furnishes an algorithm for determining whether the language,  $L(M)$ , accepted by a DFA  $M$  is infinite. For this, we determine if any string of the length between  $n$  and  $2n - 1$  is in  $L(M)$ .

## Algebraic Properties of Regular Languages

In this section we study some algebraic properties of regular languages. Let  $\Sigma$  be an alphabet and let  $R_\Sigma$  denote the set of all regular languages on  $\Sigma$ .

**Theorem 13.1.32:** If  $L_1$  and  $L_2$  are two regular languages on  $\Sigma$ , then the following assertions hold.

- (i)  $L_1 \cap L_2$  is a regular language.
- (ii)  $L_1 \cup L_2$  is a regular language.
- (iii)  $\Sigma^* - L_1$  is a regular language.

**Proof:** Because  $L_1$  and  $L_2$  are regular languages on  $\Sigma$ , there exist DFA  $M_1 = (Q_1, \Sigma, q_{01}, \delta_1, F_1)$  and DFA  $M_2 = (Q_2, \Sigma, q_{02}, \delta_2, F_2)$  such that  $L(M_1) = L_1$  and  $L(M_2) = L_2$ .

- (i) Construct a new DFA,  $M = (Q, \Sigma, q_0, \delta, F)$ , as follows:  $Q = Q_1 \times Q_2$  (the set of states),  $q_0 = (q_{01}, q_{02})$  (the initial state),  $F = F_1 \times F_2$  (the set of final states), and the transition function  $\delta : (Q_1 \times Q_2) \times \Sigma \rightarrow Q_1 \times Q_2$  is defined by

$$\delta((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))$$

for all  $(q_1, q_2) \in Q_1 \times Q_2$  and  $a \in \Sigma$ .

We show that  $L(M) = L_1 \cap L_2$ . To do this, first we show that the extended transition function  $\delta^*$  of  $\delta$  satisfies the following:

$$\delta^*((q_1, q_2), w) = (\delta_1^*(q_1, w), \delta_2^*(q_2, w))$$

for all  $(q_1, q_2) \in Q_1 \times Q_2$  and  $w \in \Sigma^*$ , where  $\delta_i^*$  denotes the extended transition function of  $\delta_i$  for  $i = 1, 2, \dots$

Let us prove this by the method of induction on the length  $|w|$  of  $w$ .

*Basis step:* Suppose  $|w| = 0$ . Then  $w = \lambda$  and  $\delta^*((q_1, q_2), \lambda) = (q_1, q_2) = (\delta_1^*(q_1, \lambda), \delta_2^*(q_2, \lambda))$ .

*Inductive hypothesis:* Assume that  $\delta^*((q_1, q_2), u) = (\delta_1^*(q_1, u), \delta_2^*(q_2, u))$  for all  $(q_1, q_2) \in Q_1 \times Q_2$  and for all  $u \in \Sigma^*$  of length  $k$ .

*Inductive step:* Let  $w$  be a string of length  $k + 1$ . Then  $w = ua$ , where  $u \in \Sigma^*$ ,  $a \in \Sigma$ , and  $|u| = k$ . Now

$$\begin{aligned} & \delta^*((q_1, q_2), w) \\ &= \delta^*((q_1, q_2), ua) \\ &= \delta(\delta^*((q_1, q_2), u), a) \\ &= \delta((\delta_1^*(q_1, u), \delta_2^*(q_2, u)), a) && \text{by the inductive hypothesis} \\ &= (\delta_1(\delta_1^*(q_1, u), a), \delta_2(\delta_2^*(q_2, u), a)) && \text{by the definition of } \delta \\ &= (\delta_1^*(q_1, ua), \delta_2^*(q_2, ua)) && \text{by the definition of } \delta_i^* \end{aligned}$$

Hence, by induction it follows that  $\delta^*((q_1, q_2), w) = (\delta_1^*(q_1, w), \delta_2^*(q_2, w))$  for all  $(q_1, q_2) \in Q_1 \times Q_2$  and  $w \in \Sigma^*$ .

Now

$$\begin{aligned} L(M) &= \{w \in \Sigma^* \mid \delta^*(q_0, w) \in F\} \\ &= \{w \in \Sigma^* \mid (\delta_1^*(q_{01}, w), \delta_2^*(q_{02}, w)) \in F_1 \times F_2\} \\ &= \{w \in \Sigma^* \mid \delta_1^*(q_{01}, w) \in F_1 \text{ and } \delta_2^*(q_{02}, w) \in F_2\} \\ &= \{w \in \Sigma^* \mid w \in L(M_1) \text{ and } w \in L(M_2)\} \\ &= L(M_1) \cap L(M_2) \\ &= L_1 \cap L_2. \end{aligned}$$

Thus, it follows that  $L_1 \cap L_2$  is a regular language.

- (ii) For  $L_1 \cup L_2$ , construct a new DFA,  $M = (Q, \Sigma, q_0, \delta, F)$ , as follows:  $Q = Q_1 \times Q_2$  (the set of states),  $q_0 = (q_{01}, q_{02})$  (the initial state),  $F = (F_1 \times Q_2) \cup (Q_1 \times F_2)$  (the set of final states), and the transition function  $\delta : (Q_1 \times Q_2) \times \Sigma \rightarrow Q_1 \times Q_2$  is defined by

$$\delta((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))$$

for all  $(q_1, q_2) \in Q_1 \times Q_2$  and  $a \in \Sigma$ . Then proceeding as in part (i), we can show that  $L_1 \cup L_2$  is a regular language. We leave the details as an exercise.

- (iii) For  $\Sigma^* - L_1$ , construct a new DFA,  $M = (Q, \Sigma, q_0, \delta, F)$ , as follows:  $Q = Q_1$ ,  $q_0 = q_{01}$ ,  $F = Q_1 - F_1$ , and  $\delta = \delta_1$ . Then we can show that  $L(M) = \Sigma^* - L_1$ . (We leave the details as an exercise.) Hence,  $\Sigma^* - L_1$  is a regular language. ■

**Corollary 13.1.33:**  $R_\Sigma$  is a Boolean algebra.

## Nondeterministic Finite Automata

**DEFINITION 13.1.34** ► A nondeterministic finite automaton (NDFA) is a quintuple  $M = (Q, \Sigma, q_0, \delta, F)$ , where

- (i)  $Q$  is a finite nonempty set of **states**,
- (ii)  $\Sigma$  is the **input alphabet** (a finite nonempty set of symbols),
- (iii)  $q_0$  is the **initial (or start) state**, a particular element of  $Q$ ,
- (iv)  $\delta$  is the **state transition function**,  $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q)$  ( $\mathcal{P}(Q)$  is the set of all subsets of  $Q$ ), and
- (v)  $F$  is the set of **final (or accepting)** states, a (possibly empty) subset of  $Q$ .

Notice that in an NDFA, the input alphabet, the set of states, the initial state, and even the set of final states are the same as in a DFA. The important difference is contained in the definition of the transition function  $\delta$ . In an NDFA the range of  $\delta$  is a subset of  $\mathcal{P}(Q)$ , the set of all subsets of  $Q$ . That is, for all  $q \in Q$  and  $a \in \Sigma$ ,  $\delta(q, a)$  is not a single state but a subset of  $Q$ . The subset defines the set of all possible states that can be reached by the transition  $\delta$ . For example if  $q$  is a state,  $a$  is an input such that  $\delta(q, a) = \{q_1, q_2\}$ , then we say that if the present state is  $q$  and the input  $a$  is read, then the next state of the NDFA is either  $q_1$  or  $q_2$ .

Like DFA, the transition function  $\delta$  of NDFA can be described by a transition table and the NDFA can be represented by a directed graph with additional

information. In the corresponding directed graph, the vertices are given by the elements of  $Q$ , while there is an arc from  $q_i$  to  $q_j$  with the label  $a$  if  $\delta(q_i, a)$  contains  $q_j$ . We illustrate this in the following example.

**EXAMPLE 13.1.35**

Consider the quintuple  $M = (Q, \Sigma, q_0, \delta, F)$ , where  $Q = \{q_0, q_1, q_2\}$ ,  $\Sigma = \{a, b\}$ ,  $F = \{q_2\}$ , and the transition function  $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q)$  is defined by the transition table

$\delta$	$a$	$b$
$q_0$	$\{q_1\}$	$\emptyset$
$q_1$	$\emptyset$	$\{q_0, q_2\}$
$q_2$	$\emptyset$	$\{q_2\}$

Then  $M$  is an NDFA. The state transition diagram of this NDFA is shown in Figure 13.5.

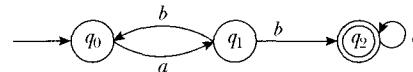


FIGURE 13.5 Transition diagram of NDFA  $M$

As in the case of a DFA, we can extend the transition function of an NDFA to process strings as follows.

**DEFINITION 13.1.36** Given an NDFA  $M = (Q, \Sigma, q_0, \delta, F)$ , the **extended state transition function** of  $M$  is the function  $\delta^* : Q \times \Sigma^* \rightarrow \mathcal{P}(Q)$  defined as follows:

- (i)  $\delta^*(q, \lambda) = \{q\}$  for all  $q \in Q$ ,
- (ii)  $\delta^*(q, wa) = \bigcup_{p \in \delta^*(q, w)} \delta(p, a)$  for all  $q \in Q$ ,  $w \in \Sigma^*$ , and  $a \in \Sigma$ .

**EXAMPLE 13.1.37**

Consider the NDFA of Example 13.1.35. Let  $w = bb$ . Let us determine  $\delta^*(q_1, bb)$ . By the definition of  $\delta^*$ ,

$$\begin{aligned} \delta^*(q_1, bb) &= \bigcup_{p \in \delta^*(q_1, b)} \delta(p, b), \\ \delta^*(q_1, b) &= \delta^*(q_1, \lambda b) = \bigcup_{p \in \delta^*(q_1, \lambda)} \delta(p, b) = \bigcup_{p \in \{q_1\}} \delta(p, b) = \delta(q_1, b) = \{q_0, q_2\}. \end{aligned}$$

Thus,

$$\delta^*(q_1, bb) = \bigcup_{p \in \{q_0, q_2\}} \delta(p, b) = \delta(q_0, b) \cup \delta(q_2, b) = \emptyset \cup \{q_2\} = \{q_2\}.$$

In the transition diagram shown in Figure 13.5, there exists only one directed walk

$$P : q_1 \xrightarrow{b} q_2 \xrightarrow{b} q_2$$

from  $q_1$  to  $q_2$  with the label  $bb$ .

Now consider the following directed walks in Figure 13.5:

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2$$

from  $q_0$  to  $q_2$  with the label  $abab$  and

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_0$$

from  $q_0$  to  $q_0$  with the label  $abab$ . Let us compute  $\delta^*(q_0, abab)$ .

$$\begin{aligned}\delta^*(q_0, abab) &= \bigcup_{p \in \delta^*(q_0, abab)} \delta(p, b), \\ \delta^*(q_0, aba) &= \bigcup_{p \in \delta^*(q_0, aba)} \delta(p, a), \\ \delta^*(q_0, ab) &= \bigcup_{p \in \delta^*(q_0, ab)} \delta(p, b), \\ \delta^*(q_0, a) = \delta^*(q_0, \lambda a) &= \bigcup_{p \in \delta^*(q_0, \lambda)} \delta(p, a) = \bigcup_{p \in \{q_0\}} \delta(p, a) = \delta(q_0, a) = \{q_1\}, \\ \delta^*(q_0, ab) &= \bigcup_{p \in \{q_1\}} \delta(p, b) = \delta(q_1, b) = \{q_0, q_2\}, \\ \delta^*(q_0, aba) &= \bigcup_{p \in \{q_0, q_2\}} \delta(p, a) = \delta(q_0, a) \cup \delta(q_2, a) = \{q_1\} \cup \emptyset = \{q_1\}, \\ \delta^*(q_0, abab) &= \bigcup_{p \in \{q_1\}} \delta(p, b) = \delta(q_1, b) = \{q_0, q_2\}.\end{aligned}$$

Note that there are only two directed walks,

$$q_0 \xrightarrow{c} q_1 \xrightarrow{b} q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2,$$

and

$$q_0 \xrightarrow{c} q_1 \xrightarrow{b} q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_0,$$

from  $q_0$  with the label  $abab$  and  $\delta^*(q_0, abab) = \{q_0, q_2\}$ .

As in a DFA, we can prove that in an NDFA  $M = (Q, \Sigma, q_0, \delta, F)$ , for any vertex  $q$ , and for any input string  $w$ ,  $\delta^*(q, w)$  contains a state  $p$  if and only if there exists a directed walk with the label  $w$  from  $q$  to  $p$ .

In Example 13.1.37, we have  $\delta^*(q_0, abab) = \{q_0, q_2\}$  and also there exist directed walks from  $q_0$  to  $q_0$  and  $q_0$  to  $q_2$  with the label  $abab$ . Moreover, these are the only directed walks with the starting vertex  $q_0$  and label  $abab$ .

---

**DEFINITION 13.1.38** ▶ Let  $M = (Q, \Sigma, q_0, \delta, F)$  be an NDFA and  $w \in \Sigma^*$ . Then  $w$  is said to be **accepted** by  $M$  if

$$\delta^*(q_0, w) \cap F \neq \emptyset.$$

---

**DEFINITION 13.1.39** ▶ Given an NDFA  $M = (Q, \Sigma, q_0, \delta, F)$ , the **language accepted** by  $M$ , written  $L(M)$ , is the set

$$L(M) = \{w \in \Sigma^* \mid \delta^*(q_0, w) \cap F \neq \emptyset\}.$$

From the definition of  $L(M)$  it follows that a string  $w \in L(M)$  if and only if there exists a directed walk labeled  $w$  from the initial state  $q_0$  to a final state.

Consider again the NDFA of Example 13.1.35. Because  $\delta^*(q_0, abab) = \{q_0, q_2\}$ , we have  $\delta^*(q_0, abab) \cap F \neq \emptyset$ . Hence,  $abab$  is accepted by  $M$ .

In the same NDFA, we find that there is a directed walk

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{b} q_2$$

with the label  $ababbb$ . Because  $q_0$  is the initial state and  $q_2$  is a final state, it follows that  $ababbb$  is accepted by  $M$ . However, for the string  $bab$  there is no directed walk from the initial state  $q_0$ . Because there is no directed walk from  $q_0$  to a final state with the label  $bab$ , the NDFA  $M$  does not accept the string  $bab$ .

In fact, if we compute

$$\delta^*(q_0, bab) = \bigcup_{p \in \delta^*(q_0, ba)} \delta(p, b),$$

$$\delta^*(q_0, ba) = \bigcup_{p \in \delta^*(q_0, b)} \delta(p, a),$$

$$\delta^*(q_0, b) = \delta^*(q_0, \lambda b) = \bigcup_{p \in \delta^*(q_0, \lambda)} \delta(p, b) = \bigcup_{p \in \{q_0\}} \delta(p, b) = \delta(q_0, b) = \emptyset,$$

$$\delta^*(q_0, ba) = \bigcup_{p \in \emptyset} \delta(p, a) = \emptyset,$$

$$\delta^*(q_0, bab) = \bigcup_{p \in \emptyset} \delta(p, b) = \emptyset,$$

then  $\delta^*(q_0, bab) \cap F = \emptyset$ . Hence, the string  $bab$  is not accepted by the given NDFA.

---

**DEFINITION 13.1.40** ► Given an NDFA  $M = (Q, \Sigma, q_0, \delta, F)$ , the corresponding DFA  $M^d = (Q^d, \Sigma, q_0^d, F^d)$  is defined as follows:

$$\begin{aligned} Q^d &= \mathcal{P}(Q), \text{ the power set of } Q, \\ q_0^d &= \{q_0\}, \\ F^d &= \{A \in Q^d \mid A \cap F \neq \emptyset\}, \end{aligned}$$

and  $\delta^d : Q^d \times \Sigma \rightarrow Q^d$  is defined by

$$\delta^d(A, a) = \bigcup_{q \in A} \delta(q, a)$$

for all  $A \in Q^d$ ,  $a \in \Sigma$ .

### EXAMPLE 13.1.41

Consider the NDFA  $M = (Q, \Sigma, q_0, \delta, F)$  of Example 13.1.35. In this example, we construct the corresponding DFA  $M^d = (Q^d, \Sigma, q_0^d, \delta^d, F^d)$ . Let

$$\begin{aligned} Q^d &= \mathcal{P}(Q) = \{\emptyset, \{q_0\}, \{q_1\}, \{q_2\}, \{q_0, q_1\}, \{q_0, q_2\}, \{q_1, q_2\}, \{q_0, q_1, q_2\}\}, \\ q_0^d &= \{q_0\}, \\ F^d &= \{\{q_2\}, \{q_0, q_2\}, \{q_1, q_2\}, \{q_0, q_1, q_2\}\}, \end{aligned}$$

and  $\delta^d : Q^d \times \Sigma \rightarrow Q^d$  is defined by the following transition table:

$\delta^d$	$a$	$b$
$\emptyset$	$\emptyset$	$\emptyset$
$\{q_0\}$	$\{q_1\}$	$\emptyset$
$\{q_1\}$	$\emptyset$	$\{q_0, q_2\}$
$\{q_2\}$	$\emptyset$	$\{q_2\}$
$\{q_0, q_1\}$	$\{q_1\}$	$\{q_0, q_2\}$
$\{q_0, q_2\}$	$\{q_1\}$	$\{q_2\}$
$\{q_1, q_2\}$	$\emptyset$	$\{q_0, q_2\}$
$\{q_0, q_1, q_2\}$	$\{q_1\}$	$\{q_0, q_2\}$

The state transition diagram of  $M^d$  is shown in Figure 13.6.

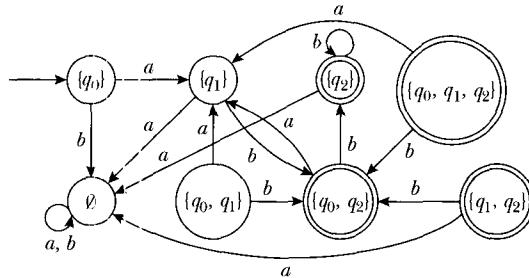


FIGURE 13.6 Transition diagram of  $M^d$

**Lemma 13.1.42:** Let  $M = (Q, \Sigma, q_0, \delta, F)$  be an NDFA and let  $M^d = (Q^d, \Sigma, q_0^d, \delta^d, F^d)$  be the corresponding DFA. Then for all  $A \in Q^d$ ,  $x \in \Sigma^*$ ,

$$(\delta^d)^*(A, x) = \bigcup_{q \in A} \delta^*(q, x),$$

where  $(\delta^d)^*$  denotes the extended state transition function for the DFA  $M^d$  and  $\delta^*$  is the extended transition function for the NDFA  $M$ .

**Proof:** We prove this lemma by induction on  $|x| = n$ .

*Basis step:* Let  $n = 0$ . Then  $x = \lambda$ . Now

$$\begin{aligned} (\delta^d)^*(A, \lambda) &= A && \text{by the definition of } (\delta^d)^* \\ &= \bigcup_{q \in A} \{q\} \\ &= \bigcup_{q \in A} \delta^*(q, \lambda) \\ &= \bigcup_{q \in A} \delta^*(q, x). \end{aligned}$$

Therefore, the result holds for  $n = 0$ .

*Inductive hypothesis:* Suppose the result holds for all  $x \in \Sigma^*$  such that  $|x| = k \geq 0$ .

*Inductive step:* Let  $x \in \Sigma^*$  be such that  $|x| = k + 1$ . Then there exists  $y \in \Sigma^*$  and  $a \in \Sigma$  such that  $x = ya$  and  $|y| = k$ . Now

$$\begin{aligned}
(\delta^d)^*(A, x) &= (\delta^d)^*(A, ya) \\
&= \delta^d((\delta^d)^*(A, y), a) && \text{by the definition of } (\delta^d)^* \\
&= \delta^d\left(\bigcup_{q \in A} \delta^*(q, y), a\right) && \text{by the induction hypothesis} \\
&= \bigcup_{p \in \bigcup_{q \in A} \delta^*(q, y)} \delta(p, a) && \text{by the definition of } \delta^d \\
&= \bigcup_{q \in A} \left( \bigcup_{p \in \delta^*(q, y)} \delta(p, a) \right) \\
&= \bigcup_{q \in A} \delta^*(q, ya) && \text{by the definition of } \delta^* \\
&= \bigcup_{q \in A} \delta^*(q, x).
\end{aligned}$$

Thus, the result is true for  $x \in \Sigma^*$  such that  $|x| = k + 1$ . The result now follows by induction. ■

**Theorem 13.1.43:** Let  $M = (Q, \Sigma, q_0, \delta, F)$  be an NDFA and let  $M^d = (Q^d, \Sigma, q_0^d, \delta^d, F^d)$  represent the corresponding DFA. Then  $L(M^d) = L(M)$ .

**Proof:** Now

$$\begin{aligned}
L(M) &= \{x \in \Sigma^* \mid \delta^*(q_0, x) \cap F \neq \emptyset\} && \text{by the definition of } L(M) \\
&= \{x \in \Sigma^* \mid \delta^*(q_0, x) \in F^d\} && \text{by the definition of } F^d \\
&= \{x \in \Sigma^* \mid (\delta^d)^*(q_0, x) \in F^d\} && \text{by Lemma 13.1.42} \\
&= L(M^d). && \blacksquare
\end{aligned}$$

**Theorem 13.1.44:** A language over  $\Sigma$  is a regular language if and only if it is accepted by some NDFA.

**Proof:** Let  $T \subseteq \Sigma^*$  be a language such that  $T$  is accepted by an NDFA  $M$ . Then  $T = L(M)$ . Let  $M^d$  represent the corresponding DFA. By Theorem 13.1.43,  $L(M^d) = L(M) = T$ . Now  $T = L(M^d)$  implies that  $L$  is a regular language.

Conversely, assume that  $T$  is a regular language. There exists a DFA  $M = (Q, \Sigma, q_0, \delta, F)$  such that  $T = L(M)$ . We now define an NDFA  $M^N = (Q^N, \Sigma, q_0, \delta^N, F^N)$  as follows:  $Q^N = Q$ ,  $F^N = F$ , and  $\delta^N : Q^N \times \Sigma \rightarrow \mathcal{P}(Q^N)$  is defined by

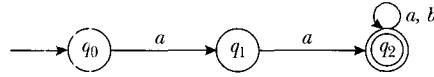
$$\delta^N(q, a) = \{\delta(q, a)\} \quad \text{for all } q \in Q \text{ and } a \in \Sigma.$$

It follows that  $L(M^N) = L(M) = T$ . ■

From Theorems 13.1.43 and 13.1.44, we find that an NDFA cannot accept any language that cannot be accepted by some DFA. So one might wonder why we bother with NDFA. We will show that NDFA are useful concepts in proving various results.

**EXAMPLE 13.1.45**

In this example, we consider the transition diagram of an NDFA  $M$  shown in Figure 13.7.



**FIGURE 13.7** Transition diagram of NDFA  $M$

The states of the NDFA  $M$  are  $q_0$ ,  $q_1$ , and  $q_2$ , where  $q_0$  is the initial state and  $q_2$  is a final state. The set of inputs of this NDFA is  $\Sigma = \{a, b\}$ . Now for the input string  $aabab$ , there exists a directed walk

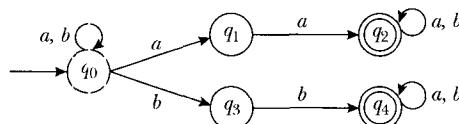
$$q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_2 \xrightarrow{b} q_2 \xrightarrow{a} q_2 \xrightarrow{b} q_2$$

from the initial state  $q_0$  to the final  $q_2$  with the label  $aabab$ . Hence, this NDFA accepts this string. Now for the string  $babaa$ , there is no directed walk from  $q_0$  to the final  $q_2$  with the label  $babaa$ . Hence, the given NDFA does not accept the string  $babaa$ . From the transition diagram of  $M$ , we find that for any string of the form  $aaw$ , where  $a \in \Sigma$  and  $w \in \Sigma^*$ , there exists a directed walk from  $q_0$  to the final  $q_2$  with the label  $aaw$  and there are no directed walks from  $q_0$  to  $q_2$  with the label of any other strings. Thus,  $L(M)$  is the set of all strings that begin with  $aa$ . Hence, it follows that the language  $L = \{aaw \mid w \in \Sigma^*\}$  is a regular language.

**EXAMPLE 13.1.46**

In this example, we show that the language  $L = \{w \in \Sigma^* \mid w \text{ contains the substring } aa \text{ or the substring } bb\}$  is a regular language. To do this we construct an NDFA  $M$  such that  $L(M) = L$ .

Let  $w \in \Sigma^*$ . If  $w$  contains the substring  $aa$  or the substring  $bb$ , then  $w$  is of the form  $uaav$  or  $ubbv$ , where  $u, v \in \Sigma^*$ . We consider the transition diagram of an NDFA  $M$ , with states  $q_0, q_1, q_2, q_3$ , and  $q_4$ , where  $q_0$  is the initial state and  $q_2$  and  $q_4$  are the final states (see Figure 13.8).



**FIGURE 13.8** Transition diagram of  $M$

In the transition diagram, we find that for any string of the form  $uaav$  or  $ubbv$ , where  $u, v \in \Sigma^*$  there exists a directed walk from the initial state to a final state. For example, for the string  $babaabba$ , there exists a directed walk

$$q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_0 \xrightarrow{a} q_0 \xrightarrow{b} q_3 \xrightarrow{b} q_4 \xrightarrow{a} q_4$$

with the label  $babaabba$ . There may exist more than one directed walk with the same label. For example,

$$q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_2 \xrightarrow{b} q_2 \xrightarrow{b} q_2 \xrightarrow{a} q_2$$

is another directed walk with the label *babaabba* from the initial state to the final state. But for strings *w* without *aa* or *bb* as substrings, (for example, strings *bababa* and *ababa*), there are no directed paths with the label *w* from the initial state to a final state. Hence,  $L(M) = \{w \in \Sigma^* \mid w \text{ has a substring } aa \text{ or a substring } bb\} = L$ . Thus, it follows that *L* is a regular language.

## WORKED-OUT EXERCISES

**Exercise 1:** Let *M* be the DFA whose transition diagram is shown in Figure 13.9.

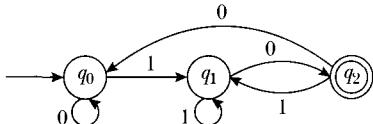


FIGURE 13.9 Transition diagram of a DFA

- What are the states of *M*?
- Write the set of input symbols.
- Which is the initial state?
- Write the set of final states.
- Write the transition table for this DFA.
- Which of the strings 0111010, 00111, 111010, 0100, and 1110 are accepted by *M*?

**Solution:**

- The states of *M* are *q*<sub>0</sub>, *q*<sub>1</sub>, and *q*<sub>2</sub>.
- The input symbols are 0 and 1.
- q*<sub>0</sub> is the initial state.
- This DFA has only one final state, which is *q*<sub>2</sub>. Thus, {*q*<sub>2</sub>} is the set of final states.
- The transition table is given by

$\delta$	0	1
<i>q</i> <sub>0</sub>	<i>q</i> <sub>0</sub>	<i>q</i> <sub>1</sub>
<i>q</i> <sub>1</sub>	<i>q</i> <sub>2</sub>	<i>q</i> <sub>1</sub>
<i>q</i> <sub>2</sub>	<i>q</i> <sub>0</sub>	<i>q</i> <sub>1</sub>

- $\delta^*(q_0, 0111010) = \delta(\delta^*(q_0, 011101), 0)$ ,  
 $\delta^*(q_0, 011101) = \delta(\delta^*(q_0, 01110), 1)$ ,  
 $\delta^*(q_0, 01110) = \delta(\delta^*(q_0, 0111), 0)$ ,  
 $\delta^*(q_0, 0111) = \delta(\delta^*(q_0, 011), 1)$ ,  
 $\delta^*(q_0, 011) = \delta(\delta^*(q_0, 01), 1)$ ,  
 $\delta^*(q_0, 01) = \delta(\delta^*(q_0, 0), 1)$ ,  $\delta^*(q_0, 0) = \delta(q_0, 0) = q_0$ .  
Then  $\delta^*(q_0, 01) = \delta(\delta^*(q_0, 0), 1) = \delta(q_0, 1) = q_1$ ,  
 $\delta^*(q_0, 011) = q_1$ ,  $\delta^*(q_0, 0111) = q_1$ ,  $\delta^*(q_0, 01110) = q_2$ ,  
 $\delta^*(q_0, 011101) = q_1$ ,  $\delta^*(q_0, 0111010) = q_2$ . Because *q*<sub>2</sub> is a final state, it follows that 0111010 is accepted by *M*.

We can also show that 0111010 is accepted by *M* by using Theorem 13.1.26 as follows: In the given transition diagram, we find a directed walk

$$P : q_0 \xrightarrow{0} q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_1 \xrightarrow{1} q_1 \xrightarrow{0} q_2 \xrightarrow{1} q_1 \xrightarrow{0} q_2$$

with the label 0111010. Hence,  $\delta^*(q_0, 0111010) = q_2$ .

Because *q*<sub>2</sub> is a final state, it follows that 0111010 is accepted by *M*.

In the given transition diagram, we find that

$$P : q_0 \xrightarrow{0} q_0 \xrightarrow{0} q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_1 \xrightarrow{1} q_1$$

is the only directed walk with the label 00111. Hence,  $\delta^*(q_0, 00111) = q_1$ . Because *q*<sub>1</sub> is not a final state, it follows that 00111 is not accepted by *M*.

In the transition diagram (see Figure 13.9), we find that

$$P : q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_1 \xrightarrow{1} q_1 \xrightarrow{0} q_2 \xrightarrow{1} q_1 \xrightarrow{0} q_2$$

is a directed walk with the label 111010. Hence,  $\delta^*(q_0, 111010) = q_2$ . Because *q*<sub>2</sub> is a final state, it follows that 111010 is accepted by *M*.

$\delta^*(q_0, 0100) = q_0$ . Because *q*<sub>0</sub> is not a final state, it follows that 0100 is not accepted by *M*.

$\delta^*(q_0, 1110) = q_2$ . Because *q*<sub>2</sub> is a final state, it follows that 1110 is accepted by *M*.

**Exercise 2:** Consider the DFA of Worked-Out Exercise 1.

- Show that any string that ends with 10 is accepted by *M*.
- Show that any string that ends with 01 is not accepted by *M*.
- Show that any string that ends with 00 is not accepted by *M*.
- Show that any string that ends with 11 is not accepted by *M*.
- Find the language, *L*(*M*), accepted by *M*.

**Solution:**

- Suppose that *w*  $\in \Sigma^*$  ends with 10. Then we can write *w* = *w*<sub>1</sub>10, where *w*<sub>1</sub>  $\in \{0, 1\}^*$ . Now  $\delta^*(q_0, w_110) = \delta^*(\delta^*(q_0, w_1), 10)$ . Because  $\delta^*(q, 10) = q_2$  for any state *q* of *M*, it follows that  $\delta^*(q_0, w_110) = q_2$ . Hence, any string that ends with 10 is accepted by *M*.
- Suppose *w*  $\in \Sigma^*$  ends with 01. Then we can write *w* = *u*01, where *u*  $\in \{0, 1\}^*$ . Now  $\delta^*(q_0, u01) = \delta^*(\delta^*(q_0, u), 01)$ . Because  $\delta^*(q, 01) = q_1$  for any state *q* of *M*, it follows that  $\delta^*(q_0, u01) = q_1$ . Because *q*<sub>1</sub> is not a final state, it follows that *u*01 is not accepted by *M*. Hence, any string that ends with 01 is not accepted by *M*.
- Suppose *w*  $\in \Sigma^*$  ends with 00. Then we can write *w* = *u*00, where *u*  $\in \{0, 1\}^*$ . Now  $\delta^*(q_0, u00) = \delta^*(\delta^*(q_0, u), 00)$ . Because  $\delta^*(q, 00) = q_0$  for any state *q* of *M*, it follows that  $\delta^*(q_0, u00) = q_0$ . Because *q*<sub>0</sub> is not a final

state, it follows that  $u00$  is not accepted by  $M$ . Hence, any string that ends with  $00$  is not accepted by  $M$ .

- (d) Suppose  $w \in \Sigma^*$  ends with  $11$ . Then we can write  $w = u11$ , where  $u \in \{0, 1\}^*$ . Now  $\delta^*(q_0, u11) = \delta^*(\delta^*(q_0, u), 11)$ . Because  $\delta^*(q, 11) = q_1$  for any state  $q$  of  $M$ , it follows that  $\delta^*(q_0, u11) = q_1$ . Because  $q_1$  is not a final state, it follows that  $u11$  is not accepted by  $M$ . Hence, any string that ends with  $11$  is not accepted by  $M$ .
- (e) The language accepted by  $M$  is  $L(M) = \{w \in \Sigma^* \mid \delta^*(q_0, w) \in F\}$ . Because  $\delta^*(q_0, \lambda) = q_0$ , it follows that  $\lambda \notin L(M)$ . Also  $\delta^*(q_0, 0) = q_0$  and  $\delta^*(q_0, 1) = q_1$ . Because  $q_0$  and  $q_1$  are not final states, it follows that  $0, 1 \notin L(m)$ . Let  $w \in \{0, 1\}^*$  be such that  $|w| \geq 2$ . Then either  $w$  ends with  $10$  or  $01$  or  $00$  or  $11$ . Hence, from (a), (b), (c), and (d), it follows that  $L(M) = \{w \in \Sigma^* \mid w \text{ ends with } 10\}$ .

**Exercise 3:** Let  $L = \{a^i b^j \mid i \geq 0, j \geq 0\}$  be a language over the alphabet  $\Sigma = \{a, b\}$ . Show that there exists a DFA  $M = (Q, \Sigma, q_0, \delta, F)$  such that  $L(M) = L$ .

**Solution:** Let  $M = (Q, \Sigma, q_0, \delta, F)$ , where

$$\begin{aligned} Q &= \{q_0, q_1, q_2\}, \\ \Sigma &= \{a, b\}, \\ F &= \{q_0, q_1\}, \end{aligned}$$

and  $\delta$  is given by

$\delta$	$a$	$b$
$q_0$	$q_0$	$q_1$
$q_1$	$q_2$	$q_1$
$q_2$	$q_2$	$q_2$

The transition diagram of  $M$  is shown in Figure 13.10.

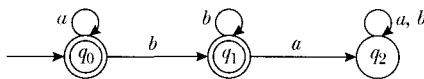


FIGURE 13.10 Transition diagram

It is easy to verify that  $L(M) = \{a^i b^j \mid i \geq 0, j \geq 0\}$ .

**Exercise 4:** Show that the language  $L = \{a^p \mid p \text{ is a prime integer}\}$  is not a regular language over  $\Sigma = \{a, b\}$ .

**Solution:** Assume that  $L$  is regular. Then there exists a DFA  $M = (Q, \Sigma, q_0, \delta, F)$  such that  $L(M) = L$ . Let  $|Q| = n$ . Because there are infinitely many prime integers (see Theorem 2.4.8), we can find a prime  $p$  such that  $p \geq n$ . Let  $w = a^p$ . Then  $|w| = p \geq n$ . Hence,  $w = a^p \in L = L(M)$ . So by the pumping lemma, Theorem 13.1.27, there exist  $x, y, z \in \Sigma^*$  such that  $w = xyz$ ,  $|xy| \leq n$ ,  $|y| \geq 1$  and  $xy^kz \in L(M)$  for all  $k = 0, 1, 2, \dots$ . In particular,  $xy^{p+1}z \in L(M)$ . Now

$$|xy^{p+1}z| = |xyz| + |y^p| = p + p|y| = p(1 + |y|).$$

This shows that  $|xy^{p+1}z|$  is not a prime integer. Therefore,  $xy^{p+1}z \notin L = L(M)$ , a contradiction. Hence,  $L$  is not regular.

**Exercise 5:** Let  $L = \{a^i b^j \mid i, j \text{ are positive integers and } \gcd(i, j) = 1\}$  be a language over the alphabet  $\Sigma = \{a, b\}$ . Show that  $L$  is not a regular language.

