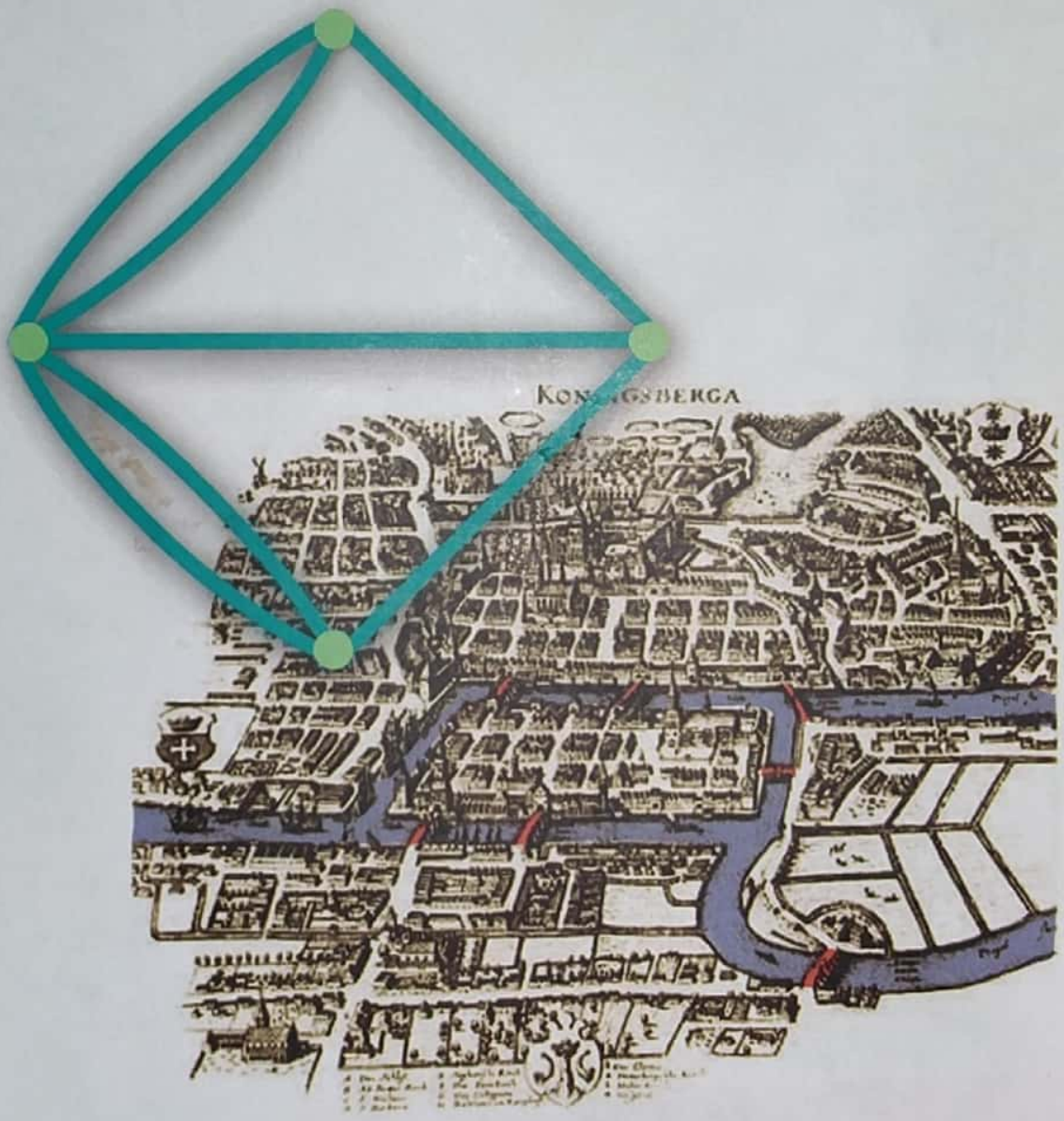
Discrete Mathematics and Discrete Models



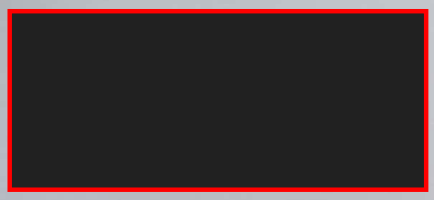
KIMMO ERIKSSON HILLEVI GAVEL

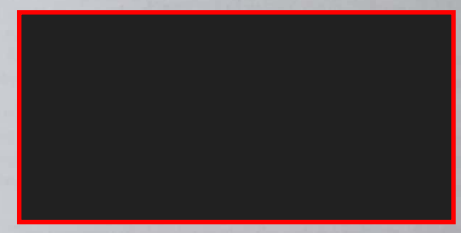
studentlitteratur

Kurs
Erik

Discrete Mathematics and Discrete Models

KIMMO ERIKSSON HILLEVI GAVEL





Studentlitteratur

Art. No. 38939
ISBN 978-91-44-10642-7

Edition 1:2

www.studentlitteratur.se
Studentlitteratur AB, Lund

Cover design by: Francisco Ortega

Printed by Eurographic Danmark A/S, Denmark 2016

Preface

Discrete mathematics is useful mathematics. It's a core subject in study programmes in computer science and is included in many other programmes as well. This book is intended for a first course in discrete mathematics at university level. It doesn't call for any previous knowledge except for high-school mathematics. There is a second part as well, *Diskret matematik, fördjupning* (currently only available in Swedish), which for those who are interested offers exciting contents closely related to current research.

To the Reader To read the book is necessary but not sufficient. To aquire mathematics you have to think for yourself as well when reading. We have tried to write the book in such a way that it will stimulate both reading and thought.

The text is characterised by lots of intersperced exercises. The intention is that these exercises should be done at the same time as the text is read. Perhaps you won't do all the exercises during the first reading, but some at the first go, some at the second one, and so on. Most of the exercises (all those not marked with *) have solutions in the answer section. Irrespective of whether you solve an exercise or choose to skip it you should look at the solution in the answers section before going on. In some exercises, theory is hidden as well. Every exercise of this kind with really important contents is marked ***Important!***.

At the end of each chapter there is a section with more exercises. These are placed under two headings: *routine work* are fairly simple exercises on basic concepts; *to ponder about* are more challenging tasks that may demand creativity, innovative thinking, or application of materials from earlier chapters.

To Teachers. The contents of this book and the second part are designed for the basic and the second courses in Discrete Mathematics at Mälardalen University. If a course is organised in some other way it's of course possible to combine chapters from the two books.

To Everyone Who has Helped. The authors would like to thank everyone who has helped with this book. The hand-drawn illustrations were made in 2001 by Alvin and Vanda Gavel. The computer scientists Kai-Mikael Jää-Aro and Viggo Kann have at different places assisted with their expert competence. In the first edition, eagle-eyed readers found a number of printing errors, that since then have been corrected.

New Edition. The second edition of the book is no dramatic modification of the first edition, but a large number of small improvements have been made. These improvements (comprising everything from layout, text, and exercises to typography and illustrations) have been based on our own experience from using this book and suggestions that others have given to us. Probably some new printing errors have been added as well. Anyone who finds some errors in this second edition is warmly encouraged to report it to the authors.

English version. This translation into English has been made to meet demands from universities that usually give a course in Swedish but occasionaly want to give it in English. The Swedish and English versions of the book are identical in contents and numbering, so the same reading instructions and exercises can be used in both cases. The English version may also be useful for students who participate in a Swedish-speaking course but are more confortable reading in English. At the end of the book there is an English-Swedish glossary.

The translation has mainly been done by Hillevi Gavel, who would like to thank Alvin Gavel for some creative suggestions at tricky places.

Västerås, 12 November 2014

Kimmo Eriksson
Hillevi Gavel

Contents

1 A First Encounter with Discrete Mathematics 1
1.1 What is Discrete Mathematics? . . . 1
1.2 What is Modelling? . . . 2
1.3 Numeracy . . . 4
1.4 Mathematical Presentation and Argumentation . . . 6
1.5 Problem-solving . . . 7
1.6 Mathematical Reading Comprehension . . . 8
1.7 Discrete Mathematics – Tool, Art, and Entertainment! . . . 8

2 Set Theory 11
2.1 The Fundaments of Set Theory . . . 12
2.2 A New Language . . . 15
2.3 New Rules . . . 18
2.4 Relations between the Sizes of Different Sets . . . 22
2.4.1 Unions of Sets . . . 22
2.4.2 Subsets of Subsets . . . 23
2.5 Pairs . . . 24
2.6 Infinite Sets . . . 26
2.6.1 Standard Sets . . . 26
2.6.2 Bijections and Cardinality . . . 28
2.7 More Exercises . . . 32
2.7.1 Routine Work . . . 32
2.7.2 To Ponder About . . . 33

3 Arithmetic 35
3.1 The Division Algorithm . . . 35
3.2 Prime Numbers and Divisors . . . 38
3.2.1 Divisors . . . 38
3.2.2 Prime Numbers . . . 39
3.2.3 The Divisor Graph . . . 41
3.2.4 Common Divisors . . . 42
3.2.5 Diofantine Equations . . . 50
3.3 Modular Arithmetic . . . 54
3.3.1 Calculations in Modular Arithmetic . . . 55
3.3.2 solving in Modular Arithmetic . . . 56
3.4 Number Bases . . . 62
3.4.1 The Binary Number System . . . 62
3.4.2 Other Number Bases . . . 64

3.5 More Exercises . . . 66
3.5.1 Routine Work . . . 66
3.5.2 To Ponder About . . . 68

4 Recursion and Induction 71
4.1 Recursion . . . 71
4.1.1 Recursive Definitions . . . 71
4.1.2 Recursive Number Sequences . . . 74
4.1.3 Recursive Algorithms . . . 78
4.2 Sums and Products . . . 79
4.2.1 Summation . . . 79
4.2.2 Arithmetical and Geometrical Sequences . . . 82
4.2.3 Products . . . 84
4.3 Proof by Induction . . . 85
4.3.1 Introductory Example . . . 85
4.3.2 The Induction Principle . . . 86
4.3.3 Application . . . 87
4.3.4 Proving Inequalities . . . 94
4.4 More Exercises . . . 99
4.4.1 Routine Work . . . 99
4.4.2 To Ponder About . . . 101

5 Combinatorics and Probability 105
5.1 Basic Probability . . . 105
5.1.1 Uniformly Distributed Probability . . . 108
5.1.2 The Addition and Multiplication Principles in Probability 110
5.1.3 Conditioned Probability . . . 113
5.2 Basic Combinatorics . . . 114
5.2.1 The Addition and Multiplication Principles in Combinatorics . . . 114
5.2.2 Permutations and Ordered Selections . . . 117
5.2.3 Unordered Selections and Binomial Numbers . . . 119
5.2.4 Permutations of Multisets . . . 123
5.3 The Pigeonhole Principle . . . 124
5.4 Partitioning of Sets . . . 126
5.4.1 Distribution of Different Objects – Stirling Numbers . 126
5.4.2 Distribution of Identical Objects . . . 127
5.5 Combinatorial Problem-solving . . . 129
5.6 More Exercises . . . 133
5.6.1 Routine Work . . . 133
5.6.2 To Ponder About . . . 136

6 Graph Theory 139
6.1 Basic Concepts in Graph Theory . . . 139
6.2 Euler and Hamilton – two Classical Graph Problems . . . 146
6.2.1 The Background of the Problems . . . 147
6.2.2 The Complexity of the Problems . . . 150
6.3 Isomorphism and Representation of Graphs . . . 151
6.4 Trees . . . 154

6.4.1 Spanning Trees . . . 155
6.5 Rooted Trees . . . 157
6.5.1 Breadth-first and Depth-first Search . . . 158
6.5.2 Binary Trees . . . 160
6.5.3 In-order, Pre-order and Post-order . . . 161
6.6 Three Examples Showing Modelling Using Graphs . . . 164
6.6.1 Scheduling . . . 165
6.6.2 Line Breaking in TEX . . . 166
6.6.3 Instant Insanity . . . 168
6.7 More Exercises . . . 171
6.7.1 Routine Work . . . 171
6.7.2 To Ponder About . . . 174

7 Logic and Boolean Algebra 177
7.1 Reflections on the Language and Meaning of Mathematics . . 178
7.2 Propositional Logic . . . 179
7.2.1 Putting Propositions Together . . . 180
7.2.2 Connectives . . . 180
7.2.3 Rules of Syntax and Calculations in Propositional Logic 185
7.2.4 Satisfiability in Propositional Logic . . . 188
7.3 Boolean Algebra . . . 191
7.3.1 General Boolean Algebra . . . 191
7.3.2 2-valued Boolean Algebra . . . 192
7.3.3 Boolean Functions . . . 192
7.3.4 Some Words about Gate Networks . . . 197
7.4 Predicate Logic . . . 198
7.4.1 Quantifiers and Predicates . . . 199
7.4.2 Truth-Values, and Rules for Syntax and Calculations in Predicate Logic . . . 201
7.4.3 Translating to Predicate-logical Notation . . . 207
7.4.4 Satisfiability in Predicate Logic . . . 210
7.5 Proof Technique . . . 210
7.5.1 Direct and Indirect Proofs . . . 211
7.5.2 Proof Strategies . . . 211
7.6 More Exercises . . . 213
7.6.1 Routine Work . . . 213
7.6.2 To Ponder About . . . 215

8 Relations and Functions 217
8.1 Relations . . . 218
8.1.1 Different Ways of Representing Relations . . . 219
8.1.2 Relations Between Sets . . . 221
8.1.3 Composite Relations . . . 221
8.1.4 Interesting Properties of Relations . . . 223
8.1.5 Special Kinds of Relations . . . 227
8.2 Functions . . . 230
8.2.1 Composite Functions . . . 232
8.2.2 Interesting Properties of Functions . . . 233
8.2.3 Inverses of Functions . . . 236

8.2.4 The Number of Functions of Different Kinds 238
8.3 More Exercises 239
8.3.1 Routine Work 239
8.3.2 To Ponder About 240

9 Artificial Languages and Finite-State Machines 243
9.1 Finite-State Machines 244
9.1.1 Mealy Machines 244
9.1.2 Automata for String Matching 248
9.1.3 Acceptors 249
9.2 Languages 250
9.2.1 Natural Languages and Artificial Languages 250
9.2.2 Regular Languages 252
9.3 More Exercises 257
9.3.1 Routine Work 257
9.3.2 To Ponder About 258

Answers 259
Chapter 1 259
Chapter 2 259
Chapter 3 263
Chapter 4 271
Chapter 5 277
Chapter 6 286
Chapter 7 298
Chapter 8 307
Chapter 9 316

English-Swedish Glossary 321

Index 325

1 A First Encounter with Discrete Mathematics

This chapter will present *discrete mathematics* regarded as a subject, but besides that we'll cover a number of things that are relevant when studying mathematics of any kind, not just this subarea. Hopefully, you'll then get aquainted with the thoughts that lie behind this book. Do return to this chapter at times when you are reading later chapters, and see if you have gained new perspectives on the contents.

Highlights from this chapter.

- The *discrete* in discrete mathematics.
- The meaning of *mathematical models*.
- The meaning of *numeracy*.
- The meaning of good *presentation and argumentation*.
- The meaning of *mathematical reading comprehension*.
- Hints for *problem solving*.
- Discussion about different aspects of discrete mathematics.

1.1 What is Discrete Mathematics?

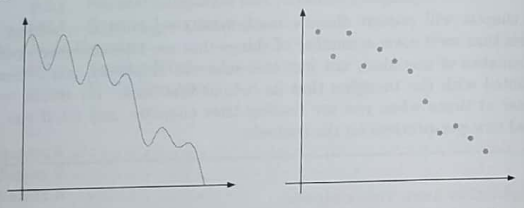
In English, there are two words that sound almost the same. One is *discreet* and means tactful and non-intrusive. *Discreet mathematics* would thus be mathematics that is well brought-up, gray, silent and insignificant. This book is not supposed to have any contents of this kind! Here we find *discrete mathematics* instead – and all the things that this entails in the form of networks of highways, juggling with coloured balls, and people who put on the wrong hats.

This word "discrete" means approximately "separate". **Discrete mathematics** deals with non-continuous phenomena: integers, dots, the truth-values

true and false, lotteries, and lots more.

An everyday example is temperature control on cookers. On electric cookers, each burner plate is normally connected to a knob with six temperature settings. On gas cookers, the control is usually without fixed settings; you can turn the knob any small amount and get a corresponding small change in temperature. The electric cooker has *discrete* temperatures while the gas cooker has a *continuous* range of temperatures.

Example 1.1: ***Stock Prices*** A more commercial example of the differences and interplay between the continous and the discrete is given by stock prices.



The left, continuous, curve may show how the price of a certain share has been changing during 2013. If instead we just measure the average price each month during this time, we get a number of discrete points that are shown in the graph on the right. To go in this way from a continuous situation to a discrete one is usually called to make a **discretisation**. ■

It's not just the case that you can go from continuous to discrete; you can just as well go in the other direction. Some types of problems exist in both a discrete and a continuous variant. Sometimes the discrete one is easier to solve, and then we can choose to use that. In other cases the continuous one is simpler, and then we can take that one instead. Other problems only exist in one of the variants.

> Exercise 1.1 Reflect on whether changes in the stock market fundamentally constitute a discrete or a continuous process. Do the prices change continuously or only at certain separate points in time? *

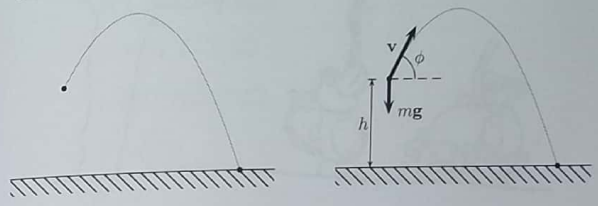
> Exercise 1.2 Three knotty philosophical problems to contemplate: Is matter discrete or continuous? (That is to say: can matter be divided anywhere or does it consist of indivisible parts?) Is space discrete or continuous? Is time discrete or continuous? *

1.2 What is Modelling?

Reality is complicated and difficult to make calculations about. By a **mathematical model** of reality is meant a simplified description where only the

most important things have been included and stated in purely mathematical terms.

Example 1.2: ***Shot Put*** In which angle to the ground should a shot putter throw his shot?



A mathematical model of shot-putting can be that the shot is regarded as a point mass that is thrown with the speed $\mathbf{v}$ from the height h at the angle ϕ to the ground, and that the shot during its passage through the air is affected only by gravity (no air resistance). The throw has ended when the shot has reached the height of zero. Given this model the length of the throw can be calculated by solving a simple differential equation. ■

> Exercise 1.3 Is the same model suitable for analysis of stone-throwing? Explain!

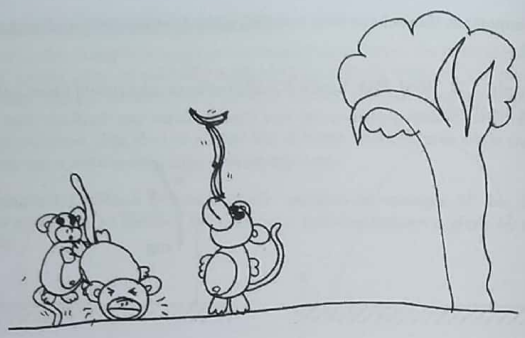
> Exercise 1.4 Is this model suitable for analysis of discus-throwing? Explain!

Note that modelling isn't the same thing as problem solving. Above, we never solved the problem about the angle at which the shot putter should aim – we formulated a model in which we later would be able to solve the problem.

What does the solution to a problem in a mathematical model tell us about the solution to the problem in reality? To this question there is no general answer. The more the model is tied to reality the better the answer ought to be, but the risk is that a model that is too advanced makes the problem impossible to solve at all. You simply need to use critical thinking to evaluate models and the validity of solutions.

The most basic mathematical model comprises the integers themselves. Take the number three, for instance. The number three can't be found anywhere where you can point at it in reality. It's thus a model that humans have introduced. The model comprehend the fact that three bananas and three monkeys have something important in common: if you pair the bananas and the monkeys no banana or monkey is left over.

One obvious flaw in the model above is that in reality some monkey may eat its banana and then they aren't the same number anyway!



Modelling is a means for making knowledge *transferable*, by which is meant that you can use your knowledge in new situations. The model should take a new situation and describe it using mathematical concepts that you already master. The important goals of this book and the exercises in particular, are to give you transferable knowledge in discrete mathematics, experience of modelling, and training in critical thinking.

Exercise 1.5 Reflect on whether you have practised modelling previously, and ask persons near you what use they have of being able to model. *

1.3 Numeracy

Mastery of the common rules of calculations is necessary in almost all science and technology. They are needed as well to make you able to make calculations on food, housing, and money. We will assume that you know how to work with the four rules of arithmetic using both numbers and letters, and how to use parentheses, square roots, and powers. The ability to carry out calculations correctly according to the rules is usually called *numeracy*.

Example 1.3: ***Simplification*** You should be able to simplify an expression such as

$$f(x) = \frac{-\big((1-3)^9 - x\big)}{2 \cdot 3 \cdot 4^2 \cdot 5 + (4\sqrt{2} + \sqrt{x})^2 - 8\sqrt{2x}}$$

where x is a positive number. Check that you understand every step in the simplification below and note in the margin which rules of calculation it is

that have been used.

$$\begin{aligned} f(x) &= \frac{-\big((-2)^9 - x\big)}{2 \cdot 3 \cdot 4^2 \cdot 5 + (4^2 \cdot 2 + x + 8\sqrt{2}\sqrt{x}) - 8\sqrt{2x}} \\ &= \frac{-(-2^9 - x)}{2 \cdot 3 \cdot 4^2 \cdot 5 + 4^2 \cdot 2 + x} \\ &= \frac{2^9 + x}{2 \cdot 4^2 \cdot (3 \cdot 5 + 1) + x} \\ &= \frac{2^9 + x}{2 \cdot (2^2)^2 \cdot (15 + 1) + x} \\ &= \frac{2^9 + x}{2^5 \cdot 16 + x} \\ &= \frac{2^9 + x}{2^9 + x} \\ &= 1 \end{aligned}$$

■

Exercise 1.6 Simplify the expression

$$3 - \sqrt{\frac{y^2}{x^2}} + \frac{x^3 - (xy^2 + 3x - y)}{x}$$

where x and y are positive numbers.

Exercise 1.7 Put together an exercise of the same kind as above by going backwards, that is to say: start with a simple answer (say 17) and reformulate it into more and more difficult expressions using calculation rules. *

Numeracy can be said to be the ability to follow algebraic rules for manipulation of expressions. In discrete mathematics (especially set theory, logic, combinatorics, and modular arithmetic) you will get a lot of new rules and principles on which to apply your numeracy. Congratulations!

But computers then? Can't they perform the calculations on our behalf so that we don't have to mess about with the rules? Of course. To be able to calculate swiftly and a lot is no important ability to train nowadays; to the greatest extent possible, large calculations should be left to machines which do them both faster and more correctly. The desired numeracy we are talking about here is the one that is a consequence of the fact that you *understand* the mathematics and if needs be can muster up the *carefulness* needed to apply the understanding so that it can be of use.

And being able to simplify may – simplify. Let's say that we are to develop a new method of calculation, or to implement someone else's. Let's say as well that we have found that a certain part is to be performed if $y \neq z$ or if $x = 0$ and if both $x \neq 0$ and $y = z$. Then it's good to know the rules of logic so that one can assure oneself that this part in fact is to be performed *every* time.

Exercise 1.8 Why do the conjugate rule, the rules for quadratic expansion, and the rules for exponents hold? Is there any rule that you have learned by heart but that you actually don't understand why it is correct? *

Exercise 1.9 Reflect on whether you have practised numeracy previously, and ask persons near you what use they have of being able to calculate. *

1.4 Mathematical Presentation and Argumentation

Assume that you have developed a model, solved a problem, worked out a proof, or made some other intellectual mathematical contribution. Now you have to pass on what you have done to others – and this step takes as much care as the earlier ones! However valuable your thoughts may be, a bad presentation can make them seem to be without meaning or even plainly wrong.

Example 1.4: ***Ice-creams*** At an ice-cream stand, n different kinds of ice-cream are sold. Explain why the number of possible ice-cream cones with two scoopes in two different flavours is $1+2+3+\cdots+(n-1)$ (if we don't think that the order in which the scoopes are added to the cone matters).

Suggested Solution 1: The first flavour can be part of $n-1$ cones, and the second one in $n-2$ ones, and so on, and that gives $1+2+3+\cdots+(n+1)$ possibilities.

Suggested Solution 2: The flavour on the top can be combined with the $n-1$ flavours below in the list. More cones than that using this flavour aren't possible. The next flavour can in the same way be combined with the $n-2$ flavours below. (The combination of this one and the one on top has already been counted.) And so on down through the list, down to the next-to-last flavour that can be combined with the last one. If we sum these numbers we get $1+2+3+\cdots+(n-1)$. ■

Exercise 1.10 Compare the solutions. In which way is one of them better than the other? *

Exercise 1.11 Try to express in general terms what makes a mathematical presentation a good one. *

What can be classified as a good mathematical presentation may vary somewhat depending on the target group and purpose, but the main features that separate good presentations from bad ones are really always the same. Some rules of thumb:

- What is unclearly said is often unclearly thought. You are supposed to describe a *clear line of thought.*
- A minimal requirement is *self-critical reflection* on whether you yourself would be able to follow your own presentation (if you didn't already know exactly what was meant).
- The presentation should be *convincing.* Mathematics uses logical arguments! In a concise way every relevant objection from the reader has to be prevented.
- Don't be afraid of using *text* and not just formulas. Formulas are exact, but hard to assimilate without explanatory text. Furthermore, text gives you opportunities for other stylistic features, such as expressions of values, feelings, and humour!
- A *figure* can often help both the author and the reader to keep track of the concepts.

In each chapter there are numerous exercises on presentation and argumentation. Use the opportunity to practise! Good examples to use as models are – we hope – included in the text and in the solutions.

Exercise 1.12 Reflect on whether you have practised presentation and argumentation previously, and ask persons near you what use they have of being able to present and argue. *

1.5 Problem-solving

You are confronted with a mathematical problem that you have never seen before. You haven't got the faintest idea about what to do. What are you to do?

Mathematical problem-solving skills are as a matter of fact something that it's possible to train. And there are rules of thumb that can be kept in your memory – the following are some selections from the bible in problem-solving, Poyla's *How to solve it*:

- Start by *understanding* the problem properly.
- Ensure that you *want* to tackle solving it.
- Consider: have you ever seen some *similar problem* before?
- Draw a *figure.*
- Start by solving a *special case* or a simpler similar problem.
- Try to *generalise* the problem – it may in fact become easier then!
- Are there any *symmetries* that can simplify the problem?

- Try to break down the problem into *sub-problems*.
- Is it possible to *reformulate* the problem in some way?

In serious problem-solving (that isn't done simply for your own amusement) you should use computers, various books, confere with colleagues, and ask experts. To be able to formulate problems, use aids, and understand other persons' solutions is because of this often more important than the ability to solve problems on one's own and without aids.

Exercise 1.13 Reflect on whether you have practised problem-solving previously, and ask persons near you what use they have of being able to solve problems. *

1.6 Mathematical Reading Comprehension

Most mathematics that is written is in English. But mathematics is to a great extent language-independent – it has a language of its own consisting of symbols such as the digits, calculation operators (such as $+$ and $-$), relation operators (such as $=$ and $\leq$), more advanced symbols (such as $\sum$, $\prod$, $\cap$, $\vee$), and structures such as matrices and graphs.

Reading mathematical text takes time. Practice is needed to *really* read, with active thought processes as a necessary component. Reading mathematics without paying attention is simply not possible.

Exercise 1.14 Reflect on whether you have practised mathematical reading comprehension before and ask persons near you what use they have of being able to comprehend mathematics. *

1.7 Discrete Mathematics – Tool, Art, and Entertainment!

We who work professionally with discrete mathematics see three clear aspects of our occupation.

The usability aspect is the one that makes more and more persons learn discrete mathematics; it's simply necessary to have some discrete-mathematical knowledge to be able to understand and deal with some phenomena in the modern world, not least in computer science.

But all mathematics has an *aesthetic and cultural* component. A mathematical truth or a mathematical proof can have a beauty of its own. A frequently cited example of this is Euclid's proof that there is an infinite number of primes. Mathematics is interesting seen as the history of ideas as well.

Lastly, discrete mathematics contains a rich flora of so-called recreational mathematics – problems that one solves for one's own amusement. There

are many of us who think that it's entertaining and relaxing to think about an amusing mathematical problem. Lewis Carroll, author of *Alice in Wonderland*, was also a mathematician and a producer of recreational problems. One of his problems is a pure exercise of logic, and reads:

A logic problem: What question should the princess ask?

A princess visits an island inhabited by two tribes. Members of one tribe always tell the truth, and members of the other tribe always lie.

The princess comes to a fork in the road. She needs to know which road leads to the castle so as to avoid the fire-breathing dragon and rescue the prince from the wizard holding him captive in the castle. (Although the princess doesn't know it, the south road leads to the castle and the north road leads to the dragon.)

Standing at this fork in the road is a member of each tribe, but the princess can't tell which tribe each belongs to. What question should she ask to find the road to the castle?

Exercise 1.15 Philosophy once more: Reflect on similarities and differences between different kinds of mathematics. What is it that makes discrete mathematics into mathematics? What makes it discrete? *

Exercise 1.16 You will see, and most of all formulate yourself, a number of proofs – logically binding arguments – in this book. To get into practice, try to recreate some proof from earlier studies of mathematics. *

Exercise 1.17 Can you make a convincing explanation as to why order is irrelevant when multiplying, that is to say that $a \cdot b = b \cdot a$ for all integers a and b? (To start with, what exactly is meant by multiplication?) *

2 Set Theory

Agenda 21 is the international action plan that has been agreed upon regarding sustainable development. From Agenda 21 the following figure is taken.

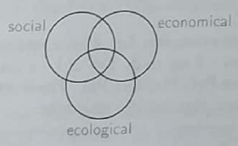


Figure 2.1: *Three dimensions of sustainable development: the social, economical, and ecological dimensions in the concept of sustainable development are, taking the long view, requisites for each other. Ideally, measures that are beneficial from several aspects and doubly profitable are prioritised.*

So the circles illustrate three categories of factors that contribute to sustainable development. That the circles intersect illustrates that some factors belong to two categories and some may even belong to all three. Clearly, this method using circles can be used to describe any kind of categorisation. The mathematical background is given by **set theory**.

Highlights from this chapter.

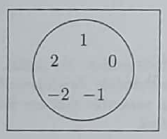
- *Elements*, *sets* and *subsets*.
- *Union* ($A \cup B$), *intersection* ($A \cap B$), and *complement* (A^c) of sets A and B.
- The language and rules of set theory.
- The *product set* $A \times B$ consisting of all pairs made up of one element from set A and one element from set B.
- The standard sets $\mathbb{Z}$ (the integers), $\mathbb{N}$ (the natural numbers) and $\mathbb{Z}_+$ (the positive integers).

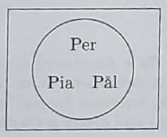
- *Bijections* (pairings of elements from two sets of the same size).
- The concept of *infinity*.

2.1 The Fundaments of Set Theory

A **set** in the mathematical sense is a collection of objects that together make up the set. Set theory counts as a part of discrete mathematics since for each thing in the universe, there are two clearly distinct alternatives: either the thing belongs to a given set or it doesn't. The objects that belong to the set are called its **elements**.

Let's look at two concrete examples: Let A be a set containing five elements, namely the integers from -2 up to 2. Let B be a set containing three elements, namely the names Per, Pia, and Pål. These sets can be described in several ways. For instance, you can use diagrams:





The frame is supposed to contain everything that exists. Inside the circle are the elements of the set; outside is everything else. A diagram of this kind is called a **Venn diagram**.

Another way of describing the sets is to list the included elements one after another:

$$A = \{-2, -1, 0, 1, 2\} \quad \text{and} \quad B = \{\text{Per, Pål, Pia}\}.$$

The important thing is *which* the elements belonging in the set are, not the way used to describe them. For instance, the elements of a set don't have any given order, so we can list the elements in whichever order we like and still describe the same set. For instance,

$$\{\text{Per, Pål, Pia}\} = \{\text{Pia, Per, Pål}\}$$

and

$$\{-2, -1, 0, 1, 2\} = \{0, \pm 1, \pm 2\},$$

because in both cases there are the same elements in the two sets. (Still, it's usually a good idea from a practical point of view to list the elements in a *systematical* way; order of size or alphabetical order can be recommended. But from a strictly mathematical point of view it doesn't matter.)

Nor does it matter how many times we repeat an element in the description – only one copy exists. For instance,

$$\{\text{Per, Per, Per, Pål, Per, Per, Per, Pia}\} = \{\text{Per, Pål, Pia}\}$$

because there are the same three names in the two lists.

A third possible way of describing a set is to specify the properties that are distinctive for the elements belonging to the set. We might describe the set A as

$$A = \{\, n \mid n \text{ is an integer and } |n| \leq 2 \,\},$$

where $|n|$ denotes the absolute value of the number n. This description is to be read as "*A is the set of all n:s such that n is an integer and the absolute value of n is less than or equal to 2.*" The structure

$$\{\, n \mid n \text{ has the property } \ldots \,\}$$

we call the **set builder**.

In ordinary language you may also succeed in expressing yourself without mentioning any names of variables: "*A is the set of all integers with an absolute value of less than or equal to 2.*" But if you have to describe a set using mathematical symbols, having a name for a typical element of the set is very practical. This name is written at the very beginning of the set specification, followed by a vertical line. You can choose an arbitrary name for the variable, like x instead of n:

$$A = \{\, x \mid |x| \leq 2 \text{ and } x \text{ is an integer} \,\}.$$

Still, you should choose a name that sounds natural and makes the expression easy to read. Often n is chosen as the name of a variable denoting an integer, since that is the initial letter of the word *number*.

There may be several ways of describing a set. Our set A could also be described as

$$A = \{\, n \mid n^2 < 5 \text{ and } n \text{ is an integer} \,\}.$$

Exercise 2.1 Find some other way of describing the set A using the set builder.

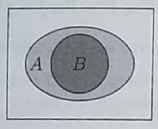
Exercise 2.2 Describe the set B using the set builder on the form $\{\, N \mid N \text{ is a Swedish name that } \ldots \,\}$.

Exercise 2.3 Try to describe the set B without introducing the name of a variable and without listing the elements.

Exercise 2.4 List the elements in the set

$$\{\, n \mid n \text{ is a positive odd number less than } 10 \,\}.$$

If we remove some elements from a set we get a **subset**. The set B in the picture is a subset of the the set A.



For instance, the set of odd numbers is a subset of the set containing all the integers. The set of all children, retired persons, and smokers is a subset of the set of humans.

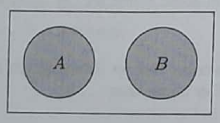
That a set B is a subset of a set A is denoted by $B \subseteq A$. The rigorous formal definition is that all elements in B also belong to A.

Exercise 2.5 We have the sets $A = \{1, 2\}$, $B = \{2, 3\}$, and $C = \{1, 2, 3\}$. Which set is a subset of which?

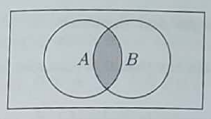
Exercise 2.6 Let A be the set of all retired persons. Let B be the set of all retired smokers. Which set is a subset of which?

Exercise 2.7: *Important!* Suppose that we have discovered that both $A \subseteq B$ and $B \subseteq A$. What conclusions can we make about the relationship between the sets A and B?

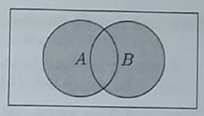
If two sets overlap, so that they have some elements in common, they are said to **intersect** each other. Two sets that don't intersect are said to be **disjoint**. Examples of disjoint sets are { children } and { old age pensioners }, that of course don't have any elements in common. Another example is { odd numbers } and { even numbers }. In the figure below the sets A and B are disjoint.



If two sets intersect, the subset of elements that belong to both sets is called the **intersection** of the sets. The set made up of all elements belonging to one or the other of two sets is called the **union** of those two sets. The diagrams on the facing page show the mental picture you should have of the concepts of intersection and union, respectively.



Intersection: $A \cap B$



Union: $A \cup B$

The union of two sets A and B can, using the set builder, be described as

$$A \cup B = \{ x \mid x \text{ belongs to } A \text{ or } B \}.$$

"Or" is here, as almost everywhere in mathematics, to be interpreted as "and/or", one or the other or both.

Exercise 2.8 Describe the intersection $\cap$ of two sets A and B using the set builder.

Exercise 2.9 We have the sets $A = \{2, 4, 6, 8, 10\}$ and $B = \{3, 6, 9\}$. Work out

(a) the union of A and B. **(b)** the intersection of A and B.

Exercise 2.10 Let S_1 be the set of all languages you master, at least somewhat. Ask some fellow student what languages she masters and let S_2 be the set of those languages. See to it that S_1 and S_2 are different in some way; be creative if you have to!

(a) What is the union of S_1 and S_2? State some situtation where this set is relevant.

(b) What is the intersection of S_1 and S_2? Same question about this set. *

Exercise 2.11 Make a diagram showing the following two sets: The set of your parents and the set of your close female relatives. (As close relatives we here count siblings, parents, grandparents, cousins, aunts, uncles, children, grandchildren, and nieces and nephews.) These two sets intersect, that is, they have some element in common, and the diagram must show this. *

2.2 A New Language

To denote all the operations and concepts in set theory, a whole language of symbols has been developed. Here is a list of the most important ones. Some of them we have already described, but it's nice to have everything assembled in one place.

Definition 2.1: ***Set Theoretical Concepts and Symbols***

$\emptyset$	the empty set that doesn't contain any elements (can also be written as an empty list: $\{\}$).
$\mathcal{U}$	the **universe**, the set of all the available elements.
$A \subseteq B$	the set A is a subset of the set B. (Some books use $A \subset B$.)
$A \not\subseteq B$	A is *not* a subset of B.
$x \in A$	The element x belongs to the set A.
$x \notin A$	x does *not* belong to the set A.
$\lvert A \rvert$	the number of elements in the set A (also called the **cardinality** or simply the size of the set).
$A \cap B$	the intersection of the sets A and B.
$A \cup B$	the union of the sets A and B.
$A \setminus B$	the **difference** between the sets A and B; the elements that belong to A but not to B.
A^c	the **complement set** of the set A, that is $\mathcal{U} \setminus A$. ■

As is the case with any other language, it can take some time to learn to read "set-theoretical" fluently.

Example 2.1 Let the universe $\mathcal{U}$ consist of the digits zero to nine. Form the sets $A = \{0, 1, 2, 3, 4, 5\}$ and $B = \{5, 7, 9\}$.

- We have $A \not\subseteq B$, since A isn't a subset of B as there are elements in A that aren't in B (for instance, 0).
- We have $3 \in A$, since 3 is an element in the set A.
- We have $7 \notin A$, since 7 isn't an element in the set A.
- We have $|A| = 6$, since there are 6 elements in the set A.
- We have $A \cap B = \{5\}$, since 5 is the only element that belongs to both set A and set B.
- We have $A \cup B = \{0, 1, 2, 3, 4, 5, 7, 9\}$, since these are all the elements that belong to A or B.
- We have $A \setminus B = \{0, 1, 2, 3, 4\}$, since these are the elements that belong to A but not to B.
- We have $A^c = \{6, 7, 8, 9\}$, since these are the elements in the universe that don't belong to the set A. ■

Exercise 2.12 Using the same sets A and B as in the previous example, which of the following statements are true?

(a) $B \not\subseteq A$ **(b)** $3 \in B$ **(c)** $7 \notin B$ **(d)** $|B| = 3$

(e) $B \cap A = \{5\}$ **(f)** $B \cup A = \{0, 1, 2, 3, 4, 5, 7, 9\}$

(g) $B \setminus A = \{0, 1, 2, 3, 4\}$ **(h)** $B^c = \{6, 7, 8, 9\}$

Exercise 2.13 For each of the concepts in definition 2.1, specify whether it is a number, a set, or a statement.

Exercise 2.14 Define fours sets as follows: $M_1 = \{$paper, rock, scissors$\}$, $M_2 = \{$rock, wood$\}$, $M_3 = \{$rock, punk, metal, jazz$\}$, and $M_4 = \{$paper, wood, metal, concrete$\}$. Work out the sets

(a) $M_1 \cap M_2$ **(b)** $M_1 \cap M_2 \cap M_3$

(c) $M_2 \cup M_3$ **(d)** $M_1 \cap (M_2 \cup M_3)$

(e) $(M_1 \cap M_2) \cup M_3$ **(f)** $M_1 \setminus M_2$

(g) $M_3 \setminus M_4$ **(h)** $(M_1 \cup M_3) \setminus (M_2 \cup M_4)$

(i) $(M_1 \setminus M_2) \cup (M_3 \setminus M_4)$

Exercise 2.15 Using the same sets as in the previous exercise, is there any way of combining two of the sets so that the intersection is empty? Same question about three of the sets! *

Exercise 2.16

(a) Express the meaning of $A \setminus B$ using the set builder.

(b) Illustrate the meaning using a Venn diagram.

(c) It is possible to express $A \setminus B$ using the other operations. Do this!

Exercise 2.17 Explain why, given arbitrary sets M_1, M_2, M_3, M_4, it is always true that

$$(M_1 \cup M_3) \setminus (M_2 \cup M_4) \subseteq (M_1 \setminus M_2) \cup (M_3 \setminus M_4)$$

Exercise 2.18 Study the subset of the symbols in definition 2.1 on the preceding page where a line through the symbol creates a meaningful negated symbol (like $\not\subseteq$). What is the size of this subset? Can it be described in some other way?

Exercise 2.19 Make up sets A and B that satisfy $|A| = 4$, $|B| = 4$, and $A \setminus B = \{$C++, Pascal, Java$\}$. What sizes must $B \setminus A$, $A \cap B$, and $A \cup B$ be?

2.3 New Rules

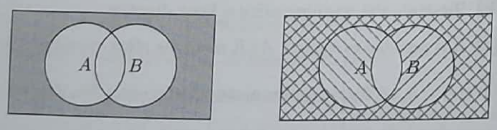
From calculations using the four rules of arithmetic, we are used to the fact that a number of calculation rules hold. These rules – or "laws" – have names that not everyone is familiar with:

- the *commutative* laws of addition and multiplication: $a + b = b + a$ and $a \cdot b = b \cdot a$.
- the *associative* laws of addition and multiplication: $a+(b+c) = (a+b)+c$ and $(a \cdot b) \cdot c = a \cdot (b \cdot c)$.
- the distributive law of addition and multiplication: $a \cdot (b+c) = a \cdot b + a \cdot c$.
- the *identity* law of addition and multiplication: $a + 0 = a$ and $a \cdot 1 = a$.
- the law of *double negation*: $-(-a) = a$.
- the law of *multiplication by zero*: $a \cdot 0 = 0$.