**Solution:** Assume that  $L$  is regular. Then there exists a DFA  $M = (Q, \Sigma, q_0, \delta, F)$  such that  $L(M) = L$ . Let  $|Q| = n$ . Because there are infinitely many prime integers, we can find a prime  $p$  such that  $p > n$ . Let  $w = a^p b^{(p-1)!}$ . Now  $1$  and  $p$  are the only positive divisors of  $p$ . Therefore,  $\gcd(p, (p-1)!) = 1$ . Hence,  $w = a^p b^{(p-1)!} \in L = L(M)$ . Now  $|w| = p + (p-1)! > n$ . So by the pumping lemma, there exist  $x, y, z \in \Sigma^*$  such that  $w = xyz$ ,  $|xy| \leq n$ ,  $|y| \geq 1$  and  $xy^kz \in L(M)$  for all  $k = 0, 1, 2, \dots$ . In particular,  $xz \in L(M)$ . Because  $|xy| \leq n < p$ , the string  $xy$  contains only  $a$ 's. Let  $x = a^i, y = a^j$  for some integers  $i$  and  $j$  less than  $p$ . Because  $|y| \geq 1$ , it follows that  $j \geq 1$ . Now  $z = a^t b^{(p-1)!}$ , where  $t$  is an integer such that  $i + j + t = p$ . Then  $xz = a^i a^t b^{(p-1)!} = a^{i+t} b^{(p-1)!} = a^{(p-j)} b^{(p-1)!}$ . Because  $p > j \geq 1$ ,  $1 < p-j \leq (p-1)!$  and  $\gcd(p-j, (p-1)!) = p-j > 1$ . Therefore,  $xz \notin L = L(M)$ . Thus, we arrive at a contradiction. Hence,  $L$  is not a regular language.

**Exercise 6:** Let  $L = \{w \in \{a, b\}^* \mid w = w^R\}$  be a language over the alphabet  $\Sigma = \{a, b\}$ . Show that  $L$  is not a regular language.

**Solution:**  $L$  is the set of all palindromes on  $\Sigma = \{a, b\}$ . Assume that  $L$  is regular. Then there exists a DFA  $M = (Q, \Sigma, q_0, \delta, F)$  such that  $L(M) = L$ . Let  $|Q| = n$ . Let  $w = a^n b a^n$ . Because  $w = w^R$ , we find that  $w \in L = L(M)$ . So by the pumping lemma, Theorem 13.1.27, there exist  $x, y, z \in \Sigma^*$  such that  $w = xyz$ ,  $|xy| \leq n$ ,  $|y| \geq 1$  and  $xy^kz \in L(M)$  for all  $k = 0, 1, 2, \dots$ . In particular,  $xz \in L(M)$ . Because  $|xy| \leq n$ , the string  $xy$  contains only  $a$ 's. Let  $x = a^i, y = a^j$  for some integers  $i$  and  $j$  less than or equal to  $n$ . Because  $|y| \geq 1$ , it follows that  $j \geq 1$ . Now  $z = a^t b a^n$ , where  $t$  is an integer such that  $i + j + t = n$ . Then  $xz = a^{i+t} b a^n$ . Because  $j \geq 1$ ,  $i + t < n$ , and hence  $a^{i+t} b a^n$  is not a palindrome. This implies that  $xz \notin L = L(M)$ . Thus, we arrive at a contradiction. Hence,  $L$  is not a regular language.

**Exercise 7:** Let  $M$  be the NDFA whose state diagram is given in Figure 13.11.

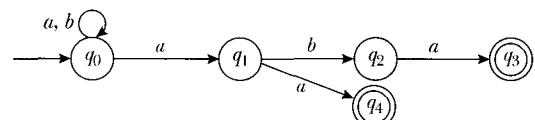


FIGURE 13.11 NDFA

- Write the transition table for this NDFA.
- Find a directed walk with the label  $bbabbaa$  from  $q_0$  to  $q_4$ .
- What are the final states?
- Find  $\delta^*(q_0, baa)$ .
- Is  $baa$  in  $L(M)$ ?
- Find  $L(M)$ .

**Solution:**

(a)	$\delta$	a	b
$q_0$		$\{q_0, q_1\}$	$\{q_0\}$
$q_1$		$\{q_4\}$	$\{q_2\}$
$q_2$		$\{q_3\}$	$\emptyset$
$q_3$		$\emptyset$	$\emptyset$
$q_4$		$\emptyset$	$\emptyset$

- (b) The following is a directed walk with the label  $bbabbaaa$  from  $q_0$  to  $q_4$ .

$$q_0 \xrightarrow{b} q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_0 \xrightarrow{b} q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_4.$$

- (c) This NDFA has two final states,  $q_3$  and  $q_4$ .

(d)

$$\begin{aligned}\delta^*(q_0, baa) &= \bigcup_{p \in \delta^*(q_0, ba)} \delta(p, a), \\ \delta^*(q_0, ba) &= \bigcup_{p \in \delta^*(q_0, b)} \delta(p, a), \\ \delta^*(q_0, b) = \delta^*(q_0, \lambda b) &= \bigcup_{p \in \delta^*(q_0, \lambda)} \delta(p, b) \\ &= \bigcup_{p \in \{q_0\}} \delta(p, b) = \delta(q_0, b) = \{q_0\}, \\ \delta^*(q_0, ba) &= \bigcup_{p \in \{q_0\}} \delta(p, a) = \delta(q_0, a) = \{q_0, q_1\} \\ \delta^*(q_0, baa) &= \bigcup_{p \in \{q_0, q_1\}} \delta(p, a) = \delta(q_0, a) \cup \delta(q_1, a) \\ &= \{q_0, q_1\} \cup \{q_4\} = \{q_0, q_1, q_4\}.\end{aligned}$$

- (e) In the transition diagram, we find a directed walk

$$q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_4$$

with the label  $baa$  from the initial state  $q_0$  to a final state. Hence,  $baa \in L(M)$ .

- (f) For all strings  $w = \{a, b\}$  of the form  $uaa, uaba$ , where  $u \in \Sigma^*$ , there exist directed walks from the initial state to some final state. Because  $q_3$  and  $q_4$  are only the final states and to reach these two vertices either we have to use the directed walk

$$q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_4$$

or the directed walk

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3,$$

it follows that  $L(M) = \{w \in \Sigma^* \mid w = uaa \text{ or } w = uaba, \text{ where } u \in \Sigma^*\}$ .

**Exercise 8:** Find an NDFA  $M$  on  $\Sigma = \{a, b\}$  such that  $L(M) = \{ab, ba\}$ .

**Solution:** Consider the NDFA  $M$  shown in Figure 13.12.

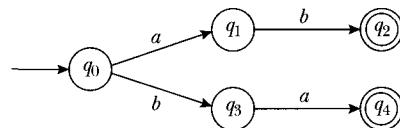


FIGURE 13.12 NDFA

This is an NDFA with initial state  $q_0$  and final states  $q_2$  and  $q_4$ . In this NDFA, there are only two directed walks,  $q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2$  and  $q_0 \xrightarrow{b} q_3 \xrightarrow{a} q_4$ , from the initial state to the final states.  $q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2$  is a directed walk with the label  $ab$ , and  $q_0 \xrightarrow{b} q_3 \xrightarrow{a} q_4$  is a directed walk with the label  $ba$ . Hence,  $L(M) = \{ab, ba\}$ .

## SECTION REVIEW

### Key Terms

length	states	accepted
language	input alphabet	language accepted by
concatenation	initial (start) state	regular language
Kleene star	state transition function	label of a directed walk
reversal string	final (accepting) states	pumping lemma for
reversal	transition table	regular languages
palindrome	state diagram	nondeterministic finite automaton
deterministic finite automaton	transition diagram	(NDFA)
deterministic finite acceptor (DFA)	extended transition function	extended state transition function

## Some Key Definitions

1. Let  $\Sigma$  be an alphabet. Any subset of  $\Sigma^*$  is called a language on  $\Sigma$ .
2. A deterministic finite automaton, or deterministic finite acceptor (DFA), is a quintuple  $M = (Q, \Sigma, q_0, \delta, F)$ , where
  - (i)  $Q$  is a finite nonempty set of states,
  - (ii)  $\Sigma$  is the input alphabet (a finite nonempty set of symbols),
  - (iii)  $q_0$  is the initial (or start) state, a particular element of  $Q$ ,
  - (iv)  $\delta$  is the state transition function,  $\delta : Q \times \Sigma \rightarrow Q$ , and
  - (v)  $F$  is the set of final (or accepting) states, a (possibly empty) subset of  $Q$ .
3. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA and  $w \in \Sigma^*$ . Then  $w$  is said to be accepted by  $M = (Q, \Sigma, q_0, \delta, F)$  if  $\delta^*(q_0, w) \in F$ .
4. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA. The language accepted by  $M$ , written  $L(M)$ , is the set  $L(M) = \{w \in \Sigma^* \mid \delta^*(q_0, w) \in F\}$ .
5. A language  $A$  on the alphabet  $\Sigma$  is said to be a regular language if there exists a DFA  $M$  such that  $L(M) = A$ .
6. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA and let  $G_M$  be its transition diagram. If

$$P : q_{i_0} \xrightarrow{a_1} q_{i_1} \xrightarrow{a_2} q_{i_2} \xrightarrow{a_3} \cdots \xrightarrow{a_{n-1}} q_{i_{n-1}} \xrightarrow{a_n} q_{i_n}$$

is a directed walk, where  $\delta(q_{i_{k-1}}, a_k) = q_{i_k}$ , then the string  $a_1 a_2 \cdots a_{n-1} a_n$  is called the label of the directed walk  $P$ .

7. A nondeterministic finite automaton (NDFA) is a quintuple  $M = (Q, \Sigma, q_0, \delta, F)$ , where
  - (i)  $Q$  is a finite nonempty set of states,
  - (ii)  $\Sigma$  is the input alphabet (a finite nonempty set of symbols),
  - (iii)  $q_0$  is the initial (or start) state, a particular element of  $Q$ , and
  - (iv)  $\delta$  is the state transition function,  $\delta : Q \times \Sigma \rightarrow P(Q)$  ( $P(Q)$  is the set of all subsets of  $Q$ ), and
  - (v)  $F$  is the set of final (or accepting) states, a (possibly empty) subset of  $Q$ .
8. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be an NDFA and  $w \in \Sigma^*$ . Then  $w$  is said to be accepted by  $M$  if  $\delta^*(q_0, w) \cap F \neq \emptyset$ .
9. Given an NDFA  $M = (Q, \Sigma, q_0, \delta, F)$ , the language accepted by  $M$ , written  $L(M)$ , is the set  $L(M) = \{w \in \Sigma^* \mid \delta^*(q_0, w) \cap F \neq \emptyset\}$ .

## Some Key Results

1. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA and let  $G_M$  be its transition diagram. Then for any two states  $q_i$  and  $q_j$ , and a string  $w \in \Sigma^*$ ,  $\delta^*(q_i, w) = q_j$  if and only if there is a directed walk  $P$  in  $G_M$  with the label  $w$  from  $q_i$  to  $q_j$ .
2. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA with  $n$  states. Let  $w \in L(M)$  be such that  $|w| \geq n$ . Then there exist strings  $x, y, z \in \Sigma^*$  such that  $w = xyz$ ,  $|xy| \leq n$ ,  $|y| \geq 1$  and  $xy^kz \in L(M)$  for all integers  $k \geq 0$ .

3. Let  $M$  be a DFA with  $n$  states. If  $L(M) \neq \emptyset$ , then there is a string  $w \in L(M)$  such that  $|w| < n$ .
4. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be a DFA with  $n$  states. Then  $L(M)$  is infinite if and only if  $L(M)$  contains a string  $x$  such that  $n \leq |x| < 2n$ .
5. A language over  $\Sigma$  is a regular language if and only if it is accepted by some NDFA.

## EXERCISES

1. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be the DFA such that  $Q = \{q_0, q_1, q_2\}$ ,  $\Sigma = \{a, b\}$ ,  $F = \{q_2\}$ ,  $q_0$  = the initial state, and  $\delta$  is given by

$\delta$	a	b
$q_0$	$q_0$	$q_1$
$q_1$	$q_2$	$q_1$
$q_2$	$q_2$	$q_0$

- a. Draw the state diagram of  $M$ .
- b. Which of the strings  $abaa$ ,  $bbbabb$ ,  $bbbaa$ , and  $bababa$  are accepted by  $M$ ?
2. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be the DFA such that  $Q = \{q_0, q_1, q_2, q_3\}$ ,  $\Sigma = \{a, b, c\}$ ,  $F = \{q_0\}$ , the initial state is  $q_0$ , and  $\delta$  is given by

$\delta$	a	b	c
$q_0$	$q_0$	$q_0$	$q_0$
$q_1$	$q_0$	$q_2$	$q_3$
$q_2$	$q_1$	$q_2$	$q_3$
$q_3$	$q_2$	$q_1$	$q_3$

- a. Draw the state diagram of  $M$ .
- b. Which of the strings  $abc$ ,  $bbb$ ,  $cbba$ , and  $caccb$  are accepted by  $M$ ?
- c. Find  $L(M)$ .
3. Let  $M$  be the DFA whose transition diagram is shown in Figure 13.13.

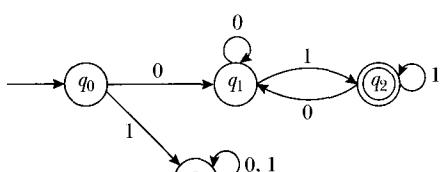


FIGURE 13.13 A DFA

- a. What are the states of  $M$ ?
- b. Write the set of input symbols.
- c. Which is the initial state?

- d. Write the set of final states.
- e. Write the transition table for this DFA.
- f. Which of the strings  $000$ ,  $0011$ ,  $0100$ , and  $1111$  are accepted by  $M$ ?
- g. Find  $L(M)$ .
4. Let  $M$  be the DFA whose state transition diagram is shown in Figure 13.14.

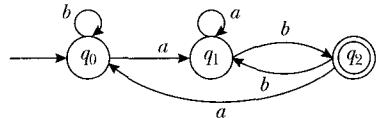


FIGURE 13.14 DFA

- a. Construct the transition table of  $M$ .
  - b. Which of the strings  $baba$ ,  $baab$ ,  $abab$ , and  $abaab$  are accepted by  $M$ ?
  - c. Find  $L(M)$ .
  5. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be the DFA such that  $Q = \{q_0, q_1, q_2\}$ ,  $\Sigma = \{a, b\}$ ,  $F = \{q_0, q_1\}$ , and  $\delta$  is given by
- | $\delta$ | a     | b     |
|----------|-------|-------|
| $q_0$    | $q_0$ | $q_1$ |
| $q_1$    | $q_0$ | $q_2$ |
| $q_2$    | $q_2$ | $q_2$ |
- a. Draw the state diagram of  $M$ .
  - b. Which of the strings  $aaaa$ ,  $bbbb$ ,  $abba$ , and  $babab$  are accepted by  $M$ ?
  - c. Find  $L(M)$ .
  6. Let  $M$  be the DFA whose transition diagram is shown in Figure 13.15.

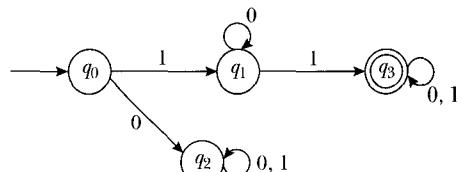


FIGURE 13.15 A DFA

- What are the states of  $M$ ?
  - Write the set of input symbols.
  - Which is the initial state?
  - Write the set of final states.
  - Which of the strings 1000, 10011, 1101, and 1111 are accepted by  $M$ ?
  - Find  $L(M)$ .
7. Let  $M$  be the DFA whose transition diagram is shown in Figure 13.16.

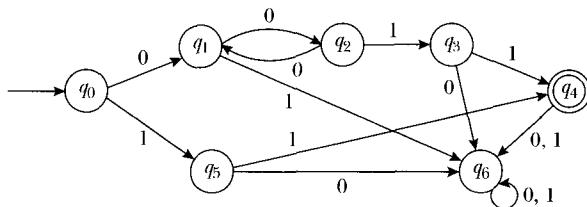


FIGURE 13.16 DFA

- Which is the initial state?
  - What are the final states?
  - Which of the strings 1, 11, 0011, and 00111 are accepted by  $M$ ?
  - Find  $L(M)$ .
8. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be the DFA such that  $Q = \{q_0, q_1, q_2\}$ ,  $\Sigma = \{a, b\}$ ,  $F = \{q_2\}$ ,  $q_0$  is the initial state, and  $\delta$  is given by

$\delta$	a	b
$q_0$	$q_0$	$q_1$
$q_1$	$q_2$	$q_2$
$q_2$	$q_2$	$q_2$

- Draw the state diagram of  $M$ .
  - Which of the strings  $ab$ ,  $abab$ ,  $abbb$ , and  $babb$  are accepted by  $M$ ?
  - Find  $L(M)$ .
9. Find the language accepted by the DFA shown in Figure 13.17 (see top of next column).
10. Find a DFA that recognizes the set of all strings on  $\Sigma = \{a, b\}$  that starts with  $ab$ .
11. Prove that the empty set  $\emptyset$ , and the language  $\{\lambda\}$  are regular languages.
12. Draw a state transition diagram of a DFA that accepts strings on  $\{0, 1\}$  that do not contain the substring 001.
13. Construct a state transition diagram of a DFA  $M$  with the input set  $\{0, 1\}$  such that  $M$  accepts only the string 101.
14. Construct a state transition diagram of a DFA  $M$  with the input set  $\{0, 1\}$  such that  $M$  accepts only the strings  $101w$ , where  $w$  is any string on  $\{0, 1\}$ .
15. Construct a state transition diagram of a DFA that accepts all strings on  $\{a, b\}$  that contain an even number of  $a$ 's and an odd number of  $b$ 's.

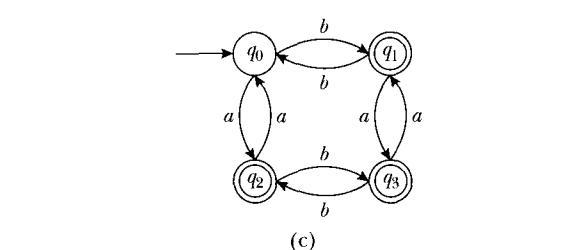
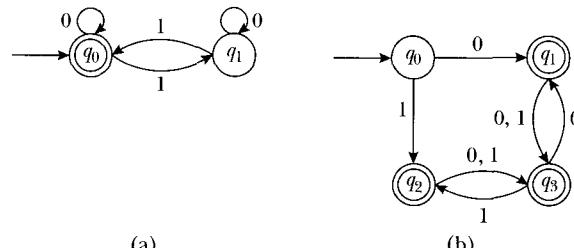


FIGURE 13.17 Various DFA

- Construct a state transition diagram of a DFA that accepts all strings on  $\{0, 1, 2, 3\}$  such that the sum of the integers in the string is divisible by 4.
  - Construct a DFA that recognizes the set of all strings on  $\Sigma = \{a, b\}$  that do not contain an even number of  $a$ 's and an odd number of  $b$ 's.
  - Construct a state transition diagram of a DFA that accepts all strings on  $\{0, 1\}$  which contain 01 or 10.
  - Construct a state transition diagram of a DFA that accepts the following languages:
    - $L = \{w \in \{a, b\}^* \mid w \text{ has } 3k + 1 \text{ } b\text{'s for some } k \geq 0\}$ .
    - $L = \{w \in \{a, b\}^* \mid w \text{ contains an even number of } a\text{'s}\}$ .
    - $L = \{w \in \{a, b\}^* \mid \text{each } a \text{ in } w \text{ is immediately preceded and followed by } b\}$ .
20. Describe the language accepted by the DFA whose transition diagram is shown in Figure 13.18.

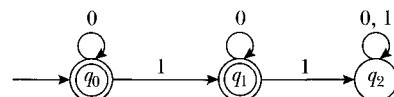


FIGURE 13.18 DFA

21. Describe the language accepted by the DFA whose transition diagram is shown in Figure 13.19.

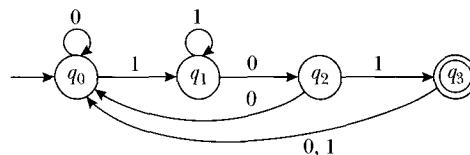


FIGURE 13.19 DFA

- Show that the language  $L = \{a^n b \mid n \geq 0\}$  is a regular language on  $\Sigma = \{a, b\}$ .
- Show that the language  $L = \{awa \mid w \in \Sigma^*\}$  is a regular language on  $\Sigma = \{a, b\}$ .

24. Show that the language  $L = \{a^n b^m \mid n < m\}$  is not a regular language on  $\Sigma = \{a, b\}$ .
25. Show that the language  $L = \{ww \mid w \in \Sigma^*\}$  is not a regular language on  $\Sigma = \{a, b\}$ .
26. Show that the language  $L = \{a^n \mid n \text{ is a nonnegative integer}\}$  is not a regular language on  $\Sigma = \{a\}$ .
27. Draw an NDFA  $M$  with input symbols 0, 1 such that the language of  $M$  is

$$L(M) = \{0^n \mid n \geq 0\} \cup \{0^n 1 \mid n \geq 0\} \cup \{0^n 11 \mid n \geq 0\}.$$

28. Suppose that  $L_1$  and  $L_2$  are two regular languages on  $\Sigma$ . Prove that  $L_1 \cup L_2$  and  $\Sigma^* - L_1$  are regular languages on  $\Sigma$ .
29. Let  $\Sigma = \{a, b\}$ . Construct an NDFA  $M$  that accepts only the language  $L = \{a^i b^j \in \Sigma^* \mid 0 \leq i \leq 3\}$ .
30. Let  $M$  be the NDFA whose state diagram is shown in Figure 13.20.

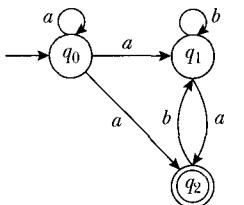


FIGURE 13.20 An NDFA

- a. Construct a transition table for  $M$ .
- b. Is  $aaabb$  in  $L(M)$ ?
- c. Find the corresponding DFA of  $M$ .
31. Let  $M$  be the NDFA whose state diagram is shown in Figure 13.21.

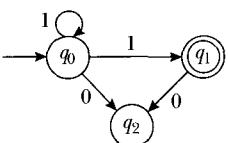


FIGURE 13.21 An NDFA

- a. Construct a transition table for  $M$ .
- b. Find  $L(M)$ .
- c. Find the corresponding DFA of  $M$ .
32. Let  $M$  be the NDFA whose state diagram is shown in Figure 13.22.

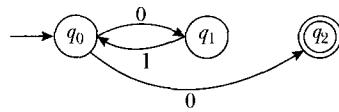


FIGURE 13.22 An NDFA

- a. Construct a transition table for  $M$ .
- b. Find  $L(M)$ .
- c. Find the corresponding DFA of  $M$ .
33. Find an NDFA  $M$  on  $\Sigma = \{a, b\}$  such that  $L(M) = \{aa, aba\}$ .
34. Let  $M = (Q, \Sigma, q_0, \delta, F)$  be the NDFA such that  $Q = \{q_0, q_1, q_2\}$ ,  $\Sigma = \{a, b\}$ ,  $F = \{q_2\}$ , the initial state is  $q_0$ , and  $\delta$  is given by

$\delta$	$a$	$b$
$q_0$	$\{q_0, q_2\}$	$\{q_0\}$
$q_1$	$\emptyset$	$\{q_0, q_2\}$
$q_2$	$\{q_1, q_2\}$	$\emptyset$

- a. Draw the transition diagram of  $M$ .
- b. Which of the strings  $aba$ ,  $bbb$ ,  $bba$ , and  $abb$  are accepted by  $M$ ?
35. Construct the NDFA  $M$  that accepts the language  $\{a^n b^m \mid n, m \geq 0\}$  over the set  $\{a, b\}$ . Find the corresponding DFA  $M^d$  of  $M$ .
36. Give a state diagram of an NDFA that accepts the following language  $T$  over  $\{a, b\}$ .

$$T = \{w \in \{a, b\}^* \mid \text{the third to last symbol in } w \text{ is } b\}.$$

37. Let  $M = (\{q_0, q_1\}, \{0, 1\}, \{q_0\}, \delta, \{q_1\})$  be an NDFA, where
- $$\delta(q_0, 0) = \{q_0, q_1\}, \quad \delta(q_0, 1) = \emptyset,$$
- $$\delta(q_1, 0) = \emptyset, \quad \text{and} \quad \delta(q_1, 1) = \{q_0, q_1\}.$$

Construct a DFA  $M^d$  that accepts  $L(M)$ .

38. Let  $L = \{w \in \{a, b\}^* \mid w = (ab)^n, n \geq 0\}$ . Show that  $L$  is a regular language.
39. Construct an NDFA that accepts the strings over  $\Sigma = \{a, b\}$  satisfying the given properties specified in the exercises (a)–(c).
- a. The strings start with the substring  $aba$ .
- b. The strings contain  $aba$  as a substring.
- c. The strings terminate with the substring  $aba$ .

## 13.2 FINITE STATE MACHINES WITH INPUT AND OUTPUT

In the preceding sections, we considered automata with input only. In this section, we consider automata with input as well as output. That is, every state has an input and corresponding to the input the state also has an output. These types of automata are commonly called *finite state machines*. As we will see, these types of automata can be used to model vending machines as well as adding binary numbers. We begin with the following definition.

**DEFINITION 13.2.1** ► A **finite state machine (FSM)** is a sextuple  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$ , where

- (i)  $Q$  is a finite nonempty set of **states**,
- (ii)  $\Sigma$  is the **input alphabet** (a finite nonempty set of symbols),
- (iii)  $\Gamma$  is the **output alphabet** (a finite nonempty set of symbols),
- (iv)  $q_0$  is the **initial (or start) state**, a particular element of  $Q$ ,
- (v)  $\delta$  is the **state transition function**,  $\delta : Q \times \Sigma \rightarrow Q$ ,
- (vi)  $\gamma$  is the **output function**,  $\gamma : Q \times \Sigma \rightarrow \Gamma$ .

A finite state machine as defined in Definition 13.2.1 is also called a **Mealy sequential machine**.

**EXAMPLE 13.2.2**

Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$ , where  $Q = \{q_0, q_1, q_2\}$ ,  $\Sigma = \{a, b\}$ ,  $\Gamma = \{0, 1\}$ ,  $q_0$  is the initial state, and the functions  $\delta$  and  $\gamma$  are defined as follows:

$$\begin{array}{ll} \delta(q_0, a) = q_1, & \delta(q_1, b) = q_2, \\ \delta(q_0, b) = q_0, & \delta(q_2, a) = q_0, \\ \delta(q_1, a) = q_2, & \delta(q_2, b) = q_1, \end{array}$$

and

$$\begin{array}{ll} \gamma(q_0, a) = 0, & \gamma(q_1, b) = 0, \\ \gamma(q_0, b) = 1, & \gamma(q_2, a) = 0, \\ \gamma(q_1, a) = 1, & \gamma(q_2, b) = 1. \end{array}$$

Then  $M$  is a finite state machine. Note that  $\delta(q_0, a) = q_1$  and  $\gamma(q_0, a) = 0$ . This means that  $M$  is in the state  $q_0$  and receives the input  $a$ , and  $M$  goes to state  $q_1$  and also outputs 0.

As in the case of DFA, we can describe the transition function and the output function using a table. For example, the table corresponding to the transition and output functions of Example 13.2.2 is

		$\delta$		$\gamma$	
		$a$	$b$	$a$	$b$
$q_0$	$q_1$	$q_0$		0	1
	$q_2$	$q_2$		1	0
$q_2$	$q_0$	$q_1$		0	1

We can also use a directed graph to describe a finite state machine. In this case, the arcs are labeled with the input as well as the output as follows: Suppose  $M$  is a finite state machine and  $p, q \in Q$ . Suppose  $\delta(p, a) = q$  and  $\gamma(p, a) = 0$ , where  $a \in \Sigma$  and  $0 \in \Gamma$ . Then the arc from  $p$  to  $q$  is labeled as  $a/0$ .

The directed graph corresponding to the finite state machine of Example 13.2.2 is shown in Figure 13.23.

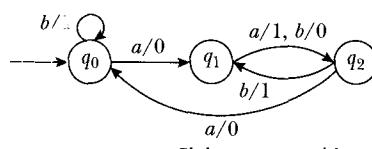


FIGURE 13.23 Finite state machine

Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM. As in the case of a DFA, we can extend the transition function  $\delta$  to  $\delta^*$  to process strings and the definition of  $\delta^*$  is the same as in the case of DFA.

**DEFINITION 13.2.3** ▶ Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM. The extended output function written  $\gamma^*$  is a function  $\gamma^* : Q \times \Sigma^* \rightarrow \Gamma^*$  defined recursively as follows:

- (i) For all  $q \in Q$ ,  $\gamma^*(q, \lambda) = \lambda$ .
- (ii) For all  $q \in Q$ ,  $x \in \Sigma^*$ ,  $a \in \Sigma$ ,

$$\gamma^*(q, ax) = \gamma(q, a)\gamma^*(\delta(q, a), x).$$

#### EXAMPLE 13.2.4

Let  $M$  be the FSM of Example 13.2.2. Let  $x = bbab$ . Then, we have

$$\begin{aligned}\gamma^*(q_0, bbab) &= \gamma(q_0, b)\gamma^*(\delta(q_0, b), bab) \\ &= 1\gamma^*(q_0, bab) \\ &= 1\gamma(q_0, b)\gamma^*(\delta(q_0, b), ab) \\ &= 11\gamma^*(q_0, ab) \\ &= 11\gamma(q_0, a)\gamma^*(\delta(q_0, a), b) \\ &= 110\gamma^*(q_1, b) \\ &= 110\gamma(q_1, b)\gamma^*(\delta(q_1, b), \lambda) \\ &= 1100\gamma^*(q_2, \lambda) \\ &= 1100\lambda = 1100.\end{aligned}$$

We can also show this transition as follows:

$$q_0 \xrightarrow[1]{b} q_0 \xrightarrow[1]{b} q_0 \xrightarrow[0]{a} q_1 \xrightarrow[0]{b} q_2$$

and we say that for the input string  $bbab$  the corresponding output string is 1100 and the output of the string is 0.

**DEFINITION 13.2.5** ▶ Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM. Let  $x = x_1 x_2 \dots x_n$  be a nonempty string over  $\Sigma$ . Then  $y = y_1 y_2 \dots y_n$ ,  $y_i \in \Gamma$ ,  $i = 1, 2, \dots, n$ , is the corresponding **output string** if  $\gamma^*(q_0, x_1 x_2 \dots x_n) = y_1 y_2 \dots y_n$ . We call  $y_n$  the **output of the string**  $x$ . In this case, we also say that  $y_n$  is the output of  $M$  for the string  $x$ , or simply  $M$  outputs  $y_n$  for the string  $x$ .

#### EXAMPLE 13.2.6

Let  $M$  be the FSM of Example 13.2.2. Let  $x = bbab$ . Then, we have

$$q_0 \xrightarrow[1]{b} q_0 \xrightarrow[1]{b} q_0 \xrightarrow[0]{a} q_1 \xrightarrow[0]{b} q_2.$$

This implies that the output string corresponding to input string  $bbab$  is 1100. Moreover, the output of  $M$  for  $bbab$  is 0.

Let us now consider the input string  $bababaa$ . Then

$$q_0 \xrightarrow[1]{b} q_0 \xrightarrow[0]{a} q_1 \xrightarrow[0]{b} q_2 \xrightarrow[0]{a} q_0 \xrightarrow[1]{b} q_0 \xrightarrow[0]{a} q_1 \xrightarrow[1]{a} q_2.$$

Thus, the output string corresponding to input string  $bababaa$  is 1000101. Moreover, the output of  $M$  for  $bababaa$  is 1.

We leave the proof of the following theorem as an exercise.

**Theorem 13.2.7:** Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM. Then for any  $u, v \in \Sigma^*$  and for any  $q \in Q$ ,

$$\gamma^*(q, uv) = \gamma^*(q, u)\gamma^*(\delta^*(q, u), v).$$

**DEFINITION 13.2.8** ▶ Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM and let  $G_M$  be its transition diagram. If

$$P : q_{i_0} \xrightarrow{a_1/b_1} q_{i_1} \xrightarrow{a_2/b_2} q_{i_2} \xrightarrow{a_3/b_3} \cdots \xrightarrow{a_n/b_n} q_{i_n}$$

is a directed walk, where  $\delta(q_{i_{k-1}}, a_k) = q_{i_k} \in Q$ ,  $\gamma(q_{i_{k-1}}, a_k) = b_k \in \Gamma$ , then the expression  $(a_1/b_1)(a_2/b_2) \cdots (a_{n-1}/b_{n-1})(a_n/b_n)$  is called the **label of the directed walk  $P$** .

**REMARK 13.2.9** ▶ Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM and let  $G_M$  be its transition diagram. Suppose

$$P : q_{i_0} \xrightarrow{a_1/b_1} q_{i_1} \xrightarrow{a_2/b_2} q_{i_2} \xrightarrow{a_3/b_3} \cdots \xrightarrow{a_n/b_n} q_{i_n}$$

is a directed walk. As in Example 13.2.6, sometimes for clarity and ease of reading, we write  $P$  as

$$P : q_{i_0} \xrightarrow[a_1]{b_1} q_{i_1} \xrightarrow[a_2]{b_2} q_{i_2} \xrightarrow[a_3]{b_3} \cdots \xrightarrow[a_n]{b_n} q_{i_n},$$

i.e., the input symbol is written on top of the arrow and the output symbol is written below the arrow.

We leave the proof of the following theorem as an exercise.

**Theorem 13.2.10:** Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM. Let  $x = a_1 a_2 \cdots a_n$  be a nonempty string over  $\Sigma$ . Let  $y = b_1 b_2 \cdots b_n$ ,  $b_i \in \Gamma$ ,  $i = 1, 2, \dots, n$ . Then  $\gamma^*(q_0, a_1 a_2 \cdots a_n) = b_1 b_2 \cdots b_n$  if and only if there is a directed walk  $P$  in  $G_M$  with label

$$(a_1/b_1)(a_2/b_2) \cdots (a_{n-1}/b_{n-1})(a_n/b_n)$$

from  $q_0$  to  $\delta^*(q_0, a_1 a_2 \cdots a_n)$ .

### EXAMPLE 13.2.11

**Unit-time delay FSM.** In this example, we design a FSM called a unit-time delay machine that outputs the input string delayed by a specific amount of time. The inputs to the machine are strings of 0's and 1's and the outputs are strings of 0's and 1's. We assume that the time delay is one-unit time. Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be the FSM with the directed graph representation shown in Figure 13.24. Note that  $Q = \{q_0, q_1, q_2\}$ ,  $\Sigma = \{0, 1\}$ ,  $\Gamma = \{0, 1\}$ , and  $q_0$  is the initial state.

Initially,  $M$  is in state  $q_0$ . If the first input is a 0, then  $M$  goes into state  $q_1$  and stays in  $q_1$  as long as the input is 0. Note that  $q_1$  with input 0 outputs 0. If the first input is 1, then  $M$  goes into state  $q_2$  and stays in  $q_2$  as long as the input is 1. Note that  $q_2$  with input 1 outputs 1.

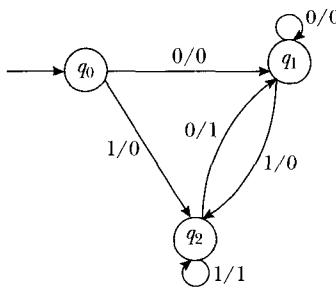


FIGURE 13.24 Unit-time delay FSM

If the input string is  $a_1 a_2 \dots a_k$ , then  $M$  outputs  $0a_1 a_2 \dots a_{k-1}$ . To output the entire string, then 0 must be appended to the input string. For example, the input string  $a_1 a_2 \dots a_k$  is input as  $a_1 a_2 \dots a_k 0$ . Then the output is  $0a_1 a_2 \dots a_k$ .