In set theory we have, instead of $+$, $\cdot$ and $-$, just been introduced to some new operations: $\cup$, $\cap$, and ^{c}. These operations must have rules as well. Which these are you can find out by pondering and experimenting.

Example 2.2: ***De Morgan's law 1*** What happens, for instance, if you take two sets A and B, form their union, and finally find the complement of this new set? Will you get the same final result as if you had instead found the complements of the sets separately, and formed the union of those? Let's draw a couple of Venn diagrams to investigate the question:

In the first picture we have marked the area outside $A \cup B$, that is $(A \cup B)^c$. In the second one we have have drawn lines in gray in one direction to mark A^c and in blue in the other direction to mark B^c. The union of those areas is everything that is marked in some way.



$(A \cup B)^c$ and $A^c \cup B^c$ are clearly not the same thing! But the cross-lined area, that belongs to both A^c and B^c, looks right. Thus it seems as if $(A \cup B)^c = A^c \cap B^c$.

An ultra-formal proof of the fact that this is the case looks like this:

Take an element x belonging to $(A \cup B)^c$. x can't belong to A, since if it did it would belong to $A \cup B$ as well, and then it can't belong to the complement of that set. So x has to belong to A^c. In the same way we find that x must belong to B^c. And if x belongs to both A^c and B^c it belongs to the intersection of those sets, that is: x belongs to $A^c \cap B^c$.

So every element that belongs to $(A \cup B)^c$ belongs to $A^c \cap B^c$ as well. Then $(A \cup B)^c$ has to be a *subset* of $A^c \cap B^c$. (It might be possible for $A^c \cap B^c$ to contain other elements besides the ones in $(A \cup B)^c$.)

Now, let's study an element y belonging to $A^c \cap B^c$. It belongs to both A^c and B^c. Could it possibly belong to $A \cup B$? It can't belong to A, since we know that it belongs to A^c. And the elements of $A \cup B$ that don't belong to A belong to B instead. And that we know that y doesn't. So y doesn't belong to $A \cup B$, which means that y belongs to $(A \cup B)^c$.

That means that $A^c \cap B^c$ is a subset of $(A \cup B)^c$.

If two sets are subsets of each other they have to be the same, according to exercise 2.7 on page 14. So $(A \cup B)^c = A^c \cap B^c$, Quod Erat Demonstrandum (Latin for *which was to be proved*). ■

The rule we found and proved here is one of **De Morgan's laws**.

> Exercise 2.20: *De Morgan's Law 2* There are two De Morgan's laws. The other one reads $(A \cap B)^c = A^c \cup B^c$.
>
> **(a)** Illustrate the law using Venn diagrams.
>
> **(b)** Check the law by working out both $(A \cap B)^c$ and $A^c \cup B^c$, given that the universe consists of the integers between 1 and 10, A of the even numbers and B of the numbers greater than 5 in this universe.
>
> **(c)** Explain why part (b) *isn't* a valid proof of the law.
>
> **(d)** Make a formal proof of the law.

If you investigate the new operations in this way you get a whole collection of rules. Some resemble the rules for $+$, $\cdot$, and $-$ that we looked at in the beginning: if we exchange everywhere $+$ for $\cup$, $\cdot$ for $\cap$, $-$ for ^{c}, 1 for $\mathcal{U}$, and 0 for $\emptyset$. (Remember this correspondence between union and addition and between intersection and multiplication, because this principle will reappear in later chapters both when the subject is probability and combinatorics and when it's Boolean algebra!) Others, like De Morgan's laws, don't have any counterparts.

A compilation of the rules for union, intersection, and complement is given in table 2.1 on the next page. (Since difference according to exercise 2.16 on page 17 can be expressed using the other operations, we have not included any of the rules pertaining to that operation.)

If you succeed in interpreting the meaning of the rules in the table, in most cases you just need some common sense to realise that they have to hold. For instance, the first associative law says that $(A \cup B) \cup C = A \cup (B \cup C)$, which follows from the fact that both expressions describe the set of all elements belonging to A or B or C. (A consequence of this is that you can skip

Rules for Union, Intersection, and Complement

associative laws

$$(A \cup B) \cup C = A \cup (B \cup C) \quad (A \cap B) \cap C = A \cap (B \cap C)$$

commutative laws

$$A \cup B = B \cup A \quad A \cap B = B \cap A$$

distributive laws

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C) \quad A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

De Morgan's laws

$$(A \cup B)^c = A^c \cap B^c \quad (A \cap B)^c = A^c \cup B^c$$

idempotence laws

$$A \cup A = A \quad A \cap A = A$$

absorption laws

$$A \cup (A \cap B) = A \quad A \cap (A \cup B) = A$$

double complement

$$(A^c)^c = A$$

inverse laws

$$A \cup A^c = \mathcal{U} \quad A \cap A^c = \emptyset$$

identity laws

$$A \cap \mathcal{U} = A \quad A \cup \emptyset = A$$

dominance laws

$$A \cap \emptyset = \emptyset \quad A \cup \mathcal{U} = \mathcal{U}$$

Table 2.1: *Rules of set theory.*

the parentheses and just write $A \cup B \cup C$. The same thing holds for the intersection of several sets.)

If you find it hard to grasp some rule you can draw Venn diagrams and establish that the two sets in a given law give the same area in the diagram. In a relation including three sets the diagram has to be drawn in the same way as "Agenda 21" on page 11. That way of drawing covers all the possibilities.

And then there is the option of writing a proof using formal reasoning. Some mathematicians only accept that proof method; others think that Venn diagrams with comments suffice.

Exercise 2.21

(a) Illustrate the distributive laws using Venn diagrams.

(b) (If you want to:) Write formal proofs of the laws.

The rules that have been proved can be used in other proofs. If you want to show that two set-algebraic expressions describe the same set, one method is to rewrite one of the expressions step by step, using the rules, into the other one.

Example 2.3: ***Application of Rules*** Show that $(A \cap C^c) \cup (C \cap B)^c = (B \cap C)^c$.

We start on the left-hand side, our aim being to simplify it into the expression on the right-hand side. Since almost none of the rules discuss the situation where there is a complement "outside" a compound expression we start by using De Morgan's law, and then we'll see how to continue:

$$\begin{aligned}
(A \cap C^c) \cup (C \cap B)^c &= (A \cap C^c) \cup (C^c \cup B^c) && \text{De Morgan's law} \\
&= ((A \cap C^c) \cup C^c) \cup B^c && \text{Associative law} \\
&= (C^c \cup (A \cap C^c)) \cup B^c && \text{Commutative law} \\
&= (C^c \cup (C^c \cap A)) \cup B^c && \text{Commutative law} \\
&= C^c \cup B^c && \text{Absorption law} \\
&= B^c \cup C^c && \text{Commutative law} \\
&= (B \cap C)^c && \text{De Morgan's law}
\end{aligned}$$

In a proof of this kind you should never do more in one step than you are able to keep track of, and you should also note which rules you are using (which we have done). If you discover that you have used a "rule" without a name it is time to consider whether you really know what you are doing.

How much you can keep track of in one step in a calculation depends on how experienced you are. An experienced person would for instance skip the first two applications of the commutative law. Their function was to make it possible to use the absorption law – as written in the table, with the absorbing set on the left. With practice, it becomes possible to do steps like this in your head, but initially it's better to write down *everything*.

Note by the way that the sets to which you apply the rules don't have to have the same names as the sets in the collection of rules! When we used the associative law $A \cup (B \cup C) = (A \cup B) \cup C$ on $(A \cap C^c) \cup (C^c \cup B^c)$, then $(A \cap C^c)$ had to act the part of "A", C^c played "B" and "B^c" was "C". ■

Exercise 2.22 Examine the calculation in the example very carefully, and study the way the rules have been applied. *

Exercise 2.23 Show the equalities below using the rules. Write down *every* step.

(a) $(A^c \cup B)^c = A \cap B^c$ **(b)** $(B \cap A) \cup (A^c \cap B) = B$

Exercise 2.24 Here we have studied a whole bunch of rules for calculations. Think of some situation where they can be useful!

2.4 Relations between the Sizes of Different Sets

In many situations, one is mainly interested in *how many* elements there are in a set; exactly which the elements are is of less interest.

2.4.1 Unions of Sets

If A and B are two sets, how many elements are there in their union $A \cup B$? Is the answer perhaps the sum of the number of elements in each of the two sets, $|A| + |B|$? No, usually the union is smaller than that.

As an example we take $A = \{1, 2, 3\}$ and $B = \{2, 3, 4, 5\}$. A has got three elements and B has four elements, but the union of A and B has only five elements: $\{1, 2, 3, 4, 5\}$.

In the sum $|A| + |B|$ the elements in common are counted twice, once as elements in A and once as elements in B. The elements in common make up the intersection of the sets. Thus, $|A \cup B| + |A \cap B|$ as well will be the number of elements in the two sets, the elements in common counted twice. We have thus proved that it is always the case that

$$|A| + |B| = |A \cup B| + |A \cap B|. \tag{1}$$

Exercise 2.25 Check that the equality holds using the sets $A = \{\text{apple, banana}\}$ and $B = \{\text{banana, muesli, yoghurt}\}$.

Exercise 2.26 A register of de-registered cars contains 1812 items. A register of cars that have been automatically photographed for speeding offences contains 1066 items. Together, the two registers contain 2001 items. How many cars that are in fact de-registered have been written up for speeding?

Exercise 2.27 At a theatre performance for children, there is an audience consisting of 120 persons, the same number of each of the sexes. Two thirds of the audience consist of children, out of which three fifths are girls. Because of smoke behind the scene the theatre has to be evacuated. *Women and children first!*, the director of the theatre yells. How many will be left in the theatre when the women and children have left the hall?

Exercise 2.28: *Important!* Study the diagram to the right and explain why for any three sets A, B and C the following has to hold:

$$|A| + |B| + |C| + |A \cap B \cap C| = \\ = |A \cup B \cup C| + |A \cap B| + |A \cap C| + |B \cap C| \tag{2}$$

*

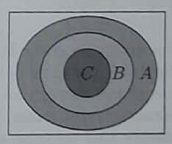
Exercise 2.29 Express the formulas (1) and (2) in such a way that $|A \cup B|$ and $|A \cup B \cup C|$ respectively are the only thing on the left-hand side.

Exercise 2.30 Based on the previous exercise, try to guess a corresponding expression giving the size of the union of four sets. (This is an example of something called the **principle of inclusion/exclusion**; more about that in example 5.9.)

2.4.2 Subsets of Subsets

If you start with a set and remove some elements you get a subset. If you remove some elements from this set you get a new subset which is a subset of the first set as well. Another way of describing this is

if $C \subseteq B$ and $B \subseteq A$ then we also have $C \subseteq A$.

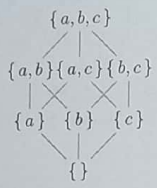


If we start with the set $A = \{a, b, c\}$ and remove one of the elements we will get one of three possible subsets with two elements: $\{a, b\}$, $\{a, c\}$, or $\{b, c\}$. If we remove another element we will get one of the subsets $\{a\}$, $\{b\}$, or $\{c\}$. Having removed the last remaining element as well we get the empty set.

The set of all subsets of a set A is called the **power set** of A and is usually denoted $\mathcal{P}(A)$. (Note that here we have a set where the elements themselves are sets as well.)

Example 2.4 If $A = \{a, b, c\}$, according to the text above $\mathcal{P}(A) = \{\{a, b, c\}, \{b, c\}, \{a, c\}, \{a, b\}, \{c\}, \{b\}, \{a\}, \{\}\}$.

The power set can be ordered in an informative way in a diagram:



■

Exercise 2.31: *Important!* According to which principle is the diagram drawn?

Exercise 2.32 Draw a corresponding diagram showing the subsets of a set with two elements. *

Exercise 2.33 Express the meaning of $\mathcal{P}(A)$ using the set builder.

Note that the set A is incuded in its own power set, and therefore counts as a subset of itself: $A \subseteq A$. Every other subset is called a **proper subset** of A. Thus, $\{a, b, c\}$ has got seven proper subsets.

Exercise 2.34 How many proper subsets does the set $\{17\}$ have?

2.5 Pairs

Now let's discuss the formation of pairs. Everyone knows the importance of the formation of pairs between women and men. But other elements than women and men can be paired, such as chairs and desks, fridges and freezers, lessons and classrooms. This seems to leave room for a definiton of an abstract concept **pair** meaning a set consisting of two elements, where we keep track of the order of the elements (that is, which element is of which kind).

We denote the pair consisting of the elements x and y by (x, y). Now we'll introduce yet another symbol: $X \times Y$ denotes the **product set** of the sets X and Y. The product set is defined as

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

that is, the set of all pairs that can be made by taking an element from the set X and pairing it with an element from the set Y.

Example 2.5: *Programme Syllabus* When a programme syllabus is made for a degree programme, for each course it is specified in which year it is placed and during which semester (autumn or spring) it will be given.

Thus every pair consisting of a year and a semester makes up a possible course date. There are ten such pairs: (year 1, autumn), (year 1, spring), (year 2, autumn), (year 2, spring), (year 3, autumn), (year 3, spring), (year 4, autumn), (year 4, spring), (year 5, autumn), (year 5, spring). ■

Exercise 2.35 $A = \{\text{white, cup}\}$, $B = \{\text{milk, egg}\}$ Write down $B \times A$.

The simplest way of structuring the product set $X \times Y$ is as a **matrix**, that is as a table where the rows are given by the elements of the set X and the columns by the elements in Y.

Example 2.6 The matrix showing course dates (according to the previous example) is:

	Autumn	Spring
year 1	(year 1, autumn)	(year 1, spring)
year 2	(year 2, autumn)	(year 2, spring)
year 3	(year 3, autumn)	(year 3, spring)
year 4	(year 4, autumn)	(year 4, spring)
year 5	(year 5, autumn)	(year 5, spring)

Another typical example of a matrix are the squares on a chess board. There are the rows 1 to 8 and the columns A to H so that each square can be identified using a pair consisting of a column and a row. A1 is the square at the lower left corner and H8 the one at the upper right.

■

From the examples above we can find an important principle: The size of the product set is the product of the sizes of the individual sets. (In the examples we have $5 \cdot 2 = 10$ and $8 \cdot 8 = 64$, respecively.)

Exercise 2.36 Explain in words why this principle holds. *

Exercise 2.37: *Important!* Express the principle as an equation using only mathematical symbols and two arbitrary sets X and Y.

Exercise 2.38 At a small dancing party for waltz and foxtrot there are four women {Viktoria, Anna, Lisa, Sanna} and three men {Fredrik, Ola, Xerxes}.

(a) How can the set of possible dance couples be described as a product set and what is the size of this set?

(b) Unfortunately Viktoria and Anna only dance the waltz, and Fredrik and Xerxes only dance the foxtrot. (The rest dance both the waltz and foxtrot.) The in practice possible couples is thus a subset of the set above. Specify this subset by marking the couples that have at least one dance in common in the matrix showing the product set. How many are they? How can this number be found as the size of a certain product set minus the size of a product set of certain subsets? *

2.6 Infinite Sets

In most of the examples and exercises up to now we have worked with fairly small sets, for purely practical reasons. But there are large sets as well, and some are even infinitely large. Now the time has come to take a closer look at that kind.

2.6.1 Standard Sets

A fundamental activity in discrete mathematics is to list things. Humanity is madly fond of numbering things – and to computers, numbers are even more important.

When you number things you normally start with one and count on upwards: one, two, three, and so on. In principle you may need to count for any length of time, because there is an infinite number of such **positive integers**. No matter what number we have arrived at, there is always a **next number**; the number that is one step bigger. The positive integers form a set that is denoted $\mathbb{Z}_+$, where the letter Z derives from the German word *Zahl* (which means number) and the plus sign of course indicates that the numbers are positive.

$$\mathbb{Z}_+ = \{1, 2, 3, \ldots\}$$

Sometimes you start the numbering with zero, for instance in some programming languages such as C++ and Java – or when you describe the score in a game of football or tennis. The integers from zero and upwards make up a standard set of their own, which is denoted $\mathbb{N}$. The letter N stands for

natural numbers, numbers that can appear as the number of objects. If there isn't anything to count then the number is zero!

$$\mathbb{N} = \{0, 1, 2, 3, \ldots\}$$

When the numbers are to indicate a direction as well, for instance if an amount is gain or loss, negative numbers are also needed. The set of all **integers** – positive, negative and zero – is denoted simply $\mathbb{Z}$. The set $\mathbb{Z}$ can be listed as well (if not in order of size):

$$\mathbb{Z} = \{0, \pm 1, \pm 2, \pm 3, \ldots\}$$

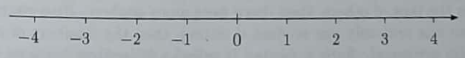
Exercise 2.39 Give examples of some phenomena in reality that are numbered according to $\mathbb{Z}_+$, $\mathbb{N}$, and perhaps even $\mathbb{Z}$. Are there any situations where there are gaps in the numbering and in that case why? How would it be to use $\mathbb{Z}$ for numbering streets (positive numbers on the right and negative on the left)? *

The infinite standard sets $\mathbb{Z}_+$, $\mathbb{N}$ and $\mathbb{Z}$ can be used in the same way as any other sets in combination with operations such as intersection, union, difference, and product.

Exercise 2.40 Describe the sets given by $\mathbb{N} \cup \mathbb{Z}_+$, $\mathbb{N} \cap \mathbb{Z}_+$, $\mathbb{N} \setminus \mathbb{Z}_+$, $\mathbb{Z} \setminus \mathbb{N}$, $\mathbb{N} \setminus \mathbb{Z}$, $\mathbb{N} \times \mathbb{N}$, and $\mathbb{Z} \times \mathbb{Z}$.

Exercise 2.41: *Important!* By $2\mathbb{Z}$ is meant the set $\{2n \mid n \in \mathbb{Z}\}$. Describe in words the sets given by $2\mathbb{Z}$ and $\mathbb{Z} \setminus 2\mathbb{Z}$.

There are more numbers than just the integers. The set of all **rational numbers** – numbers that can be written as a quotient of an arbitrary integer and a positive integer – is denoted $\mathbb{Q}$. The set of all **real numbers** – all the numbers that can be found on the number line – is denoted $\mathbb{R}$.



On the number line, we have marked the integer points, but that is not to be interpreted as meaning that these are the only numbers that exist. Every point on the line corresponds to a number, and for purely practical reasons you can't put labels on all of them.

If you want to expand even more, there are the **complex numbers** $\mathbb{C}$ as well. However, we will not bring them up here; they belong to other courses.

Exercise 2.42 Describe the set $\mathbb{Q}$ using the set builder.

Exercise 2.43 Explain why the integers are a subset of the rational numbers.

Exercise 2.44

(a) State a number belonging to $\mathbb{Z}$ but not $\mathbb{N}$.

(b) State a number belonging to $\mathbb{Q}$ but not $\mathbb{Z}$.

(c) State a number belonging to $\mathbb{R}$ but not $\mathbb{Q}$.

(d) (If you have studied complex numbers:) State a number belonging to $\mathbb{C}$ but not $\mathbb{R}$. *

Exercise 2.45 What is in fact $\mathbb{R} \times \mathbb{R}$, and have you been working with this set at times?

2.6.2 Bijections and Cardinality

When working with sets one is, as already mentioned, often mostly interested in knowing the number of elements in a set. In many cases one doesn't even want to know that, but whether it has more, less, or the same number of elements as some other set.

For instance, five ants are a larger number than four elephants. You'll notice that if you try to pair the elements in the two sets so that each ant gets an elephant of its own. There are not enough elephants, and one ant has to do without. To pair the elements of two sets that you want to compare is often a very efficient method.

If one has a set of students and a set of chairs and wants to know which kind one has got more of, students or chairs, there are at least two ways to proceed. One way is to count the students and count the chairs and see what numbers one gets. Another way is to tell the students to sit down, on different chairs if possible. If there are chairs left over, then there were more chairs than students. If there are no chairs left over, but some students sitting in the laps of others, then there were more students than chairs. If on each chair one and only one student is sitting, then the numbers of students and chairs are equal. Such a pairing is called a **bijection** between the two sets.

The usual method of counting can be regarded as a case of bijection as well. To count the elements in a set with, let's say, three elements is the same thing as pairing each element in the set with one of the numbers between one and three: one *(point at the first element)*, two *(point at the second element)*, three *(point at the third element)*. For instance the set { Per, Pål, Pia } has the cardinality of 3, since there is a bijection between this set and the set $\{1, 2, 3\}$:

$1 \mapsto$ Per
$2 \mapsto$ Pål
$3 \mapsto$ Pia

If we give this bijection the namn ϕ (the greek letter *phi*) it can be described as a **function** with the values $\phi(1) =$ Per, $\phi(2) =$ Pål, $\phi(3) =$ Pia. (More about functions in chapter 8.)

There are other bijections between the sets $\{1, 2, 3\}$ and { Per, Pål, Pia }, for instance

$3 \mapsto$ Pål
$1 \mapsto$ Pia
$2 \mapsto$ Per

and another four besides.

Exercise 2.46 Write down the four remaining possible bijections. *

Exercise 2.47 Show that the sets $A = \{$ 000, 001, 010, 011, 100, 101, 110, 111 $\}$ and $B = \mathcal{P}(\{1, 2, 3\})$ have the same cardinality by creating a bijection between them. Try to find one that is easily described!