**EXAMPLE 13.2.12**

In this example, we design a FSM that adds two binary numbers of the form  $x_1 x_2 \dots x_k$  and  $y_1 y_2 \dots y_k$ . Recall from Chapter 2 that to add these binary numbers, starting from the last bits,  $x_k$  and  $y_k$ , we add the corresponding bits from right to left.

Now  $x_i, y_i \in \{0, 1\}$ . Thus, we have four possible choices for  $x_i$  and  $y_i$  as follows:  $x_i = 0$  and  $y_i = 0$ ;  $x_i = 0$  and  $y_i = 1$ ;  $x_i = 1$  and  $y_i = 0$ ; and  $x_i = 1$  and  $y_i = 1$ . Now  $0 + 0 = 0$ ,  $1 + 0 = 1$ ,  $0 + 1 = 1$ , and  $1 + 1 = 10$ . It follows that the addition of 1 and 1 produces a sum of 0 and a carry of 1 and the addition of 0 and 0, or 0 and 1, or 1 and 0 does not produce a carry. Thus, either the sum of two bits does not produce a carry or it produces a carry. Corresponding to these two possibilities, we have two states, say  $q_0$  and  $q_1$ . The state  $q_0$  corresponds to the sum of bits that does not produce a carry. Moreover  $q_0$  is the initial state.

Now there are four possible inputs to each state: 00, 01, 10, and 11. Suppose the input to  $q_0$  is 00. Now the sum of  $0 + 0 = 0$ ; i.e., the sum is 0 and there is no carry. Thus,  $q_0$  with input 00 outputs a 0 and remains in state  $q_0$ . Consider the input 01 to  $q_0$ . Now  $0 + 1 = 1$ ; i.e., the sum is 1 and there is no carry. Thus,  $q_0$  with input 01 outputs a 1 and remains in state  $q_0$ . Similarly,  $q_0$  with input 10 outputs a 1 and remains in state  $q_0$ . Consider the input 11 to  $q_0$ . Now  $1 + 1 = 10$ ; i.e., the sum is 0 and carry is 1. In this case,  $q_0$  outputs a 0 and goes to state  $q_1$ .

Let us now consider the state  $q_1$  and the inputs 00, 01, 10, and 11. Suppose input to  $q_1$  is 00. In this case, we add the bits 0, 0 and also the carry 1. Now  $1 + 0 + 0 = 1$ ; i.e., the sum is 1 and there is no carry. Thus,  $q_1$  with input 00 outputs a 1 and goes to state  $q_0$ . Consider the input 01 to the state  $q_1$ . Here we calculate the sum  $1 + 0 + 1 = 10$ . That is, the sum is 0 and the carry is 1. Thus,  $q_1$  with input 01 outputs a 0 and remains in state  $q_1$ . Similarly,  $q_1$  with input 10 outputs a 0 and remains in state  $q_1$ . Next consider the input 11 to the state  $q_1$ . Here we add the bits 1, 1 and the carry 1; i.e.,  $1 + 1 + 1 = 11$ . That is, the sum is 1 and the carry is 1. Hence,  $q_1$  with input 11 outputs a 1 and remains in state  $q_1$ .

Considering all these possibilities, we can draw the directed graph of the required FSM as shown in Figure 13.25.

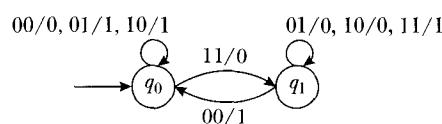


FIGURE 13.25 Serial adder

Let us add  $x = 11101$  and  $y = 01100$ . Notice that when we add the leftmost bits of  $x$  and  $y$ , to the sum of these bits we also add the carry produced by the sum of the previous bits. To get the correct sum of  $x$  and  $y$ , we input  $x$  and  $y$  as 011101 and 001100, respectively. That is, we put a 0 before each string. Now we add the corresponding bits from left to right. That is,

$$\begin{array}{r} 0 \ 1 \ 1 \ 1 \ 0 \ 1 \\ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \\ \hline & & & & & = x \\ & & & & & = y \end{array}$$

Therefore, the input bits, from right to left, are: 10, 00, 11, 11, 10, and 00. Now

$$q_0 \xrightarrow[1]{10} q_0 \xrightarrow[0]{00} q_0 \xrightarrow[0]{11} q_1 \xrightarrow[1]{11} q_1 \xrightarrow[0]{10} q_1 \xrightarrow[1]{00} q_0.$$

Thus,  $x + y = 101001$ . Note that the output string is written backward.

**DEFINITION 13.2.13** ▶ Let  $M$  be a FSM. Let  $x$  be a nonempty string in  $M$ . We say that  $x$  is accepted by  $M$  if and only if the output of  $x$  is 1.

### EXAMPLE 13.2.14

Consider the FSM  $M$ , whose transition diagram is shown in Figure 13.26.

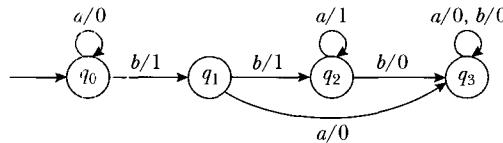


FIGURE 13.26 FSM  $M$

Let  $x = abbaa$ . Then

$$q_0 \xrightarrow[0]{a} q_0 \xrightarrow[1]{b} q_1 \xrightarrow[1]{b} q_2 \xrightarrow[1]{a} q_2 \xrightarrow[1]{a} q_2.$$

This implies that the output of string  $x$  is 1. Hence,  $x$  is accepted by  $M$ .

Let  $y = aaa$ . Then

$$q_0 \xrightarrow[0]{a} q_0 \xrightarrow[0]{a} q_0 \xrightarrow[0]{a} q_0.$$

This implies that string  $aaa$  is not accepted by  $M$ .

### EXAMPLE 13.2.15

In this example, we construct a FSM that accepts only those strings  $x$ , over the set  $\{a, b\}$ , such that  $x$  ends with  $ab$ .

Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM that accepts strings of this type. Then  $\Sigma = \{a, b\}$ . Because string  $ab$  is accepted by  $M$ , its output must be 1. This means we have at least three states in  $M$ , say  $q_0$ ,  $q_1$ , and  $q_2$ , such that  $q_0$  is the initial state and  $\delta(q_0, a) = q_1$ ,  $\gamma(q_0, a) = 0$ ,  $\delta(q_1, b) = q_2$ , and  $\gamma(q_1, b) = 1$ . Thus, so far we have the diagram shown in Figure 13.27.

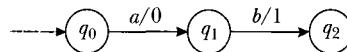


FIGURE 13.27 Partial diagram of  
FSM  $M$

Next we determine whether we need any more states. Now a string either begins with  $b$ 's or  $a$ 's. First let us consider those strings that begin with  $b$ 's. As

long as we have not seen the first  $a$ , we can stay in state  $q_0$  and continue to output 0. After seeing the first  $a$ , it is possible that the next input is  $b$ . Therefore, state  $q_0$  with input  $a$  goes to state  $q_1$  and also outputs 0. Now after processing the first  $a$ , we are in state  $q_1$ . As long as the next inputs consist of  $a$ 's, we can stay in  $q_1$  and continue to output 0. Now we are in state  $q_1$ , and the previous input was  $a$ . Suppose the next input is  $b$ . Now the last input was  $a$  and the current input is  $b$ . It is possible that the input ends here; i.e., the input ends with  $ab$ . Thus, if we are in state  $q_1$  and the next input is  $b$ , we go to state  $q_2$  and output 1. However, suppose that after the first  $ab$ , there are more inputs. Now the next input symbol is either  $a$  or  $b$ . If the next input is  $a$ , then we have the substring  $aba$ . It is quite possible that the next input is  $b$ . So if we are in state  $q_2$  and the input is  $a$ , we go to state  $q_1$  and output 0. However, if we are in state  $q_2$  and the next input is  $b$ , then we need to see string  $ab$  again. Therefore, if we are in state  $q_2$  and the next input is  $b$ , then we can go to state  $q_0$  and output 0.

If the input string starts with  $a$ 's, then with the first  $a$  we can go to state  $q_1$  and stay there until the next input is  $b$ . Following these arguments, we obtain the transition diagram of  $M$  shown in Figure 13.28.

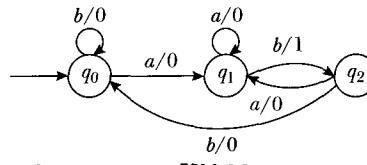


FIGURE 13.28 FSM  $M$

Notice that if a string does not end with  $ab$ , it is not accepted by  $M$ .

## WORKED-OUT EXERCISES

**Exercise 1:** Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$ , where  $Q = \{q_0, q_1, q_2, q_3\}$ ,  $\Sigma = \{a, b\}$ ,  $\Gamma = \{0, 1\}$ ,  $q_0$  is the initial state, and the functions  $\delta$  and  $\gamma$  are defined as follows:

	$\delta$		$\gamma$	
	$a$	$b$	$a$	$b$
$q_0$	$q_0$	$q_2$	0	1
$q_1$	$q_1$	$q_2$	1	0
$q_2$	$q_3$	$q_1$	1	1
$q_3$	$q_3$	$q_3$	1	1

- (a) Draw the transition diagram of  $M$ .
- (b) What is the output string if the input string is  $abbabab$ ?
- (c) What is the output of  $abbabab$ ?

### Solution:

- (a) The transition diagram of  $M$  is shown in Figure 13.29.

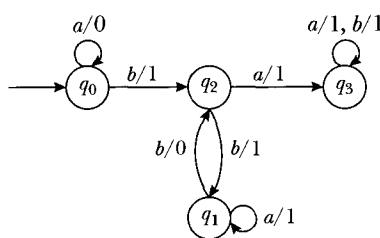


FIGURE 13.29 Transition diagram of  $M$

- (b) We have

$$q_0 \xrightarrow[0]{a} q_0 \xrightarrow[1]{b} q_2 \xrightarrow[1]{b} q_1 \xrightarrow[1]{a} q_1 \xrightarrow[0]{b} q_2 \xrightarrow[1]{a} q_3 \xrightarrow[1]{b} q_3.$$

The number below the arrow shows the output of the state. It follows that the output string is 0111011.

- (c) By part (b), the output of  $abbabab$  is 1.

**Exercise 2:** Consider a vending machine (shown in Figure 13.30) that sells newspaper.

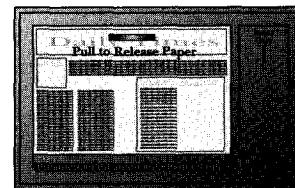


FIGURE 13.30

Suppose the cost of the newspaper is 25 cents. The machine accepts any sequence of 5-, 10-, or 25-cent coins. After inserting at least 25 cents, the customer can press the button to release the newspaper. For simplicity assume that if a customer inputs more than 25 cents, then the machine does not return the change. Moreover, after selling the newspaper,

the machine returns to the initial state; i.e., it is ready to process the next request. Construct a finite state machine that models this vending machine.

**Solution:** The input to the finite state machine is a sequence of change and the newspaper-release button. Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be the required finite state machine. Let us first determine the number of states of  $M$ , i.e., the number of elements of  $Q$ . Suppose a customer first puts in a 5-cent coin. After inputting the first 5-cent coin, the remaining required change is 20 cents. At this point, the user can input another 5-cent coin, or 10-cent coin, or 25-cent coin. If the customer inputs a 5-cent coin, then the remaining required change is 15 cents. If the customer inputs a 10-cent coin, then the remaining required change is 10 cents. If the customer inputs a 25-cent coin, then the user can press the newspaper-release button and get the newspaper. It follows that we need a state corresponding to the change needed to buy the newspaper. After inputting the first 5-cent coin, the remaining required change is 20 cents, and so on. Because  $5 \cdot 5 = 25$ , there are five states corresponding to the change required to buy the newspaper. We need one more state that corresponds to no more change required. Thus, there needs to be a total of six states. Let  $Q = \{q_0, q_1, \dots, q_5\}$ , where the

state  $q_i$  corresponds to the state that needs  $25 - 5i$  cents. The state  $q_0$  is the initial state.

The input to the vending machine is any sequence of 5-, 10-, or 25-cent coins or the newspaper-release button, say  $E$ . Hence,  $\Sigma = \{5, 10, 25, B\}$ . Corresponding to an input to a state, the output is either release the newspaper or do not release the newspaper. Suppose that  $n$  means do not release the newspaper and  $r$  means release the newspaper. Thus,  $\Gamma = \{n, r\}$ . Figure 13.31 shows the transition diagram of the FSM that models the vending machine.

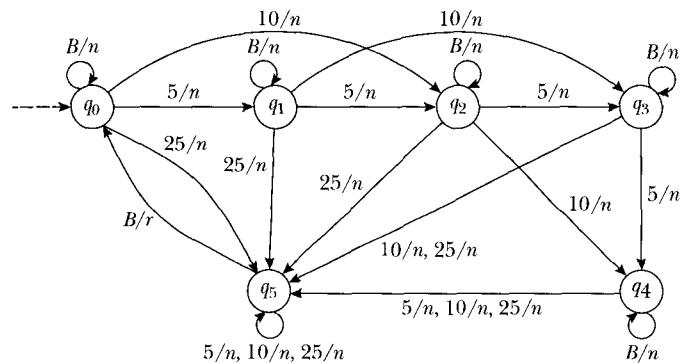


FIGURE 13.31 FSM that models newspaper-vending machine

## SECTION REVIEW

### Key Terms

finite state machine (FSM)  
states  
input alphabet  
output alphabet

initial (start) state  
state transition function  
output function  
Mealy sequential machine

output string  
output of the string  
label of a directed walk

### Some Key Definitions

1. A finite state machine (FSM) is a sextuple  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$ , where
  - (i)  $Q$  is a finite nonempty set of states,
  - (ii)  $\Sigma$  is the input alphabet (a finite nonempty set of symbols),
  - (iii)  $\Gamma$  is the output alphabet (a finite nonempty set of symbols),
  - (iv)  $q_0$  is the initial (or start) state, a particular element of  $Q$ ,
  - (v)  $\delta$  is the state transition function,  $\delta : Q \times \Sigma \rightarrow Q$ , and
  - (vi)  $\gamma$  is the output function,  $\delta : Q \times \Sigma \rightarrow \Gamma$ .
2. Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM. Let  $x = x_1 x_2 \cdots x_n$  be a nonempty string over  $\Sigma$ . Then  $y = y_1 y_2 \cdots y_n$ ,  $y_i \in \Gamma$ ,  $i = 1, 2, \dots, n$ , is the corresponding output string if  $\gamma^*(q_0, x_1 x_2 \cdots x_n) = y_1 y_2 \cdots y_n$ . We call  $y_n$  the output of the string  $x$ . In this case, we also say that  $y_n$  is the output of  $M$  for the string  $x$  or simply  $M$  outputs  $y_n$  for the string  $x$ .

3. Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM and let  $G_M$  be its transition diagram. If

$$P : q_{i_0} \xrightarrow{a_1/b_1} q_{i_1} \xrightarrow{a_2/b_2} q_{i_2} \xrightarrow{a_3/b_3} \cdots \xrightarrow{a_n/b_n} q_{i_n}$$

is a directed walk, where  $\delta(q_{i_{k-1}}, a_k) = q_{i_k} \in Q$ ,  $\gamma(q_{i_{k-1}}, a_k) = b_k \in \Gamma$ , then the expression  $(a_1/b_1)(a_2/b_2) \cdots (a_{n-1}/b_{n-1})(a_n/b_n)$  is called the label of the directed walk  $P$ .

4. Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM and let  $G_M$  be its transition diagram. Suppose

$$P : q_{i_0} \xrightarrow{a_1/b_1} q_{i_1} \xrightarrow{a_2/b_2} q_{i_2} \xrightarrow{a_3/b_3} \cdots \xrightarrow{a_n/b_n} q_{i_n}$$

is a directed walk. Sometimes for clarity and ease of reading, we write  $P$  as

$$P : q_{i_0} \xrightarrow[b_1]{a_1} q_{i_1} \xrightarrow[b_2]{a_2} q_{i_2} \xrightarrow[b_3]{a_3} \cdots \xrightarrow[b_n]{a_n} q_{i_n},$$

i.e., the input symbol is written on top of the arrow and the output symbol is written below the arrow.

5. Let  $M$  be a FSM. Let  $x$  be a nonempty string in  $M$ . We say that  $x$  is accepted by  $M$  if and only if the output of  $x$  is 1.

## Some Key Results

- Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM. Then for any  $u, v \in \Sigma^*$  and for any  $q \in Q$ ,  $\gamma^*(q, uv) = \gamma^*(q, u)\gamma^*((\delta^*(q, u), v))$ .
- Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM. Let  $x = a_1 a_2 \cdots a_n$  be a nonempty string over  $\Sigma$ . Let  $y = b_1 b_2 \cdots b_n$ ,  $b_i \in \Gamma$ ,  $i = 1, 2, \dots, n$ . Then  $\gamma^*(q_0, a_1 a_2 \cdots a_n) = b_1 b_2 \cdots b_n$  if and only if there is a directed walk  $P$  in  $G_M$  with the label

$$(a_1/b_1)(a_2/b_2) \cdots (a_{n-1}/b_{n-1})(a_n/b_n)$$

from  $q_0$  to  $\delta^*(q_0, a_1 a_2 \cdots a_n)$ .

## EXERCISES

1. Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM such that the transition table of  $M$  is

		$\delta$	$\gamma$
		$a$	$b$
$a$	$b$	$a$	$b$
$q_0$	$q_2$	$q_1$	1 1
$q_1$	$q_2$	$q_2$	0 0
$q_2$	$q_1$	$q_2$	1 1

- Draw the transition diagram of  $M$ .
  - What is the output string if the input string is  $aabb$ ?
  - What is the output string if the input string is  $ababab$ ?
  - What is the output of the string  $abbbaba$ ?
  - What is the output of the string  $bbbbabab$ ?
2. Let  $M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$  be a FSM such that the transition table of  $M$  is

		$\delta$	$\gamma$		
		$a$	$b$	$a$	$b$
$q_0$	$q_0$	$q_3$		1	0
$q_1$	$q_3$	$q_2$		0	1
$q_2$	$q_1$	$q_2$		0	1
$q_3$	$q_2$	$q_0$		0	1

- Draw the transition diagram of  $M$ .
  - What is the output string if the input string is  $bbabba$ ?
  - What is the output string if the input string is  $abbbabab$ ?
  - What is the output of the string  $bbbababa$ ?
  - What is the output of the string  $aaaaaa$ ?
3. Let  $M$  be a FSM such that the transition diagram of  $M$  is given in Figure 13.32.

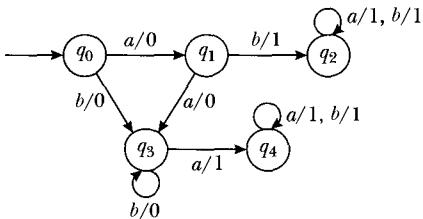


FIGURE 13.32 Transition diagram of a FSM

- Write the transition table for  $M$ .
  - What is the output string if the input string is  $aaabbbb$ ?
  - What is the output string if the input string is  $bbbaaaa$ ?
  - Is the string  $aaa$  accepted by  $M$ ?
4. Consider the FSM  $M$  of Exercise 3. Which of the following strings are accepted by  $M$ ?
- $ba$
  - $aabbba$
  - $bbbbbb$
  - $aaabbbb$
5. Construct a FSM that delays the input string by two time units. In this case, the first two bits of the output string are 00.

- The FSM of Worked-Out Exercise 2 does not return the change if the customer deposits more than 25 cents. Redesign the FSM so that if the user deposits more than 25 cents, then in addition to releasing the newspaper, the machine also returns the change.
- A vending machines sells three types of candies and the cost of a candy is 30 cents. Construct a FSM to model the candy machine. If the customer deposits more than 30 cents, then the machine must return the change.
- Design a FSM, with input alphabet  $\Sigma = \{a, b\}$ , that outputs a 1 if the input symbols read so far consist of an even number of  $a$ 's.
- Design a FSM, with input alphabet  $\Sigma = \{a, b\}$ , that outputs a 1 if the number of input symbols read so far is divisible by 3.
- Design a FSM that accepts all strings over  $\{a, b\}$  that begin with  $aa$ .
- Let  $\Sigma = \{0, 1, 2, 3\}$ . Design a FSM that accepts all strings such that the sum of the digits of the strings is divisible by 4.
- Design a FSM that accepts all strings over  $\{a, b\}$  that contain at least two  $a$ 's.

## 13.3 GRAMMARS AND LANGUAGES

In the first section, we studied regular languages. We also showed their relationship to finite automata. Every natural language, such as English, French, and Spanish, has some specific rules which are described in the grammar of the language. Regular languages have another characterization through the notion of grammars. In this section, we introduce two types of grammars—context-free grammar and regular grammar—and discuss the basic properties of the languages they generate. We begin by describing context-free grammars.

**DEFINITION 13.3.1** ▶ A **context-free grammar**  $G$  is a quadruple  $(V_N, \Sigma, P, S)$ , where

- (i)  $V_N$  is a finite set, called the set of **nonterminal symbols**,
- (ii)  $\Sigma$  is a finite set, called the set of **terminal symbols**,  $V_N \cap \Sigma = \emptyset$ ,
- (iii)  $P$  is a finite subset of  $V_N \times (V_N \cup \Sigma)^*$ , and
- (iv)  $S$  is a distinguished element of  $V_N$  called the **start symbol**.

Elements of  $P$  are called the **rules of the grammar**, or **rules**, or **productions**. A rule is an ordered pair  $(A, \alpha) \in V_N \times (V_N \cup \Sigma)^*$ , where  $A$  is a nonterminal symbol and  $\alpha$  is a string on  $V_N \cup \Sigma$ . A rule  $(A, \alpha)$  is usually written as  $A \rightarrow \alpha$ . A rule of the form  $A \rightarrow \alpha$  is called an  **$A$ -rule**, referring to the nonterminal symbol  $A$  on the left-hand side. Now the empty string  $\lambda \in (V_N \cup \Sigma)^*$ . Hence, in a grammar there may exist a rule of the form  $A \rightarrow \lambda$ . This rule is called the **null, or lambda, rule**.

### EXAMPLE 13.3.2

Let  $V_N = \{A, S\}$ ,  $\Sigma = \{0, 1\}$ , and  $P = \{A \rightarrow 0, S \rightarrow 1S0, S \rightarrow 1A0\}$ . Then  $G = (V_N, \Sigma, P, S)$  is a context-free grammar.

**REMARK 13.3.3** ► If  $A \rightarrow \alpha_1, A \rightarrow \alpha_2, \dots, A \rightarrow \alpha_n$  are the rules in a context-free grammar, then we may express them by the notation  $A \rightarrow \alpha_1 | \alpha_2 | \dots | \alpha_n$ , where the symbol “ $\alpha_i | \alpha_j$ ” is read as  $\alpha_i$  or  $\alpha_j$ .

There is an alternative way to write the rules of a context-free grammar using Backus-Naur form (BNF). In BNF, each nonterminal symbol is placed inside  $\langle \rangle$ . The rule  $A \rightarrow \alpha$  is written  $\langle A \rangle ::= \alpha$ . Rules of the form

$$\langle A \rangle ::= \alpha_1, \quad \langle A \rangle ::= \alpha_2, \quad \langle A \rangle ::= \alpha_3, \dots, \quad \langle A \rangle ::= \alpha_n,$$

may be written

$$\langle A \rangle ::= \alpha_1 | \alpha_2 | \alpha_3 | \alpha_4 | \dots | \alpha_n,$$

where the symbol “ $\alpha_i | \alpha_j$ ” is read as  $\alpha_i$  or  $\alpha_j$ .

### EXAMPLE 13.3.4

In this example, we describe a context-free grammar  $G$  whose rules are written in BNF. In this grammar,

$$V_N = \{\langle digit \rangle, \langle integer \rangle, \langle signed \text{ integer} \rangle, \langle unsigned \text{ integer} \rangle\}, \\ \Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -\},$$

$P$  consists of

$$\begin{aligned} \langle digit \rangle &::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 \\ \langle integer \rangle &::= \langle signed \text{ integer} \rangle | \langle unsigned \text{ integer} \rangle \\ \langle signed \text{ integer} \rangle &::= + \langle unsigned \text{ integer} \rangle | - \langle unsigned \text{ integer} \rangle \\ \langle unsigned \text{ integer} \rangle &::= \langle digit \rangle | \langle digit \rangle \langle unsigned \text{ integer} \rangle \end{aligned}$$

and  $S = \langle integer \rangle$ .

**DEFINITION 13.3.5** ► Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar. Suppose  $xAy$  and  $xwy$  are two strings in  $(V_N \cup \Sigma)^*$ .

- (i)  $xwy$  is said to be **directly derivable** from  $xAy$ , written  $xAy \Rightarrow xwy$ , if  $A \rightarrow w$  is a rule in  $G$ .
- (ii) If  $\alpha_1, \alpha_2, \dots, \alpha_{n+1}$  are strings in  $(V_N \cup \Sigma)^*$  such that  $\alpha_1 \Rightarrow \alpha_2 \Rightarrow \dots \Rightarrow \alpha_n \Rightarrow \alpha_{n+1}$ , then we say that  $\alpha_{n+1}$  is **derivable** from  $\alpha_1$  in  $G$  and  $\alpha_1 \Rightarrow \alpha_2 \Rightarrow \dots \Rightarrow \alpha_n \Rightarrow \alpha_{n+1}$  is a **derivation** of  $\alpha_{n+1}$  from  $\alpha_1$  of **length**  $n$ .
- (iii) Let  $\alpha_1, \alpha_2, \dots, \alpha_{n+1}$  be strings in  $(V_N \cup \Sigma)^*$ . If  $\alpha_{n+1}$  is derivable from  $\alpha_1$  by a derivation of length  $n$ , then we write  $\alpha_1 \Rightarrow^n \alpha_{n+1}$ .

**REMARK 13.3.6** ► Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar.

- (i) We assume that every string  $\alpha \in (V_N \cup \Sigma)^*$  is directly derivable from itself in  $G$ .
- (ii) Let  $\alpha_1, \alpha_2, \dots, \alpha_{n+1}$  be strings in  $(V_N \cup \Sigma)^*$ . If  $\alpha_{n+1}$  is derivable from  $\alpha_1$ , then we write  $\alpha_1 \Rightarrow^* \alpha_{n+1}$ .

### EXAMPLE 13.3.7

Consider the grammar  $G = (V_N, \Sigma, P, S)$ , where  $V_N = \{A, S\}$ ,  $\Sigma = \{0, 1\}$ ,  $P = \{A \rightarrow 0, S \rightarrow 1S1, S \rightarrow 0A0\}$ . In this grammar, we find that  $1A0 \Rightarrow 100$ , because

$A \rightarrow 0$  is a rule in  $G$ . We show that the string 011000111 is derivable in  $G$  from 0S1.

$$\begin{aligned} 0S1 &\Rightarrow 01S11 && \text{by the rule } S \rightarrow 1S1 \\ &\Rightarrow 011S111 && \text{by the rule } S \rightarrow 1S1 \\ &\Rightarrow 0110A0111 && \text{by the rule } S \rightarrow 0A0 \\ &\Rightarrow 011000111 && \text{by the rule } A \rightarrow 0 \end{aligned}$$

Hence,  $0S1 \Rightarrow 01S11 \Rightarrow 011S111 \Rightarrow 0110A0111 \Rightarrow 011000111$  is a derivation of 011000111 from 0S1 in  $G$  and the length of this derivation is 4. We can write  $0S1 \Rightarrow^4 011000111$ .

We say that the string 011000111 has been derived from 0S1 in this grammar. In symbols, we express this derivation by  $0S1 \Rightarrow^* 011000111$ .

The derivability of a string  $w$  from another string  $u$  is defined formally as follows.

---

**DEFINITION 13.3.8** ▶ Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar and  $u \in (V_N \cup \Sigma)^*$ . The set of strings **derivable** from  $u$  is defined as follows.

- (i)  $u$  is derivable from  $u$ .
- (ii) If  $v = xAy \in (V_N \cup \Sigma)^*$  is derivable from  $u$  and  $A \rightarrow w$  is a rule in  $G$ , then  $xwy$  is derivable from  $u$ .
- (iii) Only those strings of  $(V_N \cup \Sigma)^*$  that are derivable from  $u$  by finitely many applications of (ii) are derivable from  $u$ .

If  $u$  and  $v$  are two strings in  $(V_N \cup \Sigma)^*$  such that  $v$  is derivable from  $u$ , then we express it by  $u \Rightarrow^* v$ . If  $v$  is derivable from  $u$ , then we also say that  $u$  derives  $v$ . One can define a relation  $R$  on the set  $(V_N \cup \Sigma)^*$  by  $u R v$  if  $u \Rightarrow^* v$ . It follows easily that  $R$  is reflexive and transitive.

Let  $uAv \in (V_N \cup \Sigma)^*$ . We may think of  $u$  and  $v$  as providing context for the symbol  $A$ . If  $A \rightarrow \alpha$  is a rule in  $G$ , then we have a derivation  $uAv \Rightarrow u\alpha v$  in  $G$ . This derivation takes place irrespective of the context. This is where the term **context-free** comes from.

---

**DEFINITION 13.3.9** ▶ Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar.

- (i) A string  $\alpha \in (V_N \cup \Sigma)^*$  is said to be in **sentential form** of  $G$  if there is a derivation  $S \Rightarrow^* \alpha$  in  $G$ .
- (ii) A string  $w \in \Sigma^*$  is said to be a **sentence** of  $G$  if there is a derivation  $S \Rightarrow^* w$  in  $G$ .
- (iii) The set of all sentences of  $G$  is called the **language** of  $G$  or **language generated by**  $G$ .

Note that a string that may contain terminal symbols and nonterminal symbols is in sentential form if it can be derived from the starting symbol, but a string is said to be a sentence in the grammar  $G$  if it is either empty or contains only terminal symbols and if it can be derived from the starting symbol. We illustrate these concepts in Example 13.3.10.

We denote the language generated by a grammar  $G$  by  $L(G)$ . Then  $L(G)$  is a subset of  $\Sigma^*$  such that  $L(G) = \{w \in \Sigma^* \mid S \Rightarrow^* w \text{ in } G\}$ .

**EXAMPLE 13.3.10**

Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where  $V_N = \{S\}$ ,  $\Sigma = \{a, b\}$ ,  $P = \{S \rightarrow ab, S \rightarrow aSb\}$ . In this grammar, we have

$$\begin{aligned} S &\Rightarrow aSb && \text{by the rule } S \rightarrow aSb \\ &\Rightarrow aaSbb && \text{by the rule } S \rightarrow aSb \\ &\Rightarrow aaaSbbb && \text{by the rule } S \rightarrow aSb \end{aligned}$$

Hence, we find that  $S \Rightarrow^* aaaSbbb$  in  $G$ . This implies that the string  $aaaSbbb$  is in sentential form of  $G$ .

Again

$$\begin{aligned} S &\Rightarrow aSb && \text{by the rule } S \rightarrow aSb \\ &\Rightarrow aaSbb && \text{by the rule } S \rightarrow aSb \\ &\Rightarrow aaaSbbb && \text{by the rule } S \rightarrow aSb \\ &\Rightarrow aaaabbbb && \text{by the rule } S \rightarrow ab \end{aligned}$$

Thus,  $S \Rightarrow^* aaaabbbb$  in  $G$ . Now the string  $aaaabbbb \in \Sigma^*$ . This implies that the string  $aaaabbbb$  is a sentence of  $G$  and so  $aaaabbbb \in L(G)$ .

Let  $T = \{a^n b^n \in \Sigma^* \mid n > 0\}$ . Suppose  $w = a^n b^n$  for some positive integer  $n > 0$ . Now

$$\begin{aligned} S &\Rightarrow^{n-1} a^{n-1} S b^{n-1} && \text{by the repeated application of the rule } S \rightarrow aSb \\ &\Rightarrow a^n b^n && \text{by the rule } S \rightarrow ab \end{aligned}$$

Thus, we find that  $S \Rightarrow^* a^n b^n$  in  $G$ . Hence, it follows that  $T = \{a^n b^n \in \Sigma^* \mid n > 0\} \subseteq L(G)$ .

On the other hand, because  $S \rightarrow ab$  and  $S \rightarrow aSb$  are the only rules in this grammar, it follows that any sentential form of  $G$  must be of the form  $ab$  or  $a^n S b^n$ . Moreover, from  $a^n S b^n$  we can derive  $a^{n+1} b^{n+1}$  by the rule  $S \rightarrow ab$ . Thus, any sentence of  $G$  is of the form  $a^m b^m$ . Hence, it follows that  $L(G) = \{a^n b^n \in \Sigma^* \mid n > 0\}$ .

**EXAMPLE 13.3.11**

Let  $G = (V_N, \Sigma, P, S)$ , where  $V_N = \{S, A, N, V, R\}$ ,  $\Sigma = \{\text{Jim, big, cake, ate}\}$ ,  $P = \{R \rightarrow N, R \rightarrow AR, S \rightarrow RVR, A \rightarrow \text{big}, N \rightarrow \text{Jim}, N \rightarrow \text{cake}, V \rightarrow \text{ate}\}$ . Then  $G$  is a context-free grammar. In this grammar, we have the following derivation

$$\begin{aligned} S &\Rightarrow RVR && \text{by the rule } S \rightarrow RVR \\ &\Rightarrow NVR && \text{by the rule } R \rightarrow N \\ &\Rightarrow \text{JimVR} && \text{by the rule } N \rightarrow \text{Jim} \\ &\Rightarrow \text{JimateR} && \text{by the rule } V \rightarrow \text{ate} \\ &\Rightarrow \text{JimateAR} && \text{by the rule } R \rightarrow AR \\ &\Rightarrow \text{JimatebigN} && \text{by the rule } A \rightarrow \text{big} \\ &\Rightarrow \text{Jimatebigcake} && \text{by the rule } N \rightarrow \text{cake} \end{aligned}$$

Thus,  $S \Rightarrow^* \text{Jimatebigcake}$ , and because the string  $\text{Jimatebigcake} \in \Sigma^*$ , it follows that  $\text{Jimatebigcake} \in L(G)$ . Hence, loosely speaking, we can say that this context-free grammar generates the sentence *Jim ate big cake*.

**DEFINITION 13.3.12** ▶ Let  $\Sigma$  be an alphabet. A language  $L$  on  $\Sigma$  is called a **context-free language (CFL)** if there exists a context-free grammar  $G = (V_N, \Sigma, P, S)$  such that the language generated by  $G$  is  $L$ , i.e.,  $L(G) = L$ .

**REMARK 13.3.13** ▶ From Example 13.3.10, it follows that the language  $L = \{a^n b^n \in \Sigma^* \mid n > 0\}$  is a context-free language on the alphabet  $\Sigma = \{a, b\}$ . However, in Example 13.1.24, as well as in Example 13.1.28, we proved that  $L = \{a^n b^n \in \Sigma^* \mid n > 0\}$  is not a regular language.

**EXAMPLE 13.3.14**

In this example, we show that the language  $L = \{0^n 1^m \in \Sigma^* \mid n, m \text{ are nonnegative integers}\}$  is a context-free language on the alphabet  $\Sigma = \{0, 1\}$ . For this we consider the grammar  $G = (V_N, \Sigma, P, S)$ , where  $V_N = \{S\}$ ,  $\Sigma = \{0, 1\}$ , and  $P = \{S \rightarrow 0S, S \rightarrow S1, S \rightarrow \lambda\}$ . Now  $G$  is a context-free grammar. In this grammar, we obtain

$$S \Rightarrow 0S \Rightarrow 00S \Rightarrow 000S \Rightarrow \cdots \Rightarrow 0^n S$$

(by the application of the rule  $S \rightarrow 0S$ ,  $n$  times), i.e.,

$$S \Rightarrow^n 0^n S.$$

Next we apply the rule  $S \rightarrow S1$ , to obtain

$$\begin{aligned} S &\Rightarrow^n 0^n S \\ &\Rightarrow^n 0^n S1 && \text{by the rule } S \rightarrow S1 \\ &\Rightarrow^n 0^n S1^2 && \text{by the rule } S \rightarrow S1 \\ &\Rightarrow^n 0^n S1^3 && \text{by the rule } S \rightarrow S1 \\ &\vdots \\ &\Rightarrow^n 0^n S1^m && \text{by the rule } S \rightarrow S1 \\ &\Rightarrow^n 0^n 1^m && \text{by the rule } S \rightarrow \lambda \end{aligned}$$

Hence,  $\{0^n 1^m \in \Sigma^* \mid n, m \text{ are nonnegative integers}\} \subseteq L(G)$ .

Conversely, let  $w \in \Sigma^*$  be such that  $w \in L(G)$ . Then  $S \Rightarrow^* w$ . Hence, there exists a derivation

$$S \Rightarrow \alpha_1 \Rightarrow \alpha_2 \Rightarrow \alpha_3 \Rightarrow \cdots \alpha_{k-1} \Rightarrow \alpha_k = w.$$

Because  $\alpha_{k-1} \Rightarrow \alpha_k$  is derived by the rule  $S \rightarrow \lambda$ ,  $\alpha_{k-1}$  must contain  $S$  and the number of occurrences of  $S$  in  $\alpha_{k-1}$  is one. Again in each of the other steps of the above derivation, we used either the rule  $S \rightarrow 0S$  or  $S \rightarrow S1$ . It follows that  $\alpha_{k-1}$  must be of the form  $0^n S1^m$ , where  $n, m$  are nonnegative integers. Therefore,  $w \in \{0^n 1^m \in \Sigma^* \mid n, m \text{ are nonnegative integers}\}$ . Thus, it follows that  $L(G) = \{0^n 1^m \in \Sigma^* \mid n, m \text{ are nonnegative integers}\}$ . Hence,  $\{0^n 1^m \in \Sigma^* \mid n, m \text{ are nonnegative integers}\}$  is a CFL.

In the first section of this chapter, we described the pumping lemma (Theorem 13.1.27) for regular languages and effectively used it to prove that certain languages are not regular. Analogous to the pumping lemma for regular languages there is also such a result for context-free languages, described next. This theorem is known as the *pumping lemma for context-free languages*. (For a proof of this theorem see [61] in the references.)

**Theorem 13.3.15:** Let  $L$  be a context-free language. There exist positive integers  $n$  depending on  $L$ , such that if there is a string  $w$  in  $L$  with  $|w| \geq n$ , then  $w$  can be written as  $w = xyzuv$  such that  $|yu| \geq 1$ ,  $|yzu| \leq n$ , and for each integer  $i \geq 0$ ,  $xy^i zu^i v \in L$ .

Just as the pumping lemma plays an important role in studying the properties of regular languages, Theorem 13.3.15 is useful in studying the properties of context-free languages.

**EXAMPLE 13.3.16**

In this example, we use Theorem 13.3.15 to show that the language

$$L = \{a^m b^m c^m \mid m \geq 1\}$$

over  $\Sigma = \{a, b, c\}$  is not a context-free language.

Suppose  $L$  is a context-free language. Then there exists a positive integer  $n$  such that if there is a string  $w$  in  $L$  with  $|w| \geq n$ , then  $w$  can be written as  $w = xyzuv$  such that  $|yu| \geq 1$ ,  $|yzu| \leq n$ , and for each integer  $i \geq 0$ ,  $xy^i zu^i v \in L$ . Now  $L = \{a^m b^m c^m \mid m \geq 1\}$  contains an infinite number of strings. Thus, we can find a string  $w = a^m b^m c^m$  in  $L$  such that  $m > n$ . Hence, by Theorem 13.3.15,  $w$  can be written as  $w = xyzuv$  such that  $|yu| \geq 1$ ,  $|yzu| \leq n$  and for each integer  $i \geq 0$ ,  $xy^i zu^i v \in L$  for all  $i = 0, 1, 2, \dots$ . Because  $|yzu| \leq n$  and  $|b^m| = m > n$ ,  $yzu$  cannot contain both  $a$  and  $c$ . Hence,  $yu$  cannot contain both  $a$  and  $c$ .

Suppose  $yu$  contains  $a$ 's. Note that  $|yu| \geq 1$ . Then in  $xy^0 zu^0 v = xzv$ , the number of  $a$ 's cannot be equal to the number of  $c$ 's. Hence,  $xy^0 zu^0 v \notin L$ . Similarly, if  $yu$  contains  $c$ 's, then in  $xy^0 zu^0 v = xzv$ , the number of  $c$ 's cannot be equal to the number of  $a$ 's. Hence,  $xy^0 zu^0 v \notin L$ . But  $xy^0 zu^0 v \in L$ , a contradiction. Hence,  $L$  is not a CFL.

---

**DEFINITION 13.3.17** ▶ Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar. A rooted tree  $T$  is called a **derivation tree** in  $G$  if

- (i) every vertex has a label, which is a symbol of  $V_N \cup \Sigma \cup \{\lambda\}$  (two distinct vertices may have the same label);
- (ii) the label of the root of the tree is  $S$ ;
- (iii) if a vertex has label  $A$  and the children (in order from left to right) of this vertex have labels  $x_1, x_2, \dots, x_n$ , respectively, then  $A \rightarrow x_1 x_2 \dots x_n$  is a rule of  $G$ , and
- (iv) if a vertex has label  $\lambda$ , then this vertex is a leaf and it is the only child of its parent.

The following example explains Definition 13.3.17.

Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where  $V_N = \{S, A, B\}$ ,  $\Sigma = \{a, b\}$ ,  $P = \{S \rightarrow aAS, S \rightarrow bA, S \rightarrow aB, S \rightarrow a, B \rightarrow bS, A \rightarrow SS, A \rightarrow ab, A \rightarrow a\}$ . The rooted tree shown in Figure 13.33 is a derivation tree in this grammar.

Let  $T$  be a derivation tree in a context-free grammar  $G = (V_N, \Sigma, P, S)$ . This derivation tree can be assumed to be an ordered rooted tree (see Chapter 11). Hence, we have from left-to-right ordering of children of a vertex. We can extend this ordering of children to a left-to-right ordering of leaves. Given two distinct leaves  $x_1$  and  $x_2$ , we follow the paths from these vertices until we meet at a vertex, say  $w$ . Let  $y_1$  and  $y_2$  be two children of  $w$ . Suppose  $y_1$  is on the path from  $x_1$  to  $w$  and  $y_2$  is on the path from  $x_2$  to  $w$ . According to the left-to-right ordering of children of  $w$ , let  $y_1$  be to the left of  $y_2$ . Then we say that  $x_1$  is to the left of  $x_2$ .

---

**DEFINITION 13.3.18** ▶ Let  $T$  be a derivation tree for a context-free grammar  $G = (V_N, \Sigma, P, S)$ . If  $x_1, x_2, \dots, x_n$  are the labels of the leaves of  $T$  in the order from left-to-right, then the string  $\alpha = x_1 x_2 \dots x_n$  is called the **yield of the derivation tree**.

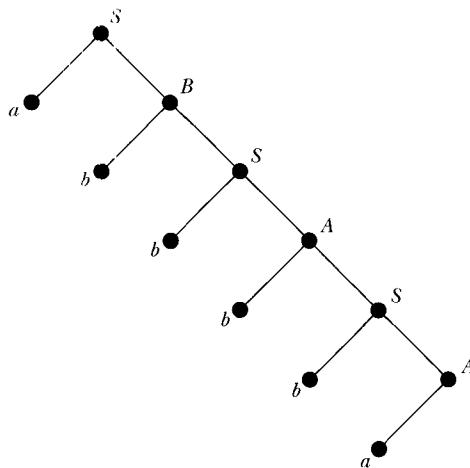


FIGURE 13.33 Derivation tree

The yield of the derivation tree of Figure 13.33 is  $abbbba$ .

We state the following theorem, without proof, which describes the relationship between derivations and derivation trees of a context-free grammar.

**Theorem 13.3.19:** Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar. Then  $S \Rightarrow^* \alpha$  in  $G$  if and only if there is a derivation tree of  $G$  with yield  $\alpha$ .

**DEFINITION 13.3.20** ► A context-free grammar  $G = (V_N, \Sigma, P, S)$  is called a **right-linear grammar** if each of the rules of  $G = (V_N, \Sigma, P, S)$  is of the form

- (i)  $A \rightarrow w,$
- (ii)  $A \rightarrow wB,$   
where  $A, B \in V_N$  and  $w \in \Sigma^*$ .

**DEFINITION 13.3.21** ► A context-free grammar  $G = (V_N, \Sigma, P, S)$  is called a **left-linear grammar** if each of the rules of  $G = (V_N, \Sigma, P, S)$  is of the form

- (i)  $A \rightarrow w,$
- (ii)  $A \rightarrow Bw,$   
where  $A, B \in V_N$  and  $w \in \Sigma^*$ .

**DEFINITION 13.3.22** ► A right-linear or left-linear grammar is called a **regular grammar**.

**REMARK 13.3.23** ► Note that in a regular grammar, at most one nonterminal symbol appears in the right side of any rule (i.e., the right side of the arrow), and that nonterminal symbol is either the rightmost or leftmost symbol of the right side of any rule.

**EXAMPLE 13.3.24**

Consider the following context-free grammar  $G = (V_N, \Sigma, P, S)$ , where  $V_N = \{S, A\}$ ,  $\Sigma = \{a, b\}$  and  $P = \{S \rightarrow aS, S \rightarrow bA, A \rightarrow bA, A \rightarrow \lambda\}$ . Then  $G$  is a right-linear grammar. In this grammar,

$$\begin{aligned} S &\Rightarrow aS \Rightarrow aaS \Rightarrow aaaS \Rightarrow aaabA \Rightarrow aaabbA \Rightarrow aaabbbA \\ &\Rightarrow aaabbbbA \Rightarrow aaabbbb\lambda = aaabbbb. \end{aligned}$$

This derivation implies that  $aaabbbb \in L(G)$ . In fact, we can show that

$$L(G) = \{a^n b^m \mid n \geq 0, m \geq 1\}.$$

From Worked-Out Exercise 3 (page 846), it follows that  $L(G) = \{a^n b^m \mid n \geq 0, m \geq 1\}$  is a regular language. Hence, the language generated by  $G$  is a regular language.

### EXAMPLE 13.3.25

Consider the context-free grammar  $G = (V_N, \Sigma, P, S)$ , where  $V_N = \{S, A\}$ ,  $\Sigma = \{a, b\}$  and  $P = \{S \rightarrow Sab, S \rightarrow b\}$ . Then  $G$  is a left-linear grammar. In this grammar,

$$S \Rightarrow Sab \Rightarrow Sabab \Rightarrow Sababab \Rightarrow bababab.$$

This implies that  $bababab \in L(G)$ . We can show that

$$L(G) = \{b(ab)^m \mid m \text{ is a nonnegative integer}\}.$$

Now consider the grammar  $G_1 = (V_N, \Sigma, P, S)$ , where  $V_N = \{S, A\}$ ,  $\Sigma = \{a, b\}$  and  $P = \{S \rightarrow bA, A \rightarrow abA, A \rightarrow \lambda\}$ . Then  $G_1$  is a right-linear grammar. In this grammar,

$$S \Rightarrow bA \Rightarrow babA \Rightarrow bababA \Rightarrow babababA \Rightarrow bababab.$$

This implies that  $bababab \in L(G_1)$ . We can show that

$$L(G_1) = \{b(ab)^m \mid m \text{ is a nonnegative integer}\}.$$

Hence,  $L(G) = L(G_1)$ .

### EXAMPLE 13.3.26

In this example, we consider the language  $L = \{abw \mid w \in \{a, b\}^*\}$ . We know that this is a regular language. Consider the DFA  $M = (Q, \Sigma, \delta, q_0, F)$ , where  $Q = \{q_0, q_1, q_2, q_3\}$ ,  $\Sigma = \{a, b\}$ ,  $F = \{q_2\}$ , and  $\delta$  is given by

$\delta$	$a$	$b$
$q_0$	$q_1$	$q_3$
$q_1$	$q_3$	$q_2$
$q_2$	$q_2$	$q_2$
$q_3$	$q_3$	$q_3$

The transition diagram of  $M$  is shown in Figure 13.34.

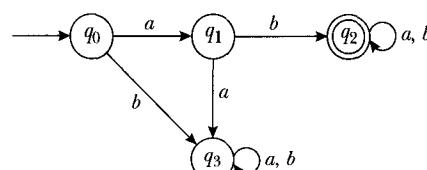


FIGURE 13.34 Transition diagram of  $M$

Note that  $L(M) = L$  and hence  $L$  is a regular language.

Next, we construct a context-free grammar  $G = (V_N, \Sigma, P, S)$ , where  $V_N = \{q_0, q_1, q_2, q_3\}$ ,  $\Sigma = \{a, b\}$ ,  $S = q_0$ , and  $P = \{q_0 \rightarrow aq_1, q_0 \rightarrow bq_3, q_1 \rightarrow aq_3, q_1 \rightarrow bq_2, q_2 \rightarrow aq_2, q_2 \rightarrow bq_2, q_3 \rightarrow aq_3, q_3 \rightarrow bq_3, q_2 \rightarrow \lambda\}$ .

Notice that the rule  $q_i \rightarrow cq_j \in P$  if and only if  $\delta(q_i, c) = q_j$  in  $M$ , for  $c \in \Sigma$ . Also,  $q_i \rightarrow \lambda \in P$  if and only if  $q_i \in F$  in  $M$ .

From the definition of  $G$ , it follows that  $G$  is a right-linear grammar.

From the definition of the DFA  $M$ , we find that  $abbbba \in L(M)$ , because there exists a directed walk

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{b} q_2 \xrightarrow{b} q_2 \xrightarrow{b} q_2 \xrightarrow{a} q_2$$

with the label  $abbbba$  from the initial state  $q_0$  to the final state  $q_2$ .

We also have the following derivation  $abbbba$ :

$$\begin{aligned} q_0 &\Rightarrow aq_1 && \text{by the rule } q_0 \rightarrow aq_1 \\ &\Rightarrow abq_2 && \text{by the rule } q_1 \rightarrow bq_2 \\ &\Rightarrow abbq_2 && \text{by the rule } q_2 \rightarrow bq_2 \\ &\Rightarrow abbbq_2 && \text{by the rule } q_2 \rightarrow bq_2 \\ &\Rightarrow abbbbq_2 && \text{by the rule } q_2 \rightarrow bq_2 \\ &\Rightarrow abbbbq_2 && \text{by the rule } q_2 \rightarrow aq_2 \\ &\Rightarrow abbbba && \text{by the rule } q_2 \rightarrow \lambda \end{aligned}$$

Hence,  $q_0 \Rightarrow^* abbbba$  and this implies that  $abbbba \in L(G)$ .

Next from the grammar  $G$ , we find that

$$\begin{aligned} q_0 &\Rightarrow aq_1 && \text{by the rule } q_0 \rightarrow aq_1 \\ &\Rightarrow abq_2 && \text{by the rule } q_1 \rightarrow bq_2 \\ &\Rightarrow abbq_2 && \text{by the rule } q_2 \rightarrow bq_2 \\ &\Rightarrow abbb && \text{by the rule } q_2 \rightarrow \lambda \end{aligned}$$

gives a derivation  $q_0 \Rightarrow^* abbb$  in  $G$ . Hence,  $abbb \in L(G)$ . Correspondingly, we have a directed walk

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{b} q_2 \xrightarrow{b} q_2$$

from the initial state to the final state. Hence,  $abbb \in L(M)$ .

In fact, we can show that  $L(M) = L(G)$ . Hence, for the given regular language  $L$  there exists a right linear grammar  $G$  such that  $L(G) = L$ .

In Example 13.3.26, we showed that a regular language  $L$  can be generated by a regular grammar. In general, we have the following theorem.

**Theorem 13.3.27:** Let  $L$  be a regular language. Then there exists a regular grammar which generates  $L$ .

**Proof:** Let  $L$  be a regular language. Then there exists a DFA  $M = (Q, \Sigma, \delta, q_0, F)$  such that  $L(M) = L$ . Suppose that  $Q = \{q_0, q_1, q_2, \dots, q_n\}$  and  $\Sigma = \{a_1, a_2, \dots, a_m\}$ . Construct the right-linear grammar  $G = (V_N, \Sigma, P, S)$  with  $V_N = \{q_0, q_1, q_2, \dots, q_n\}$ ,  $\Sigma = \{a_1, a_2, \dots, a_m\}$ ,  $S = q_0$ , and

$$P = \{q_i \rightarrow a_j q_k \mid \delta(q_i, a_j) = q_k \text{ in } M\} \cup \{q_i \rightarrow \lambda \mid q_i \in F\}.$$

We show that  $L(G) = L$ .

Let  $w \in L = L(M)$ . Suppose  $w = b_1 b_2 \cdots b_k$ ,  $b_i \in \Sigma$ . There exists a directed walk

$$q_0 \xrightarrow{b_1} q_{i_1} \xrightarrow{b_2} q_{i_2} \xrightarrow{b_3} q_{i_3} \longrightarrow \cdots \xrightarrow{b_{k-1}} q_{i_{k-1}} \xrightarrow{b_k} q_{i_k}$$

from the initial state  $q_0$  to a final state  $q_{i_k}$  with the label  $b_1 b_2 \cdots b_k$ . This implies that  $\delta(q_{i_{j-1}}, b_j) = q_{i_j}$ ,  $j = 1, 2, \dots, k$ , assuming  $q_{i_0} = q_0$ . Thus,  $q_{i_{j-1}} \xrightarrow{b_j} q_{i_j} \in P$ ,  $j = 1, 2, \dots, k$ , and also  $q_{i_k} \xrightarrow{\lambda} \lambda \in P$ . Hence, we have the derivation

$$q_0 \Rightarrow b_1 q_{i_1} \Rightarrow b_1 b_2 q_{i_2} \Rightarrow \cdots \Rightarrow b_1 b_2 \cdots b_k q_{i_k} \Rightarrow b_1 b_2 \cdots b_k.$$

This implies that  $b_1 b_2 \cdots b_k \in L(G)$ . It now follows that  $L \subseteq L(G)$ .

Conversely, assume that  $w = b_1 b_2 \cdots b_k \in L(G)$ . Then there exists a derivation

$$S = q_0 \Rightarrow b_1 q_{i_1} \Rightarrow b_1 b_2 q_{i_2} \Rightarrow \cdots \Rightarrow b_1 b_2 \cdots b_k q_{i_k} \Rightarrow b_1 b_2 \cdots b_k$$

in  $G$ . This implies that  $q_{i_{j-1}} \xrightarrow{b_j} q_{i_j} \in P$ ,  $j = 1, 2, \dots, k$ , and also  $q_{i_k} \xrightarrow{\lambda} \lambda \in P$ . Hence, we have a directed walk

$$q_0 \xrightarrow{b_1} q_{i_1} \xrightarrow{b_2} q_{i_2} \xrightarrow{b_3} q_{i_3} \longrightarrow \cdots \xrightarrow{b_{k-1}} q_{i_{k-1}} \xrightarrow{b_k} q_{i_k}$$

from the initial state  $q_0$  to a final state  $q_{i_k}$  with the label  $b_1 b_2 \cdots b_k$ . This implies that  $w = b_1 b_2 \cdots b_k \in L(M) = L$ . Thus,  $L(G) \subseteq L(M)$ . Hence, it follows that  $L(G) = L(M) = L$ . ■

### Corollary 13.3.28: Every regular language is a context-free language.

**Proof:** Let  $L$  be a regular language. Then from Theorem 13.3.27, there exists a right linear grammar  $G$  such that  $L(G) = L$ . Because every right linear grammar is a context-free grammar, it follows that  $L$  is generated by a context-free grammar. Hence,  $L$  is a context-free language. ■

---

**REMARK 13.3.29** ► The converse of Corollary 13.3.28 is not true, because  $L = \{a^n b^n \in \Sigma^* \mid n > 0\}$  is a context-free language on the alphabet  $\Sigma = \{a, b\}$  (see Example 13.3.10), but  $L = \{a^n b^n \in \Sigma^* \mid n > 0\}$  is not a regular language (see Example 13.1.24).

## WORKED-OUT EXERCISES

**Exercise 1:** Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where  $V_N = \{S, A, B\}$ ,  $\Sigma = \{a, b\}$ , and  $P = \{S \rightarrow aB, S \rightarrow bA, A \rightarrow a, A \rightarrow aS, A \rightarrow bAA, B \rightarrow b, B \rightarrow bS, B \rightarrow aBB\}$ .

- (a) For the string  $bbbbaabbbaaaa$  find a derivation.
- (b) For the string  $aabbab$  find a derivation.
- (c) Draw a derivation tree with  $abbaba$  as the yield.

**Solution:**

- (a)  $S \Rightarrow bA$  by the rule  $S \rightarrow bA$   
 $\Rightarrow bbAA$  by the rule  $A \rightarrow bAA$   
 $\Rightarrow bbbAAA$  by the rule  $A \rightarrow bAA$   
 $\Rightarrow bbbASAA$  by the rule  $A \rightarrow aS$   
 $\Rightarrow bbbaaBAA$  by the rule  $S \rightarrow aB$

- $\Rightarrow bbbaabSAA$  by the rule  $B \rightarrow bS$   
 $\Rightarrow bbbaabAAA$  by the rule  $S \rightarrow bA$   
 $\Rightarrow bbbaabbbAAAA$  by the rule  $A \rightarrow bAA$   
 $\Rightarrow bbbaabbbbaAAA$  by the rule  $A \rightarrow a$   
 $\Rightarrow bbbaabbbbaAA$  by the rule  $A \rightarrow a$   
 $\Rightarrow bbbaabbbaaaA$  by the rule  $A \rightarrow a$   
 $\Rightarrow bbbaabbbaaaa$  by the rule  $A \rightarrow b$
- (b)  $S \Rightarrow aB$  by the rule  $S \rightarrow aB$   
 $\Rightarrow aaBB$  by the rule  $B \rightarrow aBB$   
 $\Rightarrow aabSB$  by the rule  $B \rightarrow bS$   
 $\Rightarrow aabbAB$  by the rule  $S \rightarrow bA$   
 $\Rightarrow aabbaB$  by the rule  $A \rightarrow a$   
 $\Rightarrow aabbab$  by the rule  $B \rightarrow b$

- (c) The derivation tree with  $abbaba$  as the yield is shown in Figure 13.35.

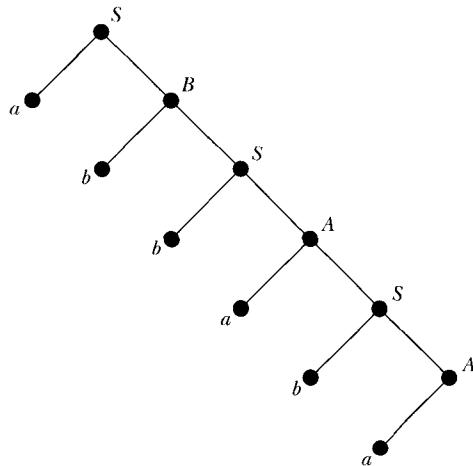


FIGURE 13.35 Derivation tree for  $abbaba$

**Exercise 2:** Let  $\Sigma = \{0, 1\}$ . Find a context-free grammar  $G = (V_N, \Sigma, P, S)$  such that  $S \Rightarrow^* 010010$  and  $S \Rightarrow^* 00010$ .

**Solution:** Let  $G = (V_N, \Sigma, P, S)$ , where  $V_N = \{S, A\}$  and  $P = \{S \rightarrow AA, A \rightarrow AAA, A \rightarrow 0, A \rightarrow 1A, A \rightarrow A1\}$ . This is a context-free grammar. In this grammar, we see that

$$\begin{aligned}
 S &\Rightarrow AA && \text{by the rule } S \rightarrow AA \\
 &\Rightarrow AAAA && \text{by the rule } A \rightarrow AAA \\
 &\Rightarrow 0AAA && \text{by the rule } A \rightarrow 0 \\
 &\Rightarrow 01AAA && \text{by the rule } A \rightarrow 1A \\
 &\Rightarrow 010AA && \text{by the rule } A \rightarrow 0 \\
 &\Rightarrow 010A1A && \text{by the rule } A \rightarrow A1 \\
 &\Rightarrow 01001A && \text{by the rule } A \rightarrow 0 \\
 &\Rightarrow 010010 && \text{by the rule } A \rightarrow 0
 \end{aligned}$$

Hence,  $S \Rightarrow^* 010010$ . Also we see that

$$\begin{aligned}
 S &\Rightarrow AA && \text{by the rule } S \rightarrow AA \\
 &\Rightarrow AAAA && \text{by the rule } A \rightarrow AAA \\
 &\Rightarrow 0AAA && \text{by the rule } A \rightarrow 0 \\
 &\Rightarrow 00AA && \text{by the rule } A \rightarrow 0 \\
 &\Rightarrow 000A && \text{by the rule } A \rightarrow 0 \\
 &\Rightarrow 0001A && \text{by the rule } A \rightarrow 1A \\
 &\Rightarrow 00010 && \text{by the rule } A \rightarrow 0
 \end{aligned}$$

Hence,  $S \Rightarrow^* 00010$ .

**Exercise 3:** Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where

$$\begin{aligned}
 V_N &= \{\langle \text{digit} \rangle, \langle \text{integer} \rangle, \langle \text{signed integer} \rangle, \langle \text{unsigned integer} \rangle\}, \\
 T &= \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -\},
 \end{aligned}$$

$P$  consists of

$$\langle \text{digit} \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$$

$$\langle \text{integer} \rangle ::= \langle \text{signed integer} \rangle \mid \langle \text{unsigned integer} \rangle$$

$$\begin{aligned}
 \langle \text{signed integer} \rangle &::= +\langle \text{unsigned integer} \rangle \mid -\langle \text{unsigned integer} \rangle \\
 \langle \text{unsigned integer} \rangle &::= \langle \text{digit} \rangle \mid \langle \text{digit} \rangle \langle \text{unsigned integer} \rangle
 \end{aligned}$$

and  $S = \langle \text{integer} \rangle$ . Find a derivation that derives the integer 352.

**Solution:** In this grammar, the starting state is  $\langle \text{integer} \rangle$ . Thus,

$$\begin{aligned}
 \langle \text{integer} \rangle & \\
 \Rightarrow \langle \text{unsigned integer} \rangle & \quad \text{by the rule } \langle \text{integer} \rangle ::= \langle \text{unsigned integer} \rangle \\
 \Rightarrow \langle \text{digit} \rangle \langle \text{unsigned integer} \rangle & \quad \text{by the rule } \langle \text{unsigned integer} \rangle ::= \langle \text{digit} \rangle \langle \text{unsigned integer} \rangle \\
 \Rightarrow 3 \langle \text{unsigned integer} \rangle & \quad \text{by the rule } \langle \text{digit} \rangle ::= 3 \\
 \Rightarrow 3 \langle \text{digit} \rangle \langle \text{unsigned integer} \rangle & \quad \text{by the rule } \langle \text{unsigned integer} \rangle ::= \langle \text{digit} \rangle \langle \text{unsigned integer} \rangle \\
 \Rightarrow 35 \langle \text{unsigned integer} \rangle & \quad \text{by the rule } \langle \text{digit} \rangle ::= 5 \\
 \Rightarrow 35 \langle \text{digit} \rangle & \quad \text{by the rule } \langle \text{unsigned integer} \rangle ::= \langle \text{digit} \rangle \\
 \Rightarrow 352 & \quad \text{by the rule } \langle \text{digit} \rangle ::= 2
 \end{aligned}$$

Hence, we have the derivation

$$\begin{aligned}
 \langle \text{integer} \rangle &\Rightarrow \langle \text{unsigned integer} \rangle \Rightarrow \\
 \langle \text{digit} \rangle \langle \text{unsigned integer} \rangle &\Rightarrow 3 \langle \text{unsigned integer} \rangle \Rightarrow \\
 3 \langle \text{digit} \rangle \langle \text{unsigned integer} \rangle &\Rightarrow 35 \langle \text{unsigned integer} \rangle \Rightarrow \\
 35 \langle \text{digit} \rangle &\Rightarrow 352.
 \end{aligned}$$

**Exercise 4:** Show that the language  $L = \{w \in \{a, b\}^* \mid w = w^R\}$  is a context-free language but not a regular language.

**Solution:** The language  $L$  is the set of all palindromes. We have

1.  $\lambda \in L$ ;
2.  $a, b \in L$ ;
3. if  $w \in L$ , then  $awa \in L$  and  $bwb \in L$ ;
4.  $u \in L$  if and only if  $u$  is one of the forms (1), (2), or (3).

Hence, to show that  $L = \{w \in \{a, b\}^* \mid w = w^R\}$  is a CFL we consider the following context-free grammar  $G$ :

$$G = (\{S\}, \{a, b\}, P, S),$$

where  $P$  consists of  $S \rightarrow aSa, S \rightarrow bSb, S \rightarrow a, S \rightarrow b, S \rightarrow \lambda$ .

Let  $w$  be a palindrome. We prove by induction on length  $|w|$  of  $w$  that  $w \in L(G)$ .

**Basis step:** Let  $|w| = 0$ . Then  $w = \lambda$  and the derivation  $S \Rightarrow \lambda$  (by the rule  $S \rightarrow \lambda$ ) shows that  $\lambda \in L(G)$ .

**Inductive hypothesis:** Assume that for any palindrome  $w$  of length less than  $n$ ,  $w \in L(G)$ .

**Inductive step:** Let  $w$  be a palindrome of the length  $n \geq 1$ . If  $|w| = 1$ , then  $w = a$  or  $b$ . Now  $S \Rightarrow a$  (by the rule  $S \rightarrow a$ ) or  $S \Rightarrow b$  (by the rule  $S \rightarrow b$ ). Hence,  $a, b \in L(G)$ .

Let  $|w| \geq 2$ . Then  $w = aua$  or  $bub$  for some palindrome  $u$ . Because  $|u| < n$ , by the inductive hypothesis,  $u \in L(G)$ . Hence, there exists a derivation  $S \Rightarrow^* u$ .

Suppose  $w = aua$ . Then

$$\begin{aligned} S &\Rightarrow aSa \quad \text{by the rule } S \rightarrow aSa \\ &\Rightarrow^* aua \quad \text{by the derivation } S \Rightarrow^* u \end{aligned}$$

If  $w = bub$ , then

$$\begin{aligned} S &\Rightarrow bSb \quad \text{by the rule } S \rightarrow bSb \\ &\Rightarrow^* bub \quad \text{by the derivation } S \Rightarrow^* u \end{aligned}$$

Thus,  $w \in L(G)$ . Hence, by induction  $L \subseteq L(G)$ .

Conversely, let  $w \in L(G)$ . Then there exists a derivation  $S \Rightarrow^* w$  in  $G$  of length, say  $n$ . Because  $S \rightarrow aSa$ ,  $S \rightarrow bSb$ ,  $S \rightarrow a$ ,  $S \rightarrow b$ , and  $S \rightarrow \lambda$  are the only rules in  $G$ , by applying induction on the length of the derivation  $S \Rightarrow^* w$  we can show that  $w$  is a palindrome. Thus, it follows that  $L = L(G)$ . Hence,  $L$  is a context-free language. In Worked-Out Exercise 6, page 846, we proved that  $L = \{w \in \{a, b\}^* \mid w = w^R\}$  is not a regular language.

**Exercise 5:** Find a right-linear grammar, if it exists, for the language  $\{a^n b \in \{a, b\}^* \mid n \text{ is a positive integer}\}$ .

**Solution:** Let  $L = \{a^n b \in \{a, b\}^* \mid n \text{ is a positive integer}\}$ . Construct a context-free grammar  $G = (V_N, \Sigma, P, S)$  with  $V_N = \{S, A\}$ ,  $\Sigma = \{a, b\}$ ,  $P = \{S \rightarrow aS, S \rightarrow aA, A \rightarrow b\}$ . Clearly this is a right-linear grammar.

We find that any derivation from  $S$  in  $G$  is of the form

$$\begin{aligned} S &\Rightarrow^* a^k S \quad \text{by repeated application of } S \rightarrow aS \\ &\Rightarrow a^{k+1} A \quad \text{by } S \rightarrow aA \\ &\Rightarrow a^{k+1} b \quad \text{by } A \rightarrow b, k \geq 1 \end{aligned}$$

or of the form

$$\begin{aligned} S &\Rightarrow aA \quad \text{by } S \rightarrow aA \\ &\Rightarrow ab \quad \text{by } A \rightarrow b \end{aligned}$$

Thus,  $L(G) \subseteq L$ . We can also show that  $L \subseteq L(G)$ . Hence,  $L = L(G)$ .

## SECTION REVIEW

### Key Terms

context-free grammar	null (lambda) rule	language generated
nonterminal symbols	directly derivable	context-free language (CFL)
terminal symbols	derivable	derivation tree
start symbol	derivation	yield of the derivation tree
rules of a grammar	length	right-linear grammar
rules	sentential	left-linear grammar
productions	sentence	regular grammar
$A$ -rule	language	

### Some Key Definitions

1. A context-free grammar  $G$  is a quadruple  $(V_N, \Sigma, P, S)$ , where
  - (i)  $V_N$  is a finite set, called the set of nonterminal symbols;
  - (ii)  $\Sigma$  is a finite set, called the set of terminal symbols,  $V_N \cap \Sigma = \emptyset$ ;
  - (iii)  $P$  is a finite subset of  $V_N \times (V_N \cup \Sigma)^*$ ; and
  - (iv)  $S$  is a distinguished element of  $V_N$ , called the start symbol.
2. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar. Suppose  $xAy$  and  $xwy$  are two strings in  $(V_N \cup \Sigma)^*$ .
  - (i)  $xwy$  is said to be directly derivable from  $xAy$ , written  $xAy \implies xwy$ , if  $A \rightarrow w$  is a rule in  $G$ .
  - (ii) If  $\alpha_1, \alpha_2, \dots, \alpha_{n+1}$  are strings in  $(V_N \cup \Sigma)^*$  such that  $\alpha_1 \Rightarrow \alpha_2 \Rightarrow \dots \Rightarrow \alpha_n \Rightarrow \alpha_{n+1}$ , then we say that  $\alpha_{n+1}$  is derivable from  $\alpha_1$  in  $G$  and  $\alpha_1 \Rightarrow \alpha_2 \Rightarrow \dots \Rightarrow \alpha_n \Rightarrow \alpha_{n+1}$  is a derivation of  $\alpha_{n+1}$  from  $\alpha_1$  of length  $n$ .

- (iii) Let  $\alpha_1, \alpha_2, \dots, \alpha_{n+1}$  be strings in  $(V_N \cup \Sigma)^*$ . If  $\alpha_{n+1}$  is derivable from  $\alpha_1$  by a derivation of length  $n$ , then we write  $\alpha_1 \Rightarrow^n \alpha_{n+1}$ .
3. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar.
    - (i) A string  $\alpha \in (V_N \cup \Sigma)^*$  is said to be in sentential form of  $G$  if there is a derivation  $S \Rightarrow^* \alpha$  in  $G$ .
    - (ii) A string  $w \in \Sigma^*$  is said to be a sentence of  $G$  if there is a derivation  $S \Rightarrow^* w$  in  $G$ .
    - (iii) The set of all sentences of  $G$  is called the language of  $G$  or language generated by  $G$ .
  4. Let  $\Sigma$  be an alphabet. A language  $L$  on  $\Sigma$  is called a context-free language (CFL) if there exists a context-free grammar  $G = (V_N, \Sigma, P, S)$  such that the language generated by  $G$  is  $L$ , i.e.,  $L(G) = L$ .
  5. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar. A rooted tree  $T$  is called a derivation tree in  $G$  if
    - (i) every vertex has a label, which is a symbol of  $V_N \cup \Sigma \cup \{\lambda\}$  (two distinct vertices may have the same label),
    - (ii) the label of the root of the tree is  $S$ ,
    - (iii) if a vertex has label  $A$  and the children (in order from left to right) of this vertex have labels  $x_1, x_2, \dots, x_n$ , respectively, then  $A \rightarrow x_1 x_2 \dots x_n$  is a rule of  $G$ ,
    - (iv) if a vertex has the label  $\lambda$ , then this vertex is a leaf and it is the only child of its parent.
  6. A context-free grammar  $G = (V_N, \Sigma, P, S)$  is called a right-linear grammar if each of the rules of  $G = (V_N, \Sigma, P, S)$  is of the form
    - (i)  $A \rightarrow w$ , and
    - (ii)  $A \rightarrow wB$ , where  $A, B \in V_N$  and  $w \in \Sigma^*$ .
  7. A context-free grammar  $G = (V_N, \Sigma, P, S)$  is called a left-linear grammar if each of the rules of  $G = (V_N, \Sigma, P, S)$  is of the form
    - (i)  $A \rightarrow w$ , and
    - (ii)  $A \rightarrow Bw$ , where  $A, B \in V_N$  and  $w \in \Sigma^*$ .
  8. A right-linear or left-linear grammar is called a regular grammar.