Remark: The elements in the set A are called *binary strings of length 3.*

But how do you go about it if you want to compare the sizes of two *infinite* sets? You do it the same way! It has been decided that "the same cardinality" means "it is possible to make up a bijection", and that definition is meaningful both when applied to finite sets and to infinite ones.

Example 2.7: ***The Cardinality of the Integers*** Which set has the highest cardinality, $\mathbb{N}$ or $\mathbb{Z}$?

Spontaneously, one might think that since $\mathbb{N}$ is a subset of $\mathbb{Z}$ it has to be smaller. But if we arrange the sets like this:

$\mathbb{N}$:	0	1	2	3	4	5	6	7	8	9	10	...
$\mathbb{Z}$:	0	1	−1	2	−2	3	−3	4	−4	5	−5	...

we see that a bijection between the sets can be made without any problems using the formula

$$\phi(n) = \begin{cases} -n/2 & \text{if } n \text{ is even} \\ (n+1)/2 & \text{if } n \text{ is odd} \end{cases}$$

Thus $\mathbb{N}$ and $\mathbb{Z}$ have the *same* cardinality!

(A lesson that can be learned from this is that "common sense" doesn't work all that well when you are handling infinite sets. Common sense is based on experience, and most people haven't worked with infinite things all that frequently.) ■

A set which has the same cardinality as $\mathbb{N}$ is said to be **countably infinite**, since the elements in it can be put into a numbered list (in the way we did with $\mathbb{Z}$ in the example.)

Exercise 2.48 Show that the set of even integers is countably infinite by making a bijection from $\mathbb{N}$. *

Quite a lot of ingenuity may be needed to confirm that two sets have the same cardinality, since the fact that you have failed to find a bijection isn't a proof that none can be found. Here are two classic examples, one where a bijection is found by applying enough cunning and one where it is possible to argue that there *really* isn't one.

Example 2.8: ***The Cardinality of the Rational Numbers*** Is $\mathbb{Q}_+$ countably infinite or not?

$\mathbb{Q}_+$, that is the positive rational numbers, the ones that can be written as a quotient of two positive integers. That the set is infinite is obvious, but is it *countable*? A first effort is to make a list in order of size. However, that is doomed to failure, since whichever positive rational number q we take it is possible to find another one that is smaller ($q/2$, for instance), so it isn't possible to decide which of the numbers that is to be the first.

Instead we try writing the numbers in a square, using the numerator as the row number and the denominator as the column number.

	1	2	3	4	5	6	⋯
1	$1/1$	$1/2$	$1/3$	$1/4$	$1/5$	$1/6$	⋯
2	$2/1$	$2/2$	$2/3$	$2/4$	$2/5$	$2/6$	⋯
3	$3/1$	$3/2$	$3/3$	$3/4$	$3/5$	$3/6$	⋯
4	$4/1$	$4/2$	$4/3$	$4/4$	$4/5$	$4/6$	⋯
5	$5/1$	$5/2$	$5/3$	$5/4$	$5/5$	$5/6$	⋯
6	$6/1$	$6/2$	$6/3$	$6/4$	$6/5$	$6/6$	⋯
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋱

If we now follow the indicated path across the matrix, we find that it passes all the rational numbers one at a time, so we can number them in passing. (In fact, we will pass every fraction several times, since for instance $1/1 = 2/2 = 3/3 = \ldots$, but we can fix that by only assigning a number at the first passage.)

It is thus possible to make a numbering of the rational numbers, where every fraction is assigned a number, and where no two different fractions are assigned the same number. A numbering of this kind is the same thing as a bijection between $\mathbb{Q}_+$ and $\mathbb{N}$. So $\mathbb{Q}_+$ is countably infinite. ■

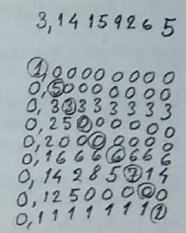
Exercise 2.49 Is $\mathbb{Q}$ countably infinite? *

Exercise 2.50 Is $\mathbb{Z} \times \mathbb{Z}$ countably infinite? *

Example 2.9: ***Cantor's Diagonal Argument*** The German mathematician Georg Cantor (1845–1918) founded set theory and was the first person who studied different kinds of infinite sets. One of his most well-known efforts is his ingenious proof that $\mathbb{R}$, the set of real numbers, *isn't* countably infinite.

All real numbers can be written using an infinite decimal expansion. (On the numbers that have a finite decimal expansion we can add an infinite number of zeros at the end.)

If the real numbers were countable we could imagine an (infinitely long) list containing all the numbers. Assume that we have that list. Then we can put together a new number by taking all the decimals on the *diagonal* of the list (first decimal in the first number, second decimal in the second number, etcetera) and change each decimal to some other digit.



The number we have constructed this way can't be equal to any number in the list, since it differs from every number in the list in at least one decimal!

So, if the real numbers were countable we would be able to include all the numbers in an infinite list. Since we have seen that no list containing real numbers is complete the real numbers are not countable! We say that $\mathbb{R}$ is **uncountably infinite**. ■

Remark For the sake of completeness we should here point out that not all numbers have a *unique* decimal expansion. Those with a finite expansion can be rewritten to infinite expansions in two ways: for instance the number 1 can be written both as $1.000\ldots$ and as $0.999\ldots$. But in Cantor's diagonal proof we can always choose to change the decimals so that the number we generate doesn't end with either an infinite number of zeros or an infinite number of nines, and then it can't be an alternative representation of a number included in the list.

Exercise 2.51 Show that $\mathbb{R}_+$ and the interval $0 < x < 1$ have the same cardinality by finding a bijective function between them.

2.7 More Exercises

2.7.1 Routine Work

Exercise 2.52 We have the set $A = \{1, 2, \{1, 3\}\}$. Start by making a list of all elements in A and another one listing all subsets of A. Then determine whether the following statements are true or false:

(a) $1 \in A$. **(b)** $1 \subseteq A$. **(c)** $\{1\} \in A$. **(d)** $\{1\} \subseteq A$.

(e) $\{1, 2\} \in A$. **(f)** $\{1, 2\} \subseteq A$. **(g)** $\{1, 3\} \in A$ **(h)** $\{1, 3\} \subseteq A$

Exercise 2.53 Express "the set of all numbers that are squares" using the set builder.

Exercise 2.54 Determine the following sets:

(a) $\{2, 3, 5, 7\} \cup \{1, 3, 5, 7, 9\}$ **(b)** $\{2, 3, 5, 7\} \cap \{1, 3, 5, 7, 9\}$

(c) $\{2, 3, 5, 7\} \setminus \{1, 3, 5, 7, 9\}$ **(d)** $\{1, 3, 5, 7, 9\} \setminus \{2, 3, 5, 7\}$

(e) $\{x \mid x \text{ is an even natural number}\} \cup \{x \mid x \text{ is an odd natural number}\}$

(f) $\{x \mid x \text{ is an even natural number}\} \cap \{x \mid x \text{ is an odd natural number}\}$

(g) $\{x \mid x \text{ is an even natural number}\} \setminus \{x \mid x \text{ is an odd natural number}\}$

Exercise 2.55 The following equality holds:

$$(A \cup B) \cap (A \cap B)^c = (A \cap B^c) \cup (A^c \cap B)$$

(a) Show the equality using the rules. Note in every step which rule you are using.

(b) Illustrate the equality using Venn diagrams.

(c) Describe in words the set in question.

(d) Prove the equality using a formal set-theoretical argument.

Exercise 2.56 We have the set $A = \{1, 2, \ldots, 10\}$ and the set $B = \{a, b, \ldots, j\}$. C is the subset of $A \times B$ where the number is odd, D is the subset of $A \times B$ where the letter is a vowel. Calculate $|C \cup D|$.

Exercise 2.57 Determine whether the following statements are true or false:

(a) $1 \in \mathbb{R}$ **(b)** $3.14 \in \mathbb{Q}$ **(c)** $\pi \in \mathbb{Q}$ **(d)** $-5 \in \mathbb{N}$ **(e)** $64/8 \in \mathbb{Z}$

Exercise 2.58 Describe in words the set $\mathbb{Q} \setminus \mathbb{Z}$.

2.7.2 To Ponder About

Exercise 2.59 At different places in the chapter there are Venn diagrams including one, two, or three sets. Try in analogy with this to draw corresponding diagrams containing four and five sets. The diagrams must cover the general case, that is, the case where there are elements in all possible combinations of the sets in question.

Exercise 2.60 Suppose that we have a countable set of sets, and that each of the sets in this set is countable. Show that the union of these sets is countable.

Exercise 2.61: ***Database*** In a database at a health clinic there is information about all patients, such as name, age, number of doctor visits, employer and name of family doctor (if a family doctor is chosen, otherwise this field contains 000). Information about a certain patient, let's say Kimmo, is retrieved using point notation: Kimmo.name, Kimmo.age, Kimmo.number-of-visits, etc. From this database, sets can be put together; for instance the expression

$$\{p \mid p.\text{employer} = \text{MDH}\}$$

gives the set of all patients who have MDH as employer. Make up sets so that the questions "Have all patients who have made less than ten visits to the doctor and are under 35 years of age chosen a family doctor?", "How many are these patients?" (the ones that have made less than ten visits to the doctor and are under 35 years of age), and "Is Kimmo such a person?" can be be answered using set-theoretical notation (that is, using the symbols $\cap, \cup, \in, \notin, \subseteq, |\cdot|$), and write down those expressions.

3 Arithmetic

In elementary school the four rules of arithmetic (addition, subtraction, multiplication, and division) were taught. At first, only integers were used, but in connection with division decimals were introduced, and then that road was followed.

But it would have been possible to follow another road, continuing the work with integers. The way of handling division would then have been different, since the quotient of two integers isn't usually an integer. This chapter will follow this alternative road, so that we can see where it leads!

Highlights from this chapter.

- Integer division with *quotient* and *remainder*: if you divide 17 by 6 you get the quotient 2 and the remainder 5.
- *Prime numbers* $(2, 3, 5, 7, 11, \dots)$ and *prime factorisation* of integers $(90 = 2 \cdot 3 \cdot 3 \cdot 5)$.
- *Greatest common divisor* (gcd) and *least common multiple* of integers: $\gcd(90, 20) = 10$ and $\operatorname{lcm}(90, 20) = 180$.
- The *Euclidian algorithm* for calculating the greatest common divisor of two integers.
- Solving *diofantine equations* of the type $ax + by = c$ (where all letters represent integers) using the Euclidian algorithm.
- *Modular arithmetic*: counting modulo n (and in $\mathbb{Z}_n$) where you restart at zero every time you reach n.
- *Binary numbers* and other number bases.

3.1 The Division Algorithm

Division is something that most people learn around the age of ten. During the years thereafter calculators have usually been in use, so the knowledge may have fallen into oblivion. Because of this we have this introductory

section, since division with remainders is the real foundation of the applications that this chapter will cover.

The following type of calculation is probably familiar:

$$\frac{11}{4} = 2 + \frac{3}{4}$$

An alternative way of writing the same thing is:

$$11 = 2 \cdot 4 + 3$$

Both calculations can be read as "if you divide 11 by 4 the quotient is 2 and the remainder is 3".

> Exercise 3.1 Note that we *don't* think that "if you divide 11 by 4 the result is 2.75". Think of some concrete situation where "quotient 2, remainder 3" is a meaningful answer while 2.75 isn't.

The idea of division can be expressed as a theorem:

Theorem 3.1: ***Division with remainders*** Given two integers p and $d \neq 0$ it is possible to find two *unique* integers q and r such that

$$p = qd + r, \quad \text{where} \quad 0 \leq r < |d| \qquad \blacksquare$$

q in the theorem is called the **quotient** and r the **principal remainder**.

The reason that not just the word "remainder" is used is that in our example it is also possible to write

$$\frac{11}{4} = 1 + \frac{7}{4} \quad \text{or} \quad \frac{11}{4} = 3 - \frac{1}{4}$$

so one might as well state that 7 or -1 is the remainder of the division. Every number that can be obtained in this way is counted as a remainder, but usually it is the smallest positive one that is wanted. If someone says just "remainder" it is usually (but not always) the principal remainder the person means.

> Exercise 3.2 Describe "the set of all remainders when 11 is divided by 4" using the set builder.

Finding the quotient and the principal remainder isn't very hard (even if it can sometimes be laborious). If $0 \leq p < |d|$ we are done, for then the quotient is zero and the remainder p. Otherwise reduce p by $|d|$, if p is too large, or increase p by $|d|$ if p is to small, and see what is left. If that number lies between zero and $|d|$ we have finished, otherwise we repeat the procedure. A systematic method of this kind to solve a problem is called an **algorithm**; this one is called the **division algorithm**. We demonstrate using an example:

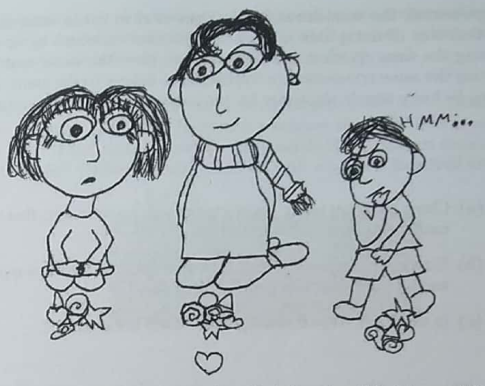


Figure 3.1: *"It isn't possible to share this alike! One piece'll be left over." "Yes, but if I eat one it will work out!*

Method 3.1: ***The Division Algorithm*** Find the quotient and remainder when 1234 is divided by 567.

Since we don't know the 567-table by heart (which is otherwise a shortcut) we follow the description:

$$1234 = 1 \cdot 567 + 667 = 2 \cdot 567 + 100$$

Since 100 is less than 567 we are done. The quotient is 2 and the remainder is 100. $\blacksquare$

> Exercise 3.3 Calculate the quotient and the principal remainder when 35 is divided by 4 and when 40 is divided by -3. Write down all steps in the calculations.

"Long division" is a systematical way of performing a division.

Example 3.1: ***Application to Programming*** A student is trying to solve exercise 5.83 on page 135. The question concerns the probability of getting a certain set of cards when playing poker. The student and the teacher have different opinions regarding the answer, and to show that he is right the student decides to write a program that generates all possible poker hands and checks how many of them there really are that satisfy the conditions.

Since it is easy to manipulate numbers when programming, the student decides to represent the 52 cards with the numbers 0–51 in the computations.

When presented, the numbers are to be converted to cards, with suit and rank. One way of doing this is to divide the card numbers by 4. Cards generating the same quotient are considered to have the same rank, cards generating the same remainder are considered to belong to the same suit. In this way, he has a simple algorithm for the conversion between numbers and cards. ■

Exercise 3.4

(a) Check that you really get thirteen cards in each suit, four of each rank, using this method.

(b) How would you choose to make the connection between remainder and suit and between quotient and rank?

(c) Is there any way of making this a little bit smarter?

3.2 Prime Numbers and Divisors

Some programs (for instance Maple) work using *exact arithmetic*. Among other things, this means that they don't approximate fractions using a decimal expansion, but represent them as *fractions* (which means that there are no round-off errors). These calculations demand that the smallest common denominator can be found very swiftly when adding and subtracting, and above all that it is possible to simplify the fractions in a reasonable amount of time; otherwise the numbers used would soon reach astronomical size. How do the programs do it?

This is a question that this section will answer.

3.2.1 Divisors

If you get the remainder 0 when dividing the number m by the number d, d is said to be a **divisor** of or **factor** of m, and m is said to be a **multiple** of d. You can also say that d **divides** m, or that m is **divisible** by d.

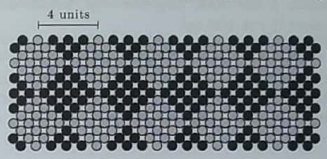
That d divides m is denoted $d \mid m$, and that d *doesn't* divide m is denoted $d \nmid m$. For instance,

$$6 \mid 24,\ -17 \mid 34,\ 5 \mid -100,\ \text{and}\ 4711 \mid 0\ \ \text{but}\ \ 0 \nmid 37\ \text{and}\ 3 \nmid 1$$

If d is a divisor of m then $-d$ is also a divisor of m, and d also divides $-m$. Because of this you can restrict yourself to the study of divisors of positive numbers. Then you just have to add a great number of $\pm$:s to the investigation when you want to generalise. Therefore, in the remainder of the text we will only work with positive numbers and positive divisors (if we don't explicitly say anything else).

Exercise 3.5: *Important!* Which numbers are divisors of zero, and why? And which numbers does zero itself divide? Which numbers have one as a divisor, and which numbers divide one?

Exercise 3.6 A person wants to make a bracelet out of pearls with a repeating pattern. The person measures 40 pearl units around the wrist. Which pattern lengths are possible if the pattern should come out even? (Below we see a bracelet with a pattern length of 4.)



3.2.2 Prime Numbers

If an integer greater than 1 doesn't have any other divisors than ± 1 and $\pm$(itself) it is called a **prime number** or simply **prime**. For instance, 11 is a prime number since it has only the divisors ± 1 and ± 11.

Exercise 3.7 Which is the smallest prime number?

An integer greater than 1 that isn't a prime number is called a **composite number**. For instance, 12 is a composite number, since it has the divisors ± 1, ± 2, ± 3, ± 4, ± 6 and ± 12.

Exercise 3.8 Which is the smallest composite number?

Note that the number 1 isn't counted as either a prime number or a composite number; it's sorted in a class of its own. The reason for this will be explained a bit further along in this section.

All integers from two and upwards can be written as products of primes (at least if we accept that "products" can consist of only one factor!). Either the number is prime, and then we are done, or it is composite, and then we can divide it into two parts which are both smaller than the initial number and repeat the line of argument on them. Since the numbers we study get smaller all the time, sooner or later we have to reach something that can't be divided further (since there is only a limited number of natural numbers smaller than our initial number). For instance we have $12 = 3 \cdot 4 = 3 \cdot 2 \cdot 2$, which is a product of primes.

As a matter of fact, it's not just the case that you are guaranteed that you can prime factorise numbers, but there is an important addition:

Theorem 3.2: ***Fundamental Theorem of Arithmetic*** Every integer greater than one can be written as a product of prime numbers in *one and only one* way (if the order of the factors doesn't matter). ∎

The proof is given in a section of its own on page 48. (It requires some things that we haven't covered yet.)

Anyway, this shows why one doesn't want to include the number one among the primes, because if one does "only one way" breaks down. You can always ladle on more ones if you like to!

Note that the theorem says only that the prime factorisation *exists*, not how to *find it*. As a matter of fact, no one has yet found any really good way to prime factorise large numbers in a reasonable time, something that is utilised for instance in encryption. (Search the web for "RSA algorithm" for more information.)

Exercise 3.9 Prime factorise 60, 61, 62, 63, and 64.

Exercise 3.10 How would you go about it if you were ordered to list all prime numbers less than 1000? Use your method to list all prime numbers less than 50, to verify that it works!

Exercise 3.11 How would you go about it to test whethera given number is a prime number? Try to estimate the number of subtests that you have to do using your method before you can be *sure* that the number is prime. If every subtest takes one nano-second, how long will it take to check if a hundred-digit number is prime?

The prime numbers are packed rather densely at the beginning of the natural numbers, but later on they thin out; that can be seen if you solve exercise 3.11. How is it, will you eventually pass all the prime numbers, or is there an infinite number of them?

There is an infinite number of primes, and that was proven by Euclid (as early as in the 4th century B.C.). We will present the formal proof, mainly because it is a good example of an important proof technique: **proof by contradiction**. (More about proof techniques in Chapter 7.)

Theorem 3.3: ***The Number of Primes*** There is an infinite number of primes.

Proof If you get stuck when trying to argue something there is always the possibility of pulling "well, assume that it isn't like that", and then showing that the consequences are absurd.

So let's assume that there is *not* an infinite number of primes. In that case the number has to be finite. Then we can take all the primes there are, multiply them, and add one to the result. That gives us a number. This number isn't divisible by the first prime, since you get the remainder of one

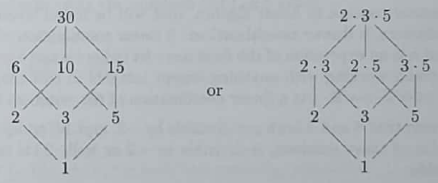
if you divide. Neither is it divisible by the second prime, for the same reason. It's not divisible by any of the primes we started with! But all numbers can be factorised into primes. The number we have found must thus either be a prime that we didn't include in our list of all the primes that exist, or it's the product of several primes that we have missed.

So: If we have all the prime numbers that exist there exist some more. This doesn't make sense! The reasoning is faultless, so the fault has to be in the starting point: that there is a finite number of primes.

Thus there is *not* a finite number of primes. Consequently, there is an infinite number. ∎

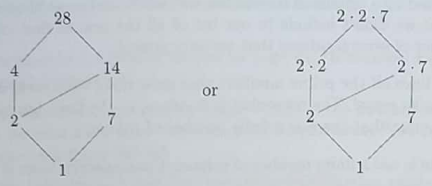
3.2.3 The Divisor Graph

There are contexts where the complete collection of divisors of a number is of interest. (Exercise 3.6 on page 39 showed one of those.) For instance the number 30 has the positive divisors 1, 2, 3, 5, 6, 10, 15, and 30. One way of illustrating the relationships between the numbers of this set is to draw a **divisor graph**. The divisors of 30 are drawn like this:



On top of the graph is the starting number. On the next "level" are the numbers that are divisors of the starting number, but don't divide any of the other divisors. On the level below are those which divide the level above but not anything else. To indicate which elements on the level above they divide we have drawn lines. And so it continues. It may be easier to see the system of you look at the graph on the right, where the number is prime-factorised. On level zero there are zero prime factors (since one isn't a prime number), on level one we have one factor, on level two two factors, and so on. Anyway, we can see that that "30 is divisible by 10, which in its turn is divisible by 5, which is divisible by 1". (These graphs are examples of something called *Hasse diagrams*; more about that in Chapter 8.)