## Some Key Results

1. Let  $L$  be a context-free language. There exist positive integers  $n$  depending on  $L$ , such that if there is a string  $w$  in  $L$  with  $|w| \geq n$ , then  $w$  can be written as  $w = xyzuv$  such that  $|yu| \geq 1$ ,  $|yzu| \leq n$ , and for each integer  $i \geq 0$ ,  $xy^izu^iv \in L$ .
2. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar. Then  $S \implies^* \alpha$  in  $G$  if and only if there is a derivation tree of  $G$  with yield  $\alpha$ .
3. Let  $L$  be a regular language. Then there exists a regular grammar which generates  $L$ .
4. Every regular language is a context-free language.

## EXERCISES

1. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where  $V_N = \{S\}$ ,  $\Sigma = \{0, 1\}$ ,  $P = \{S \rightarrow 0S1, S \rightarrow 01\}$ .
    - a. Write a derivation of  $0^31^3$  from  $S$  in  $G$ .
    - b. Draw a derivation tree of the derivation of (a).
    - c. Write a derivation of  $0^4S1^4$ .
  2. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where  $V_N = \{S, A\}$ ,  $\Sigma = \{0, 1\}$ ,  $P = \{S \rightarrow 0S, S \rightarrow 1A, A \rightarrow 1\}$ .
    - a. Write a derivation of  $0^51^2$  from  $S$  in  $G$ .
    - b. Draw a derivation tree of the derivation of (a).
    - c. Find the language  $L(G)$  generated by this grammar.
  3. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where  $V_N = \{S, A\}$ ,  $\Sigma = \{a, b\}$ ,  $P = \{S \rightarrow Ab, A \rightarrow aAb, A \rightarrow \lambda\}$ .
    - a. Write a derivation of  $aaabbbb$  from  $S$  in  $G$ .
    - b. Draw a derivation tree of the derivation of  $G$  with yield  $aaaabbbb$ .
    - c. Find the language  $L(G)$  generated by this grammar.
  4. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where  $V_N = \{S, A\}$ ,  $\Sigma = \{a, b\}$ ,  $P = \{S \rightarrow bA, A \rightarrow aS, S \rightarrow \lambda\}$ .
    - a. Write a derivation of  $bababa$  from  $S$  in  $G$ .
    - b. Draw a derivation tree of the derivation of  $G$  with yield  $babababa$ .
    - c. Find the language  $L(G)$  generated by this grammar.
  5. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where  $V_N = \{S\}$ ,  $\Sigma = \{a\}$ ,  $P = \{S \rightarrow bA, A \rightarrow a\}$ . Find  $L(G)$ .
  6. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where  $V_N = \{S, A\}$ ,  $\Sigma = \{a, b\}$ ,  $P = \{S \rightarrow aAa, A \rightarrow aAa, A \rightarrow b\}$ .
    - a. Write a derivation of  $aaabaaa$  from  $S$  in  $G$ .
    - b. Find the language  $L(G)$  generated by this grammar.
  7. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where  $V_N = \{\langle digit \rangle, \langle integer \rangle, \langle signed integer \rangle, \langle unsigned integer \rangle\}$ ,  $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -\}$ ,
- and  $P$  consists of
- $$\langle digit \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$$
- $$\langle integer \rangle ::= \langle signed integer \rangle \mid \langle unsigned integer \rangle$$
- $$\langle signed integer \rangle ::= + \langle unsigned integer \rangle \mid - \langle unsigned integer \rangle$$
- $$\langle unsigned integer \rangle ::= \langle digit \rangle \mid \langle digit \rangle \langle unsigned integer \rangle$$
- and  $S = \langle integer \rangle$ . Find a derivation that derives the integer  $-502$ .
8. Let  $G = (V_N, \Sigma, P, S)$  be a context-free grammar, where  $V_N = \{S, A, B\}$ ,  $\Sigma = \{a, b, c, +, *, (), ()\}$ ,  $P = \{S \rightarrow S + A, S \rightarrow A, A \rightarrow A * B, A \rightarrow B, B \rightarrow (S), B \rightarrow a, B \rightarrow b, B \rightarrow c\}$ . Write a derivation of  $a * (b + (a * c))$  from  $S$  in  $G$ .
  9. Let  $\Sigma = \{a, b\}$ . Is  $L = \{a, ab, b\}$  a CFL on  $\Sigma = \{a, b\}$ ? Justify your answer.
  10. Let  $\Sigma = \{0, 1\}$  and  $L = \{w \in \Sigma^* \mid \text{number of occurrences of } 0 \text{ is an even positive integer}\}$ . Show that  $L$  is a CFL.
  11. Let  $\Sigma = \{a, b\}$ . Is  $L = \{a^n \mid n \text{ is a nonnegative integer}\}$  a CFL on  $\Sigma = \{a, b\}$ ? Justify your answer.
  12. Let  $\Sigma = \{a, b\}$  and  $L = \{a^n b \mid n \text{ is a nonnegative integer}\}$ . Show that there exists a regular grammar  $G$  such that  $L(G) = L$ .
  13. Let  $\Sigma = \{a, b\}$  and  $L = \{a^n b^m \mid n > 0, m \geq 0\}$ . Show that there exists a regular grammar  $G$  such that  $L(G) = L$ .
  14. Let  $\Sigma = \{a, b, c\}$  and let  $L = \{a^n b c a^n \mid n \text{ is a nonnegative integer}\}$ . Show that  $L$  is a CFL on  $\Sigma = \{a, b\}$ .
  15. Let  $\Sigma = \{0, 1\}$  and let  $L = \{01^n 0 \mid n \text{ is a nonnegative integer}\}$ . Show that  $L$  is a CFL on  $\Sigma = \{0, 1\}$ .
  16. Prove that the language  $L = \{a^n b^m a^n \mid n > 0, m \geq 0\}$  on  $\Sigma = \{a, b\}$  is a CFL.
  17. Show that the language  $L = \{a^n b^m a^n b^m \mid n \geq 0, m \geq 0\}$  on  $\Sigma = \{a, b\}$  is not a CFL.
  18. Show that the language  $L = \{a^p \mid p \text{ is a prime integer}\}$  is not a CFL over  $\Sigma = \{a, b\}$ .
  19. Consider the language  $L = \{0^{i^2} \mid i \text{ is a positive integer}\}$  over the alphabet  $\{0, 1\}$ . Show that  $L$  is not a context-free language.
  20. Let  $\Sigma = \{0, 1\}$ .
    - a. Let  $L = \{0^n 1^m \mid n \geq 1, m \geq 3\}$ . Show that  $L$  is a CFL on  $\Sigma$ .
    - b. Let  $L = \{0^{2n+1} \mid n \geq 0\} \cup \{0^{2n} 1 \mid n \geq 0\}$ . Show that  $L$  is a CFL on  $\Sigma$ .
  21. Find a right-linear grammar, if it exists, for the language  $\{a^n \in \{a, b\}^* \mid n \text{ is a positive integer}\}$ .
  22. Show that there exists a regular grammar  $G$  on  $\Sigma = \{a, b\}$  such that  $L(G) = \{a^n b a^m \in \Sigma^* \mid n \geq 0, m \geq 1\}$ .
  23. Find a regular grammar that generates the language accepted by the DFA shown in Figure 13.36.

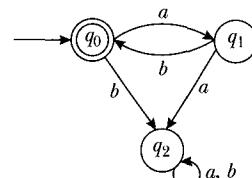


FIGURE 13.36 A DFA

24. Prove that the languages  $L_1 = \{a^n b^n c^m \mid n \geq 0, m \geq 0\}$  and  $L_2 = \{a^n b^m c^m \mid n \geq 0, m \geq 0\}$  on  $\Sigma = \{a, b, c\}$  are context-free languages. Show that  $L = L_1 \cap L_2$  is not a context-free language.
25. Prove that the union of two context-free languages is a context-free language.

 **PROGRAMMING EXERCISES**

- 
1. Write a program that implements the DFA of Worked-Out Exercise 1, page 845. Test your program to determine whether a string is accepted by the DFA.
  2. Write a program that takes as input a DFA and then determine whether a string is accepted by the DFA.

## EXONENTS AND LOGARITHMS

In this section, we review basic properties of exponents and logarithms.

### Exponents

Let  $a \in \mathbb{R}$ , and let  $m$  and  $n$  be integers. Then

$$a^m = \begin{cases} 1, & \text{if } m = 0, \\ a \cdot a^{m-1}, & \text{if } m > 0, \\ \frac{1}{a^{-m}}, & \text{if } m < 0. \end{cases}$$

If  $a = 0$ , then we assume that  $m \neq 0$ .

**Theorem A.0.1:** Let  $a, b \in \mathbb{R}$ , and let  $m$  and  $n$  be integers. Then

- (i)  $a^m a^n = a^{m+n}$ ,
- (ii)  $(a^m)^n = a^{mn}$ ,
- (iii)  $(ab)^m = a^m b^m$ ,
- (iv)  $\left(\frac{a}{b}\right)^m = \frac{a^m}{b^m}$  is  $b \neq 0$ .

**REMARK A.0.2** ▶ Let  $a$  be a fixed real number and  $x$  be any real number. Using the techniques from calculus, we can also define  $a^x$ . Moreover, the previous theorem holds if  $m$  and  $n$  are real numbers. Of course, if  $a = 0$ , then  $m$  and  $n$  are nonzero real numbers.

### Logarithms

Let  $a > 1$ ,  $y > 0$ , and  $x > 0$  be real numbers such that

$$a^x = y.$$

Then we define

$$\log_a y = x$$

and say that logarithm  $y$  to the base  $a$  is  $x$  or simply  $\log y$  to the base  $a$  is  $x$ .

#### EXAMPLE A.0.3

- (i) Because  $2^4 = 16$ ,  $\log_2 16 = 4$ .
- (ii) Because  $3^5 = 243$ ,  $\log_3 243 = 5$ .
- (iii) Because  $4^{-2} = \frac{1}{16}$ ,  $\log_4 \left(\frac{1}{16}\right) = -2$ .

**Theorem A.0.4:** Let  $a > 1$ ,  $y > 0$ , and  $x > 0$  be real numbers. The function  $\log$  is increasing. That is, if  $x < y$ , then  $\log_a x < \log_a y$ .

**Theorem A.0.5:** Let  $a > 1, b > 1$  and  $x > 0, y > 0$ . Then

- (i)  $\log_a 1 = 0$ ,
- (ii)  $\log_a a = 1$ ,
- (iii)  $\log_a xy = \log_a x + \log_a y$ ,
- (iv)  $\log_a \frac{x}{y} = \log_a x - \log_a y$ ,
- (v)  $\log_a x^y = y \log_a x$ ,
- (vi)  $a^{\log_a x} = x$ ,
- (vii)  $x^{\log_a y} = y^{\log_a x}$ ,
- (viii) (Base Change formula)  $\log_a x = \frac{\log_b x}{\log_b a}$ .

### EXAMPLE A.0.6

- (i)  $\log_5 1 = 0$ .
- (ii)  $\log_3(9 \cdot 11) = \log_3 9 + \log_3 11 = \log_3 3^2 + \log_3 11 = 2 \cdot \log_3 3 + \log_3 11 = 2 + \log_3 11$ .
- (iii)  $\log_5 \frac{125}{25} = \log_5 125 - \log_5 25 = \log_5 5^3 - \log_5 5^2 = 3 - 2 = 1$ .

In algorithm analysis, we often encounter logarithm to the base 2. We usually write  $\log_2 x$  as  $\lg x$ . Logarithms to the base 10 are known as **common logarithms**, and we usually write  $\log_{10} x$  as  $\log x$ . Let  $e = 2.718281828459\dots$ . The real number  $e$  is known as an Euler number and is an irrational number. Logarithms to the base  $e$  are known as **natural logarithms**, and we usually write  $\log_e x$  as  $\ln x$ . Every scientific calculator contains the common logarithm and natural logarithm functions.

## POLYNOMIALS

Let  $p(x)$  be a polynomial with coefficients as real numbers. Let  $r$  be a root of  $p(x)$  and let  $m$  be a positive integer such that  $(x - r)^m$  divides  $p(x)$  but  $(x - r)^{m+1}$  does not divide  $p(x)$ . Then  $m$  is called the **multiplicity** of  $r$ .

### EXAMPLE A.0.7

- (i) Let  $p(x) = x - 3$ . Then 3 is a root of  $p(x)$  and it is a root of multiplicity 1.
- (ii) Let  $p(x) = x^2 - 2x - 8$ . Then  $p(x) = (x - 2)(x + 4)$ . Thus, 2 and -4 are roots of  $p(x)$  and both of these roots are of multiplicity 1.
- (iii) Let  $p(x) = x^2 + 6x + 9$ . Then  $p(x) = (x + 3)^2$ . It follows that -3 is a root of  $p(x)$  and it is a root of multiplicity 2.
- (iv) Let  $p(x) = (x - 1)^4(2x - 3)^5(3x + 1)$ . Then the roots of  $p(x)$  are 1,  $\frac{3}{2}$ , and  $-\frac{1}{3}$ . Notice that 1 is a root of multiplicity 4,  $\frac{3}{2}$  is a root of multiplicity 5, and  $-\frac{1}{3}$  is a root of multiplicity 1.

## COMPLEX NUMBERS

In this text, there are a few places where we have used complex numbers to give some examples. Therefore, in this section, we review some basic properties of complex numbers. Let  $\mathbb{C}$  denote the set of complex numbers. Then

$$\mathbb{C} = \{a + ib \mid a, b \in \mathbb{R}\},$$

where  $i^2 = -1$  or  $i = \sqrt{-1}$ . Sometimes we write  $a + ib$  as  $a + bi$ .

Notice that  $i = \sqrt{-1}$ ,  $i^2 = -1$ ,  $i^3 = -i$ , and  $i^4 = 1$ .

Let  $a + ib$ ,  $c + id \in \mathbb{C}$ . The addition, subtraction, and multiplication of complex numbers is defined by

$$(a + ib) + (c + id) = (a + c) + i(b + d),$$

$$(a + ib) - (c + id) = (a - c) + i(b - d),$$

and

$$(a + ib)(c + id) = (ac - bd) + i(ad + bc).$$

$$|a + ib| = \sqrt{a^2 + b^2}.$$

Also  $a + ib$  and  $a - ib$  are called *conjugates* of each other.

**REMARK A.0.8** ▶ Let  $a \in \mathbb{R}$ . Then  $a = a + 0i \in \mathbb{C}$ . It follows that  $\mathbb{R} \subseteq \mathbb{C}$ .

### EXAMPLE A.0.9

- (i)  $(2 + 3i) + (4 + 5i) = 6 + 8i$
- (ii)  $(3 - 2i) + (-7 + 6i) = (3 - 7) + (-2 + 6)i = -4 + 4i$
- (iii)  $(6 + 7i) - (3 + 5i) = 3 + 2i$
- (iv)  $(-3 + 4i) - (2 - 3i) = -3 + 4i - 2 + 3i = -5 + 7i$
- (v)  $(2 + 3i)(4 + 2i) = (8 - 6) + (4 + 12)i = 2 + 16i$
- (vi)  $4(3 - 4i) = 12 - 16i$ . Note that  $4 = 4 + 0i$
- (vii) The conjugate of  $4 + 3i$  is  $4 - 3i$ . The conjugate of  $-2 - 3i$  is  $-2 + 3i$ .

### EXAMPLE A.0.10

- (i) Consider the equation  $x^2 + 1 = 0$ . This equation has no real roots. However,  $x^2 + 1 = (x + i)(x - i)$ . Thus,  $i$  and  $-i$  are the roots of this equation. That is,  $x^2 + 1 = 0$  has two complex roots.
- (ii) Consider the equation  $x^4 - 1 = 0$ . Now  $x^4 - 1 = (x^2 - 1)(x^2 + 1) = (x + 1)(x - 1)(x + i)(x - i) = 0$ . Thus, the roots of  $x^4 - 1 = 0$  are  $1$ ,  $-1$ ,  $i$ , and  $-i$ . It follows that the equation  $x^4 - 1 = 0$  has only two roots that are real numbers.

## Answers and Hints to Selected Exercises

Student Solutions Manual containing answers to all odd-numbered exercises is available at the Web site accompanying this book. Use your password to access the Solutions Manual.

### CHAPTER 1: FOUNDATIONS: SETS, LOGIC, AND ALGORITHMS

#### 1.1 Sets

- |             |           |           |
|-------------|-----------|-----------|
| 1. (a) True | (b) True  | (c) False |
| (d) False   | (e) True  | (f) True  |
| (g) True    | (h) False | (i) True  |
| (j) True    | (k) True  | (l) True  |
| (m) True    | (n) True  |           |

3. (a)  $(A \cup B)' = \{c\}$       (b)  $A \cap B = \{e\}$   
       (c)  $A - B = \{a, d, f\}$       (d)  $B - A = \{b, g\}$

5. (a)  $P \cup R = \{x \in \mathbb{N} \mid 1 \leq x \leq 10\}$   
       (b)  $Q \cap R = \{x \in \mathbb{N} \mid 1 \leq x \leq 4\}$   
       (c)  $P \Delta R = \{x \in \mathbb{N} \mid 1 \leq x \leq 2 \text{ or } 8 < x \leq 10\}$   
       (d)  $Q' = \{x \in \mathbb{Z} \mid -2 \leq x < 0 \text{ or } 5 \leq x < 12\}$

7.  $A \cup B = \{x \in \mathbb{R} \mid 1 < x \leq 8\};$   
 $A \cap B = \{x \in \mathbb{R} \mid 1 < x \leq 5 \text{ and } 3 \leq x \leq 8\}$   
 $= \{x \in \mathbb{R} \mid 3 \leq x \leq 5\};$   
 $A - B = \{x \in \mathbb{R} \mid x < 1 \text{ or } x > 5\};$   
 $B - A = \{x \in \mathbb{R} \mid 5 < x \leq 8\}.$

9. Yes. Let  $A = \{1\}$ . Then  $\emptyset$  is a proper subset of  $A$ .  
 11.  $\{\emptyset, \{\emptyset\}, \{\{\emptyset\}\}, \{\{\emptyset, \{\emptyset\}\}\}\}.$

21. (b)

$$\begin{aligned} A - (B \cap C) &= A \cap (B \cap C)' \\ &= A \cap (B' \cup C') \\ &= (A \cap B') \cup (A \cap C') \quad \text{by distributivity} \\ &= (A - B) \cup (A - C). \end{aligned}$$

(c)

$$\begin{aligned} A - (B \cup C) &= A \cap (B \cup C)' \\ &= A \cap (B' \cap C') \\ &= (A \cap A) \cap (B' \cap C') \quad \text{because } A = A \cap A \\ &= (A \cap B') \cap (A \cap C') \\ &= (A - B) \cap (A - C). \end{aligned}$$

23.  $B - C = \{b\}$ . Therefore,  $A \times (B - C) = \{(a, b), (d, b), (e, b), (f, b)\}$ . Also  $A \times B = \{(a, b), (d, b), (e, b), (f, b), (a, e), (d, e), (e, e), (f, e), (a, g), (d, g), (e, g), (f, g)\}$  and  $A \times C = \{(a, a), (d, a), (e, a), (f, a), (a, c), (d, c), (e, c), (f, c), (a, e), (d, e), (e, e), (f, e), (a, g), (d, g), (e, g), (f, g)\}$ . Hence,  $(A \times B) - (A \times C) = \{(a, b), (d, b), (e, b), (f, b)\}$ . Consequently,  $A \times (B - C) = (A \times B) - (A \times C)$ .

29. (a) 31 (b) 3 (c) 194 (d) 185 (e) 379

31. 300 people appeared in the examination.

35. (a)  $s_A = 0101 0101 0101 0101 0101 0101 0101 01$   
       (b)  $s_{A'} = 1010 1010 1010 1010 1010 1010 1010 10$   
       (c)  $s_B = 0010 0100 1001 0010 0100 1001 0010 01$   
       (d)  $0111 0101 1101 0111 0101 1101 0111 01$   
       (e)  $0000 0100 0001 0000 0100 0001 0000 01$   
 37. (a) False (b) True (c) False (d) False

#### 1.2 Mathematical Logic

1. (a), (b), (c), (d), and (e), are statements.  
 3. (a) 13 is not an even integer.  
       (b)  $5 + 8 \not< 18$ . This can also be written as  $5 + 8 \geq 18$ .  
       (c) The flower is not beautiful.

$p$	$q$	$\sim p$	$\sim p \vee q$	$(\sim p \vee q) \wedge p$
T	T	F	T	T
T	F	F	F	F
F	T	T	T	F
F	F	T	T	F

$p$	$q$	$\sim p$	$p \rightarrow q$	$\sim p \rightarrow (p \rightarrow q)$
T	T	F	T	T
T	F	F	F	T
F	T	T	T	T
F	F	T	T	T

Hence,  $\sim p \rightarrow (p \rightarrow q)$  is a tautology.

15.  $(p \vee q) \rightarrow (p \wedge q)$  is not a tautology.  
 19.  $(p \wedge q) \wedge \sim q$

23. (a) We construct the truth table for  $A \rightarrow B$ .

$p$	$q$	$p \rightarrow q$	$\sim(p \rightarrow q)$	$\sim q$	$p \wedge (\sim q)$	$A \rightarrow B$
T	T	T	F	F	F	T
T	F	F	T	T	T	T
F	T	T	F	F	F	T
F	F	T	F	T	F	T

Hence,  $A$  logically implies  $B$ .

33. The given expressions are not logically equivalent.

## 1.3 Validity of Arguments

1. We construct the truth table for the statement formula  $A = (p \rightarrow q) \wedge (p \rightarrow r) \rightarrow (p \rightarrow (q \vee r))$ .

$p$	$q$	$r$	$p \rightarrow q$	$p \rightarrow r$	$(p \rightarrow q) \wedge (p \rightarrow r)$	$q \vee r$	$p \rightarrow (q \vee r)$	$A$
T	T	T	T	T	T	T	T	T
T	T	F	T	F	F	T	T	T
T	F	T	F	T	F	T	T	T
T	F	F	F	F	F	F	F	T
F	T	T	T	T	T	T	T	T
F	T	F	T	T	T	T	T	T
F	F	T	T	T	T	T	T	T
F	F	F	T	T	F	T	T	T

From the truth table we find that  $A$  is a tautology. Hence, the given argument is valid.

5. The truth table for the statement formula  $((p \rightarrow q) \wedge \sim p) \rightarrow \sim q$  is the following.

$p$	$q$	$\sim p$	$p \rightarrow q$	$(p \rightarrow q) \wedge \sim p$	$\sim q$	$(p \rightarrow q) \wedge \sim p \rightarrow \sim q$
T	T	F	T	F	F	T
T	F	F	F	F	T	T
F	T	T	T	T	F	F
F	F	T	T	T	T	T

Because  $((p \rightarrow q) \wedge \sim p) \rightarrow \sim q$  is not a tautology, it follows that the given argument is not a valid argument.

15. To check the validity of the argument, we symbolize it using statement letters. Let

- $p$  : I do all exercises in this chapter.
- $q$  : I understand the material.
- $r$  : I pass the examination.
- $s$  : I do well on the exam.

Then the whole argument may be symbolized as

$$\begin{aligned} & p \rightarrow q \\ & q \rightarrow s \\ & s \rightarrow r \\ & r \\ \therefore & p \end{aligned}$$

There is an assignment  $T$  for  $r$ ,  $F$  for  $s$ ,  $F$  for  $q$ , and  $F$  for  $p$  such that each of  $p \rightarrow q$ ,  $q \rightarrow s$ ,  $s \rightarrow r$ , and  $r$  is  $T$  but  $p$  is  $F$ . Hence, the above argument is not valid.

## 1.4 Quantifiers and First-Order Logic

1. (a) Let  $P(x) : x$  is an integer.  $Q(x) : x$  is a rational number. Then in symbols, the given sentence takes the form  $\forall x(P(x) \rightarrow Q(x))$ .
- (b) Let  $P(x) : x$  is an integer. Then in symbols, the given sentence takes the form  $\exists x P(x)$ , the domain of discourse is the set of rational numbers.

- (c) Let  $P(x) : x$  is a positive integer,  $Q(x) : x$  is multiple of 5. Then in symbols, the given sentence takes the form  $\forall x(P(x) \rightarrow Q(x))$ .

- (d) Let  $P(x) : x$  is a rectangle,  $Q(x) : x$  is a square. Then in symbols, the given sentence takes the form  $\exists x(P(x) \rightarrow Q(x))$ , the domain of discourse is the set of all rectangles.

- (e) Let  $P(n) : n$  is an integer.  $Q(n) : n$  is an odd integer. Then in symbols, the given sentence takes the form  $\forall n(P(n) \rightarrow Q(2n+1))$ .

- (f) Let  $P(n) : n$  is an integer.  $E(n) : n$  is an even integer.  $Q(n) : n$  is an odd integer. Then in symbols, the given sentence takes the form  $\forall n(P(n) \rightarrow E(n) \vee Q(n))$ .

- (g) Let  $P(x) : x$  is a multiple of 6,  $Q(x) : x$  is multiple of 3,  $R(x) : x$  is a multiple of 2. Then in symbols, the given sentence takes the form  $\forall x(P(x) \leftrightarrow Q(x) \wedge R(x))$ , the domain of discourse is the set of all integers.

- (h) Let  $P(n) : n$  is an integer.  $Q(n) : n^2$  is 5. Then in symbols, the given sentence takes the form  $\forall n(P(n) \rightarrow \sim Q(n))$ .

3. Let  $P(x)$  be the predicate given by  $P(x) : x^2 + x$  is an even integer and the domain is the set of all odd integers. The universal quantification of  $P(x)$  is  $\forall x P(x)$ . The universal quantification  $\forall x P(x)$  is a true statement.

9. (a) Because  $x + (x - 7) = 7$ , the truth value of  $\forall x \exists y P(x, y)$  is  $T$ .  
(b) If we take  $x = 5$  and  $y = 3$ , then  $5 + 3 \neq 7$ . Hence, the truth value of  $\forall x \forall y P(x, y)$  is  $F$ .  
(c) If we take  $x = 5$  and  $y = 2$ , then  $5 + 2 = 7$ . Hence, the truth value of  $\exists x \exists y P(x, y)$  is  $T$ .

11. (a) true      (b) false      (c) true

19. Let  $P(x) : x$  is a employer and  $Q(x) : x$  pays to their employees. Then in symbols, the given argument takes the form

$$\forall x(P(x) \rightarrow Q(x)).$$

$$P(\text{Juan})$$

$$\therefore Q(\text{Juan})$$

To verify the validity we now consider the following sequence of formulas:

$$\begin{array}{ll} B_1 : \forall x(P(x) \rightarrow Q(x)) & \text{hypothesis} \\ B_2 : P(\text{Juan}) \rightarrow Q(\text{Juan}) & \text{by the rule of inference US} \\ B_3 : P(\text{Juan}) & \text{hypothesis} \\ B_4 : Q(\text{Juan}) & \text{by modus ponens} \end{array}$$

Hence, the given argument is a valid argument.

27. (a) If we take  $x = 1$ , then  $1 \not< 1^2$ . This is a counterexample. Hence, the given proposition is false.

## 1.5 Proof Techniques

1. (a) Let  $P(n) : n$  is an even integer and  $Q(n) : n^2$  is an even integer. Then  $\forall n(P(n) \rightarrow Q(n))$ . Let  $n$  be an even integer. This implies that  $n = 2k$  for some integer  $k$ . Now  $n^2 = (2k)^2 = 4k^2 = 2(2k^2) = 2t$ ,

where  $t = 2k^2$ . Because  $k$  is an integer,  $t$  is an integer. It now follows that  $n^2$  is an even integer. Therefore,  $Q(n)$ . We have thus shown that  $P(n) \rightarrow Q(n)$ . Hence, by the method of direct proof,  $\forall n (P(n) \rightarrow Q(n))$ .

- (c) Let  $P(n) : n$  is an integer and  $n$  is odd;  $Q(n, m) : n$  and  $m$  are integers and  $n + m$  is even. Then  $\forall n \forall m (P(n) \wedge P(m) \rightarrow Q(n, m))$ .

Let  $n$  and  $m$  be odd integers. Then  $n = 2r + 1$  and  $m = 2s + 1$  for some integers  $r$  and  $s$ . Now  $n + m = 2r + 1 + 2s + 1 = 2r + 2s + 2 = 2(r + s + 1) = 2k$ , where  $k = r + s + 1$ . Because,  $r, s$ , and 1 are integers,  $k$  is an even integer. It now follows that  $n + m$  is an even integer.

3. Suppose  $x$  and  $y$  are even integers. Then  $x = 2m$  and  $y = 2n$  for some integers  $m$  and  $n$ . This implies that  $x + y = 2m + 2n = 2(m + n)$ . Let  $t = m + n$ . Then because  $m$  and  $n$  are integers,  $t$  is an integer. Hence,  $x + y = 2t$  is an even integer.  
 7. Suppose  $x$  and  $y$  are even integers. Then  $x = 2m$  and  $y = 2n$  for some integers  $m$  and  $n$ . This implies that  $xy = 2m2n = 4mn = 2(2mn) = 2t$ , where  $t = 2mn$ . Because  $m$  and  $n$  are integers,  $t$  is an integer. Hence,  $xy$  is an even integer.  
 17. Let  $x = \sqrt{2}$  and  $y = \sqrt{2}$ . Then  $x$  and  $y$  are irrational numbers. However,  $xy = (\sqrt{2})^2 = 2$ , so  $xy$  is a rational number. Hence, the statement is false.  
 29. Because  $x$ ,  $y$ , and  $z$  are real numbers,  $\max(y, z)$  is  $y$  or  $z$ , so  $\max(x, \max(y, z))$  is  $x$  or  $y$  or  $z$ . Suppose that  $\max(x, \max(y, z)) = x$ . Then  $x \geq \max(y, z)$ , so  $x \geq y$  and  $x \geq z$ . Now  $x \geq y$  implies that  $\max(x, y) = x$  and

$x \geq z$  implies that  $\max(x, z) = x$ . Hence,  $\max(\max(x, y), z) = \max(x, z) = x$ . Hence,  $\max(x, \max(y, z)) = \max(\max(x, y), z)$ .

In a similar way we can show that if  $\max(x, \max(y, z))$  is  $y$  or  $z$ , then  $\max(x, \max(y, z)) = \max(\max(x, y), z)$ .

## 1.6 Algorithms

1. This function computes  $x^3$ .
- 7.

**ALGORITHM:** Determine the position of the last occurrence of the smallest element in a list.

*Input:*  $L$ —a list of  $n$  elements,  
 $n$ —the size of the list  
*Output:* Position of the last occurrence of the smallest element of  $L$

1. **function** **lastSmallestIndex**( $L$ ,  $n$ )
2. **begin**
3.   minIndex :=  $n$ ;
4.   **for**  $i := n - 1$  **to** 1 **do**
5.     **if**  $L[\text{minIndex}] > L[i]$  **then**
6.       minIndex :=  $i$ ;
7.     **return** minIndex;
8. **end**

# CHAPTER 2: INTEGERS AND MATHEMATICAL INDUCTION

## 2.1 Integers

1. (a)  $q = 22$  and  $r = 6$   
 (b)  $q = -23$  and  $r = 21$   
 (c)  $q = 23$  and  $r = 21$   
 (d)  $q = -22$  and  $r = 6$
3.  $3092 \text{ div } 5 = 618$  and  $-7308 \text{ div } 11 = -665$ .
9. Because  $a | (5x + 4)$  and  $a | (65x + 9)$ , we have  $a | (13(5x + 4) - (65x + 9))$ , by Theorem 2.1.14(iii). Thus  $a | 43$ . The only positive divisors of 43 are 1 and 43. Hence,  $a = 43$ .
13. Because  $a | b$  and  $a | c$ , there exist integers  $k$  and  $t$  such that  $b = ak$  and  $c = at$ . Thus,  $bc = akat = a^2(kt)$ . Now  $kt$  is an integer and so  $a^2 | bc$ .
15. Because  $a | b$  and  $b | a$ , there exist integers  $u$  and  $v$  such that  $b = au$  and  $a = bv$ . Hence,  $b = au = (bv)u = buv$ . Hence,  $b(1 - uv) = 0$  and because  $b \neq 0$ , we have  $1 - uv = 0$ , i.e.,  $uv = 1$ . This implies that  $u = v = 1$  or  $u = v = -1$ . Hence,  $b = a$  or  $b = -a$ . Because  $a > 0$  and  $b > 0$ , we must have  $a = b$ .
25. Suppose  $\gcd(5k + 3, 3k + 2) = d$ . Then  $d | (5k + 3)$  and  $d | (3k + 2)$ . This implies that  $d | (2(3k + 2) - (5k + 3))$ ,

i.e.,  $d | (k + 1)$ . Next  $d | (3k + 2)$  and  $d | (k + 1)$  and so  $d | (3(k + 1) - (3k + 2))$ , i.e.,  $d | 1$ . Thus,  $d = 1$ . Hence,  $\gcd(5k + 3, 3k + 2) = 1$ .

## 2.2 Representation of Integers in Computer

1.  $(11000111111)_2$
5.  $(11110111111001)_2$
9.  $260873_{10}$
13.  $111_2$
17.  $10011100_2$
21.  $18869_{10}$
31. The addition of the given binary strings can be performed.
33. The addition of the given binary strings cannot be performed.
3. 145
7.  $(13D9)_{16}$
11.  $1010011_2$
15.  $111_2$
19.  $-19$

## 2.3 Mathematical Induction

1. *Basis step:* Let  $n = 1$ . Then  $n^2 = 1$ , so  $1 = n^2$ , when  $n = 1$ . Thus, the result is true for  $n = 1$ .

*Inductive hypothesis:* Suppose  $1 + 3 + 5 + \cdots + (2k - 1) = k^2$  for some positive integer  $k$ .

*Inductive step:* Let  $n = k + 1$ . We want to show that  $1 + 3 + 5 + \cdots + (2k - 1) + (2k + 1) = (k + 1)^2$ . Now  $1 + 3 + 5 + \cdots + (2k - 1) + (2k + 1) = (1 + 3 + 5 + \cdots + (2k - 1)) + (2k + 1) = k^2 + (2k + 1) = k^2 + 2k + 1 = (k + 1)^2$ . Thus, the result is true for  $n = k + 1$ . The result now follows by induction.

7. (a) *Basis step:* Let  $n = 1$ . Then  $2^{2n} - 1 = 2^{2 \cdot 1} - 1 = 3$ , which is divisible by 3. Thus, the result is true for  $n = 1$ .

*Inductive hypothesis:*  $3 \mid (2^{2k} - 1)$  for some positive integer  $k \geq 1$ .

*Inductive step:* Let  $n = k + 1$ . We want to show that  $2^{2(k+1)} - 1$  is divisible by 3.

Now  $2^{2(k+1)} - 1 = 2^{2k+2} - 1 = 2^{2k} \cdot 2^2 - 1 = 4 \cdot 2^{2k} - 1 = 4 \cdot 2^{2k} - 4 + 3 = 4(2^{2k} - 1) + 3$ . By the inductive hypothesis, 3 divides  $2^{2k} - 1$ , so 3 divides  $4(2^{2k} - 1)$ . Also 3 divides 3. It now follows that 3 divides  $4(2^{2k} - 1) + 3$ . That is, 3 divides  $2^{2(k+1)} - 1$ . Thus, the result is true for  $k+1$ . Hence, by induction, the result is true for all  $n \geq 0$ .

11. (a) *Basis step:* For  $n = 7$ ,  $7! = 5040 > 2187 = 3^7$ . Thus, the result is true for  $n = 7$ .

*Inductive hypothesis:* Suppose  $3^k < k!$  for some  $k \geq 7$ .

*Inductive step:* Let  $n = k + 1$ . We want to show that  $(k+1)! \geq 3^{k+1}$ . Now  $(k+1)! = (k+1)k! > (k+1)3^k > 3 \cdot 3^k = 3^{k+1}$ . Thus, the result is true for  $k+1$ . Hence, by induction, the result is true for all  $n \geq 7$ .

# CHAPTER 3: RELATIONS AND POSETS

### 3.1 Relations

- (a)  $R_1 = \{(2, 4), (2, 10), (2, 16), (5, 10), (8, 16)\}$   
 (b)  $R_2 = \emptyset$   
 (c)  $R_3 = \{(1, 5), (1, 6), (1, 10), (2, 5), (3, 5), (3, 10), (4, 5)\}$   
 (d)  $R_4 = \{(2, 2), (2, 3), (2, 4), (2, 5), (2, 10), (5, 5), (5, 10), (7, 10)\}$   
 (e)  $R_5 = \{(2, 1), (10, 9), (12, 11)\}$
  - $R^2 = \{(1, 3), (1, 4), (2, 4)\}$ .  $R^3 = \{(1, 4)\}$
  - The relations (a) and (f) of Exercise 6 are equivalence relations. The partition of  $\{1, 2, 3, 4\}$  corresponding to the equivalence relation of (a) is  $\{\{1, 2\}, \{3\}, \{4\}\}$  and the partition of  $\{1, 2, 3, 4\}$  corresponding to the equivalence relation of (f) is  $\{\{1\}, \{2\}, \{3\}, \{4\}\}$ .
  - $R = \{(a, a), (b, b), (c, c), (d, d), (e, e), (b, c), (c, b), (d, e), (e, d)\}$
  - (1, 1), (2, 2), (3, 3), (2, 1), (3, 2), (1, 3), and (3, 1)
  - (a), (c), and (e) are equivalence relations.

## 2.4 Prime Numbers

- (a) 391 is not a prime integer  
 (b) 1999 is prime  
 (c) 2033 is not a prime integer
  - (a) The divisors of 2502 are 1, 2, 3, 6, 9, 18, 139, 278, 417, 1251, 834, and 2502.  
 (b) The divisors of 399 are 1, 3, 7, 19, 21, 57, 133 and 399.  
 (c) The divisors of 2177 are 1, 7, 311, and 2177.
  - Now  $p \mid a^6$  implies  $p \mid a$  because  $p$  is a prime. This implies that  $p \mid a^3$ . Now  $p \mid a^3$  and  $p \mid a^3 + b^7$ . Hence,  $p \mid b^7$  and so  $p \mid b$ .
  - 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89
  - Because  $7k + 1$  is prime,  $k \geq 2$ . Now there exists an integer  $t \geq 1$ , such that  $k = 2t$  or  $k = 2t + 1$ . Then  $7k + 1 = 14t + 1$  or  $7k + 1 = 14t + 7 + 1 = 14t + 8$ . Because  $14t + 8$  is not prime, it follows that  $7k + 1 = 14t + 1$ .
  - $8127 = 129 \cdot 63$ ;  $17653 = 139 \cdot 127$ ;  $9970716 = 3282 \cdot 3038$ .

## 2.5 Linear Diophantine Equations

1. (b) and (d)
  3.  $x = 228 - 19n$ ,  $y = 342 - 29n$  for any integer  $n$ .
  5.  $x = -50 + 13n$ ,  $y = 20 - 5n$ , for any integer  $n$ .
  7.  $x = -200 - 5n$ ,  $y = -300 - 7n$ , for any integer  $n$ .
  9. There is one positive solution,  $x = 5$ ,  $y = 4$ .
  11. (1, 16), (4, 14), (7, 12), (10, 10), (13, 8), (16, 6), (19, 4), and (22, 2)
  13. (6, 10) and (13, 5)
  15. There are 8 ways John can place the order.
  19. 8



### 3.2 Partially Ordered Sets

1. (a), (b), and (c) are antisymmetric.
  3.  $R$  is not a partial order.
  5.  $R$  is antisymmetric.
  9. On  $S = \{1, 2, 3\}$  we take  $R = \{(1, 1), (2, 2), (1, 2)\}$ . Then clearly  $R$  is antisymmetric but not reflexive.

11. Because  $1 = (-1)(-1)$  and  $-1 = 1(-1)$ , we see that  $1 \leq -1, -1 \leq 1$  but  $1 \neq -1$ . Hence,  $R$  is not antisymmetric and so  $R$  is not a poset.
17.  $\{(1, 1), (1, 2), (1, 3), (1, 6), (2, 1), (2, 2), (2, 3), (2, 6), (4, 1), (4, 2)\}$
21. (a) invitation, invite, real, reason, relation, relative, reliable.  
(b) orange, organize, partial, party, pond, poset, posses.
23.  $4 \wedge (5 \vee 9) = 4$  and  $(3 \vee (3 \wedge 8)) = 3$ . It is not a Boolean algebra.
27. (a) Because lub $\{a, c\}$  does not exist, it is not a lattice.  
(b) It is a lattice. (c) It is a lattice.
29.  $4 \wedge (6 \vee 14) = 4$  and  $(2 \vee (2 \wedge 8)) \vee 21 = 42$

### 3.3 Application: Relational Database

1.  $\text{Supplier} = \{(100, J \& M, 222 - 2222), (200, A \text{ Soft}, 333 - 3333), (300, Adams \& Co, 444 - 4444), (400, Ware D, 555 - 5555)\}$ .

5.  $ID$

7.  $PCode$

9. select PName, PrdPrice, UnitsInStock  
from Product;

11. select PName  
from Product  
where UnitsInStock > 75;

## CHAPTER 4: MATRICES AND CLOSURES OF RELATIONS

### 4.1 Matrices

1. (a)  $a_{11} = 7$  (b)  $a_{23} = 9$   
(c)  $b_{13} = -4$  (d)  $b_{22} = -49$   
(e)  $b_{24}$  does not exist (f)  $c_{12}$  does not exist  
(g)  $c_{21} = 23$
5. Here  $a + 2b = 5$ ,  $2a - b = 0$ ,  $c + d = -4$  and  $c - 2d = 17$ . Solving we get  $a = 1$ ,  $b = 2$ ,  $c = 3$ , and  $d = -7$ .
7. (a)  $AB = \begin{bmatrix} -1 & -2 \\ -12 & -16 \end{bmatrix}$  (b)  $BA = \begin{bmatrix} -1 & -8 \\ -3 & -16 \end{bmatrix}$   
(c)  $AC = \begin{bmatrix} -4 \\ -36 \end{bmatrix}$  (d)  $CA$  does not exist,  
(e)  $BD = \begin{bmatrix} 0 & -14 & 17 \\ -2 & -28 & 37 \end{bmatrix}$   
(f)  $A^2 + B^2 = \begin{bmatrix} 8 & 10 \\ 15 & 38 \end{bmatrix}$
9. (a)  $AB$  does not exist  
(b)  $BC = \begin{bmatrix} 10 & 7 \\ 28 & 19 \\ 46 & 31 \end{bmatrix}$   
(c)  $DA$  does not exist  
(d)  $DA - 5BC$  does not exist,  
(e)  $D^2 + E^2 = \begin{bmatrix} 52 & -4 \\ 16 & 19 \end{bmatrix}$ ,  
(f)  $CE + CD = \begin{bmatrix} 38 & -17 \\ 34 & 1 \\ 12 & -10 \end{bmatrix}$ ,  
(g)  $C(E + D) = \begin{bmatrix} 38 & -17 \\ 34 & 1 \\ 12 & -10 \end{bmatrix}$ .
13. (a) The size of  $C$  is  $3 \times 3$  and that of  $D$  is  $2 \times 2$ .  
(b)  $c_{22} = 3$ ,  $c_{32} = 43$ ,  $d_{11} = 9$ ,  $d_{21} = -20$ ,  $d_{32}$  does not exist.
31. Let  $A = [a_{ij}]$ , where  $a_{ij} = 0$  for all  $i \neq j$ . Let  $A^T = [b_{ij}]$ . Then  $b_{ii} = a_{ii}$  and  $b_{ij} = a_{ji} = 0 = a_{ij}$  for all  $i \neq j$ . Hence  $A^T = A$ .

39. (a)  $A \wedge B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ , (b)  $A \vee B = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$ ,  
(c)  $B \vee C = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$ ,  
(f)  $A \vee (B \wedge C) = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ ,  
(g)  $A \wedge (B \vee C) = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ .

### 4.2 The Matrix of a Relation and Closures

1.  $M_R = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$ ,  $M_S = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$ ,

$M_{R \cup S} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$ ,  $M_{R \cap S} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ ,

$M_{R^{-1}} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ .

5. (a)  $R = \{(a, 2), (b, 1), (b, 2), (c, 1), (c, 2), (d, 2)\}$  and  $S = \{(1, y), (2, z), (3, y)\}$

(b)  $\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$

- (c)  $S \circ R = \{(a, z), (b, y), (b, z), (c, y), (c, z), (d, z)\}$

9. (a)  $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$  (b)  $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$

(c)  $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$  (d)  $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$

11.  $R$  is not transitive. Transitive closure of  $R$  using Warshall's algorithm is  $R^\infty = \{(1, 1), (1, 3), (2, 2), (3, 1), (3, 3)\}$ .
13.  $R^\infty = \{(a_1, a_2), (a_1, a_3), (a_1, a_4), (a_2, a_2), (a_2, a_3), (a_2, a_4), (a_3, a_3), (a_3, a_4), (a_4, a_4)\}$

15.  $R^\infty = \{(a_1, a_1), (a_1, a_2), (a_2, a_2), (a_3, a_1), (a_3, a_2), (a_3, a_3), (a_4, a_1), (a_4, a_2), (a_4, a_4)\}$

## CHAPTER 5: FUNCTIONS

### 5.1 Functions

1. (c), (d), (e), and (f) are functions.
3. The range is all those real numbers that are greater than or equal to  $\frac{3}{4}$ .
7. (Recall that  $m(\text{mod } n)$  is the remainder when  $m$  is divided by  $n$ .) Now  $(2 \cdot 3) + 3(\text{mod } 5) = 4$ ,  $(2 \cdot 4) + 3(\text{mod } 5) = 1$ ,  $(2 \cdot 5) + 3(\text{mod } 5) = 3$ ,  $(2 \cdot 6) + 3(\text{mod } 5) = 0$ ,  $(2 \cdot 7) + 3(\text{mod } 5) = 2$ ,  $(2 \cdot 8) + 3(\text{mod } 5) = 4$ . Hence,  $f = \{(3, 4), (4, 1), (5, 3), (6, 0), (7, 2), (8, 4)\}$ . Now  $3 \neq 8$  and  $f(3) = 4 = f(8)$ . Thus,  $f$  is not one-one. Again the range of  $f$ ,  $\text{Im}(f) = \{0, 1, 2, 3, 4\} = Y$ , so  $f$  is onto  $Y$ .
13.  $\text{Im}(f)$  is all those real numbers that are greater than or equal to  $-9$ .  $f$  is not onto  $\mathbb{R}$ .  $f$  is not one-one.
17. If  $A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ , then  $f(A) = 0 - 0 = 0$ . Again if  $A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ , then  $f(A) = 1 - 1 = 0$ .  $f$  is not one-one.  $f$  is onto  $\mathbb{R}$ .
23.  $(f \circ g)(x) = \sqrt{3x+1}$ .  $(g \circ f)(x) = 3\sqrt{x}+1$ .  $f \circ g \neq g \circ f$ .
25.  $(g \circ f)(x) = 16x^2 - 24x + 11$ .  
 $(f \circ g)(x) = 16x - 15$ .  $(g \circ g)(x) = x^4 + 4x^2 + 6$ .

### 5.2 Special Functions and Cardinality of a Set

1. To show  $f$  is one-one, let  $x, y \in X$  and  $f(x) = f(y)$ . Now  $f(x) = f(y)$  implies  $x + 3 = y + 3$ . This implies that  $x = y$ . Thus,  $f$  is one-one. Let  $a \in Y$ . Then  $1 < a \leq 8$ , and so  $-2 < a - 3 \leq 5$ . So there exists  $a - 3 \in X$  such that  $f(a - 3) = (a - 3) + 3 = a$ . Hence,  $f$  is onto. Consequently,  $f$  is a one-to-one correspondence.  
We define  $g : Y \rightarrow X$  by  $g(a) = a - 3$ . Then  $g$  is the inverse of  $f$  and so  $g = f^{-1}$ .
3. Define  $g : \mathbb{Q} \rightarrow \mathbb{Q}$  by  $g(a) = \frac{a}{8}$  for all  $a \in \mathbb{Q}$ . Then  $g$  is the inverse of  $f$ .
5. Define  $g : \mathbb{R} \rightarrow \mathbb{R}$  by  $g(a) = \frac{2a+10}{3}$  for all  $a \in \mathbb{R}$ . Then  $g$  is the inverse of  $f$ .
7. We show that  $f$  is not one-one. For this let  $x$  be a positive real number. Then  $x \neq -x$  but  $f(x) = x^2 = (-x)^2 = f(-x)$ . Thus,  $f$  is not one-one. Hence, an inverse of  $f$  does not exist.
9. An inverse of  $f$  does not exist.
15. (a) Define  $g : \mathbb{Z} \rightarrow \mathbb{Z}$  by  $g(x) = x + 2$  for all  $x \in \mathbb{Z}$ . Then  $g$  is a right inverse of  $f$ .

- (b) A right inverse of  $f$  does not exist.
- (c) An inverse of  $f$  does not exist.
21.  $\lfloor 8.13 \rfloor = 8$ ;  $\lfloor \sqrt{221} \rfloor = 14$ ;  $\lfloor -11.23 \rfloor = -12$ ;  $\lfloor -1.23 \rfloor + 1 = -2 + 1 = -1$
23.  $f(9.11) = 11$  and  $g(-13.02) = -12$
25.  $(f \circ g)(x) = \lfloor x \rfloor + 3$  and  $(g \circ f)(x) = \lceil x \rceil + 3$
27. Define  $f : \mathbb{N} \rightarrow S$  by  $f(n) = n + 99$ . Then  $f$  is a one-to-one correspondence. Because  $\mathbb{N}$  is countable so is  $S$ .
29. Define  $f : \mathbb{R}^+ \rightarrow S$  by  $f(x) = \frac{x}{x+1}$ . Then  $f$  is a one-to-one correspondence.
33. False

### 5.3 Sequences and Strings

1.  $a_1 = 6$ ,  $a_2 = 8$ ,  $a_3 = 10$ ,  $a_4 = 12$ , and  $a_5 = 14$
3.  $b_1 = -1$ ,  $b_2 = -6$ ,  $b_3 = 9$ ,  $b_4 = \frac{12}{3} = 4$ , and  $b_5 = 3$
5.  $s_0 = 0$ ,  $s_1 = -1$ ,  $s_2 = 4$ ,  $s_3 = -9$ , and  $s_4 = 16$
7.  $b_1 = 1$ ,  $b_2 = 2$ ,  $b_3 = \frac{3}{2}$ ,  $b_4 = \frac{2}{3}$ , and  $b_5 = \frac{5}{24}$
9. Let  $\{a_n\}_{n=1}^\infty$  denote this sequence. Then  $a_1 = 1$ ,  $a_2 = -1$ ,  $a_3 = 1$ ,  $a_4 = -1$ ,  $a_5 = 1$ ,  $a_6 = -1$   $a_n = (-1)^{n-1}$ .
11. Let  $\{a_n\}_{n=1}^\infty$  denote this sequence. Then  $a_1 = 3$ ,  $a_2 = 8$ ,  $a_3 = 15$ ,  $a_4 = 24$ ,  $a_5 = 35$ ,  $a_6 = 48$ .  $a_n = (n)^2 + 2n$ .
13. Let  $\{a_n\}_{n=1}^\infty$  denote this sequence. Then  $a_1 = \frac{1}{8}$ ,  $a_2 = \frac{4}{27}$ ,  $a_3 = \frac{9}{64}$ ,  $a_4 = \frac{16}{125}$ ,  $a_5 = \frac{25}{216}$ .  $a_n = \frac{n^2}{(n+1)^3}$ .
17. (a)  $-30, -49, -68$ , and  $-87$  (b)  $\frac{4}{3}, 3, \frac{14}{3}$ , and  $\frac{19}{3}$
19.  $75 \quad 21. \quad 6 \quad 23. \quad 256 \quad 25. \quad 4, 6, 9$  or  $9, 6, 4$
27.  $28 \quad 29. \quad 95 \quad 31. \quad \frac{7}{8} \quad 33. \quad 1$
35.  $\frac{25200}{143}$
37.  $\sum_{n=1}^7 a_n$ , where  $a_n = (-1)^{n-1} n^2$
39.  $\sum_{i=1}^n a_i$ , where  $a_i = \frac{i+1}{i}$
41.  $\sum_{i=0}^n a_i$ , where  $a_i = a^i$
43.  $\prod_{n=1}^5 a_n$ , where  $a_n = \frac{n}{n+1}$
45.  $\sum_{j=0}^{n-1} j^2$
49.  $\sum_{k=1}^n (k^2 - k - 3)$
55. (i)  $|s_1| = 6$ , (ii)  $|s_2| = 6$ , (iii)  $|s_3| = 9$ ,  
(iv)  $s_1 s_2 = aabbccabcabc$ , (v)  $s_3 s_2 = abccbacababcabc$ ,  
(vi)  $s_1 s_2 s_3 = aabbccabcabcabccbabcab$
47.  $\sum_{j=2}^{n+1} \frac{2n-j}{j^2}$
51.  $\prod_{k=1}^n \frac{2k}{k+2}$
47. All are associative.
49. Only (ii) is commutative.
55. (c) is commutative.

### 5.4 Binary Operations

1. All are associative.
3. Only (ii) is commutative.
7. (c) is commutative.

## **CHAPTER 6: CONGRUENCES**

## 6.1 Congruences

1. (a) False (b) True (c) False (d) True (e) False (f) True  
3. 13                            5. 1                            7. 2518  
11. (c)  
13. (a), (b), and (d) are divisible by 3. (d) is divisible by 9.  
15. (c)

## 6.2 Check Digits

1. 7      3. 4      5. (c) and (f) are valid.      9. No  
11. (a) 3 (b) 3 (c) 5  
15. 344207 000310 4  
17. (a) Visa card (b) 563 98 (c) 10 3862 540 (d) 8 (e) Yes  
19. (a) Not valid (b) Valid (c) Valid

- (d) All integers  $x$  such that  $x \equiv 4 \pmod{19}$ .  
(e) All integers  $x$  such that  $x \equiv (8 + \frac{9}{3}j) \pmod{9}$ , where  $j = 0, 1$  and  $2$ .  
(f) All integers  $x$  such that  $x \equiv (9 + \frac{42}{6}j) \pmod{42}$ , where  $j = 0, 1, 2, 3, 4, 5$ .

3. (a) All integers  $x$  such that  $x \equiv 139 \pmod{153}$ .  
(b) All integers  $x$  such that  $x \equiv 499 \pmod{665}$ .  
(c) All integers  $x$  such that  $x \equiv 206 \pmod{210}$ .  
(d) All integers  $x$  such that  $x \equiv 1226 \pmod{2145}$ .

5.  $30n + 23$ ,  $n = 0, 1, 2, \dots$

7. 117

9.  $a + b \Leftrightarrow (0, 3, 4)$  and  $a \cdot b \Leftrightarrow (2, 2, 3)$

13.  $\text{HashTable}[0] \leftarrow 907354864$ ,  $\text{HashTable}[2] \leftarrow 193318595$ ,  
 $\text{HashTable}[5] \leftarrow 132489973$ ,  $\text{HashTable}[8] \leftarrow 134052056$ ,  
 $\text{HashTable}[6] \leftarrow 316500307$ ,  $\text{HashTable}[3] \leftarrow 106500306$ ,  
 $\text{HashTable}[4] \leftarrow 116510307$ ,  $\text{HashTable}[7] \leftarrow 107354865$ .

## 6.3 Linear Congruences

- (a) All integers  $x$  such that  $x \equiv 3 \pmod{9}$ .  
 (b) All integers  $x$  such that  $x \equiv 21 \pmod{26}$ .  
 (c) All integers  $x$  such that  $x \equiv (12 + \frac{15}{3}j) \pmod{15}$ , where  $j = 0, 1$  and  $2$ .

- ## 6.4 Special Congruence Theorems

1. (a) 3 (b) 1 (c) 1 (d) 17 (e) 4 (f) 4  
 3. 8 5. 11 7. 7  
 13. (a) 1998 (b) 100 (c) 162 (d) 648  
 21. 4

# CHAPTER 7: COUNTING PRINCIPLES

## **7.1 Basic Counting Principle**

1. 60      3. 60      5. 5  
 7. 7  
 11. 20,280,000  
 15. 64  
 19. 63  
 23. (a) 8    (b) 3    (c) 5    (d) 5  
 25. 90  
 27. 27,216  
 29. (a) 166    (b) 100    (c) 71    (d) 33    (e) 119    (f) 286  
 31. (a) 0    (b) 40320    (c) 1,814,400    (d) 5,079,110,400  
 33. (a) 100    (b)  $2^{100} - 101$   
 35. (a)  $26^5$     (b)  $26^6$

(c) If  $n$  is even, say  $n = 2m$  for some positive integer  $m$ , then there are  $26^m$  palindromes of length  $n = 2m$ . If  $n$  is odd, say  $n = 2m + 1$ , then  $26^{m+1}$  palindromes of length  $n = 2m + 1$ .

37.  $10^9 - 1$   
39. (a) 55  
(b) The inner loop executes 10 times and it has 55 iterations.  
(c) The outer loop executes once and it has 10 iterations.

## 7.2 Pigeonhole Principle

1. There 365 or 366 days in a year. You can think of days as pigeonholes and students as pigeons. Now there are 400 students and 365 or 366 days, so there are 400 pigeons and 365 or 366 pigeonholes. Because  $400 > 366$ , at least two students must be assigned the same birthday. Hence, at least two students are born on the same day of the year. This also shows that at least two students are born on the same day of a month.

5. True

11. 16

5. 4      17. 3

21. (a) 288 (b) 114 (c) 290

## 7.3 Permutations

- $P(10, 3) = 720$ ,  $P(15, 10) = 10897286400$ ,  $P(6, 0) = 1$ ,  $P(6, 6) = 720$ .
  - $P(n, n - 1) = \frac{n!}{(n - (n - 1))!} = \frac{n!}{(n - n + 1)!} = \frac{n!}{1!} = n!$
  - 24      7. 210      9. 60      11. (a) 720 (b) 96
  13. 24      15. 111      17.  $P(9, 6) \cdot 8! = 2,438,553,600$
  9. 840
  21. (a) 720      (b) 1440      (c) 24      (d) 96
  23. 360

## 7.4 Combinations

1.  $C(10, 3) = 120$ ,  $C(15, 10) = 3003$ ,  $C(6, 0) = 1$ ,  $C(6, 6) = 1$
3. 15    5. 8    7. 1820    9. 31824
11. (a) 117,600    (b) 566,320    (c) 654,320
13. 441    15. 200
17. (a) 6336    (b) 6831
19. 344

## 7.5 Generalized Permutations and Combinations

1. 151,200    3. 4,989,600    5. 360,360
7. 210    9. 1,771    11. 91
13. 13    15. 1,001    17. 165

## 7.6 Binomial Coefficients

1. (a)  $243a^5 + 405a^4b + 270a^3b^2 + 90a^2b^3 + 15ab^4 + b^5$   
(b)  $x^6 - 12x^5y + 60x^4y^2 - 160x^3y^3 + 240x^2y^4 - 192xy^5 + 64y^6$   
(c)  $x^4 + 8x^2 + 24 + \frac{32}{x^2} + \frac{16}{x^4}$   
(d)  $1 - 6x^2 + 15x^4 - 20x^6 + 15x^8 - 6x^{10} + x^{12}$
3. -165
5. (a) The first two terms are  $x^{15}$  and  $15x^{14}y$   
(b) The last two terms are  $15xy^{14}$  and  $y^{15}$ , respectively.

(c) The seventh term is  $5005x^9y^6$ . The eighth term is  $6435x^8y^7$ .

7. (a)  $x^5 + 5x^2 + \frac{10}{x} + \frac{10}{x^4} + \frac{5}{x^7} + \frac{1}{x^{10}}$
- (b)  $x^{10} - 5x^7 + 10x^4 - 10x + \frac{5}{x^2} - \frac{1}{x^5}$
9. 21    11. 1365    13.  $1120x^8y^4$
15. 1 7 21 35 35 21 7 1

## 7.7 Generating Permutations and Combinations

1. 13267548, 13587462, 26753184, 26754381, 37284165, 53728164
3. 51234, 51243, 51324, 51342, 51423, 51432, 52134, 52143, 52314, 52341, 52413, 52431, 53124, 53142, 53214, 53241, 53412, 53421, 54123, 54132, 54213, 54231, 54312, 54321
5. (a) {3, 5, 8} (b) {2, 3, 6, 7, 9} (c) {1, 3, 5, 6, 7, 8}

## 7.8 Discrete Probability

1. (a)  $\frac{15}{36}$     (b)  $\frac{5}{18}$     (c)  $\frac{35}{36}$
3. The event  $E$  consists of  $HHT$ ,  $HTH$ , and  $THH$ . The probability of obtaining exactly one tails is  $\frac{3}{8}$ .
5.  $S = \{abc, abd, abe, acd, ace, ade, bcd, bce, bde, cde\}$
7.  $\frac{1}{4}$     9.  $\frac{1}{2}$     11.  $\frac{1}{9}$     13.  $\frac{1}{56}$
15. (a)  $\frac{56}{1287}$     (b)  $\frac{1246}{1287}$     (c)  $\frac{560}{1287}$
17.  $\frac{15}{1001}$     19.  $\frac{20}{63}$     21.  $\frac{1}{4}$
23. (a)  $\frac{1}{18}$     (b)  $P(A) = \frac{5}{9}$   $P(B) = \frac{1}{2}$

# CHAPTER 8: RECURRENCE RELATIONS

## 8.1 Sequences and Recurrence Relations

1. (a)  $a_0 = 1$ ,  $a_1 = 5$ ,  $a_2 = 9$ ,  $a_3 = 13$ ,  $a_4 = 17$   
(b)  $a_0 = 5$ ,  $a_1 = 7$ ,  $a_2 = 11$ ,  $a_3 = 17$ ,  $a_4 = 25$   
(c)  $a_0 = 1$ ,  $a_1 = 3$ ,  $a_2 = 10$ ,  $a_3 = 29$ ,  $a_4 = 74$   
(d)  $a_0 = 1$ ,  $a_1 = 2$ ,  $a_2 = 6$ ,  $a_3 = 14$ ,  $a_4 = 38$   
(e)  $a_0 = 5$ ,  $a_1 = 2$ ,  $a_2 = -40$ ,  $a_3 = -298$ ,  $a_4 = -1600$   
(f)  $a_0 = -1$ ,  $a_1 = 1$ ,  $a_2 = 2$ ,  $a_3 = 6$ ,  $a_4 = 12$   
(g)  $a_0 = 2$ ,  $a_1 = 1$ ,  $a_2 = 1$ ,  $a_3 = 5$ ,  $a_4 = 12$
3.  $a_0 = 1$ ,  $a_1 = 2$ ,  $a_2 = 0$ ,  $a_3 = 1$ ,  $a_4 = 3$ ,  $a_5 = 3$ ,  $a_6 = 4$ ,  $a_7 = 7$
5.  $a_n = 2a_{n-1} + 1$ ,  $n \geq 1$ ,  $a_0 = 0$
7.  $a_n = a_{n-1} + 1$ ,  $n \geq 2$  and  $a_1 = 1$
9.  $a_n = 1.0009a_{n-1} - 4000$ ,  $n \geq 1$ , and  $a_0 = 440,000$
11. (a)  $A_n = 1.075A_{n-1}$ ,  $n \geq 1$ ,  $A_0 = 7000$  (b) \$10049.39
13.  $B_n = B_{n-1} + B_{n-2}$ ,  $n \geq 3$  and the initial conditions are  $B_1 = 2$  and  $B_2 = 3$ .
15. (a)  $A_n = 1.09A_{n-1}$ ,  $n \geq 1$  and  $A_0 = 8500$   
(b)  $A_0 = 8500$ ;  $A_1 = 9265$ ;  $A_2 = 10098.85$ , and  $A_3 = 11007.75$

## 8.2 Linear Homogeneous Recurrence Relations

1. (b) and (e)
3.  $a_n = (8+n)2^{n-1}$ ,  $n \geq 0$
5.  $a_n = 2 \cdot 4^n + (-3)^n$ ,  $n \geq 0$
7.  $a_n = \left(\frac{1+\sqrt{2}}{4}3^n + \frac{3-\sqrt{2}}{4}(-1)^n\right)^2$ ,  $n \geq 0$

## 8.3 Linear Nonhomogeneous Recurrence Relations

1.  $S_n = S_{n-1} + n$ ,  $n > 1$ ,  $S_1 = 1$ . This is a linear nonhomogeneous recurrence relation.
3.  $a_n = 9 + \frac{1}{2}n + \frac{1}{2}n^2$ ,  $n \geq 0$
5.  $a_n = \frac{1}{2}8^n + \frac{1}{2}10^n$ ,  $n \geq 0$
7.  $a_n = \frac{459}{180} + \frac{136}{180}(-4)^n + \frac{125}{180} \cdot 5^n$ ,  $n \geq 0$
9.  $a_n = -\frac{29}{20}4^n + \frac{268}{180}(-1)^n + \frac{11}{18}2^n + \frac{17}{6}2^n$ ,  $n \geq 0$
11.  $a_n = \frac{1}{3}n3^n + 4 \cdot 2^n = 3^{n-1} \cdot n + 2^{n+2}$ ,  $n \geq 0$

## CHAPTER 9: ALGORITHMS AND TIME COMPLEXITY

### 9.1 Algorithm Analysis

1. (a)  $f(n) = 2n^4 + 7n + 5 \leq 2n^4 + n^4 + n^4 \leq 4n^4$  for all  $n \geq 2$ . Let  $c = 4$  and  $n_0 = 2$ . Then  $|f(n)| \leq cn^4$  for all  $n \geq n_0$ . Hence,  $f(n) = O(n^4)$ .
- (b)  $f(n) = 2n^4 + 7n + 5 \geq 2n^4$  for all  $n \geq 0$ . Let  $c = 2$  and  $n_0 = 0$ . Thus,  $|f(n)| \geq cn^4$  for all  $n \geq n_0$ . Hence,  $f(n) = \Omega(n^4)$ . By (a),  $f(n) = O(n^4)$ . It follows that  $f(n) = \Theta(n^4)$ .
3.  $f(n) = \Theta(n^2)$
5. By Exercise 4, Section 2.3 (Chapter 2),  

$$f(n) = \left(\frac{n(n+1)}{2}\right)^2 = \frac{1}{4}(n^2 + n)^2 = \frac{1}{4}(n^4 + 2n^3 + n^2).$$
By Theorem 9.1.18,  $f(n) = \Theta(n^4)$ .
13.  $\Theta(n)$ . The number of additions is  $2n$ .
15.  $\Theta(n)$  17.  $\Theta(n^2)$

21. Let  $f(n) = n$ . We know that  $f(n)$  is eventually nondecreasing for  $n > 0$ . Now  $f(2n) = 2n = \Theta(n) = \Theta(f(n))$ . Thus, the function  $f(n) = n$  is smooth.

27. Here  $a = 35$ ,  $b = 3$ ,  $c = 7$ , and  $k = 3$ . Now  $b^k = 3^3 = 27 < 35 = a$ . Hence,  $f(n) = \Theta(n^{\log_3 35})$ .

### 9.2 Various Algorithms

1. (a) 8 (b) 6 (c) 1 (d) 8
3. 3
7. 9
13. The optimal number of scalar multiplications is 16133. Thus, the optimal order to multiply  $A_1, A_2, A_3, A_4, A_5$  is:  $(A_1(A_2A_3))(A_4A_5)$ .

## CHAPTER 10: GRAPH THEORY

### 10.1 Graph Definition and Notations

1. The graph is shown in the accompanying figure. The number of edges is 2.

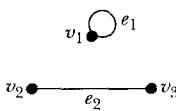


Figure Ch10Sec1Ex1

3. The graph is shown in the accompanying figure. The number of edges is 8.



Figure Ch10Sec1Ex3

9. (a) 3, 3, 4, 4  
(b) 0, 1, 2, 2, 3, 6, 6  
(c) 2, 2, 2, 2, 2, 3, 3  
(d) 2, 2, 3, 3, 4  
(e) 2, 2, 3, 3, 3, 3, 4  
(f) 1, 2, 2, 2, 2, 3, 4
11. Graphs (c), (e), and (f) are simple.
13. 10
15. No
17. 0, 1, 1, 2, 2; 0, 0, 2, 2, 2; 0, 1, 1, 1, 3; and 1, 1, 1, 1, 2
21. No
23. No
31. We know that a complete graph with  $n$  vertices has  $\frac{n(n-1)}{2}$  edges. Thus, a complete graph with 20 vertices must have  $\frac{20(20-1)}{2} = 190$  edges. Hence, there is no such graph.

35. (a) The number of edges in  $K_{2,3}$  is  $2 \cdot 3 = 6$ .  
(b) The number of edges in  $K_{4,3}$  is  $4 \cdot 3 = 12$ .  
(c) The number of edges in  $K_{4,4}$  is  $4 \cdot 4 = 16$ .  
(d) The number of edges in  $K_{n,n}$  is  $n \cdot n = n^2$ .

### 10.2 Walks, Paths, and Cycles

1. (a)  $(v_2, e_3, v_3, e_4, v_5, e_5, v_6, e_6, v_7, e_7, v_2)$  is a walk of length 4. Yes, it is a trail, because it has no repeated edges. It is a path.  
(b)  $(v_2, e_2, v_2, e_7, v_4, e_8, v_7, e_{10}, v_1, e_1, v_2)$  is a closed walk of length 5. Yes, it is a circuit.  
(c)  $(v_2, e_3, v_3, e_4, v_4, e_5, v_5, e_6, v_6, e_7, v_7, e_8, v_8, e_9, v_9, e_{10}, v_2)$  is a circuit of length 6. Yes, it is a cycle.  
(d)  $(v_2, e_7, v_4, e_8, v_7, e_{10}, v_1, e_1, v_2)$  is a 4 cycle.
3.  $G_1 = (\{v_3, v_4\}, \{e_3\})$ ,  $G_2 = (\{v_4, v_5, v_6\}, \{e_4, e_5, e_6\})$ , and  $G_3 = (\{v_4, v_5, v_6, v_1\}, \{e_5, e_6\})$  are subgraphs of the given graph.
7. Let  $P = (u = v_1, e_1, v_2, e_2, \dots, v_n = u)$  be a circuit. If this is not a cycle, then  $v_i = v_j$  for some  $1 \leq i < j \leq n$ . This shows that there is a closed walk  $Q$  from  $v_i$  to  $v_j$ . We reduce  $P$  to  $P - Q$ . Now  $P - Q$  is a new circuit from  $u$  to  $v$ . If this circuit is not a cycle, we repeat the above process. Because the number of closed walks in  $P$  is finite, we eventually obtain a cycle from  $u$  to  $u$ .
15. Let  $V_1 = \{v_1, v_2, v_3, v_4\}$  and  $V_2 = \{u_1, u_2, u_3\}$ . Then  $V_1 \cup V_2$  is a bipartition of the given graph. Now  $N(V_1) = \{u_1, u_2, u_3\}$ . Because  $|V_1| = 4 \not\leq 3 = |N(V_1)|$ , there is no matching that saturates the vertices  $v_1, v_2, v_3, v_4$  by Hall's Marriage Theorem.