Not all numbers have a graph as symmetrical as the one for 30; we can for instance take a look on the graph over the divisors of 28:



Exercise 3.12 What is it that makes the graph over the divisors of 30 more symmetrical than the graph over the divisors of 28?

Exercise 3.13 How does the divisor graph of a power of a prime look? (You can test with the graph for $2^{500}\ldots$)

Exercise 3.14 Draw the divisor graphs for the numbers in exercise 3.9 on page 40. (Hint: save 60 for last!)

3.2.4 Common Divisors

If the number d divides both the number m and the number n, d is called a **common divisor** of m and n. For instance, -2 is a common divisor of 6 and 4, since both numbers are divisible by -2.

A fundamental concept in linear algebra, that will be useful when we study common divisors, is **linear combination**. A linear combination of two numbers m and n is an expression of the form $am+bn$ (where a and b are integers, since we aren't working with anything except integers in this chapter). For instance, $14 = 1 \cdot 6 + 2 \cdot 4$ is a linear combination of the numbers 6 and 4.

We have seen that 6 and 4 both are divisible by -2, and 14, which is a linear combination of these numbers, is divisible by -2 as well. This relationship always holds:

Theorem 3.4: ***Divisors of Linear Combinations*** If m and n have the common divisor d, every linear combination of m and n will have d as a divisor as well.

Proof If $d \mid m$ then $m = pd$ (where $p \in \mathbb{Z}$) and if $d \mid n$ then $n = qd$ (where $q \in \mathbb{Z}$). So

$$am + bn = apd + bqd = \underbrace{(ap + bq)}_{\text{an integer}}d \quad \Rightarrow \quad d \mid am + bn \qquad \blacksquare$$

Very often one is primarily interested in the **greatest common divisor** of two numbers. A typical application is simplification of fractions, but there is

a surprising number of other areas of use. The greatest common divisor of the numbers m and n is denoted $\gcd(m,n)$.

For instance, we find that the greatest common divisor of 16 and 20 is $\gcd(16,20) = 4$. (2 is a common divisor as well, but not the *greatest* common divisor.) The numbers 15 and 19 don't have any factors in common, so $\gcd(15,19) = 1$. Such pairs of numbers that lack common factors are called **coprime numbers**.

One way of finding the greatest common divisor is to prime factorise the numbers. The greatest common divisor is the intersection between the prime factorisations of the two numbers. For instance, $252 = 2 \cdot 2 \cdot 3 \cdot 3 \cdot 7$ and $60 = 2 \cdot 2 \cdot 3 \cdot 5$, so $\gcd(252, 60) = 2 \cdot 2 \cdot 3 = 12$. But if you don't already have the prime factorisations at hand this is a very laborious method, at least if the numbers are large. (As stated earlier, no really good way of finding prime factorisations is known.)

The interesting thing is that there exists an efficient method of finding the gcd that *doesn't* demand that the numbers are factorised. The method was invented by Euclid in the 4th century B.C. and still holds its own.

Method 3.2: ***The Euclidian Algorithm*** We are trying to find the greatest common divisor of two numbers. Start by dividing the greater number by the smaller one. Then divide the smaller number by the resulting remainder. That gives us a new remainder. Repeat the proceedings until the remainder is zero. The remainder in the previous step, the **last "non-vanishing" remainder**, the number we divided by in the last step, is equal to the greatest common divisor!

We demonstrate using the numbers 1914 and 4734:

$$\begin{aligned}
4734 &= 2 \cdot 1914 + 906 \\
1914 &= 2 \cdot 906 + 102 \\
906 &= 8 \cdot 102 + 90 \\
102 &= 1 \cdot 90 + 12 \\
90 &= 7 \cdot 12 + \boxed{6} \quad \leftarrow \text{Last "non-vanishing" remainder} \\
12 &= 2 \cdot 6 + 0
\end{aligned}$$

$\gcd(4734, 1914) = 6$. We can do a fast check of the calculation by dividing the numbers by 6:

$$\frac{4734}{6} = 789 \qquad \frac{1914}{6} = 319$$

Here we see that 6 divides both the numbers, so it is a common divisor. (But this test doesn't tell if 6 is the *greatest* common divisor.)

Exercise 3.15 Use the Euclidian algorithm on the numbers 100 and 70, to check that it really gives the answer 10 (which you can see is the greatest common divisor).

Why the Euclidian algorithm works (theoretical answer): The calculation on the previous page was based on the idea

$$\begin{aligned}\gcd(4734, 1914) &= \gcd(1914, 906) = \gcd(906, 102)\\ &= \gcd(102, 90) = \gcd(90, 12) = \gcd(12, 6) = \boxed{6}\end{aligned}$$

The last step, that $\gcd(12, 6) = 6$, is fairly obvious, but why do the other equalities hold? We have to show that $\gcd(m, n) = \gcd(n, r)$, where $m = qn + r$. This follows from the fact that m and n have exactly the same common divisors as n and r, a relationship we can explain like this:

If d is a common divisor of n and r, then we can write the numbers as $n = n_1 d$ and $r = r_1 d$. Then $m = qn + r = qn_1 d + r_1 d = (qn_1 + r_1)d$, so d a divisor of m as well. Then we can repeat the argument on a common divisor of m and n, using the fact that $r = m - qn$.

Thus all common divisors of m and n are common divisors of n and r as well, and vice versa, which shows that the numbers must have exactly the same common divisors. Then of course, among other things, their greatest common divisor has to be the same.

Why the Euclidian algorithm works (concrete answer): An alternative way of gaining understanding of the algorithm is to analyse a concrete example in depth. In this analysis we will also rewrite expressions in ways that will be used in problem-solving later in this chapter.

So the question is: How do we know that the result from the Euclidian algorithm, the last "non-vanishing" remainder, really is the greatest common divisor of the given numbers?

Step 1 First, we can show that the result really is a common divisor, which requires that it is a factor of both numbers. We will go through the algorithm from the bottom and upwards, and insert the results from the lower steps into the higher ones:

$$\begin{aligned}
12 &= 2 \cdot 6 \\
90 &= 7 \cdot 12 + 6 && \text{Replace 12 with } 2 \cdot 6\\
&= 7 \cdot (2 \cdot 6) + 6 = 15 \cdot 6 && \text{Factor out 6}\\
102 &= 1 \cdot 90 + 12 && \text{Replace 90 and 12}\\
&= 1 \cdot (15 \cdot 6) + 2 \cdot 6 = 17 \cdot 6 && \text{Factor out 6}\\
906 &= 8 \cdot 102 + 90 && \text{Replace 102 and 90}\\
&= 8 \cdot (17 \cdot 6) + 15 \cdot 6 = 151 \cdot 6 && \text{Factor out 6}\\
1914 &= 2 \cdot 906 + 102 && \text{Replace 906 and 102}\\
&= 2 \cdot (151 \cdot 6) + 17 \cdot 6 = 319 \cdot 6 && \text{Factor out 6}\\
4734 &= 2 \cdot 1914 + 906 && \text{Replace 1914 and 906}\\
&= 2 \cdot (319 \cdot 6) + 151 \cdot 6 = 789 \cdot 6 && \text{Factor out 6}
\end{aligned}$$

Both numbers evidently contain the factor 6, which is therefore a common divisor. That this always is the case when using the Euclidian algorithm

follows from the fact that it's always possible to reverse the algorithm and in this way express the initial numbers as multiples of the last non-vanishing remainder.

Step 2 Now we show that the result must be a multiple of the greatest common divisor. Again we reverse the algorithm, but now in a different way. First we extract all the remainders:

$$\begin{aligned}
6 &= 90 - 7 \cdot 12\\
12 &= 102 - 1 \cdot 90\\
90 &= 906 - 8 \cdot 102\\
102 &= 1914 - 2 \cdot 906\\
906 &= 4734 - 2 \cdot 1914
\end{aligned}$$

Then we insert them step by step into the expression giving the first remainder. We'll show the first step in detail so that it is clear exactly what happens:

$$\begin{aligned}
6 &= 90 - 7 \cdot 12\\
&= 90 - 7 \cdot (102 - 90) && \text{Replace 12 with } 102 - 1 \cdot 90\\
&= 90 - 7 \cdot 102 + 7 \cdot 90 && \text{Expand the expression}\\
&= 8 \cdot 90 - 7 \cdot 102 && \text{Collect the multiples of 90}
\end{aligned}$$

We can go on like this and replace one number at a time:

$$\begin{aligned}
&= 8 \cdot (906 - 8 \cdot 102) - 7 \cdot 102 && \text{Replace 90}\\
&= 8 \cdot 906 - 71 \cdot 102 && \text{Collect 102}\\
&= 8 \cdot 906 - 71 \cdot (1914 - 2 \cdot 906) && \text{Replace 102}\\
&= 150 \cdot 906 - 71 \cdot 1914 && \text{Collect 906}\\
&= 150 \cdot (4734 - 2 \cdot 1914) - 71 \cdot 1914 && \text{Replace 906}\\
&= 150 \cdot 4734 - 371 \cdot 1914 && \text{Collect 1914}
\end{aligned}$$

At last we have succeeded in writing 6 (the result from the Euclidian algorithm) as a linear combination of 4734 and 1914 (the initial numbers). According to theorem 3.4 on page 42, 6 has then to be a multiple of the greatest common divisor. This calculation as well is possible to carry out regardless of the numbers given by the algorithm, so the conclusion is universal.

Conclusion The last non-vanishing remainder is thus firstly a common divisor and secondly a multiple of the greatest common divisor. The only number that satisfies both these requirements *at the same time* is precisely the greatest common divisor.

Thus we have shown that the result from the Euclidian algorithm is the greatest common divisor of the intial numbers, which was to be proven! ■

Exercise 3.16 The numbers 100 and 70 have the greatest common divisor 10 according to exercise 3.15 on page 43. Follow the method above to express 10 as a linear combination of 100 and 70.

The Euclidian algorithm will be used in several applications in the rest of the chapter, with more examples with complete solutions.

One area of use for the gcd is as already mentioned fractions, and in this context what is sought is often the **least common multiple**, lcm for short, of two integers. The reason is that when adding fractions, one usually wants to use the **least common denominator**, as in

$$\frac{3}{4}+\frac{1}{6}=\frac{3\cdot 3}{3\cdot 4}+\frac{2\cdot 1}{2\cdot 6}=\frac{9}{12}+\frac{2}{12}=\frac{11}{12}.$$

The least common denominator (12) is the least common multiple of the original denominators (6 and 4).

> Exercise 3.17 Write a formal definition of the meaning of "least common multiple" of two numbers m and n.

Some pages back we noted that the greatest common divisor of two numbers is given by the intersection of their prime factorisations. In the same way, the least common multiple is given by the union of the prime factorisations. For instance, $252 = 2\cdot 2\cdot 3\cdot 3\cdot 7$ and $60 = 2\cdot 2\cdot 3\cdot 5$ have the least common multiple $2\cdot 2\cdot 3\cdot 3\cdot 5\cdot 7 = 1260$. But neither in this case is prime factorisation a *good* way of solving the problem in the universal case.

You get the union of two sets if you write down the elements of the two sets and then remove the ones included in the intersection, since they have been included twice. The intersection of the prime factorisations is the greatest common divisor, so the set-theoretical line of argument transfered to numbers is

$$\operatorname{lcm}(m,n)=\frac{m\cdot n}{\gcd(m,n)}$$

Example 3.2: ***Least Common Multiple*** Find the least common multiple of 4711 and 777.

We start by finding the gcd using the Euclidian algorithm.

$$\begin{aligned}4711 &= 6\cdot 777+49\\ 777 &= 15\cdot 49+42\\ 49 &= 1\cdot 42+\boxed{7}\\ 42 &= 6\cdot 7+0\end{aligned}$$

The last remainder that isn't zero is 7, so $\gcd(4711,777)=7$. Then we have

$$\operatorname{lcm}(4711,777)=\frac{4711\cdot 777}{7}=\left\{\begin{array}{l}4711\cdot\dfrac{777}{7}=4711\cdot 111\\ \dfrac{4711}{7}\cdot 777=673\cdot 777\end{array}\right\}=522\,921$$

Note that the calculations are simplified if in this way you divide *before* multiplying! ■

By going through the Euclidian algorithm backwards (as we did in the concrete explanation) we found that it's not just the case that every linear combination of the numbers is a multiple of the gcd but also that it is possible to express the gcd as a linear combination of the numbers. This will prove to have very interesting applications (as soon as in the next section). We repeat this as a theorem:

Theorem 3.5: gcd ***as a Linear Combination*** $\gcd(m,n)$ can be written as a linear combination of the numbers m and n. ■

We repeat the procedure:

Method 3.3: ***To Write the*** **gcd** ***as a Linear Combination*** We want to write $\gcd(4711,777)$ as a linear combination of 4711 and 777.

In example 3.2 we found that $\gcd(4711,777)=7$. Now we'll go through the running of the Euclidian algorithm there, and extract the remainders:

$$\begin{aligned}7 &= 49-1\cdot 42\\ 42 &= 777-15\cdot 49\\ 49 &= 4711-6\cdot 777\end{aligned}$$

Now we'll insert the remainders in the first expression step by step:

$$\begin{aligned}7 &= 49-1\cdot 42 && \\ &= 49-1\cdot(777-15\cdot 49) && \text{Replace}\\ &= 16\cdot 49-1\cdot 777 && \text{Collect}\\ &= 16\cdot(4711-6\cdot 777)-1\cdot 777 && \text{Replace}\\ &= 16\cdot 4711-97\cdot 777 && \text{Collect}\end{aligned}$$

So we have found that $16\cdot 4711-97\cdot 777=7$.

But is this the *only* way of writing $\gcd(4711,777)$ as a linear combination of 4711 and 777? No, using the equality $\operatorname{lcm}(4711,777)=522{,}921=111\cdot 4711=673\cdot 777$ we can find other possible linear combinations:

$$\begin{aligned}7 &= 16\cdot 4711-97\cdot 777 && \\ &= 16\cdot 4711-97\cdot 777+k\cdot 0 && \text{Zero doesn't change the value}\\ &= 16\cdot 4711-97\cdot 777+k(522{,}921-522{,}921) && \text{Insert lcm,}\\ &= 16\cdot 4711-97\cdot 777+k(111\cdot 4711-673\cdot 777) && \text{in two different ways}\\ &= 16\cdot 4711-97\cdot 777+111k\cdot 4711-673k\cdot 777 && \text{Expand}\\ &= (16+111k)\cdot 4711-(97+673k)\cdot 777 && \text{Collect}\end{aligned}$$

Thus there is an infinite number of ways of writing 7 as a linear combination of 4711 and 777, one for each possible value of k.

What we did in the third step was to add *and* subtract $\operatorname{lcm}(4711,777)$, and since this number can be written both as a multiple of 4711 and of 777 it gave us the "extra" possibilities we were looking for. ■

Exercise 3.18: *Important!* Explain why by using this method we'll get *all* the ways of writing the gcd as a linear combination of the numbers.

Exercise 3.19 Find the gcd and lcm for these numbers, and write the gcd as a linear combination of the numbers in at least two different ways.

(a) 408 and 672. **(b)** 527 and 300.

Exercise 3.20 Mark 6, 10, $\gcd(6,10)$, and $\text{lcm}(6,10)$, and 20, 15, $\gcd(20,15)$, and $\text{lcm}(20,15)$ in the graph depicting the divisors of 60 that you drew in exercise 3.14 on page 42. Comments?

Exercise 3.21 Work out this fraction, and simplify the answer as much as possible. Do not use a calculator, nor prime factorisation!

$$\frac{277}{777} - \frac{136}{399}$$

Proof of the Fundamental Theorem of Arithmetic

Now we have the foundations needed to prove the fundamental theorem of aritmethic. But first we need some lemmas (theorems used to prove more interesting theorems).

Theorem 3.6: ***Divisibility of a Product*** If a prime p divides a product ab it has to divide one of the factors of the product.

Proof If p divides a we are done; p divides one of the factors. What happens if p *doesn't* divide a?

If p doesn't divide a then p and a have to be coprime, since p is a prime only divisible by 1 and itself, and the last factor we assumed wasn't part of a. So $\gcd(p,a) = 1$. Using the Euclidian algorithm we can write the gcd as a linear combination of the numbers, so there are integers x and y such that

$$px + ay = 1 \quad \Rightarrow \quad b(px + ay) = b \cdot 1 \quad \Rightarrow \quad pbx + \underbrace{ab}_{\text{divisible by } p,\text{ according to premises}} y = b$$

b can be written as a sum of two terms that are both divisible by p, so b is divisible by p.

So if the first factor isn't divisible by p then the other one will be. Then we are guaranteed that at least one (perhaps both)of the factors is divisible by p. ■

Note that the theorem only says that this holds if p is a prime number!

Exercise 3.22

(a) Find three numbers p, a, and b where p isn't a prime and where the conclusion of the theorem isn't true.

(b) Find three numbers p, a, and b where p isn't a prime but where the conclusion of the theorem is true anyway.

The theorem can be generalised to products of an arbitrary number of factors:

Theorem 3.7: ***Corollary***[1] If a prime number p divides a product $a_1a_2a_3\ldots a_n$ then p divides one of the factors a_1, a_2, a_3, $\ldots$, a_n. (It is quite possible that p divides more than one factor, or all of them, but of main interest to us is that it divides *at least* one of them.)

Proof Either p divides the first number a_1, and then we are done, or p *doesn't*, and then p (according to the previous theorem) has to divide the other half of the expression: $a_2a_3\ldots a_n$. Repeat the line of argument until only a_n is left. ■

Now we have the tools needed for our main purpose: to prove the fundamental theorem itself.

Theorem 3.2: ***Fundamental Theorem of Arithmetic*** Every integer greater than one can be written as a product of prime numbers in *one and only one* way (if the order of the factors doesn't matter).

Proof We have already (on page 39) shown that it can be done in *at least* one way. Now let's assume that we have divided the number a into prime factors in two ways: as $a = p_1p_2p_3\ldots p_m$ and as $a = q_1q_2q_3\ldots q_n$. We are going to show that these factorisations as a matter of fact consist of the same numbers (though possibly in a different order).

From the first factorisation we see that p_1 divides a. Since a is a product of a lot of q:s and p_1 is a prime number, according to the theorem we just proved, p_1 has to divide one of the q:s, let's say q_i. Since q_i is a prime that only has the divisors one and itself, and since $p_1 \neq 1$, we must have $p_1 = q_i$. We can then delete these two identical factors and repeat the procedure on what's left of the factorisations. At last we will have paired all p-values with q-values. Therefore, the factorisations consist of the same numbers, which was to be proven. ■

Here as well it might be best to point out that the argument only holds for factorisation into *primes*. For instance, 36 kan be factorised both as $6 \cdot 6$ and as $4 \cdot 9$, but these factorisations definitely don't consist of the same numbers! (But if we keep on factorising, we get $2 \cdot 3 \cdot 2 \cdot 3$ and $2 \cdot 2 \cdot 3 \cdot 3$ respectively, and those factorisations do consist of the same numbers, but in a different order.)

[1] A corollary is a theorem that follows directly from another theorem.

3.2.5 Diofantine Equations

A diofantine equation is an equation where only *integer solutions* are accepted. Usually, diofantine equations are more difficult to solve than equations without this restriction, but there is a class that is easily solved using the Euclidian algorithm. It consists of **linear equations** of the type

$$ax + by = c$$

where a, b, c, x, and y are integers. We will here show how to proceed:

Method 3.4: ***Diofantine Equations*** Since complications can occur, we'll demonstrate the method using three different equations.

Equation 1 Solve the equation

$$69x + 49y = 5, \qquad x, y \in \mathbb{Z}$$

We start by looking for the gcd of the **coefficients** 69 and 49 with the help of the Euclidian algorithm, and we'll also extract the remainders:

$$\begin{aligned} 69 &= 1 \cdot 49 + 20 & 20 &= 69 - 49 \\ 49 &= 2 \cdot 20 + 9 & 9 &= 49 - 2 \cdot 20 \\ 20 &= 2 \cdot 9 + 2 & 2 &= 20 - 2 \cdot 9 \\ 9 &= 4 \cdot 2 + \boxed{1} & 1 &= 9 - 4 \cdot 2 \\ 2 &= 2 \cdot 1 \end{aligned}$$

$\gcd(69, 49) = 1$, that is: the numbers are coprime. Now we express the gcd as a linear combination of the numbers using method 3.3 on page 47:

$$\begin{aligned} 1 &= 9 - 4 \cdot 2 = 9 - 4 \cdot (20 - 2 \cdot 9) = 9 \cdot 9 - 4 \cdot 20 \\ &= 9 \cdot (49 - 2 \cdot 20) - 4 \cdot 20 = 9 \cdot 49 - 22 \cdot 20 = 9 \cdot 49 - 22 \cdot (69 - 49) \\ &= 31 \cdot 49 - 22 \cdot 69 \end{aligned}$$

(By the way, it is a safety precaution to check that what you have found – in this case $31 \cdot 49 - 22 \cdot 69$ – really equals the gcd before basing any further steps in the calculation on that result. It's very easy to make computational errors in these calculations.)

We rearrange the linear combination a bit, so that we get it on the form "69(some number) + 49(some number) = some number", just like the equation:

$$1 = 31 \cdot 49 - 22 \cdot 69 \qquad \Leftrightarrow \qquad 69 \cdot (-22) + 49 \cdot 31 = 1$$

If we multiply this with the desired right-hand side 5 we get

$$69 \cdot \underbrace{(-110)}_{x} + 49 \cdot \underbrace{155}_{y} = 5$$

so *one* solution to the equation is

$$x = -110, \qquad y = 155$$

but is this the only solution? No, just as in method 3.3 we can add and subtract the least common multiple (which in this case is simply $69 \cdot 49$, since the numbers are coprime).

$$\begin{aligned} 69 \cdot (-110) + 49 \cdot 155 + k \cdot 0 &= 5 \\ 69 \cdot (-110) + 49 \cdot 155 + k \cdot (69 \cdot 49 - 69 \cdot 49) &= 5 \\ 69 \cdot (-110) + 49 \cdot 155 + 49 \cdot 69k - 69 \cdot 49k &= 5 \\ 69 \cdot \underbrace{(-110 - 49k)}_{x} + 49 \cdot \underbrace{(155 + 69k)}_{y} &= 5 \end{aligned}$$

So the set of pairs (x, y) where

$$x = -110 - 49k, \quad y = 155 + 69k \qquad k \in \mathbb{Z}$$

is the **complete solution** to the equation. (A *solution* is something that fits into the equation. A *complete solution* contains *everything* that fits into the equation.) That there aren't any more solutions than these we can motivate in the same way as in exercise 3.18 on page 48.

Note that we mustn't add the lcm until *after* we have scaled the equation to get the correct right-hand side, otherwise we'll miss a number of solutions!

Exercise 3.23 Which solutions would we miss?

Exercise 3.24 List the solutions to the equation that correspond to $-5 \leq k \leq 5$.

Equation 2 Solve the equation

$$69x + 48y = 5, \qquad x, y \in \mathbb{Z}$$

We look for the gcd of the coefficients 69 och 48:

$$\begin{aligned} 69 &= 1 \cdot 48 + 21 \\ 48 &= 2 \cdot 21 + 6 \\ 21 &= 3 \cdot 6 + \boxed{3} \\ 6 &= 2 \cdot 3 \end{aligned}$$

$\gcd(69, 48) = 3$, so every linear combination of these number has to include the factor 3. So $69x + 48y$ will be a multiple of 3, which the desired right-hand side 5 isn't. Thus the equation is *unsolvable*.

Equation 3 Solve the equation

$$69x + 48y = 6, \qquad x, y \in \mathbb{Z}$$

We look for the gcd:

$$\begin{aligned} 69 &= 1 \cdot 48 + 21 & 21 &= 69 - 48 \\ 48 &= 2 \cdot 21 + 6 & 6 &= 48 - 2 \cdot 21 \\ 21 &= 3 \cdot 6 + \boxed{3} & 3 &= 21 - 3 \cdot 6 \\ 6 &= 2 \cdot 3 \end{aligned}$$

$\gcd(69, 48) = 3$, but since the right-hand side is a multiple of 3 this isn't a problem. We extract the gcd:

$$\begin{aligned} 3 = 21 - 3 \cdot 6 = 21 - 3 \cdot (48 - 2 \cdot 21) &= 7 \cdot 21 - 3 \cdot 48 \\ &= 7 \cdot (69 - 48) - 3 \cdot 48 = 7 \cdot 69 - 10 \cdot 48 \end{aligned}$$

We have

$$69 \cdot 7 + 48 \cdot (-10) = 3 \quad \Rightarrow \quad 69 \cdot \underbrace{14}_{x} + 48 \cdot \underbrace{(-20)}_{y} = 6$$

So now we have *one* solution. Now we augment it with $\operatorname{lcm}(69, 48) = (69 \cdot 48)/3 = 69 \cdot 16 = 23 \cdot 48$. That gives us

$$\begin{aligned} 69 \cdot 14 + 48(-20) + k \cdot (69 \cdot 16 - 23 \cdot 48) &= 6 \\ 69 \cdot 14 + 48(-20) + 69 \cdot 16k + 48 \cdot (-23k) &= 6 \\ 69 \cdot \underbrace{(14 + 16k)}_{x} + 48 \cdot \underbrace{(-20 - 23k)}_{y} &= 6 \end{aligned}$$

The complete solution to the equation is

$$x = 14 + 16k, \quad y = -20 - 23k \qquad k \in \mathbb{Z}$$

■

Exercise 3.25 If in equation 3 we had instead added $k(23 \cdot 48 - 69 \cdot 16)$ we would get the answer $x = 14 - 16k$, $y = -20 + 23k$, $k \in \mathbb{Z}$. Can both this answer and the one given in the description of the method be correct?

Exercise 3.26 Solve the equations:

(a) $8x + 9y = 3$ **(b)** $36x + 78y = 18$ **(c)** $84z + 119w = 134$

In some cases, not quite all the solutions to the equation are solutions to the problem modelled with the help of the equation:

Example 3.3: ***Diofantine Equation in Accountancy*** The treasurer in a housing association is pondering about a very old receipt. The final sum is 1284 crowns and 50 öre, and the items purchased were light-bulbs and floorcloths. These items are to be registered on different accounts, but unfortunately the receipt doesn't tell how much of the sum that represents light-bulbs. From two remaining packages it can be found that a packet of light-bulbs did cost 33:50, and a packet of floorcloths 14:50. Is it possible based on this to find out how many packages of either kind the purchase must have included?

The numbers have to be positive integers; everything else is absurd. Unfortunately, the prices aren't integers. But if we change the currency to 50 öre coins, we find that the cost of a packet of light-bulbs was 67 coins, a packet of floorcloths 29 coins, and the total sum was 2569 coins. If we use x to denote the number of packages of light-bulbs and y the number of packages of floorcloths, we find that we have to solve the diofantine equation

$$67x + 29y = 2569$$

We look for the gcd of the coefficients:

$$\begin{aligned} 67 &= 2 \cdot 29 + 9 & 9 &= 67 - 2 \cdot 29 \\ 29 &= 3 \cdot 9 + 2 & 2 &= 29 - 3 \cdot 9 \\ 9 &= 4 \cdot 2 + 1 & 1 &= 9 - 4 \cdot 2 \\ 2 &= 2 \cdot 1 \end{aligned}$$

and express it as a linear combination:

$$\begin{aligned} 1 = 9 - 4 \cdot 2 = 9 - 4(29 - 3 \cdot 9) &= 13 \cdot 9 - 4 \cdot 29 \\ &= 13 \cdot (67 - 2 \cdot 29) - 4 \cdot 29 = 13 \cdot 67 - 30 \cdot 29 \end{aligned}$$

Now we have

$$67 \cdot 13 + 29(-30) = 1 \quad \Rightarrow \quad 67 \cdot 33{,}397 + 29 \cdot (-77{,}070) = 2569$$

and expanding using $\operatorname{lcm} = 67 \cdot 29$ we get

$$67 \cdot (33{,}397 - 29k) + 29 \cdot (-77{,}070 + 67k) = 2569$$

so

$$\begin{cases} x = 33{,}397 - 29k \\ y = -77{,}070 + 67k \end{cases} \qquad \text{where } k \in \mathbb{Z}$$

is the complete solution to the equation.

Now we have to choose a suitable value of k, so that both x and y will be positive (which we realised that they had to be). $33{,}397/29 \approx 1151.6$ means that k can be at most 1151 if x is to be positive. Further $77{,}070/67 \approx 1150.3$,

which means that k must be at least 1151 if y is to be positive. The only number that is both at least and at most 1151 is 1151 itself, so the answer is

$$\begin{cases} x = 33{,}397 - 29 \cdot 1151 = 18 \\ y = -77{,}070 + 67 \cdot 1151 = 47 \end{cases}$$

Since we got a unique answer we can, with a clear conscience, enter 18 packages of light-bulbs at 33:50 each and 47 packages of floorcloths at 14:50. ∎

Exercise 3.27 Can you think of some other way of solving the problem in the example above?

Exercise 3.28 How many solutions are there to the last equation in method 3.4 on page 50 where both the numbers are positive?

Exercise 3.29 Try to think of a way to proceed if you want to solve a linear diofantine equation containing *three* unknowns. If you want to, try out your method on the equation $15x + 20y + 9z = 1$.

3.3 Modular Arithmetic

Example 3.4: ***Counting Time*** "Now the time is 23.00. What will it be in 5 hours?"

A first effort to solve this problem will probably give the answer "$23 + 5 = 28$ o'clock". Studying this answer, one will swiftly classify it as absurd, since the clock won't show anything higher than 23.59, and then it restarts at 00.00. Because of this we reduce the answer by 24 hours, and answer $28 - 24 = 4$. ∎

This example showed something called **modular arithmetic** (also known as **clock arithmetic**, for fairly obvious reasons). The part of the answer that was of interest to us was the *remainder* after division by 24. The quotient (which is 1) we just threw away. This phenomenon, that one cares about the remainders and ignores the quotients, is fairly common. When working with the remainders from divison with the integer n one is said to count **modulo** n or to be **counting in** $\mathbb{Z}_n$.

Two numbers having the same remainder are called **equivalent modulo** n or **congruent modulo** n. Equivalence/congruence is denoted $\equiv$ (even if carelessness is common and $=$ is used, especially if $\equiv$ can't be found on the keyboard). You can for instance write

$$17 \equiv 7 \equiv 2 \pmod 5 \quad \text{or} \quad 17 \equiv 7 \equiv 2 \quad \text{in } \mathbb{Z}_5$$

since

$$17 = 3 \cdot 5 + 2 \quad \text{and} \quad 7 = 1 \cdot 5 + 2 \quad \text{and} \quad 2 = 0 \cdot 5 + 2$$

Usually when working modularly one wants to have the answers "scaled down" to the principal remainder, which lies between 0 and $n - 1$, but nothing prevents the usage of numbers outside this interval in the calculations.

Exercise 3.30 When someone is saying $x \equiv 7 \pmod 8$, what is meant is "x is one of all the numbers having the remainder 7 when divided by 8". Write the same thing using set notation.

3.3.1 Calculations in Modular Arithmetic

Three out of the four rules of arithmethic – addition, subtraction, and multiplication – work excellently in modular arithmetic. We'll take a look at them; division we'll get back to later.

Example 3.5: ***Modular addition*** We want to add 27 and 35 modulo 12.

We can start either by adding the numbers and then reduce modularly

$$27 + 35 = 62 = 5 \cdot 12 + 2 \equiv 2 \pmod{12}$$

or first reduce modularly, afterwards add, and then reduce again

$$\begin{aligned} 27 + 35 &= (2 \cdot 12 + 3) + (2 \cdot 12 + 11) \\ &\equiv 3 + 11 = 14 = 1 \cdot 12 + 2 \equiv 2 \pmod{12} \end{aligned}$$

The advantage of this version was that we got smaller numbers to work with in the calculation, the disadvantage being that we had to reduce modularly several times.

We could also do it like this:

$$27 + 35 = (2 \cdot 12 + 3) + (3 \cdot 12 - 1) \equiv 3 + (-1) = 2 \pmod{12}$$

It is fully permitted to switch to negative numbers in the intermediate results. Here, we had easier numbers to work with, and on top of that we didn't have to do the final reduction.

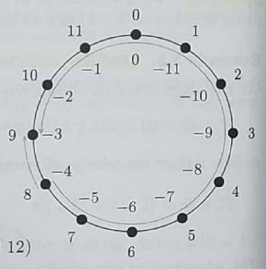
Concerning the notation: We have been using an ordinary "$=$" for equalities that hold in normal arithmetic as well, and "$\equiv$" in those that only hold when working modularly. ∎

Usually, all intermediate results are reduced when doing modulo calculations, so that the numbers used in the work are small. But no matter how we organise the calculations here, we are going to get the same final result: 2. And in exactly the same way subtraction and multiplication are done.

Exercise 3.31 Evaluate $27 \cdot 35 \pmod{12}$ in the same way as in the example, trying all three of the ways. Which one was in this case the simplest and which one the most complicated? (Do *not* use a calculator.)

"Normal arithmetic" can be illustrated using the number line. Modular arithmetic can be illustrated using a **number circle**. If you are going to work modulo 24, you can draw a clock with 24 markings, 0–23. If you then want to calculate $17+11$ you can start in 17 and the go 11 steps clockwise, and see where you end up. If you are going to make a lot of calculations modulo a small number it can be worth it to draw the number circle. It can be of great help, for instance when you have to convert negative numbers to positive ones swiftly.

Example 3.6: ***Number Circle*** We are going to work on a number of problems modulo 12, and therefore we draw a number circle. Since we are going to both add and subtract, at every point we note both which positive number it represents and which negative one.



Now we want to compute $8-11$. Graphically, that represents starting at 8 and going 11 steps backwards. (The inner arrow shows this.) We end up at 9. -11 is equivalent to 1, which means that we can get to the same point by going 1 step forwards instead. (The outer arrow shows this.)

Computationally we can do it like this:

$$8-11=8+(-11)\equiv 8+1=9 \pmod{12}$$

or

$$8-11=-3\equiv 9 \pmod{12}$$

In both cases we are helped by the fact that we can convert negative numbers to positive ones using the number circle. ∎

Exercise 3.32 Mark the two versions of $5-10 \pmod{12}$ in the number circle. *

Exercise 3.33 Model the days of the week using modular arithmetic. Use your model to answer the question "what day of the week will it be in 100 days if today is Tuesday?".

Exercise 3.34 Try to think of some application of modular arithmetic that doesn't have anything to do with time.

3.3.2 solving in Modular Arithmetic

Now we arrive at the fourth rule of arithmetic: division. Since modular arithmetic is a variant of integer calculation while division (in most cases) gives fractions as results some sort of problem has to arise. They are in fact

so large that the decision has been made not to talk about division at all in modular arithmetic, but to talk about *equationsolving* instead. One doesn't try to calculate "b/a" but instead one "solves the equation $ax=b$". We shall take a look at what can happen then:

Example 3.7: ***Calculations in $\mathbb{Z}_6$*** We write an addition table and a multiplication table for computations modulo 6. (Actually, we are not going to use the addition table for anything, but it can be fun to see what it looks like.) There is no reason to include any numbers outside the inteval 0–5, since every number outside this interval is equivalent to some number inside it, and gives a result equivalent to something inside the table.

$+$	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

$\cdot$	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

In the addition table we have for instance 1 in row 3, column 4 because $3+4=7=6+1\equiv 1 \pmod 6$.

Now let's say that we want to solve the equation $2x\equiv 4 \pmod 6$. That can be done with the help of the multiplication table, by scanning row 2 to see where we find a 4. That we do in two places! The equation has two different solutions; the normal one $x\equiv 2$ but also $x\equiv 5$.

If instead we want to solve the equation $2x\equiv 5 \pmod 6$, that one is unsolvable, since there is no 5 in row 2.

The equation $5x\equiv 2 \pmod 6$, finally, has just one solution: $x\equiv 4$, since there is only one 2 in row 5. ∎

This shows fairly well why one has chosen not to use division. An expression like $4/2$ should only mean *one* thing; if there are several alternative values the normal methods for calculation stop working.

Example 3.8: ***Calculations in $\mathbb{Z}_7$*** We can try doing the same thing, but for calculations modulo 7:

$+$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	2	3	4	5	6	0
2	2	3	4	5	6	0	1
3	3	4	5	6	0	1	2
4	4	5	6	0	1	2	3
5	5	6	0	1	2	3	4
6	6	0	1	2	3	4	5

$\cdot$	0	1	2	3	4	5	6
0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6
2	0	2	4	6	1	3	5
3	0	3	6	2	5	1	4
4	0	4	1	5	2	6	3
5	0	5	3	1	6	4	2
6	0	6	5	4	3	2	1

Here the multiplication table doesn't look at all as weird as in $\mathbb{Z}_6$. All numbers in a row are different (except in row 0), and that also means that every number can be found in the row. If we want to solve an equation of the type $ax \equiv b \pmod 7$ we can count on it having one and only one solution. ∎

Exercise 3.35 Can you find something that the "weird" rows in the multiplication table for $\mathbb{Z}_6$ have in common, and that distinguish them from the "normal" rows?

Exercise 3.36 The last row and column in both the multiplication tables look quite tidy. Why?

Exercise 3.37 Can you see some more symmetries in the multiplication table? *

The reason why calculations in $\mathbb{Z}_6$ proved to be so much stranger than in $\mathbb{Z}_7$ we'll get back to after finding out how to solve a problem of this type computationally. (If you have to solve an equation in $\mathbb{Z}_{2^{1024}}$ – an in practical applications completely normal modulo – you will prefer not to have to do it by writing down the multiplication table...)

Method 3.5: ***Solving a Modular Equation*** We want to solve

$$6x \equiv 8 \quad \text{in } \mathbb{Z}_{17}$$

Saying that $6x \equiv 8 \pmod{17}$ is by definition the same thing as saying that $6x = 8 + 17y$. Thus we have to solve the equation

$$6x = 8 + 17y \quad \Leftrightarrow \quad 6x - 17y = 8 \quad \text{where } x, y \in \mathbb{Z}$$

This is a diofantine equation, of a kind that we know how to solve!

We look for the gcd of the coefficients, and express it as a linear combination:

$$\begin{aligned} 17 &= 2 \cdot 6 + 5 & 1 &= 6 - 5 \\ 6 &= 1 \cdot 5 + 1 & &= 6 - (17 - 2 \cdot 6) \\ 5 &= 5 \cdot 1 & &= 3 \cdot 6 - 1 \cdot 17 \end{aligned}$$

The numbers are coprime (which isn't a big surprise, since 17 is a prime, and 6 – which is smaller than 17 – can't possibly be a multiple of 17). Now we have

$$6 \cdot 3 - 17 \cdot 1 = 1 \quad \Leftrightarrow \quad 6 \cdot 24 - 17 \cdot 8 = 8$$

The equation has the solution $x = 24$. If we add the lcm the answer will be expanded to $24 + 17k$, but since we count modulo 17, where 17 is the same thing as 0, this doesn't matter. And the value of y is of no importance!

The answer can be scaled down to a number in the interval between 0 and 16, like

$$x = 24 = 17 + 7 \equiv 7 \pmod{17}$$

We can check the answer (which is a good habit to acquire):

$$6 \cdot 7 = 42 = 2 \cdot 17 + 8 \equiv 8 \pmod{17}$$

The answer is correct. ∎

We test the method on two similar equations:

Example 3.9 Solve the equation

$$6x \equiv 7 \quad \text{in } \mathbb{Z}_{16}$$

The equation can be rewritten as the diofantine equation

$$6x - 16y = 7$$

Here we can see directly that since both 6 and 16 are even numbers the left-hand side has to be even, and that doesn't fit so well combined with the odd right-hand side 7. This equation is *unsolvable*. ∎

Example 3.10 Solve the equation

$$6x \equiv 8 \quad \text{in } \mathbb{Z}_{16}$$

The equation can be rewritten as

$$6x - 16y = 8$$

We note directly that $\gcd(6, 16) = 2$, but we run the Euclidian algorithm anyway to get the gcd as a linear combination of the numbers.

$$\begin{aligned} 16 &= 2 \cdot 6 + 4 & 2 &= 6 - 4 \\ 6 &= 1 \cdot 4 + 2 & &= 6 - (16 - 2 \cdot 6) = 3 \cdot 6 - 16 \\ 4 &= 2 \cdot 2 \end{aligned}$$

This gives us $\operatorname{lcm}(6, 16) = 6 \cdot 16 / \gcd(6, 16) = 48 = 6 \cdot 8 = 3 \cdot 16$. Now we have

$$\begin{aligned} 6 \cdot 3 - 16 \cdot 1 &= 2 \\ 6 \cdot 12 - 16 \cdot 4 &= 8 \\ 6 \cdot 12 - 16 \cdot 4 + k(48 - 48) &= 8 \\ 6 \cdot 12 - 16 \cdot 4 + 6 \cdot 8k - 16 \cdot 3k &= 8 \\ 6(12 + 8k) - 16(4 + 3k) &= 8 \end{aligned}$$

All x-values on the form $12 + 8k$ are solutions to the equation. Out of those, two (the same number as the gcd), namely 12 and 4, belong to the interval 0–15. Thus the equation has two solutions in $\mathbb{Z}_{16}$:

$$x \equiv 4 \quad \text{and} \quad x \equiv 12$$

We can verify the solutions like this:

$$\begin{aligned} 6 \cdot 4 &= 24 = 16 + 8 \equiv 8 \pmod{16} \\ 6 \cdot 12 &= 72 = 4 \cdot 16 + 8 \equiv 8 \pmod{16} \end{aligned}$$

∎

Exercise 3.38 Solve the equation $8x = 10$

(a) in $\mathbb{Z}_{14}$ **(b)** in $\mathbb{Z}_{15}$ **(c)** in $\mathbb{Z}_{16}$

Exercise 3.39 Solve the equation $136x = 119$

(a) in $\mathbb{Z}_{255}$ **(b)** in $\mathbb{Z}_{256}$ **(c)** in $\mathbb{Z}_{257}$

Exercise 3.40 Explain why the number of solutions to a solvable equation of the type $ax \equiv b \pmod{n}$ is always equal to $\gcd(a, n)$. *

Exercise 3.41 Six children are going to play tag and use a counting-out rhyme to decide who is going to be "it". The chosen one is the person where the counting started. How many words did the rhyme consist of?