## 10.3 Matrix Representation of Graphs

1. (a)  $\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 2 & 1 \\ 1 & 2 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}$  (c)  $\begin{bmatrix} 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 2 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 2 & 0 & 0 \end{bmatrix}$

3.  $A_G = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

5. (a) 5 (b) 4 (c) 0

## 10.4 Special Circuits

- In graph (a), vertex  $v_4$  is of degree 3, which is odd. Thus, this graph has a vertex of odd degree. Hence, this graph has no Euler circuit. In graph (b), every vertex is of even degree. Hence, this graph has an Euler circuit. One such circuit is:  $(v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_5, e_5, v_6, e_6, v_7, e_7, v_8, e_8, v_9, e_9, v_{10}, e_{14}, v_2, e_{15}, v_4, e_{11}, v_6, e_{12}, v_8, e_{13}, v_{10}, e_{10}, v_1)$ . In graph (c), every vertex is of even degree. Hence, this graph has an Euler circuit. One such circuit is:  $(v_3, e_{12}, v_1, e_{11}, v_5, e_6, v_2, e_3, v_6, e_7, v_3, e_8, v_4, e_{10}, v_4, e_9, v_5, e_5, v_6, e_2, v_1, e_1, v_2, e_4, v_3)$ . In graph (d), vertex  $v_3$  is of degree 3, which is odd. Thus, this graph has a vertex of odd degree. Hence, this graph has no Euler circuit. In graph (e), every vertex is of even degree. Hence, this graph has an Euler circuit. One such circuit is:  $(v_1, e_1, v_2, e_2, v_3, e_3, v_1, e_4, v_4, e_5, v_5, e_6, v_1, e_7, v_6, e_8, v_7, e_9, v_1, e_{10}, v_8, e_{11}, v_9, e_{12}, v_1)$ .
- Let  $v$  be a vertex in  $K_n$ . Then  $\deg(v) = n - 1$ . Suppose  $n$  is odd. Then  $n = 2k + 1$  for some positive integer  $k$ . Thus,  $\deg(v) = n - 1 = 2k + 1 - 1 = 2k$ , which is an even integer. Hence, every vertex of  $K_n$  is of even degree if  $n$  is odd. This implies that  $K_n$  has an Euler circuit if  $n$  is odd.
- (a) Because  $\deg(v_4) = 1$ ,  $v_4$  cannot be a member of any cycle. Hence, this graph has no Hamiltonian cycle.
- (b) A Hamiltonian cycle of this graph is the following:  $(v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_5, e_5, v_6, e_6, v_7, e_7, v_1)$ .
- (c) A Hamiltonian cycle of this graph is the following:  $(v_1, e_2, v_4, e_5, v_3, e_6, v_2, e_3, v_5, e_1, v_1)$ .
- Because the graph  $G$  is a simple connected graph,  $\deg(u) \leq n - 1$  for any vertex  $u$  of  $G$ . Thus, from the given condition,  $\deg(u) = n - 1$ . This shows that  $G$  is  $K_n$ . Hence, from Exercise 11, it follows that  $G$  has a Hamiltonian cycle.

## 10.5 Isomorphism

- The pairs of graphs in parts (c), (d), and (f) are isomorphic.

3. Let  $G_1 = (V_1, E_1)$  and  $G_2 = (V_2, E_2)$ . Because  $G_1$  is isomorphic to  $G_2$ , there exists a one-to-one correspondence  $f : V_1 \rightarrow V_2$  and a one-to-one correspondence  $h : E_1 \rightarrow E_2$  such that if any two vertices  $v_i, v_j \in V_1$  are end vertices of some edge  $e_k$  in  $G_1$ , then  $f(v_i)$  and  $f(v_j)$  are end vertices of the edge  $h(e_k)$  in  $G_2$ . Let  $G_2$  be not a simple graph. Then  $G_2$  contains a loop or parallel edges. Suppose  $G_2$  contains a loop  $e$  at a vertex  $u$  of  $G_2$ . Then there exists a vertex  $v$  of  $G_1$  and an edge  $e'$  of  $E_1$  such that  $f(v) = u$ ,  $h(e') = e$ . Let  $v_1, v_2$  be the end vertices of  $e'$ . Then  $f(v_1)$  and  $f(v_2)$  are the end vertices of  $h(e') = e$ . But  $e$  is a loop at  $u$ . Hence,  $f(v_1) = f(v_2) = u$ . Because  $f$  is one-one it follows that  $v_1 = v_2$ . This implies that  $e'$  is a loop at  $v_1$ . This contradicts that  $G_1$  is a simple graph. Hence,  $G_2$  has no loop. Similarly, we can show that  $G_2$  has no parallel edges. Hence,  $G_2$  is a simple graph.

- Let  $G_1$  be Eulerian. Then it follows that  $G_1$  is a connected graph and every vertex is of even degree. From Worked-Out Exercise 2, it follows that  $G_2$  is a connected graph. Again,  $G_1$  has a vertex of degree  $k$  if and only if  $G_2$  has a vertex of degree  $k$ . Thus, every vertex of  $G_2$  is of even degree. Hence, it follows that  $G_2$  is Eulerian.

## 10.6 Graph Algorithms

1. (a)

$a$	0	10	6	3	$\infty$	$\infty$	$\infty$
$v_1$	10	0	$\infty$	$\infty$	$\infty$	6	$\infty$
$v_2$	6	$\infty$	0	$\infty$	$\infty$	4	8
$v_3$	3	$\infty$	$\infty$	0	11	$\infty$	$\infty$
$v_4$	$\infty$	$\infty$	$\infty$	11	0	$\infty$	$\infty$
$v_5$	$\infty$	6	4	$\infty$	$\infty$	0	10
$z$	$\infty$	$\infty$	8	$\infty$	$\infty$	10	0

7.  $v_6, v_{10}, v_1, v_7, v_2, v_3, v_8, v_5, v_9, v_4$

## 10.7 Planar Graphs and Graph Coloring

- Graph (a) divides the plane into four regions  $R_1, R_2, R_3$ , and  $R_4$ . The boundary of  $R_1$  consists of the edges  $e_1, e_6, e_7$ , and  $e_4$ . The boundary of  $R_2$  consists of the edges  $e_6, e_5$ , and  $e_3$ . The boundary of  $R_3$  consists of the edges  $e_2, e_3$ , and  $e_5$ . The boundary of  $R_4$  consists of the edges  $e_1, e_2, e_3$ , and  $e_4$ . This planar graph has four faces,  $R_1, R_2, R_3$  and  $R_4$ .
- Graph (a) divides the plane into five regions,  $R_1, R_2, R_3, R_4$ , and  $R_5$ . The boundary of  $R_1$  consists of the edges  $e_1, e_2$ , and  $e_8$ . The boundary of  $R_2$  consists of the edges  $e_3, e_7$ , and  $e_2$ . The boundary of  $R_3$  consists of the edges  $e_5, e_7$ , and  $e_6$ . The boundary of  $R_4$  consists of the edges  $e_6, e_3, e_4$ , and  $e_5$ . The boundary of  $R_5$  consists of the edges  $e_1, e_4$ , and  $e_5$ . This planar graph has five faces,  $R_1, R_2, R_3, R_4$ , and  $R_5$ .
5. 3  
7. 3  
11. 3  
13.  $\chi'(K_{2,3}) = 3$  and  $\chi'(C_6) = 2$

# CHAPTER 11: TREES AND NETWORKS

## 11.1 Trees

1. Only (a) is a tree.
3. 15
5. No
7. Yes
13. Suppose  $K_{m,n}$  is a tree. Now the number of vertices of  $G$  is  $m + n$  and the number of edges is  $mn$ . Then  $m + n - 1 = mn$ , i.e.,  $mn - m - n + 1 = 0$ . Thus,  $(m-1)(n-1) = 0$ . This implies that either  $m = 1$  or  $n = 1$ . Because  $n \geq 2$ ,  $m = 1$ . Conversely, suppose that  $m = 1$ . Then  $K_{1,n}$  is a connected graph with  $n + 1$  vertices and  $n$  edges. Hence, it is a tree.

15.

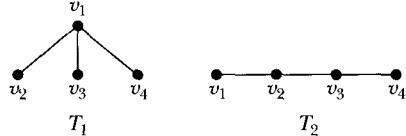


Figure Ch11Sec1Ex15

## 11.2 Rooted Trees

1.

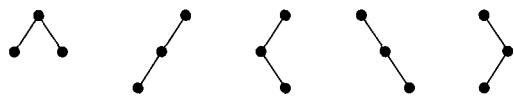


Figure Ch11Sec2Ex1

3. (a) BACED     (b) DBGEACHF
5. (a) BEDCA     (b) DGEBHFCA
15. 45
17.  $AB + CD + E -$
19.  $AB + CD - /E + F \cdot G -$
21.  $a \cdot b + c$
23.  $((a - b) - c) \cdot d$
25. (a)  $(x - y) \cdot (z + w)$      (b)  $(x/(y \cdot z)) \cdot (w/t - s)$ .
27. (a)  $x \cdot y - z \cdot w + \cdot$      (b)  $x \cdot y \cdot z \cdot / w \cdot t / s - \cdot$
29. 1430

## 11.3 Spanning Trees

1.

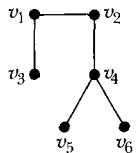


Figure Ch11Sec3Ex1

9.

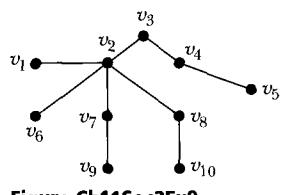


Figure Ch11Sec3Ex9

11.  $n$ 

17.

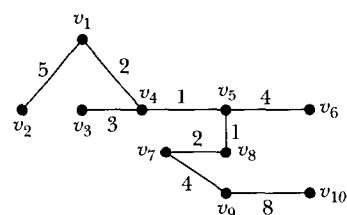


Figure Ch11Sec3Ex17

23.

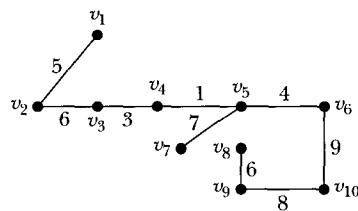


Figure Ch11Sec3Ex23

## 11.4 Networks

1. It is an  $s-t$  network and each arc is assigned a nonnegative number, so it is a transport network. The value of the flow is 12.
3. The value of the flow is 9. The current flow is not maximal as there are various  $F$ -unsaturated  $s-t$  paths. For example,  $s - v_1 - t$  is an  $F$ -unsaturated  $s-t$  path. A maximal flow is given by the accompanying figure and the value of this maximal flow is 12.

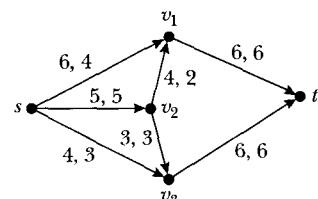


Figure Ch11Sec4Ex3

7. Let  $Q$  be the path  $s - v_1 - v_3 - t$ . Then  $i(Q) = \min\{5 - 3, 7 - 2, 4 - 1\} = \min\{2, 5, 3\} = 2$ . Thus, the path  $s - v_1 - v_3 - t$  is unsaturated. Hence, the flow is not maximal. A maximal flow is shown in the accompanying figure. The value of this maximal flow is 11.

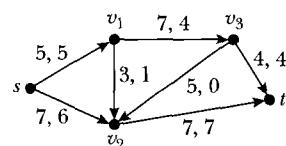


Figure Ch11Sec4Ex07

## CHAPTER 12: BOOLEAN ALGEBRA AND COMBINATORIAL CIRCUITS

### 12.1 Two-Element Boolean Algebra

1. (a) 1 (b) 1 (c) 0 (d) 1

3. (a)

$x_1$	$x_2$	$x_1 \cdot x_2$	$x'_1$	$x'_2$	$x'_1 \cdot x'_2$	$x_1 \cdot x_2 + x'_1 \cdot x'_2$
1	1	1	0	0	0	1
1	0	0	0	1	0	0
0	1	0	1	0	0	0
0	0	0	1	1	1	1

(b)

$x_1$	$x_2$	$x'_2$	$x_1 + x'_2$	$x_1 \cdot (x_1 + x'_2)$
1	1	0	1	1
1	0	1	1	1
0	1	0	0	0
0	0	1	1	0

5. (a)

$x_1$	$x_2$	$x_3$	$x'_2$	$x_2 \cdot x'_2$	$x_1 + x_2 \cdot x'_2$	$\alpha$	$\beta$
1	1	1	0	0	1	1	1
1	1	0	0	0	1	1	1
1	0	1	1	0	1	1	1
1	0	0	1	0	1	1	1
0	1	1	0	0	0	1	1
0	1	0	0	0	0	0	0
0	0	1	1	0	0	1	1
0	0	0	1	0	0	0	0

Hence,  $\alpha = \beta$ .

7. (a)  $(x_1 + x'_2) \cdot (x_1 + x_2)$

(b)  $(x_1 + x'_2) \cdot (x_1 + x_2)$

(c)  $(x'_1 + x'_2) \cdot (x'_1 + x_2)$

(d)  $x_1 + x_2$

(e)  $(x_1 + x'_2) \cdot (x_1 + x_2)$

9. (a)  $(x'_1 + x'_2 + x_3) \cdot (x'_1 + x_2 + x_3) \cdot (x_1 + x_2 + x_3)$

(b)  $(x'_1 + x'_2 + x'_3) \cdot (x'_1 + x_2 + x'_3) \cdot (x'_1 + x_2 + x_3)$

(c)  $(x'_1 + x'_2 + x_3) \cdot (x'_1 + x_2 + x_3) \cdot (x_1 + x_2 + x'_3) \cdot (x_1 + x_2 + x_3)$

(d)  $(x'_1 + x'_2 + x'_3) \cdot (x_1 + x'_2 + x'_3) \cdot (x_1 + x_2 + x_3)$

(e)  $(x'_1 + x'_2 + x_3) \cdot (x'_1 + x_2 + x_3) \cdot (x_1 + x_2 + x'_3) \cdot (x_1 + x_2 + x_3)$

11. (a)  $x \cdot y + x \cdot y' + x' \cdot y$

(b)  $xyz + xyz' + xy'z + xy'z' + x'y'z + x'y'z'$

(c)  $xyz + xyz' + xy'z + x'y'z$

### 12.2 Boolean Algebra

1. (a)

$$\begin{aligned}
 & (a + b') + ac \\
 &= a + ac + b' \\
 &= a + b' \quad \text{because } a + ac = a \text{ by absorption} \\
 & \quad \text{property, Theorem 12.2.11(vii)}
 \end{aligned}$$

- (c)

$$\begin{aligned}
 ab + ab' &= a(b + b') \\
 &= a1 \\
 &= a
 \end{aligned}$$

3.  $(ab + ab') + (a'b + a'b') = a(b + b') + a'(b + b') = a + a' = 1$

7. If  $a = 0$  and  $b = 0$ , then  $a + b = 0 + 0 = 0$ . Suppose  $a + b = 0$ . By Theorem 12.2.11(vii),  $a(a + b) = a$ . Thus,  $a = a(a + b) = a0 = 0$ . Similarly,  $b = b(a + b) = b0 = 0$ .

15. No

### 12.3. Logical Gates and Combinatorial Circuits

- 1.

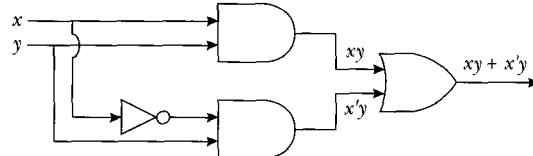


Figure Ch12Sec3Ex1(a)

- 3.

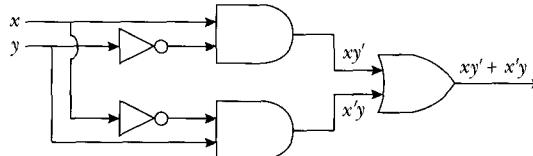


Figure Ch12Sec3Ex3

7. 1

9. 0

11. (a) The Boolean expression corresponding to the first circuit is  $(x + y)y'$ , and the Boolean expression corresponding to the second circuit is  $x'y$ . The input-output tables corresponding to these circuits are:

	$x$	$y$	$x + y$	$y'$	$(x + y)y'$
Row 1	1	1	1	0	0
Row 2	1	0	1	1	1
Row 3	0	1	1	0	0
Row 4	0	0	0	1	0

	$x$	$y$	$x'$	$x'y$
Row 1	1	1	0	0
Row 2	1	0	0	0
Row 3	0	1	1	1
Row 4	0	0	1	0

From the input-output tables it follows that for  $x = 1$  and  $y = 0$ ,  $(x + y)y' \neq x'y$ . Hence, the given circuits are not equivalent.

- (b) The Boolean expression corresponding to the first circuit is  $(x + y)x$ , and the Boolean expression corresponding to the second circuit is  $xy + x$ . The input-output tables corresponding to these circuits

are:

	$x$	$y$	$x + y$	$(x + y)x$
Row 1	1	1	1	1
Row 2	1	0	1	1
Row 3	0	1	1	0
Row 4	0	0	0	0

	$x$	$y$	$xy$	$xy + x$
Row 1	1	1	1	1
Row 2	1	0	0	1
Row 3	0	1	0	0
Row 4	0	0	0	0

From the input-output tables it follows that for each assignment of  $x$  and  $y$ ,  $(x + y)x = xy + x$ . Hence, the given circuits are equivalent.

15. A minimized Boolean expression is  $x + y'$ .  
 17. A minimized Boolean expression is  $xz + x'z'$ .  
 19. A minimized Boolean expression is  $y'z + x'yz'$ .  
 21. A minimized Boolean expression is  $x'z + xyz'w' + x'y'$ .

## CHAPTER 13: FINITE AUTOMATA AND LANGUAGES

### 13.1 Finite Automata and Regular Languages

1. (a) The state diagram is shown in the accompanying figure.

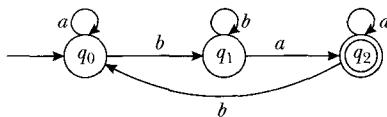


Figure Ch13Sec1Ex1

- (b)  $abaa, bbaa, bababa$
7. (a) The initial state is  $q_0$ .  
 (b) The final state is  $q_4$ .  
 (c) 11 and 0011.  
 (d) Set of all strings of the form  $0^{2n}11$ ,  $n \geq 0$ .
9. (a) Set of all strings on  $\{0, 1\}$  that contain an even number of 1's.  
 (b) Set of all nonempty strings on  $\{0, 1\}$ .  
 (c) Set of all strings on  $\{a, b\}$  that contains an odd number of  $a$ 's or an odd number of  $b$ 's.

15.

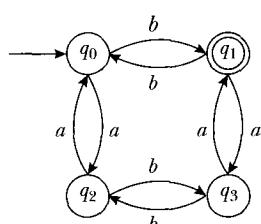


Figure Ch13Sec1Ex15

21. The language accepted by this DFA is the set of all strings that end with 101.  
 33. The required NDFA is shown in the accompanying figure.

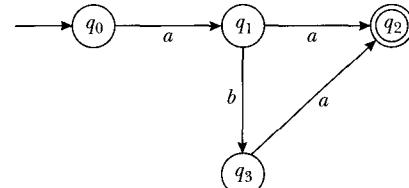


Figure Ch13Sec1Ex35

37. The corresponding DFA  $M^d = (Q^d, \Sigma, q_0^d, \delta^d, F^d)$ , where  $Q^d = \mathcal{P}(Q) = \{\emptyset, \{q_0\}, \{q_1\}, \{q_0, q_1\}\}$ ,  $q_0^d = q_0$ ,  $F^d = \{\{q_1\}, \{q_0, q_1\}\}$ , and  $\delta^d : Q^d \times \Sigma \rightarrow Q^d$  is defined by the following transition table.

$\delta^d$	0	1
$\emptyset$	$\emptyset$	$\emptyset$
$\{q_0\}$	$\{q_0, q_1\}$	$\emptyset$
$\{q_1\}$	$\emptyset$	$\{q_0, q_1\}$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_1\}$

## 13.2 Finite State Machines with Input and Output

1. (a) The transition diagram of  $M$  is shown in the accompanying figure.

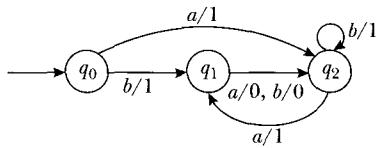


Figure Ch13Sec2Ex1

- (b) 11011  
(c) 111010  
(d) 1111101  
(e) 10110101

3. (a)

	$\delta$		$\gamma$	
	$a$	$b$	$a$	$b$
$q_0$	$q_1$	$q_3$	0	0
$q_1$	$q_3$	$q_2$	0	1
$q_2$	$q_2$	$q_2$	1	1
$q_3$	$q_4$	$q_3$	1	0
$q_4$	$q_4$	$q_4$	1	1

- (b) 0011111 (c) 0001111 (d) Yes

- 9.

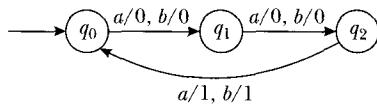


Figure Ch13Sec2Ex9

## 13.3 Grammars and Languages

3. (a)

$$\begin{aligned}
 S &\Rightarrow Ab && \text{by the rule } S \rightarrow Ab \\
 &\Rightarrow aAbb && \text{by the rule } A \rightarrow aAb \\
 &\Rightarrow aaAbbb && \text{by the rule } A \rightarrow aAb \\
 &\Rightarrow aaaAbbbb && \text{by the rule } A \rightarrow aAb \\
 &\Rightarrow aaa\lambda bbbb && \text{by the rule } A \rightarrow \lambda \\
 &= aaabbbb
 \end{aligned}$$

- (b) The derivation tree of the derivation of (a) is shown in the accompanying figure.

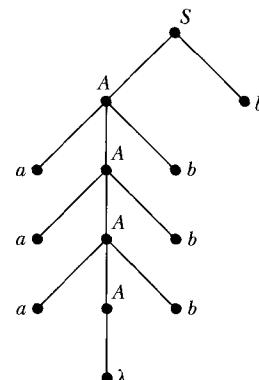


Figure Ch13Sec3Ex3

- (c) The language  $L(G)$ , of  $G$ , is the set of all strings of the form  $a^m b^{m+1}$ ,  $m \geq 0$ .

5.  $L(G) = \{b^n a \mid n \geq 1\}$ .  
 9. This is a CFL, because it is generated by a context-free grammar  $G = (V_N, \Sigma, P, S)$  with  $V_N = \{S, A\}$ ,  $\Sigma = \{a, b\}$ ,  $P = \{S \rightarrow a, S \rightarrow ab, S \rightarrow b\}$ .  
 11. Yes  
 13. Consider the context-free grammar  $G = (V_N, \Sigma, P, S)$  with  $V_N = \{S, A, B\}$ ,  $\Sigma = \{a, b\}$ ,  $P = \{S \rightarrow aA, A \rightarrow aA, A \rightarrow \lambda, A \rightarrow bB, B \rightarrow bB, B \rightarrow \lambda\}$ . Clearly this is a regular grammar and  $L(G) = \{a^n b^m \mid n > 0, m \geq 0\}$ .  
 21. Let  $L = \{a^n \in \{a, b\}^* \mid n \text{ is a positive integer}\}$ . Consider the context-free grammar  $G = (V_N, \Sigma, P, S)$  with  $V_N = \{S, A\}$ ,  $\Sigma = \{a, b\}$ ,  $P = \{S \rightarrow aS, S \rightarrow a\}$ . This is right linear grammar. Any derivation from  $S$  in  $G$  is of the form

$$\begin{aligned}
 S &\Rightarrow^* a^k S && \text{by repeated application of } S \rightarrow aS \\
 &\Rightarrow a^{k+1} && \text{by } S \rightarrow a
 \end{aligned}$$

$L(G) \subseteq L$ . We can also show that  $L \subseteq L(G)$ . Hence,  $L = L(G)$ .

## R e f e r e n c e s

1. A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.
2. A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *Data Structures and Algorithms*, Addison-Wesley, Reading, MA, 1983.
3. K. Appel and W. Haken, "Every planar graph is four-colorable," *Illinois L. Math.* 21 (1977) 429–567.
4. C. Berge, *Graphs and Hypergraphs*, North-Holland, Amsterdam, 1979.
5. K. P. Bogart, *Introductory Combinatorics*, Pitman, Boston, 1983.
6. J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, Elsevier Science, New York, 1979.
7. G. Boole, *The Laws of Thoughts*, reprinted, Dover, New York, 1951.
8. G. Booch, *Object-Oriented Analysis and Design*, 2nd ed., Addison-Wesley, Reading, MA, 1995.
9. R. C. Bose and R. J. Nelson, "A sorting problem," *J. Assoc. Computing Machinery* 9 (1962): 282–296.
10. D. M. Burton, *Elementary Number Theory*, 3rd ed., McGraw-Hill, New York, 1997.
11. R. C. Buck, "Mathematical induction and recursive definitions," *American Mathematical Monthly* 70 (1963): 128–135.
12. C. L. Cheng and R. C. Lee, *Symbolic Logic and Mechanical Theorem Proving*, Academic Press, New York, 1973.
13. E. F. Codd, "A relational model of data for large shared databanks," *Comm. ACM* 13 (1970): 377–387.
14. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed., MIT Press and McGraw-Hill, New York, 2001.
15. P. Cull and E. F. Ecklund, "Towers of Hanoi and analysis of algorithms," *American Mathematical Monthly* 92 (1985): 407–422.
16. C. J. Date, *An Introduction to Database Systems*, 6th ed., Addison-Wesley, Reading, MA, 1995.
17. J. Edmonds and R. M. Karp, "Theoretical improvements in algorithmic efficiency for network flow problems," *J. Assoc. Computing Machinery* 19 (1972): 248–264.
18. S. Even, *Algorithmic Combinatorics*, Macmillian, New York, 1973.
19. S. Even, *Graph Algorithms*, Freeman, New York, 1984.
20. L. R. Ford and D. R. Fulkerson, *Flows in Networks*, Princeton University Press, Princeton, N.J., 1962.
21. L. R. Ford and D. R. Fulkerson, "Maximal flow through a network," *Canad. J. Math.* 18 (1956): 399–404.
22. P. A. Fowlers, "The Königsberg bridges—250 years later," *American Mathematical Monthly* 95 (1988): 42–43.

23. A. D. Friedman and P. R. Menon, *Theory and Design of Switching Circuits*, Computer Science Press, Rockville, MD, 1975.
24. J. A. Gallian, "Assigning driver's licence numbers," *Mathematics Magazine* 64 (1991): 13–22.
25. J. A. Gallian, "The mathematics of identification numbers," *The College Mathematics Journal* 22 (1991): 194–202.
26. J. L. Goldberg, *Matrix Theory with Applications*, McGraw-Hill, New York, 1991.
27. P. Hall, "On representations of subsets," *J. London Math. Soc.* 10 (1935): 26–30.
28. P. R. Halmos, *Naive Set Theory*, Springer, New York, 1994.
29. F. Harary, *Graph Theory*, Addison-Wesley, Reading, MA, 1994.
30. L. Henken, "On mathematical induction," *American Mathematical Monthly* 67 (1960): 323–337.
31. J. E. Hopcroft and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computations*, Addison-Wesley, Reading, MA, 1979.
32. E. Horowitz, S. Sahni, and S. Rajasekaran, *Computer Algorithms C++*, Computer Science Press, New York, 1997.
33. T. C. Hu, *Combinatorial Algorithms*, Addison-Wesley, Reading, MA, 1982.
34. S. K. Jain and A. D. Gunawardena, *Linear Algebra: An Interactive Approach*, Brooks/Cole, CA, 2004.
35. C. Jordan, *Calculus of Finite Difference*, Chelsea, New York, 1965.
36. D. E. Knuth, *The Art of Computer Programming*, Vols. 1–3, Addison-Wesley, Reading, MA, 1973, 1969, 1973.
37. D. E. Knuth, Algorithms, *Scientific American*, 1977 (April): 63–80.
38. D. E. Knuth, "Algorithmic thinking and mathematical thinking," *American Mathematical Monthly* 92 (1985): 170–181.
39. Z. Kohavi, *Switching and Finite Automata Theory*, 2nd ed., McGraw-Hill, New York, 1978.
40. H. W. Kuhn, "The Hungarian method for assignment problem," *Naval Res. Logist. Quart.* 2 (1955): 83–97.
41. H. R. Lewis and C. H. Papadimitriou, "The efficiency of algorithms," *Scientific American* 1978 (March): 96–109.
42. G. S. Lueker, "Some techniques for solving recurrences," *Computing Survey* 12 (1980): 419–436.
43. D. S. Malik, *C++ Programming: From Problem Analysis to Program Design*, 2nd ed., Course Technology, Boston, 2004.
44. D. S. Malik, *Data Structures Using C++*, Course Technology, Boston, 2003.
45. D. S. Malik and J. N. Mordeson, *Fuzzy Discrete Structures, Studies in Fuzziness and Soft Computing*, Physicia-Verlag, New York, 58: 2000.
46. D. S. Malik, J. N. Mordeson, and M. K. Sen, *Fundamentals of Abstract Algebra*, McGraw-Hill, New York, 1996.
47. T. R. McCalla, *Digital Logic and Computer Design*, Merrill, New York, 1992.
48. E. Mendelson, *Boolean Algebra and Switching Circuits*, Schaum, New York, 1970.
49. R. E Mickens, *Difference Equations*, Van Nostrand Reinhold, New York, 1987.

50. R. Neapolitan and K. Naimipour, *Foundations of Algorithms: Using C++ Pseudocode*, 2nd ed., Jones and Bartlett, Boston, 1998.
51. J. R Newman, “Leonhard Euler and the Koenigsberg bridges,” *Scientific American*, 1953 (July): 66–70.
52. O. Ore, *Graphs and Their Uses*, Mathematical Association of America, Washington, D.C., 1963.
53. N. Pipenger, “Complexity theory,” *Scientific American*, 1978 (June): 114–124.
54. R. C. Read and D. G. Corneil, “The graph isomorphism disease,” *J. Graph Theory* 1 (1977): 339–363.
55. E. Reingold, J. Nievergelt, and N. Deo, *Combinatorial Algorithms*, Prentice-Hall, Englewood Cliffs, N.J., 1977.
56. P. Ribenboim, “Catalan’s conjecture,” *American Mathematical Monthly* 103 (1996): 529–538.
57. P. Rob and C. Coronel, *Database Systems: Design, Implementation, and Management*, 4th ed., Course Technology, Boston, 2000.
58. R. Sedgewick, *Algorithms in C*, 3rd ed., Addison-Wesley, Boston, Parts 1–4, 1998; Part 5, 2002.
59. C. E. Shannon, “A symbolic analysis of relay and switching circuits,” *Trans. Amer. Inst. Electr. Engrs.* 47 (1938): 713–723.
60. R. R. Stoll, *Set Theory and Logic*, Dover, New York, 1979.
61. T. A. Sudkamp, *Languages and Machines: An introduction to the Theory of Computer Science*, Addison-Wesley, Reading, MA, 1996.
62. P. M. Tuchinsky, “International standard book numbers,” *The UMAP Journal* 67 (1985): 41–53.
63. A. Tucker, *Applied Combinatorics*, 3rd ed., Wiley, New York, 1995.
64. D. Wood, *Theory of Computations*, Harper & Row, New York, 1987.