Now we can try to analyse the calculations that we have been doing:

The fact that no problems occurred when solving the equation $6x \equiv 8 \pmod{17}$ in the demonstration of the method was because the coefficient $a = 6$ and the modular base $n = 17$ were coprime. Then $\gcd(a, n) = 1$, and there will never be any problems scaling to the correct right-hand side. At the same time, $\text{lcm}(a, n) = a \cdot n$, which causes the different solutions for x to be placed at the distance n – one revolve around the number circle – from each other. From a practical point of view, we then have only one solution.

If a and n *aren't* coprime, the situation will be more complicated. If $\gcd(a, n)$ isn't a divisor of the right-hand side b the equation is unsolvable, like $6x \equiv 7 \pmod{16}$. Otherwise, the equation is solvable, but there will be several solutions, as with $6x \equiv 8 \pmod{16}$.

The equation $ax \equiv 1 \pmod{n}$ is only solvable if $\gcd(a, n) = 1$. And if it is, then all other equations of the type $ax \equiv b \pmod{n}$ are solvable as well, and furthermore they have only one solution in $\mathbb{Z}_n$. Because of this, some terminology has been created concerning exactly this equation:

Definition 3.1: ***Inverse in Modular Arithmetic*** A number a in $\mathbb{Z}_n$ is called **invertible** if there exists some number in $\mathbb{Z}_n$ that multiplied with a gives 1. The number in question is then called the **inverse** of a and is denoted a^{-1}. Otherwise, a is said to be non-invertible. ■

For instance, in method 3.5 we saw that $6 \cdot 3 \equiv 1 \pmod{17}$. Then 6 is invertible in $\mathbb{Z}_{17}$, and $6^{-1} = 3$. If you know this, the equation $6x \equiv 8 \pmod{17}$ can be solved as follows:

$$\begin{aligned} 6x &\equiv 8 \\ 3 \cdot 6x &\equiv 3 \cdot 8 \\ 1x &\equiv 24 \\ x &\equiv 24 \pmod{\mathbb{Z}_{17}} \end{aligned} \qquad \begin{aligned} ax &\equiv b \\ a^{-1}ax &\equiv a^{-1}b \\ 1x &\equiv a^{-1}b \\ x &\equiv a^{-1}b \pmod{\mathbb{Z}_n} \end{aligned}$$

(In the column to the right we have shown the general principle.) If a has an inverse then the equation $ax \equiv b \pmod{n}$ has exactly one solution, namely $x \equiv a^{-1}b$.

Furthermore, we have:

Theorem 3.8: ***Invertibility in Modular Arithmetic*** A number a in $\mathbb{Z}_n$ is invertible if and only if a and n are coprime (that is: if $\gcd(a, n) = 1$). ■

If you actually want to *find* the inverse (if it exists) you solve the equation $ax \equiv 1 \pmod{n}$. But otherwise inverses and invertibility are mainly theoretical concepts, that are used in arguments. If you want to solve a concrete problem with given numbers, it is usually simpler to just find the answer directly, without bothering about the inverse.

Anyway, we can now explain why the calculations in $\mathbb{Z}_6$ are more strange than the ones in $\mathbb{Z}_7$. Since 7 is a prime all the numbers except 0 in $\mathbb{Z}_7$ are coprime to the base; in $\mathbb{Z}_6$ a great number are not.

Exercise 3.42 Study the multiplication tables in examples 3.7 and 3.8 on page 57.

(a) Which numbers have an inverse when calculating in $\mathbb{Z}_6$, and which are the inverses?

(b) The same question about $\mathbb{Z}_7$.

Exercise 3.43: *Important!* In "normal" arithmetic we have the important rule of the **zero factor law**. This says that if a product is zero then one of the factors has to be zero; it's impossible to get zero by multiplication in any other way. If you look at the multiplication tables on page 57 you'll see that this is not a given fact in modular arithmetic.

(a) In the table showing $\mathbb{Z}_6$ there are a number of **zero divisors**, that is to say, numbers other than zero that can be factors of zero. Explain why there always exist zero divisors if you calculate modulo a composite number.

(b) In the table showing $\mathbb{Z}_7$, on the other hand, there are only zeros in row and column zero. Explain why there can't exist any zero divisors when you calculate modulo a prime number.

Those who have studied linear algebra may note the similarities to matrix algebra. If the matrix A has an inverse then the equation $\mathsf{AX} = \mathsf{B}$ has a unique solution: $\mathsf{X} = \mathsf{A}^{-1}\mathsf{B}$. Otherwise, there will be either no solution or several ones, depending on the value of B. Furthermore, you *can* solve a matrix equation by calculating the inverse (if it exists). But that is usually not done, since there are much more efficient methods, that work besides, even if the inverse doesn't exist.

The concept *inverse* is thus used in many different contexts, not just in modular arithmetic. It will for example appear again in Chapter 8.

3.4 Number Bases

It isn't just in problems concerning divisors and the Euclidian algorithm that division with remainders can be used. Here comes another completely different application: representation of numbers.

When a number is written as 12,345 it is understood that the number is given in the **number base** 10. 12,345 can be read as $1 \cdot 10{,}000 + 2 \cdot 1000 + 3 \cdot 100 + 4 \cdot 10 + 5 \cdot 1 = 1 \cdot 10^4 + 2 \cdot 10^3 + 3 \cdot 10^2 + 4 \cdot 10^1 + 5 \cdot 10^0$.

That 10 has been chosen as the number base is probably caused by the fact that the truely basic way of counting is counting on one's fingers, and the usual number of those is 10. Apart from that, there is nothing that says that 10 has any especially magical properties that makes it superiorly suitable as a base for a number system. On the other hand, approximately the same problems arise irrespective of the base chosen, so there is no pressing need to change the standards. This common number system is called **decimal**.

> Exercise 3.44 Exactly what are the problems that can arise, and that are connected to the number base?

> Exercise 3.45 This way of writing numbers is called the **positional system** or **place-value system**. Why, and what other kinds of notations can be imagined?

3.4.1 The Binary Number System

In the computer industry components that have two modes are often used: A switch can be open or closed, a lamp can be on or off, a picture element on the screen can be black or white, and so on. In these contexts, it is natural to use the number 2 as the base. For the rest, the digits 0 and 1 are used, with the same meaning as usual. This is called the **binary number system**. A **binary digit** is often called a **bit**.

Since we aren't computers, methods are needed to get between the binary and decimal representation of numbers.

Method 3.6: ***Converting from Binary to Decimal*** What is the binary number 1001101 converted to decimal form?

1001101 is to be interpreted as

$$\begin{aligned} &1 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \\ &\quad = 1 \cdot 64 + 0 \cdot 32 + 0 \cdot 16 + 1 \cdot 8 + 1 \cdot 4 + 0 \cdot 2 + 1 \cdot 1 = 77 \end{aligned}$$

It's exactly the same principle as for decimal numbers, but every digit is multiplied with a power of the number base 2 instead. ■

It is, as shown here, easy to convert binary notation to decimal. The other direction is a bit trickier, and you have to use the division algorithm.

Method 3.7: ***Conversion from Decimal to Binary*** Convert the decimal number 777 to binary notation.

We halve the number repeatedly, until nothing is left. (The point of this will appear shortly.)

$$\begin{aligned} 777 &= 2 \cdot 388 + 1 \\ 388 &= 2 \cdot 194 + 0 \\ 194 &= 2 \cdot 97 + 0 \\ 97 &= 2 \cdot 48 + 1 \\ 48 &= 2 \cdot 24 + 0 \\ 24 &= 2 \cdot 12 + 0 \\ 12 &= 2 \cdot 6 + 0 \\ 6 &= 2 \cdot 3 + 0 \\ 3 &= 2 \cdot 1 + 1 \\ 1 &= 2 \cdot 0 + 1 \end{aligned}$$

Now there is no point in continuing, since after this every step will look like "$0 = 2 \cdot 0 + 0$".

By piecing together the steps in the calculation we see that

$$\begin{aligned} 777 &= 2 \cdot 388 + 1 \\ &= 2(2 \cdot 194 + 0) + 1 && \text{Replace 388 with } 2 \cdot 194 + 0 \\ &= 2(2(2 \cdot 97 + 0) + 0) + 1 && \text{Replace 194 with } 2 \cdot 97 + 0 \\ &= 2(2(2(2 \cdot 48 + 1) + 0) + 0) + 1 && \text{And so on...} \\ &= 2(2(2(2(2 \cdot 24 + 0) + 1) + 0) + 0) + 1 \\ &= 2(2(2(2(2(2 \cdot 12 + 0) + 0) + 1) + 0) + 0) + 1 \\ &= 2(2(2(2(2(2(2 \cdot 6 + 0) + 0) + 0) + 1) + 0) + 0) + 1 \\ &= 2(2(2(2(2(2(2(2 \cdot 3 + 0) + 0) + 0) + 0) + 1) + 0) + 0) + 1 \\ &= 2(2(2(2(2(2(2(2(2 \cdot 1 + 1) + 0) + 0) + 0) + 0) + 1) + 0) + 0) + 1 \\ &= 2^9 \cdot 1 + 2^8 \cdot 1 + 2^7 \cdot 0 + 2^6 \cdot 0 + 2^5 \cdot 0 + 2^4 \cdot 0 \\ &\qquad + 2^3 \cdot 1 + 2^2 \cdot 0 + 2^1 \cdot 0 + 2^0 \cdot 1 \end{aligned}$$

777 is thus written as 1100001001 in binary, that is, as the sequence of remainders we got while dividing, read bottom-up. If you know this there is no reason to carry out this part of the computation. ■

Exercise 3.46 Convert the binary number 101010 to decimal notation and the decimal number 101 to binary. Then convert the numbers back to the original notation and check that you get what you started with.

Exercise 3.47 Can you figure out some way other than the one described in method 3.7 on the previous page to convert to binary form? Try to convert 777 using your method. Which one of the methods would you say is easier?

Exercise 3.48 In the description of the method, there is the remark

> Now there is no point in continuing, since after this every step will look like "$0 = 2 \cdot 0 + 0$".

Are there some situations when one would like to continue anyway, and what will that mean for the way the number is written?

3.4.2 Other Number Bases

Even if numbers in binary representation are practical for computers, they are very hard to read for the human eye, which isn't able to sort the long tangle of digits. Because of this, there are a couple of alternative representations that are both very easy to convert to and from binary form and fairly easy to read. One is **octal form**, using the number base 8 (that is: 2^3). Then the digits 0–7 are used.

The methodes described in the previous section for conversion between decimal and binary notation are applicable to any number base. 8 is handled in the same way as 2.

Example 3.11: ***From Octal*** What does 777 in octal correspond to in decimal notation?

$$7 \cdot 8^2 + 7 \cdot 8^1 + 7 \cdot 8^0 = 7 \cdot 64 + 7 \cdot 8 + 7 \cdot 1 = 511$$ ∎

Example 3.12: ***To Octal*** Convert 777 in decimal to octal notation.

We can use the same method as we used for conversion to binary, but dividing by 8 instead of 2:

$$\begin{aligned} 777 &= 8 \cdot 97 + 1 \\ 97 &= 8 \cdot 12 + 1 \\ 12 &= 8 \cdot 1 \ + 4 \\ 1 &= 8 \cdot 0 \ + 1 \end{aligned}$$

so 777 is written 1411 in octal.

Alternatively, we can use the fact that we have already converted the number into binary form, and use that as a start. $8 = 2^3$, so three binary digits correspond to one octal one:

$$\underbrace{1}_{1}\ \underbrace{100}_{4}\ \underbrace{001}_{1}\ \underbrace{001}_{1}$$ ∎

Another one is **hexadecimal form** (also known as **sedecimal form**), using the number base 16 (that is: 2^4). In this case, the digits 0–9 and the letters A–F, where A represents 10, B 11, and so on, are used.

Example 3.13: ***To Hexadecimal*** Convert 777 to hexadecimal notation.

Either we start from scratch and divide by 16, like this:

$$\begin{aligned} 777 &= 16 \cdot 48 + 9 \\ 48 &= 16 \cdot 3 \ + 0 \\ 3 &= 16 \cdot 0 \ + 3 \end{aligned}$$

or we start with the binary form (where four binary digits correspond to one hexadecimal one).

$$\underbrace{1100}_{3}\ \underbrace{0010}_{0}\ \underbrace{1001}_{9}$$

No matter how we go about it, we get the result that the number is represented as 309. ∎

Exercise 3.49 Convert the octal number 105 to decimal form and the decimal number 105 to hexadecimal form.

Exercise 3.50 Figure out one (or several) way(s) of converting between octal and hexadecimal form. Make a trial run of your idea on some number.

One observation that can be made here is that if you see for instance the number 100, there is nothing that tells you whether it's in decimal, binary, octal, hexadecimal, or some other notation. You just have to hope that it will be apparent from the context. There are some different ways of adding an index that give the number base, but they are not very standardised.

3.5 More Exercises

3.5.1 Routine Work

The Division Algorithm

Exercise 3.51 Find the quotient and principal remainder when dividing

(a) 34 by 6 **(b)** 34 by -6 **(c)** -34 by 6 **(d)** -34 by -6

Prime Numbers and Divisors

Exercise 3.52 Factorise 6125 into primes. Draw the divisor graph and mark the numbers 125, 245, $\gcd(125, 245)$, and $\operatorname{lcm}(125, 245)$ in it.

Exercise 3.53 Factorise 2565 into primes. Draw the divisor graph and mark the numbers 45, 57, $\gcd(45, 57)$, and $\operatorname{lcm}(45, 57)$ in it.

Common Divisors

Exercise 3.54 Calculate the gcd and lcm of the numbers below, and express the gcd as a linear combination of the numbers.

(a) 84 and 91 **(b)** 901 and 425 **(c)** 533 and 617 **(d)** 2376 and 1260

Exercise 3.55 Calculate the gcd and lcm of the numbers below, and express the gcd as a linear combination of the numbers.

(a) 70 and 97 **(b)** 693 and 220 **(c)** 598 and 918 **(d)** 2356 and 6235

Exercise 3.56 If A is a set of numbers, $\max A$ denotes "the greatest element in the set A". For instance, $\max\{\,1, -5, 17, 3\,\} = 17$. Use max in combination with the set builder and the divide symbol $\mid$ to make a formal definition of $\gcd(m, n)$.

Exercise 3.57 Can you define "greatest common divisor" in a way that doesn't make use of the concepts "greatest" or "greater than"?

Diofantine Equations

Exercise 3.58 Solve the following diofantine equations:

(a) $27x + 37y = 101$ **(b)** $51y + 68z = -221$ **(c)** $39z + 42w = 17$

Exercise 3.59 Solve the following diofantine equations:

(a) $57x - 33y = 11$ **(b)** $44x + 68y = 20$ **(c)** $26x + 33y = 144$

Exercise 3.60 The shop is selling eggs in 6-packs and 10-packs. For the great cake-making 88 eggs are needed. How many packs of each size should we buy, if we don't want any eggs left over? (Make a complete calculation even if you are able to solve the problem in your head.)

Modular Arithmetic

Exercise 3.61 Calculate

(a) $15 \cdot 16 - 7(9 + 10) + 11$ in $\mathbb{Z}_{17}$. **(b)** $13(17 + 18) - 8(3 + 10)$ in $\mathbb{Z}_{19}$.

Do not use a calculator!

Exercise 3.62 Draw a number circle representing $\mathbb{Z}_8$. Then find $6 + 5$ and $2 - 7$ using the circle. *

Exercise 3.63

(a) Write down the addition and multiplication tables for $\mathbb{Z}_8$.

(b) Which numbers have an inverse, and what is the inverse?

(c) How many solutions does the equation $x^2 \equiv 1$ have?

(d) Any comments on the previous excercise?

Exercise 3.64

(a) Write down the addition and multiplication tables for $\mathbb{Z}_9$.

(b) Which numbers are zero dividers (see exercise 3.43 on page 61)?

(c) How many solutions does the equation $x^2 \equiv 0$ have?

(d) Factorise the polynomial x^2 in a couple of different ways. Comments? (If you haven't studied polynomials you can skip this subexercise.)

Exercise 3.65 Solve the following equations in $\mathbb{Z}_{1024}$:

(a) $17x \equiv 116$ **(b)** $52x \equiv 401$ **(c)** $40x \equiv 888$

Exercise 3.66 Solve the following equations in $\mathbb{Z}_{576}$:

(a) $24x \equiv 54$ **(b)** $18x \equiv 54$ **(c)** $35x \equiv 54$

Number Bases

Exercise 3.67

(a) Convert the decimal number 254 to binary, octal, and hexadecimal notation.

(b) Convert the binary number 101110101101 to decimal, octal, and hexadecimal notation.

(c) Convert the octal number 170,720 to binary, decimal, and hexadecimal notation.

(d) Convert the hexadecimal number BADA to binary, octal, and decimal notation.

3.5.2 To Ponder About

Exercise 3.68 If you look at the multiplication tables for $\mathbb{Z}_6$ and $\mathbb{Z}_7$ (see page 57) you'll find at most one 1 in each row. That means that here no number exists that has several inverses. Does this hold in general, or is it possible to find some number base where one number has several inverses? Explain!

Exercise 3.69 What is the inverse of the inverse of a number in modular arithmetic?

Exercise 3.70 Solve the diofantine equation $x + y = 17$

(a) using the methods taught in this chapter.

(b) in some other way.

Comments? *

Exercise 3.71 Solve the diofantine equation $11x + 13y = 202$ with the constraint that both x and y have to be positive numbers.

Exercise 3.72 Two acquaintances of yours come to disagreement. Both have tried to solve the diofantine equation $11x + 19y = 37$ and ended up with completely different answers. They want you as an arbitrator. One has got the answer $x = 278 + 38k$, $y = -159 - 22k$. The other has got the answer $x = 240 - 57k$, $y = -137 + 33k$. Who is in the right?

Exercise 3.73 (If you have studied polynomial division:) Calculate the gcd for the polynomials $x^3 + 2x^2 - 5x - 6$ and $x^3 + 2x^2 - 19x - 20$.

Exercise 3.74 In "normal" mathematics coordinate systems are often used, put together from two number lines crossing each other at one point. Every point in the coordinate system corresponds to a pair of real numbers, the **coordinates** of the point, and calculations in a coordinate system are often called calculations in $\mathbb{R}^2$, where $\mathbb{R}^2$ is shorthand for $\mathbb{R} \times \mathbb{R}$. In modular arithmetic, the correspondence to the number line is the number circle.

(a) $\mathbb{Z}_n^2 = \mathbb{Z}_n \times \mathbb{Z}_n$ must thus consist of two number circles crossing each other at one point. Together, they are to generate a network of squares, where the different pairs of number can be marked. What will this look like?

(b) If you succeed in solving the previous subexercise, do you have any idea how to extend this to number triplets? How will $\mathbb{Z}_n \times \mathbb{Z}_n \times \mathbb{Z}_n$ look?

Exercise 3.75 Among the advantages of the positional system is not just the fact that it's possible to write any number. It's also possible to subdivide calculations so that you work with one position at a time.

(a) Add the binary numbers 1011 and 1101, using the same technique that is used when adding multi-digit decimal numbers.

(b) Subtract the octal numbers 4321 and 567.

(c) Multiply the hexadecimal numbers 3D and 31.

You may also ponder about why these calculation methods work!

Exercise 3.76: ***Rules of Divisibility*** There exists a number of **divisibility rules**, with the help of which it's possible to check whether a number is divisible by another without involving the division algorithm.

(a) If a decimal integer ends with 5, it is divisible by 5. Explain why!

(b) If the sum of the digits in a decimal integer is divisible by 9, the integer is divisible by 9. Actually, the sum of the digits of an integer is always equivalent to the number modulo 9. Divisibilty by 9 is a special case of this rule, and means that both the number and the sum are equivalent to 0. Explain why!

(c) Consider for which other numbers the method in (a) works, and express it as a universal principle.

(d) Consider for which other numbers the method in (b) works, and express it as a universal principle.

(e) For which numbers is the method in (a) useful if you are calculating in octal?

(f) For which numbers is the method in (b) useful if you are calculating in hexadecimal?

4 Recursion and Induction

Recursion is a very powerful and useful idea. Those with experience from programming have usually been in contact with the concept; for others it can be a new thing. The point of the technique is to solve problems with the help of simpler versions of the *same* problem.

The **induction** proof method is based on the same idea. Proof by induction is a fantastic way of proving an infinite number of statements at the same time, and especially useful when analysing algorithms.

Highlights from this chapter.

- *Recursive definitions* of well-known mathematical concepts.
- Recursively defined *number sequences*, such as the *Fibonacci numbers*.
- Calculating sums and products with the help of the *sum symbol* $\sum$ and the *product symbol* $\prod$.
- Proofs using the *induction principle*.

4.1 Recursion

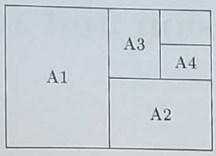
4.1.1 Recursive Definitions

A **recursive definition** is a definition that (at least in parts) refers back to itself.

Everyone has probably seen a picture where somewhere in the picture is a picture of the picture itself, the picture of the picture included, and so on. This can be said to be a recursive picture.

Example 4.1: ***The A-formats*** The A-formats for papers have a recursive definition:

- A0 has the of area 1 m^2 and the proportions $1 : \sqrt{2}$.
- Otherwise, you get format n:o i by dividing format n:o $i - 1$ along the middle.