## Special Characters

( $i, j$ )th element, defined, 237  
/(forward slash), 99  
% (percent sign), 99

## A

absorption laws, 37  
absorptivity, laws of, 10  
abstract visualization, 7  
Academy of Science in Berlin, 602  
*Acta Eruditorum*, 279  
acyclic graphs, 704  
addition  
    congruence classes and, 353–355  
    modular representation and, 385–387  
    principles, 417–418, 422–423  
additive inverse, 92  
Adelman, Leonard, 407  
adjacency matrix, 636–640  
adjacent, use of the term, 177, 604  
Agrawal, Manindra, 160  
al-Khwārizmī, Muhammad ibn Mūsā, 74  
Alexandria, 152  
algebra  
    Boolean, 221  
    regular languages and, 837–838  
algorithms  
    analysis of, 548–564  
    binary search, 566–567  
    comparison-based, 574–575  
    complexity functions and, 558  
    computing the factorial, 462–467  
    counting principles and, 462–467  
    described, 73–85  
    graph, 669–682  
    greedy, 670  
    insertion sort, 570–574  
    matrices and, 245  
    merge sort, 575–581  
    multiplication and, 245  
    pseudocode conventions, 75  
    selection sort, 567–570  
    sequential search, 564–567  
    shortest path, 670–678  
    strings and, 326–327  
    time complexity and, 547–600  
    Warshall's, 266–271  
alphabet, defined, 325  
*American Journal of Mathematics*, 237  
analog signals, 111  
AND gate, 795–797, 799, 809–810  
Annals of Mathematics, 91

antisymmetric relations, 207  
Apple, Kenneth, 603  
Applied Data Researchers, 266  
arc

    backward, 750  
    defined, 608  
    forward, 750  
    slack on, 750

argument(s)  
    premises of, 44  
    valid forms, 46–49  
    validity of, 44–49  
Armengaud, Joel, 160  
arrays, 76  
arrow diagram, 176  
assignment operator, 75  
assignment statement, 75  
associative, use of the term, 332  
associative laws, 37, 92  
associativity, 39  
    laws of, 10  
asymptotic, defined, 553  
atoms, defined, 791  
automata. *See* finite automata  
axiomatic approach, 481–483

## B

backward arc, 750  
band, 336  
Bell Laboratories, 771, 811  
belongs to, 3  
Bernoulli, Johann, 602  
big-O notation, 548–564  
biimplication(s), 32, 68–69  
binary code  
    converting decimals to, 116–117  
    converting to decimals, 117–118  
    described, 112, 114  
binary numbers  
    operations on, 118–130  
    two's complement and, 122–130  
binary operations  
    Cayley tables and, 333  
    described, 331–336  
binary relation. *See* relations  
binary searches, 566–567  
binary strings, recurrence relations, 493–495  
binomial coefficients, described, 455–467  
binomial expression  
    defined, 456  
    of order  $n$ , 456  
binomial theorem, 457–459

bipartition, 611  
bit string  
    defined, 18  
    length of, 18  
bits  
    leading, 125  
    weight of, 117  
block of statement, 76  
blocks, of partitions, 189  
Boole, George, 221, 770  
Boolean algebra, 221  
    combinatorial circuits and, 794–818  
    described, 769–823  
    logical gates and, 794–818  
    two-element, 770–785  
Boolean complementation, 771  
Boolean expressions, 771, 776–777  
    involving four variables, 816–818  
    involving three variables, 813–816  
    minimization of, 810–818  
Boolean function, 776, 779–781  
Boolean join, 265  
Boolean matrices, 246–251  
Boolean product, 249–250, 771  
Boolean sum, 771  
Boolean variable, 771  
booleanExpression, 75, 76, 139  
boundary, 685  
Brown, A. Crum, 704  
bubble sort, 79–80  
Buhler, Joe, 91

## C

C++ (high-level language), 2, 98, 99, 326  
Caen University, 477  
Caesar, Julius, 405  
Cambridge Mathematical Journal, 770  
Cambridge University, 7  
cancellation law(s), 93, 100  
Cantor, Georg, 2  
Cantor's paradox, 2  
capacity  
    constraints, 745  
    defined, 744  
    networks and, 744–745, 748–749  
cardinality, 6, 298–310  
Cartesian products, 17–18, 210  
Catalan's number, 728  
Cayley, Arthur, 236, 333  
Cayley table, 236, 333  
Cayley-Hamilton theorem, 236

ceiling functions, 304–307  
**chainedMatrixMultiplication**  
 function, 595–597  
 chains, 208  
 characteristic equation  
   of recurrence relations, 516,  
   522  
 check digits  
   credit cards and, 371–373  
   described, 358–373  
   EAN 13 and, 365  
   unequal interchange of,  
   363–365  
   UPC and, 365  
 child  
   left, 714  
   right, 714  
   vertices, 713, 714  
 Chinese Remainder Theorem,  
   382–384  
 chromatic index, 696  
 chromatic number, 693  
 ciphertext. *See also* cryptography  
 circuits  
   Euler, 644–652  
   special, 644–657  
 classes  
   addition of, 353–355  
   congruences and, 353–355  
   multiplication of, 353–355  
 closed under, use of the term, 332  
 closures, 92, 192–200  
   described, 192–200, 257–276  
   reflexive, 193  
   symmetric, 193  
   transitive, 195–196, 265–271  
 CNF (conjunctive normal form), 781  
 codomain, defined, 282  
 coefficients  
   constant, 512  
   recurrence relations and, 512  
 College de France, 278  
 collisions, hashing functions and,  
   393–394  
 coloring  
   edge, 695  
   graph, 692–698  
   proper vertex, 692  
   vertex, 692  
 combinations  
   described, 442–445  
   generalized, 448–452  
   generating, 469–475  
   next largest  $r$ , 472  
 combinatorial circuits, described,  
   769–823  
 comments, 77  
 commutative, use of the term, 332,  
   334  
 commutative laws, 10, 37, 92  
 comparison tree, 574–575  
 comparison-based sort algorithms,  
   574–575  
 complement, 15, 220, 785  
 complementary event, 479–480  
 complete bipartite graph, 611

complex numbers, described,  
   876–877  
 complexity functions, 557  
 components, subgraphs as, 624  
 composition, 181–185  
   of functions, 289  
 compound event, 479–480  
 compound interest, 495  
 concatenation, 325, 826–827  
 conclusion, 44  
 condition, 31  
 congruence modulo, 137  
 congruences  
   check digits and, 358–373  
   Chinese Remainder Theorem  
    and, 382–384  
   classes, 353–355  
   defined, 837  
   described, 341–413  
   divisibility tests and, 349–353  
   hashing functions and, 390–396  
   left/right, 837  
   linear, 378–382  
   modular integer representation  
    and, 384–387  
   properties of, 342–349  
   regular languages and, 837–840  
   round-robin tournaments and,  
    388–390  
   theorems, 401–409  
 conjunction, 28–29  
 conjunctive addition, 48  
 conjunctive simplifications, 47  
 constant polynomial, 81  
 contradiction, 35  
   method of, 94, 96  
   proof by, 67–68  
 contrapositive, 31  
 control structures, 75–76  
 converse, 31  
 correctness, of programs, 138–142  
 countable sets, 309  
 counterexample, 60  
 counting principles  
   addition principles, 417–418,  
    422–423  
   basic, 416–427  
   binomial coefficients and,  
    455–467  
   combinations and, 442–445,  
    448–452, 469–475  
   computing the factorial,  
    462–467  
   described, 415–488  
   discrete probability, 477–484  
   multiplication principle,  
    418–423  
   permutations and, 438–440,  
    448–452, 469–475  
   pigeonhole principle, 431–435  
   principle of inclusion–  
    exclusion, 423–427  
 covers, use of the term, 212  
 Crandall, Richard, 91  
 Cray I supercomputer, 150  
 credit cards, check digits, 371–373

cryptography. *See also* encryption  
   described, 405–409  
   RSA, 406–409  
 cycles, 619–631

**D**

data types, strings and, 326  
 decimal representation  
   converting, to binary code,  
    116–118  
   described, 114–115  
 decomposition, 623  
 decryption keys, 406  
 degree, 81  
 degree of nilpotency, 256  
 DeMorgan’s laws, 15, 16, 37, 39, 58  
 denying, method of, 47  
 derivation tree, 865  
 DFA (deterministic finite automata),  
   827–835  
 diagonal, 18  
 dictionary order, 210  
 difference, matrices and, 241  
 digital signals, 111  
 digraphs. *See also* graph  
   representations  
    defined, 177–178, 608–612  
    of posets, 210–212  
 Dijkstra, Edsger Wybe, 671–678  
 dilemma, 47  
 Diophantine equation  
   defined, 163  
   integral solutions of, 167  
   in two variables, 163  
   linear, 163–167  
 Diophantus, 163  
 directed arc, 177  
 directed edges  
   defined, 177  
   loops and, 178  
 directed walk, label of, 854  
 Dirichlet, Johann, 277, 278  
 disjoint, 9  
 disjunction, 30  
 disjunctive additions, 47  
 disjunctive syllogisms, 47  
 disproof, 60  
*Disquisitiones Arithmeticae* (Gauss), 278,  
   341  
 distributive lattices, 218–220  
 distributive laws, 37, 92  
 distributivity, 39  
   laws of, 10  
 div operator, 98–99  
 divisibility, 99–101  
 divisibility tests, 349–353  
 division algorithm, 95–99, 103, 112  
 divisor(s), 99  
   common, 101  
   greatest common, 101–106  
 DNF (disjunctive normal form),  
   777–779  
 do/while loop, 76  
 does not belong to, 3  
 domain, 54, 282

dot product, 368  
 double hashing, 394  
 double negation law, 37, 39  
 duality principle, 775, 787  
 dummy variable, 320

**E**

EAN 13, 365–371  
 École Militaire, 477  
 edges  
     coloring, 695  
     directed, 608  
     parallel, 605  
     sets of, 604, 685  
     subgraphs and, 613  
     weight of, 669  
 elements  
     complementary, 220  
     diagonal, 239  
     greatest, 214  
     least, 214  
     maximal, 214–221  
     minimal, 214–221  
 empty set, 6  
 encryption. *See also* cryptography  
     function, 406  
     keys, 406  
     RSA, 406–409  
 endpoints, 604  
 entry, defined, 237  
 epsilon, 3  
 equal sets, 6  
 equality of sets, 11  
 equivalence class, 188–192  
 equivalence relations  
     described, 183–188  
     equivalence classes and, 188–192  
     partitions and, 188–192  
 equivalent sets, 308  
 equivalent statements, proving, 69–70  
 Eratosthenes, 150, 160  
 errors  
     detection of, 362–365, 369–371  
     ISBNs and, 362–365  
     UPC and, 369–371  
 Euclid, 149, 150, 152  
 Euclidean algorithm, 103–106  
 Euler, Leonhard, 91, 277, 602  
 Euler Circuit algorithm, 650–652  
 Euler circuits, 644–652  
 Euler phi-function, 403  
 Euler trail, 651–652  
 events  
     complementary, 479–480  
     compound, 479–480  
     equally likely, 480  
     mutually exclusive, 480  
 existential generalization (EG), 59  
 existential quantifier, 57  
 existential specification (ES), 59  
 experiment  
     probabilistic, 478  
     random, 478  
     use of the term, 477

explicit formula, for sequences, 490  
 exponents, described, 875  
 expressions

    join of meet, 247–249  
     matrices and, 247–249  
     trees and, 724–726

extended state transition function, 839  
 extensions, 302–304  
 exterior face, 685

**F**

face  
     exterior, 685  
     interior, 685  
 factorial, 140–141  
     computing, 462–467  
     counting principles, 462–467  
 factoring  
     Fermat's factorization method and, 158  
     positive integers, 157–158

Fermat, Pierre de, 90–92, 158–160  
 Fermat's factorization method, 158  
 Fermat's Last Theorem, 90, 91, 158, 160, 278

Fermat's Little Theorem, 150, 401–402, 404

Fibonacci, Leonardo, 491, 498

Fibonacci sequence, 150, 491

Fichte-Gymnasium, 212

final state, 827, 838

finite automata  
     described, 825–874  
     deterministic (DFA), 827–835  
     nondeterministic (NDFA), 838–845  
     regular languages and, 826

finite sets, 6  
     recurrence relations, 495  
     subsets of, 495  
 finite state machines (FSM), 851–857  
 first-order logic, 53–60  
 floor functions, 304–307  
 flow

    augmentation algorithm, 754–760  
     augmenting path, 751  
     conservation, 745, 746  
     defined, 745  
     maximal, 749, 754–760  
     terminology, 746

for loops, 139  
     matrix multiplication and, 584, 595–597

    sequential searches and, 565

formulas, explicit, 490

forward arc, 750

forward slash (/), 99

France, 91

Frederick the Great, 602

free monoid, 336

free semigroup, 336

free variable, 54

*F*-saturated quasipath, 750

FSM (finite state machines), 851–857

full adder, 807–809

functions

    ceiling, 304–307  
     composition of, 289  
     constant, 285  
     described, 277–340  
     domain of, 282  
     eventually nondecreasing, 558  
     extensions and, 302–304  
     floor, 304–307  
     identity, 285  
     images and, 281, 302–304  
     inverse of, 298–302  
     nondecreasing, 558  
     numeric, 284  
     one-one correspondence and, 286–293  
     one-to-one correspondence and, 286–293  
     onto correspondence and, 286–293  
     preimages and, 281, 302–304  
     restriction and, 302–304  
     special, 298–310  
     strictly increasing, 558  
     Fundamental Theorem of Arithmetic, 150, 155–156  
     *F*-unsaturated quasipath, 750

**G**

Gage, Paul, 160  
 Gauss, Carl Friedrich, 342  
 general term, 320  
 Germain, Sophie, 91  
 Germany, 91, 212, 342  
 Goldbach, Christian, 401  
 Goldbach's Conjecture, 27, 401  
 grammar

    context-free, 860  
     described, 860–869  
     left-linear, 866  
     regular, 866  
     right-linear, 866  
     rules of the, 860

graph representations, directed, 177  
*See also* digraphs, graphs, graph theory

graph theory. *See also* graphs

    cycles and, 619–631  
     definitions, 603–614  
     described, 601–702  
     isomorphism and, 661–666  
     notation, 603–614  
     paths and, 619–631  
     special circuits and, 644–657  
     walks and, 619–631

graphs. *See also* graph theory

    acyclic, 704  
     algorithms, 669–682  
     bipartite, 611  
     coloring, 692–698  
     complements of, 613  
     complete, 611  
     connected, 624

defined, 604  
 disconnected, 624  
 Hamiltonian, 652  
 homeomorphic, 690  
 planar, 685–692  
 representation of, in computer memory, 636–641  
 simple, 610  
 sub-, 612–614  
 weighted, 669  
 Great Britain, 91  
 greedy algorithm, 670  
 Guthrie, Francis, 602

**H**

Hajratwala, Nayan, 160  
 Haken, Wolfgang, 603  
 half adder, 805, 809  
 Hamilton, William Rowan, 652–653  
 Hamiltonian cycle, 652–657  
 Hamiltonian graph, 652  
 Harvard University, 266  
 hash address, 391  
 hash tables, 391  
 hashing functions  
     congruences and, 390–396  
     hash tables and, 391  
 Hasse, Helmut, 212–213  
 Hasse diagram, 212–213  
 hexadecimal number system, 112, 115  
 homeomorphic graphs, 690  
 Horner's method, 597  
 hypothetical syllogism, 47, 49

**I**

IBM (International Business Machines), 367, 811  
 idempotency, laws of, 10, 37, 336  
 idempotent semigroup, 336  
 identity, laws of, 10  
 identity elements, 92  
 if statements  
     matrix multiplication and, 595  
 images, 302–304  
     defined, 281  
     direct, 303  
 immediate predecessor, 754  
 implication, 31–32  
 incidence function, 604  
 incidence matrices, 640–641  
 incidence table, 604  
 incidents, defined, 604  
 inclusion-exclusion, principle of, 423–427  
 index  
     chromatic, 696  
     of products, 324  
     of summations, 320–321  
     variables, 321–324  
 induction. *See* mathematical induction  
 inferences, additional rules of, 59–60  
 infinite set, 6

infix notation, 724  
 initial conditions, 492  
 initial state, 827, 838  
 initial vertex, 196  
 inorder sequence, 718–720  
 input alphabet, 827, 838  
 input-output table, 795–796, 799  
 insertion sorts, 570–574  
 integers  
     composite, 149  
     factoring positive, 157  
     modular representation of, 384–387  
     overview of, 92–111  
     prime, 149–162  
     relatively prime, 106  
     representation of, 111–133  
 integral solution, 164  
 interest rates, 495  
 interior face, 685  
 internal vertices, 196  
 International Article Numbering Association EAN, 366  
 intersection, 9  
 inverse, 31, 298–302  
 inverse property, 93  
 inverter, 795  
 ISBN (International Standard Book Number), 359–365  
     described, 359–365  
 isomorphism, 661–666, 690  
     of trees, 708–710, 727–728  
 ISSN (International Standard Serial Number), 364–365  
 Italy, 90  
 iteration, solving recurrence relation by, 499–506

**J**

Java, 2, 98, 99, 326  
 Jesuit College, 278  
 Johns Hopkins University, 237  
 join  
     Boolean, 247, 265  
     matrices and, 247, 265  
     of meet expression, 247–249

**K**

Karnaugh maps, 803, 810–818  
 Karnaugh, Maurice, 811  
 Kayal, Neeraj, 160  
*k*-cycle, 621  
 Kleene star, 827  
 Königsberg bridge problem, 601, 644  
 Kummer, Ernst, 91  
 Kuratowski, Kazimierz, 691–692

**L**

label of the directed walk, 854  
 lambda rule, 860  
 Lamé, Gabriel, 91

language generated, use of the term, 862  
 languages. *See also* grammar, programming languages  
 Laplace, Pierre Simon de, 477  
 lattices  
     defined, 218  
     distributive, 218–220  
 Laurer, George, 367  
 leaf, 713  
 left inverse, defined, 301  
 Leibnitz, Gottfried Wilhelm, 277, 279  
 lexicographic order, 209–210, 470  
 linear probing, 393  
 linearly ordered set, 208  
 lists  
     as arrays, 76  
     finding the largest element in, 581–583  
     finding the smallest element in, 568–569, 581–583  
     insertion sorts and, 570–574  
     merge sort and, 575–581  
     selection sorts and, 568–570  
     swapping elements in, 569  
 logarithms, described, 875–876  
 logical connectives, 33  
 logical gates, 794  
 logically equivalent, 36–37  
 logically imply, 35  
 logically valid, 44  
 loopBody, 139  
 loops  
     algorithm analysis and, 550–551  
     defined, 604  
     directed edges and, 178  
     invariant, 138–142  
     matrix multiplication and, 595  
     sequential searches and, 565  
 lower bound  
     defined, 217  
     greatest, 217  
 lower limit, 320  
 Ludwig, Georg, 279

**M**

machine language, 112  
 Maple, 98  
 mapped, use of the term, 281  
 Master Theorem, 560–561  
 matching  
     described, 627–631  
     perfect, 629  
     problems, 761–762  
 Mathematica, 98  
 mathematical induction, 74  
     described, 133–142  
     principle of, 135–137  
     program correctness and, 138–142  
     second principle of, 137–138  
 Mathematical Institute of the Polish Academy of Sciences, 691  
 mathematical logic, 26–39

mathematical system, defined, 333–334  
 matrices, 238–240  
   adjacency, 636–640  
   Boolean (zero-one), 246–251  
   chain multiplication, 589–597  
   described, 236–257  
   diagonal, 239  
   idempotent, 256  
   identity, 239–240  
   incidence, 640–641  
   multiplication, 583–597  
   nilpotent, 256  
   of relations, 257–276  
   square, 238–239  
   symmetric, 246  
   transpose of, 245–246  
   weight, 670  
   zero, 239  
 Max-flow, min cut theorem, 753–754  
 maximal elements, 214–221  
 Mealy sequential machine, 852  
 meet  
   Boolean, 247  
   matrices and, 247  
 memory  
   binary code and, 112  
   graphs and, 636–641  
   strings and, 326–327  
 Mencke, Otto, 279  
 merge sort, 575–581  
 mergeLists procedure, 579–581  
 Mersenne numbers, 150, 160  
 minimal elements, 214–221  
 minimization problem, 803  
 minLocation function, 570  
 minterm, 776–777  
 MIT (Massachusetts Institute of Technology), 771  
 mod operator, 98–99  
 modular representation  
   of integers, 384–387  
   sum/product of, 385–387  
 modulo  
   congruences and, 379  
   solution is unique, 379–382  
 modus ponens, 47, 48  
 modus tollens, 47  
 monoid  
   defined, 334  
   free, 336  
 multiples, least common, 107–108  
 multiplication  
   chain, 589–597  
   commutative, 243  
   congruence classes and, 353–355  
   matrices and, 242–245, 250–251  
   matrix, 583–597  
   modular representation and, 385–387  
   principle, 418–423  
   scalar, 583  
 multiplicative identity, 92  
 mutually disjoint, 13

**N**

NAND gate, 803–804, 809–810  
 Napoleonic wars, 278  
 Nazi Party, 212  
 NDFA, (nondeterministic finite automata), 838  
 negation, 28, 785  
 networks  
   matching problems and, 761–762  
   single-source single-sink, 744–745  
   s-t cut of, 747–748  
   transport, 744  
   trees and, 703–767  
 New York University, 811  
 Newton, Isaac, 279  
 n-fold Cartesian product, 18  
 nilpotency, degree of, 256  
 nonisomorphic trees, 727–728  
 nonterminal symbols, 860  
 NOR gate, 804, 809–810  
 Normandy, 477  
 NOT gate, 795, 797, 799, 809–810  
 n-place predicate, 55  
 null rules, 860  
 null set, 6

**O**

octal number system, 112, 114  
 omega, defined, 556  
 one's complement, 122  
 one-one correspondence, 286–293  
 one-to-one correspondence, 286–293  
 onto correspondence, 286–293  
 Operation Research Office, 266  
 operators  
   div, 98–99  
   mod, 98–99  
 OR gate, 795–797, 799  
 order  
   dictionary, 210  
   lexicographic, 209–210  
   product partial, 209–210  
 ordered *n*-tuples, 18  
 ordered pair, 17  
 ordered sets  
   linearly, 208  
   partially, 207–221  
   totally, 208  
 output alphabet, 852  
 output function, 852

**P**

pairwise disjoint, 13  
 palindrome, 827  
 parent, vertices, 754  
 partially ordered sets, 207–221  
   described, 207–221  
   digraphs and, 210

Hasse diagrams and, 212  
 lexicographic order and, 209–210  
 minimal/maximal elements and, 214–217  
 posets and, 210, 214–217

partitions, 188–192  
 Pascal, Blaise, 459–462  
 Pascal's identity, 458–459  
 Pascal's triangle, 459–462  
 paths, 619–631  
   defined, 196, 620  
   Hamiltonian, 652  
   length of, 670  
   nontrivial, 620  
   shortest, 670–678  
 Peano, Giuseppe, 3  
 Peirce arrow, 804  
 percent sign (%), 99  
 permutations  
   described, 438–440  
   generalized, 448–452  
   generating, 469–475  
   next largest, 470

Petersen 3-regular graph, 606  
 pigeonhole principle, 431–435  
 plaintext, 405  
 planar representation, 685  
 Plato, 152

Polish notation, 724

polynomials  
   add, 81, 83  
   described, 876  
   multiply, 81, 84  
   operations, 81–85  
   subtract, 81, 83  
 posets, 207, 210–212, 214–217  
 postcondition, 139  
 postfix expressions, 724, 726  
 postorder sequence, 718  
 Poussin, Vallee, 150  
 power set, 7  
 precondition, 139  
 predicates, 54, 58  
 preimages, 281, 302–304  
 preorder sequence, 718  
 Prim's algorithm, 736–738  
 prime number theorem, 150  
 prime numbers  
   described, 149–162  
   factoring positive, 157–158  
   Fundamental Theorem of Arithmetic and, 155–156  
   search for, 150–152  
   uniqueness and, 156–157

Princeton University, 91

print statements, 76  
 printOptimalOrder function, 596–597  
 probability  
   axiomatic approach, 481–483  
   conditional, 483–484  
   discrete, 477–484  
 probe sequence, 393  
 procedures, 77

*Proceedings of American Mathematical Society*, 364

### product

- Boolean, 249–250
- index of, 324
- matrices and, 249–250
- of the terms, 324
- sequences and, 324–325
- symbol, 324

product partial order, 209

### programming languages

- algebraic properties of, 837–838
- finite automata and, 825–874

programs. *See also* programming languages

- correctness of, 138–142
- defined, 139

### proof(s)

- direct, 64–66
- fallacies/errors in, 70
- indirect, 66–67
- techniques, 63–71

proof by direct method, 64

proper subset, 5

propositional function, 54

propositional logic, 53

propositions, 27, 787

- defined, 787
- dual of, 787

pseudocode conventions, 75

Ptolemy I, 152

### pumping lemma

- applications of, 836–837
- described, 834–837

Pythagoras, 89, 90, 149

Pythagorean triple, 89

## Q

quantification, 55

quantifiers, 53–60

quasipath, 750

Queens College, 770

### queues

- described, 679
- front of, 679
- rear of, 679

Quine-McCluskey method, 803

quotient, 97, 99

## R

rabbits on an island problem, 498–499

Ramsey numbers, 614

range, defined, 282

read statements, 76

recMergeSort procedure, 580

### recurrence relations

- characteristic equation of, 516, 522
- described, 489–545
- initial conditions, 492
- linear homogeneous, 512–524

linear nonhomogeneous,

527–541

satisfying, 513

sequences and, 490–499

solution of, 499–506, 513

recursive definition, 492

reduction, 622

regular languages

- algebraic properties of, 837–838

- finite automata and, 826

- pumping lemma for, 834–837

Reign of Terror, 477

relations, 174–200. *See also* functions

- antisymmetric, 207

- arrow diagrams and, 177

- closures and, 192–200

- compatibility with, 215

- constructing new relations from existing, 180–183

- described, 174–200

- domain and range of, 178–183

- empty, 175

- equality, 184

- equivalence, 183–188

- equivalence classes and, 188–192

- matrices of, 257–276

remainder, 97, 99

R-equivalence class, 188–192

restriction, 302–304

return statement, 76

reversal string, 827

right inverse, defined, 301

Rivest, Ronald, 407

root node, 574

roster method, 3

round-robin tournaments, 387–390

Royal Academy, 237

RSA cryptosystem, 406–409

## S

Samos, 90

sample points, 478

sample space

- described, 477

- discrete, 478

Saxena, Nitin, 160

scalar multiplication, 584

Scheffer stroke, 804

searches

- binary, 566–567

- sequential, 564–565

selection sorts, 567–570

semigroup

- free, 336

- idempotent, 336

sequences

- described, 315–327

- explicit formula for, 490

- Fibonacci, 491

- products and, 324–325

- recurrence relations and, 490–499

- special, 318–320

strings and, 325–326

summation and, 320–321

words and, 325–326

sequential search, 78

set-builder method, 4

sets, 2–26

- cardinality of, 298–310

- computer representation of, 18–21

- countable, 309

- described, 2

- equivalent, 308

- operations on, 8–17

- uncountable, 310

Shamir, Adi, 407

Shannon, Claude Elwood, 771

Shannon's Theorem, 771

shortest path algorithm, 670–678

Sieve of Eratosthenes, 150

single valued, use of the term, 281

singleton set, 6

slack, on the arc, 750

Slowinski, David, 160

sorts

- algorithms, 567–581

- insertion, 570–574

- merge, 575–581

- selection, 567–570

- topological, 216

spanning trees, 731–740

Spence, Gordon, 160

St John's College, 237

s-t networks, 744–745

St. Petersburg Academy of Science, 602

standard factorization, 158

start symbol, 860

state diagrams, 829

state transition function, 827, 838

statement, defined, 27

statement formulas, 33–40

- follow logically from, 48

statement logic, 53

statement variables, 33

Strassen's matrix multiplication, 583–597

strings

- described, 315–327

- empty, 325

- length of, 325

- output, 853

- representing, 326–327

- reversal, 827

- sequences and, 325–326

subgraphs, 612–614

subprogram, 77

subscript. *See* index

subset, 5

substitution. *See also* iteration

- principle of, 38

sum. *See also* summation

- matrices and, 240–241

- of-product form, 777–779

- of the terms, defined, 320

summation. *See also* sum

- described, 320–321

general term of, 320  
 index of, 320–321  
 lower limit of, 320  
 symbol, 320  
 upper limit of, 320  
 supercomputers, 150  
 superset, 5  
 swap function, 570  
 Sylvester, James Joseph, 236  
 symmetric difference, 16

**T**

target, defined, 282  
 tautology, 35  
 Taylor, Richard, 91  
 Technical University of Lvov, 691  
 terminal symbols, 860  
 terminal vertex, 196  
 theorems  
     congruences and, 401–409  
     defined, 26, 64  
 theta notation, 556  
 time complexity, algorithms and, 547–600  
 topological ordering, 216, 678–682  
 totally ordered set, 208  
 Tower of Hanoi problem, 496–498, 512  
 trails, nontrivial, 620  
 transition diagrams, 829  
 transition tables, 828  
 transitions  
     finite automata and, 828–830  
     functions and, 830  
 traversal  
     inorder, 717, 719–720  
     postorder, 718  
     preorder, 717  
     tree, 717–720  
 trees  
     analysis of, 723–724  
     binary, 714  
     described, 703–767  
     expression, 724–726  
     full, 715–716  
     height of, 717  
     insertion into, 722  
     isomorphism of, 708–710, 727–728  
     left, 714–715  
     minimal, 735–740  
     nonisomorphic, 727–728  
     ordered, 713  
     right, 714–715  
     rooted, 712–728  
     search, 720–724  
     spanning, 731–740  
     sub-, 714–715  
     traversal of, 717–720  
     trivial, 714  
     weighted, 736

triangles  
     defined, 611  
 Trinity College, 236  
 trivial positive divisors, 149  
 trivial, use of the term, 620  
 truth table, 28  
 truth value  
     defined, 27–28  
     logical, 27–28  
 two's complement, 122–130

**U**

uncountable sets, 310  
 Underground University of Warsaw, 691  
 Uniform Code Council (UCC), 366  
 union, 8  
 uniqueness, 156–157  
 universal  
     generalization (UG), 59  
     quantifier, 55  
     relation, 175  
     set, 7  
     specification (US), 59  
 University of Basel, 602  
 University of Berlin, 2, 278  
 University of Glasgow, 691  
 University of Halle, 2  
 University of Helmstedt, 342  
 University of Kiel, 212  
 University of Leipzig, 279  
 University of London, 237  
 University of Maryland, 367  
 University of Michigan, 771  
 University of Toulouse, 158  
 UPC (Universal Product Code), 365–371  
 UPC-A (Universal Product Code A), 365–371  
 upper bound  
     defined, 217  
     least, 217  
 upper limit, 320  
 U.S. Army, 266

**V**

variables  
     bound, 57  
     congruences and, 378  
     index, 321–324  
 Venn diagram, 7–9, 14, 15  
     complement of a set, 15  
     difference of sets, 14  
     intersection of sets, 9  
     union and intersection of three sets, 12  
 Venn, John, 7

vertices  
     child, 713, 714  
     connected, 623  
     defined, 177  
     distance between, 625  
     endpoints, 604  
     even (odd) degree, 606  
     in-degree of, 609  
     initial, 196, 619  
     internal, 196, 713  
     isolated, 605  
     M-saturated, 629  
     M-unsaturated, 629  
     out-degree of, 609  
     parent, 754  
     sets of, 604  
     starting, 609  
     subgraphs and, 612  
     terminal, 196, 713  
     terminating, 609

**W**

walks  
     closed, 620  
     described, 619–631  
     directed, 196, 619–620  
     nontrivial, 620  
     open, 620  
     sub-, 621  
     vertices of, 196  
 Warshall, Stephen, 266  
 Warshall's algorithm, 266–271  
 weight matrix, 670  
 weighted trees, 736  
 weighting vector, 371  
 well defined, use of the term, 281  
 well-formed formula, 33  
 well-ordering principle, 94–95, 102  
 while loops, 75, 104, 139, 550–551  
 while statement, 139  
 Wiles, Andrew, 91–92  
 words  
     empty, 325  
     sequences and, 325–326  
 World War II, 212

**Y**

yield of the derivation tree, 865

**Z**

zero-one (Boolean matrices), 246–251

# Photo Credits

## Chapter 1

Cantor, © CORBIS

Venn, *Courtesy of The Master and Fellows of Gonville and Caius College, Cambridge*  
Al-Khowarizmi, *No credit*

## Chapter 2

Pythagorus, © Bettmann/CORBIS

Euclid, © Bettmann/CORBIS

Diophantus, *No credit*

De Fermat, *No credit*

Andrew Wiles, *Photo courtesy of Princeton University, Office of Communications*  
Postage meter image, *Courtesy of Chris K. Caldwell, primes.utm.edu*

13 digit EAN code, *Courtesy of Symbol Technologies*

12 digit UPS code, *Courtesy of Symbol Technologies*

## Chapter 3

Helmut Hasse, *Photo Deutsches Museum Munchen*

Edgar Codd, *Courtesy of National Academy of Engineering*

## Chapter 4

Arthur Cayley, © Hulton-Deutsch Collection/CORBIS

JJ Sylvester, © Bettmann/CORBIS

Stephen Warshall, *Courtesy of Robert M. McClure*

## Chapter 5

Johann Wilhelm Liebniz, *No credit*

Johann Peter Gustave Lejeune Dirichlet, *Courtesy of Staatliche Museen zu Berlin - Preussischer Kulturbesitz*

## Chapter 6

Johann Carl Friedrich Gauss, *No credit*

George Laurerm, *Courtesy of George Laurer*

RSA-Rivest, Shamir, Adelman, *Source: www.usc.edu/dept/molecular-science*

## Chapter 7

Blaise Pascal, © Archivo Iconografico, S.A./CORBIS

Pierre Simon de Laplace, © Bettmann/CORBIS

## Chapter 10

Leonhard Euler, © Bettmann/CORBIS

Sir William Rowan Hamilton, *No credit*

Edsger Wybe Dijkstra, Photo © 2002 Hamilton Richards

Kazimierz Kuratowski, *Courtesy of the Institute of Mathematics of the Polish Academy of Sciences*

## Chapter 12

George Boole, © CORBIS

CE Shannon, *Courtesy of AT&T Shannon Labs*

$n_v$	number of vertices	687
$n_e$	number of edges	687
$n_f$	number of faces	687
$\chi(G)$	chromatic number of $G$	693
$\Delta(G)$	$\Delta(G) = \max\{\deg(v_i) \mid i = 1, 2, \dots, n\}$	694
$\chi'(G)$	chromatic index of $G$	696

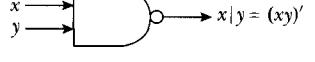
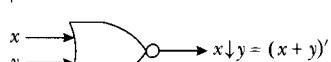
## Trees and Networks

---

$L_v$	vertex set of the left subtree of $v$	714
$R_v$	vertex set of the right subtree of $v$	714
$V(T)$	vertex set of a spanning tree $T$	734
$E(T)$	edge set of $T$	734
$head(e)$	head of arc $e$	743
$tail(e)$	tail of arc $e$	743
$N = (V, E)$	$s-t$ network	744
$C(e)$	capacity of arc $e$	744
$C(v_i v_j)$	capacity of the arc $v_i - v_j$	744
$C_{ij}$	capacity of arc $v_i - v_j$	744
$C_{v_i v_j}$	capacity of arc $v_i - v_j$	744
$out(v)$	set of arcs going out of $v$	744
$in(v)$	set of arcs going into $v$	744
$A(X, Y)$	set of arcs $e$ such that $tail(e) \in X$ and $head(e) \in Y$	745
$F : E \rightarrow \mathbb{N} \cup \{0\}$	flow in a network	745
$F(e)$	flow in arc $e$	745
$F(v_i v_j)$	flow in edge $v_i v_j$	745
$F_{ij}$	flow in edge $v_i v_j$	745
$F_{v_i v_j}$	flow in edge $v_i v_j$	745
$\sum_{e \in in(v)} F(e)$	flow into vertex $v$	745
$\sum_{e \in out(v)} F(e)$	flow out of vertex $v$	745
$\sum_{v_i \in V} F_{ij}$	flow into vertex $v_j$	745
$\sum_{v_i \in V} F_{ji}$	flow out of vertex $v_j$	746
$A(V_t, V_t)$	$s-t$ cut of a network $N$	747
$i(e)$	slack on the arc $e$	750
$i(Q)$	slack of the path $Q$	750
$p(v)$	parent of $v$	754
$value(v)$	value of $v$	754
$label(v)$	lable of $v$	754

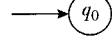
## Boolean Algebra and Combinatorial Circuits

---

$\alpha(x_1, x_2, \dots, x_n)$	Boolean expression in $n$ distinct variables	772
$\alpha^d$	dual of $\alpha$	775
$(B, +, \cdot, ', 0, 1)$	Boolean algebra	786
	Not gate or inverter	795
	AND gate	795
	OR gate	795
	NAND gate	803
	Scheffer stroke	804
	NOR gate	804
↓	Peirce arrow	804

# Finite Automata and Languages

---

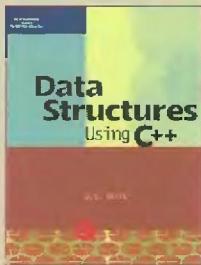
$\Sigma$	alphabet	826
$\Sigma^+$	set of all nonempty strings on $\Sigma$	826
$\lambda$	empty string	826
$\Sigma^*$	set of all strings on $\Sigma$	826
$ w $	length of the string $w$	826
$uv$	concatenation of $u$ and $v$	826
$XY$	concatenation of $X$ and $Y$	826
$X^n$	concatenation of $X$ with itself $n$ times	826
$X^*$	$X^* = \cup_{i=0}^{\infty} X^i$ , Kleene's star of $X$	827
$w^R$	reversal string or reversal of $w$	827
$M = (Q, \Sigma, q_0, \delta, F)$	deterministic finite automaton (DFA)	827
$\delta$	$\delta : Q \times \Sigma \rightarrow Q$ , state transition function	827
	starting state	829
	an accepting state	829
$G_M$	transition diagram of $M$	829
$\delta^*$	$\delta^* : Q \times \Sigma^* \rightarrow Q$ , extended transition function for a DFA	830
$L(M)$	language accepted by a DFA $M$	831
$M = (Q, \Sigma, q_0, \delta, F)$	nondeterministic finite automaton (NDFA)	838
$\delta$	$\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q)$ , state transition function for an NDFA	838
$L(M)$	language accepted by an NDFA $M$	840
$M^d$	DFA corresponding to an NDFA	841
$M = (Q, \Sigma, \Gamma, q_0, \delta, \gamma)$	finite state machine (FSM)	852
$(V_N, \Sigma, P, S)$	context-free grammar	860
$A \rightarrow \alpha$	a rule or production	860
$A \rightarrow \lambda$	null or lambda rule	860
$\alpha_i \mid \alpha_j$	$\alpha_i$ or $\alpha_j$	861
$\alpha_1 \Rightarrow^n \alpha_{n+1}$	derivation of length $n$	861
$\alpha_1 \Rightarrow^* \alpha_{n+1}$	$\alpha_{n+1}$ is derivable from $\alpha_1$	861
$L(G)$	language generated by the grammar $G$	862

# DISCRETE MATHEMATICAL STRUCTURES:

Theory and Applications

D.S. MALIK AND M.K. SEN

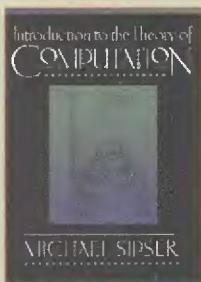
*Discrete Mathematical Structures: Theory and Applications* offers a refreshing alternative for the undergraduate Discrete Mathematics course. In this text, Dr. Malik and Dr. Sen employ a classroom-tested, student-focused approach that is conducive to effective learning. Each chapter motivates students through the use of real-world, concrete examples. Ample exercise sets provide additional practice, while programming exercises in each chapter allow opportunities for computer science application. This new text is a true blend of theory and applications.



**Data Structures Using C++**

0-619-15907-3

D.S. Malik



**Introduction to the Theory of Computation**

0-534-94728-X

Michael Sipser

## FEATURES

- Includes free access to a 120-day trial version of Maple software with every student copy!
- Designed for an undergraduate course in Discrete Mathematics.
- Provides over 100 exercises in each chapter.
- Features Worked-Out Exercises throughout the text designed to demonstrate problem-solving techniques.
- Supplies a rich collection of examples and visual diagrams that clearly define and illustrate key concepts.
- Contains a passcode to access a student Web site that includes additional practice questions, Web links, a solutions manual, and a guide to using Maple with the text.

**maple**<sup>TM</sup>

COURSE  
•com

THOMSON  
COURSE TECHNOLOGY™

Course Technology is part of the Thomson Learning family of companies—dedicated to providing innovative approaches to lifelong learning. Thomson is learning.

Visit Course Technology online at [www.course.com](http://www.course.com)

For your lifelong learning needs, [www.thomsonlearning.com](http://www.thomsonlearning.com)



9 780619 215583

ISBN 0-619-21558-5