A0

An A4 is thus half an A3, which is half an A2, which is half an A1, which is half an A0, for which the dimensions can be figured out from the definition. ■

Exercise 4.1 What are the dimensions of an A4?

The definition of the A-formats is put together from two parts:

1. A direct description of the simplest case. This part is called the **base part**.
2. A description of how to solve a more complicated case with the help of simpler cases of the same thing. This part is the **recursive** one.

All recursive descriptions consist of these two parts.

We can express some well-known mathematical concepts in a recursive way:

Example 4.2: ***Multiplication*** First we have to ponder about whether there is some case that is simple enough to be solved directly. Multiplication by zero fits well; it will always give zero.

Then we have to ponder about how a simple case of multiplication can be used to solve a more difficult one. That is something that one does fairly frequently; if one knows that $12 \cdot 10 = 120$ it's easy to find $12 \cdot 11$; just add 12.

Put together this gives that $a \cdot b$, where $b \in \mathbb{N}$, can be defined as:

$$\begin{cases} a \cdot 0 = 0 & \text{Base part} \\ a \cdot b = a + a \cdot (b-1) \text{ if } b > 0 & \text{Recursive part} \end{cases}$$

This means that for instance $3 \cdot 4$ can be evaluated as

$$\begin{aligned} 3 \cdot 4 &= 3 + 3 \cdot 3 && \text{Recursive part} \\ &= 3 + 3 + 3 \cdot 2 && \text{Recursive part} \\ &= 3 + 3 + 3 + 3 \cdot 1 && \text{Recursive part} \\ &= 3 + 3 + 3 + 3 + 3 \cdot 0 && \text{Recursive part} \\ &= 3 + 3 + 3 + 3 + 0 && \text{Base part} \\ &= 3 + 3 + 3 + 3 \\ &= 3 + 3 + 6 \\ &= 3 + 9 \\ &= 12 \end{aligned}$$

(When solving a problem with the help of recursive methods one very frequently gets, as in this case, a whole stack of half-finished calculations while working towards the base part.) ■

Example 4.3: ***Remainder in Division*** The principal remainder r when dividing the non-negative number n by the positive number d can be defined as

$$\begin{cases} r(n,d) = n & \text{if } n < d \\ r(n,d) = r(n-d,d) & \text{otherwise} \end{cases}$$

The upper part is the base part, the lower one the recursive part.

The remainder when dividing 19 by 3 can, according to this formula, be calculated as

$$r(19,3) = r(16,3) = r(13,3) = r(10,3) = r(7,3) = r(4,3) = r(1,3) = 1$$

In the first six steps we used the recursive part of the formula; in the last step the base part. ■

Exercise 4.2: *Integer Powers* Write a recursive definition of integer exponentiation.

(a) What is a suitable base case? (That is, is there any power that can be found without calculations?)

(b) What is a suitable recursive part? (That is: How can you figure out a "difficult" power with the help of a somewhat simpler one?)

(c) Put the parts together to make a complete definition.

(d) Calculate 5^3 using this.

Exercise 4.3: *Quotient* Write a recursive definition of ***quotient*** in integer division, and use it to calculate the quotient then dividing 19 by 3.

The base case (or cases), that soesn't refer back to the concept you are about to define, is essential. Otherwise you end up in an **infinite recursion**, where *every* step demands that you work out something else first.

By the way, it's not enough to include the base case. You also have to ensure that you actually end up there eventually. (If you for instance put in a negative value of b in the definition of multiplication, you'll get further and further away from the base case. And if you put in a positive non-integer b you'll get closer to the base case, skip over it, and then get away from it.)

Exercise 4.4

(a) Try to calculate $2 \cdot (-3)$ using the definition of multiplication, as written in example 4.2 on page 72. What happens?

(b) Extend the definition of multiplication so that it will be able to handle negative values of b as well.

(c) Calculate $2 \cdot (-3)$ using your extended definition.

Exercise 4.5 Enhance the definition of principal remainder in example 4.3 on the preceding page, so that it works for negative values of n and d as well.

Recursive definitions should not be confused with **circular definitions**. In a circular definition one refers back to *exactly* the concept one is trying to define, as in

> Child: "Which hand is the right hand?"
>
> Adult: "The hand you use when greeting people."
>
> Child: "Alright, but which hand do you use when greeting people, then?"
>
> Adult: "The right hand."

This "definition" doesn't define anything. In a recursive definition, on the other hand, you refer back to a *simpler case* of the same thing. Via the more and more simple cases you'll finally reach the directly defined base case.

4.1.2 Recursive Number Sequences

In mathematics, one often works with **number sequences**. These are, as hinted by the name, sequences of numbers, such as 2, 4, 6, 8, ... The difference between a number sequence and a set is in part the fact that the numbers in the sequence are ordered (so that it is possible to talk about "element number 17") and in part that the same number can be included several times.

If you want to discuss a number sequence in general it's handy to denote the elements with numbered letters, like a_1, a_2, a_3, a_4, ... The complete sequence is denoted

$$\{a_n\}_1^\infty$$

where the indices state at which number the sequence starts and stops. (In this case, the sequence is infinitely long.) The sequence that starts 2, 4, 6, 8, ... can now be described as the sequence $\{a_n\}_1^\infty$ where $a_n = 2n$ for all integers $n \geq 1$.

Here we'll discuss a certain kind of number sequences, namely **recursively defined number sequences**.

Example 4.4: ***Recursive Number Sequence*** A sequence $\{a_n\}_0^\infty$ is defined like this:

$$\begin{cases} a_0 = 1 \\ a_{n+1} = 2a_n + 1 \quad n \geq 0 \end{cases}$$

This can be read as "the first number in the sequence has the number 0 and the value 1, and for the rest every number in the sequence is calculated by doubling the previous number and adding 1". The first 5 numbers in the sequence are

$$\begin{aligned} a_0 &= 1 \\ a_1 &= 2a_0 + 1 = 2 \cdot 1 + 1 = 3 \\ a_2 &= 2a_1 + 1 = 2 \cdot 3 + 1 = 7 \\ a_3 &= 2a_2 + 1 = 2 \cdot 7 + 1 = 15 \\ a_4 &= 2a_3 + 1 = 2 \cdot 15 + 1 = 31 \end{aligned}$$

The numbers in the sequence can be computed using the following C-function:

```
int a(int n)
{
    if (n == 0)
        return 1;
    else
        return 2*a(n-1) + 1;
}
```

More about this sequence in example 4.16 on page 88! ■

Exercise 4.6 Write down the first ten numbers in the sequence $\{a_n\}_0^\infty$ defined as $a_0 = 1$; $a_n = 2a_{n-1}$ when $n > 0$. Comments?

Combinatorics, which will be covered in chapter 5, is about finding out in how many ways something can be done. Many combinatorial problems can be solved by making a recursive number sequence:

Example 4.5: ***Sweet Shopping*** A child has been given n crowns as a birthday present. Every day on the way home from school, the child buys either a chocolate toffee costing one crown, a chewing-gum costing two crowns or a lollipop costing two crowns, and keeps on like this until no money remains. How many shopping sequences are possible?

We can use a_n to denote "the number of ways of spending n crowns". If the child hasn't got any money there is only *one* thing to do: refrain from buying sweets. Because of this, $a_0 = 1$. If the child has one crown at the start there is also only one alternative: buy a chocolate toffee the first day. So $a_1 = 1$. Two crowns give several choices: Chewing gum the first day, lollipop the first day, or chocolate toffee two days running. So $a_2 = 3$.

We'll write the possible shopping sequences and values of a_n in a somewhat more systematical way. We denote "buy chocolate toffee" T, "buy chewing gum" G, and "buy lollipop" L. A sequence consisting of "lollipop the first day, chocolate toffee the second and chewing gum the last two days" is then denoted $LTGG$.

$n = 0$:	{empty purchase}	$a_0 = 1$
$n = 1$:	$\{T\}$	$a_1 = 1$
$n = 2$:	$\{G, L, TT\}$	$a_2 = 3$
$n = 3$:	$\{TG, TL, GT, LT, TTT\}$	$a_3 = 5$
$n = 4$:	$\{GG, LG, TTG, GL, LL, TTL,$ $TGT, TLT, GTT, LTT, TTTT\}$	$a_4 = 11$

We can get a sequence for $n = 3$ by appending a G or an L to a sequence for $n = 1$ (which concretely put means "spend one crown, and then buy chewing gum/lollipop") or by appendning a T to a sequence for $n = 2$ ("spend two crowns and then finish off with a toffee"). In the same way, we can get sequences for $n = 4$ from the sequences for $n = 2$ and $n = 3$. In this way we can go on putting together sequences. This makes us able to set up the following relationship:

$$\begin{cases} a_0 = 1 \\ a_1 = 1 \\ a_{n+2} = a_{n+1} + 2a_n \quad n \geq 0 \end{cases}$$

(It would have been just as correct to write the last case as $a_{n+1} = a_n + 2a_{n-1}$, $n \geq 1$ or as $a_n = a_{n-1} + 2a_{n-2}$, $n \geq 2$, since all the expressions mean "every number in the sequence equals the previous one plus twice the one before that". Which version one finds the best-looking is a matter of taste.)

We will return to this example a bit further along this text, in example 4.19 on page 93. ∎

A very famous example of a recursive number sequence are the **Fibonacci numbers**, which are defined as

$$\begin{cases} f_0 = 0 \\ f_1 = 1 \\ f_n = f_{n-1} + f_{n-2} \quad n \geq 2 \end{cases}$$

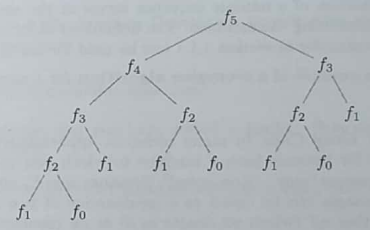
This recursion was set up by the mathematician Fibonacci (whose real name was Leonardo from Pisa and who by the way introduced the positional system in Europe) in 1201. To Fibonacci the recursion described a population of rabbits that reproduced, but it has many more applications than that. The Fibonacci numbers have a tendency to appear in many strange contexts, such as in the number of petals in the spirals on ordinary pine cones. There is a magazine solely dedicated to Fibonacci numbers! The first numbers in the sequence are 0, 1, 1, 2, 3, 5, 8, 13, 21,...

Example 4.6: ***Fibonacci Computation*** A swift but not very thoughtful student has to write a function that computes the Fibonacci numbers, and hereby implements the definition straight off:

```
int fibo(int n)
{
    if (n == 0)
        return 0;
    else if (n == 1)
        return 1;
    else
        return fibo(n-1)+fibo(n-2);
}
```

The question is: how efficient is this program?

Here we have drawn a picture showing the calling sequence when computing f_5.



To calculate f_5 we have to calculate f_4 and f_3, and to calculate f_4 we have to calculate f_3 and f_2, and so on.

We let a_n denote "the number of times the function `fibo` has to be called for f_n to be computed".

We can quickly see that $a_0 = a_1 = 1$, because then we make one call and get one answer, whereas if we are to calculate values for n:s greater than that the function, when called, will call itself twice: once to find out f_{n-1} and another time to find out f_{n-2}. The number sequence will thus be

$$\begin{cases} a_0 = 1 \\ a_1 = 1 \\ a_n = a_{n-1} + a_{n-2} + 1, \quad n \geq 2 \end{cases}$$

This resembles the Fibonacci sequence in a striking way, just augmented with a one in each step. If we write down the start of the sequence we get the numbers { 1, 1, 3, 5, 9, 15, 25, 41, 67, 109, 177, ... }.

This is a fairly swiftly growing sequence. For instance, $a_{100} \approx 1.15 \cdot 10^{21}$. That so many function calls should be needed to compute element n:o 100

in a sequence seems strictly speaking absurd! This was in fact an example showing how *not* to write a program. Some compilers detect this kind of stupidity, but far from all. Exercise 4.8 below discusses a better way of writing this program. ■

Exercise 4.7 Write down a sequence $\{b_n\}$, where b_n is the number of times this program will have to find f_0 when trying to calculate f_n. Write down the first ten numbers in that sequence as well.

Exercise 4.8 Can you suggest a somewhat smarter way of organising the calculation? (You may implement your solution, and make a trial run of that and the given program for some different values of n and compare.)

4.1.3 Recursive Algorithms

A recursive definition of a number sequence serves at the same time as an algorithm for calculating the numbers. The definitions of for instance multiplication and remainder in section 4.1.1 can be used för calculations as well.

Here follows an example of a **recursive algorithm** for a more complicated problem:

Example 4.7: ***Gray Code*** In many technical applications messages are sent. You can for instance have a machine to which you want to be able to send the messages "stop", "slow speed", "medium speed", and "full speed". These four messages can be coded as a combination of two signals, where both can be either *off* (which we denote as 0) or *on* (denoted by 1). With the help of these two signals we can send the signal combinations 00, 01, 10, and 11. Now we have to decide which combination is to signify what. Our choice here will prove to be relevant, since an unsuitable coding can cause serious problems.

If we for instance choose the order in which we have written messages and code words, that is stop–00, slow–01, medium–10, and full–11 (which looks completely reasonable, since the code words interpreted as binary numbers then directly correspond to increased speed), the following can happen:

We run at medium speed, and decide to go down to slow speed. Then the signal has to change from 10 to 01, so both the insignals have to be changed. If they then don't change at exactly the same moment we will, for a fraction of a second, send either 00 (stop!) or 11 (full speed!) and that can have disastrous consequences, if the machine manages to react to the incorrect signal.

Because of this it is of interest to find some way of ordering code words so that consecutive words only differ in one digit, because then the problem can't occur. (A coding of this kind is called a **Gray code** after the person who patented its use in a shaft encoder in 1953.)

You can put together such a code using n binary digits with the help of a recursive algorithm.

- For $n = 1$ the code looks like $\{0, 1\}$
- For higher values of n: Write the code for $n-1$ twice, one after the other, the second time backwards. Add a zero in front of every code word in the first set and add a one in front of every code word in the second part.

If we start with the code $\{0, 1\}$ and write it twice, second time backwards, we get $\{0, 1, 1, 0\}$. By adding a zero in front of every code word in the first part ($\{0, 1\}$ becomes $\{00, 01\}$) and a one in front of every code word in the second part ($\{1, 0\}$ becomes $\{11, 10\}$) we find that the code for $n = 2$ is $\{00, 01, 11, 10\}$. ■

Exercise 4.9 Write the codes for $n = 3$ and $n = 4$.

Exercise 4.10: *Sorting* Develop an algorithm for sorting the books in a bookcase.

(a) What is a suitable base case?

(b) Suppose that you have sorted n books. How do you sort $n + 1$ books?

(c) Find an unsorted bookcase and sort it using your algorithm.

There will be more examples of recursive algorithms in later chapters.

4.2 Sums and Products

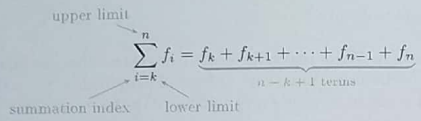
4.2.1 Summation

A symbol that most people have encountered in one context or another is Σ. That is *sigma*, capital S in Greek, and it usually denotes "sum". The **summation symbol** can be used to express large summations in a compact way.

The meaning of the summation symbol is easily expressed recursively:

$$\begin{cases} \sum_{i=k}^{n} f_i = 0 & \text{if } n < k \\ \sum_{i=k}^{n} f_i = \left(\sum_{i=k}^{n-1} f_i\right) + f_n & \text{otherwise} \end{cases}$$

A way of writing exactly the same thing that may be more adapted for humans is



The **upper** and **lower limits** are usually numbers, and as the summation index you can choose any letter you like that isn't already used to denote something else.

Example 4.8: ***A Couple of Sums*** We calculate some sums:

$$\sum_{i=1}^{5} i = 1+2+3+4+5 = 15$$

We have let the summation index i take all the values starting with the lower limit 1 and ending with the upper limit 5, and added the results.

$$\sum_{k=0}^{3} (-2)^k = (-2)^0 + (-2)^1 + (-2)^2 + (-2)^3 = -5$$

Here k has been assigned all the values from 0 up to 3, 0 and 3 included.

$$\sum_{m=10}^{11} \sin m\pi = \sin 10\pi + \sin 11\pi = 0$$

m was assigned values between 10 and 11. ∎

Example 4.9: ***Two Programs*** Here are two C-functions for calculating sums, one that implements the recursive definition directly and one that organises the computation in an iterative way. (The latter will probably be faster, but that depends a bit on the way the compiler is designed.)

```
int sumrec(int n, int k)
{
    if (n < k)
        return 0;
    else
        return sumrec(n-1, k) + f(n);
}

int sumit(int n, int k)
{
    int acc, i;
```



```
    acc = 0;
    for (i = k; i <= n; i++)
        acc += f(i);
    return acc;
}
```

That the programs are this simple indicates that these kinds of calculations are very common, and that because of this the programming language is designed so that they can be easily described. ∎

Exercise 4.11 Write down all the terms in the sums below:

(a) $\sum_{k=3}^{7} (-k)^3$ **(b)** $\sum_{i=0}^{4} (2i+1)$

Exercise 4.12 Write the following sums using the summation sign:

(a) $1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4}$ **(b)** $2-2+2-2+2-2$

One given summation can be written in several different ways. For instance

$$\sum_{i=0}^{3} 2(i+2) = \sum_{k=0}^{3} 2(k+2) = \sum_{n=0}^{3} 2(n+2) = \sum_{n=1}^{4} 2(n+1) = \sum_{n=2}^{5} 2n$$

because all the formulas simply mean "calculate $2\cdot 2 + 2\cdot 3 + 2\cdot 4 + 2\cdot 5$".

Which one of the formulas is the "best" one completely depends on what you intend to use it for. The last version does undoubtedly look the simplest, but in some contexts there is a great advantage in starting at zero, or at one, and then the other versions may be better. And if you are working with complex numbers or electricity i isn't a well-chosen name on the summation index, since that letter is already used for other things. (Which things, by the way?)

Exercise 4.13 Rewrite the following summations, so that they start at zero:

(a) $\sum_{i=3}^{7} i^4$ **(b)** $\sum_{k=-2}^{5} \cos kx$

Exercise 4.14 If one sings the song "The twelve Days of Christmas" slowly the line "On the first day of Christmas my true love sent to me" takes about 10 seconds, and "a partridge in a pear tree" about 8 seconds. Write an expression giving the time it takes to sing n verses of this song. (If you don't know it: The "my true love"-line is sung once every verse, while in every verse another set of gifts is added, so that every verse is longer than the previous one.)

4.2.2 Arithmetical and Geometrical Sequences

There are some types of sums that appear fairly frequently, and that can be calculated in a very simple way. One of those is the sum of an **arithmetical number sequence**, a sum that can be written as

$$\sum_{i=0}^{n-1}(a+bi) = \underbrace{a+(a+b)+(a+2b)+\cdots+(a+(n-1)b)}_{n \text{ terms}}$$

Example 4.10: ***An Arithmetical Sequence***

$$\sum_{k=2}^{8} 3k$$

is arithmetical, since the sum can be rewritten as

$$\begin{aligned}\sum_{k=2}^{8} 3k &= 3\cdot 2+3\cdot 3+\cdots+3\cdot 7+3\cdot 8\\ &= (6+3\cdot 0)+(6+3\cdot 1)+\cdots+(6+3\cdot 5)+(6+3\cdot 6)\\ &= \sum_{i=0}^{6}(6+3i)\end{aligned}$$

■

If you want to calculate the value of a sum of this kind you find that the whole process becomes a lot simpler if instead you write down the sum *twice*, the second time backwards:

$$\begin{array}{ccccccc} a & + & (a+b) & +\cdots+ & (a+(n-2)b) & + & (a+(n-1)b)\\ (a+(n-1)b) & + & (a+(n-2)b) & +\cdots+ & (a+b) & + & a \end{array}$$

$$\underbrace{(2a+(n-1)b)+(2a+(n-1)b)+\cdots+(2a+(n-1)b)+(2a+(n-1)b)}_{n \text{ terms}}$$

To find the value of just one sum we halve, which gives us:

Theorem 4.1: ***The Sum of an Arithmetical Sequence***

$$\begin{aligned}\sum_{i=0}^{n-1}(a+bi) &= \frac{n(2a+(n-1)b)}{2} = \frac{a+(a+(n-1)b)}{2}\,n\\ &= (\text{mean value of first and last term})\cdot(\text{number of terms})\end{aligned}$$

■

The last version, in ordinary language, is probably easiest to remember. But remember as well that it only applies to arithmetical sequences!

Example 4.11 We use the formula we derived to calculate the first of the sums in example 4.8. (It can, if desired, be written on the form $\sum_{k=0}^{5-1} 1\cdot(1+k)$, so it is arithmetical.)

$$\sum_{i=1}^{5} i = \underbrace{\frac{1+5}{2}}_{\text{first term, last term}} \cdot \underbrace{5}_{\text{number of terms}} = 15$$

■

Exercise 4.15 Calculate using the formula:

(a) The sum in exercise 4.11(b).

(b) The time it takes to sing 10 verses of "The Twelve Days of Christmas " (see exercise 4.14).

Another common sum is the sum of a **geometrical sequence**. Those can be written on the form

$$\sum_{i=0}^{n-1} ak^i = \underbrace{ak^0+ak^1+ak^2+\cdots+ak^{n-1}}_{n \text{ terms}}$$

Here the trick is to spot that if you multiply the sum by k you get another sum, with *almost* the same terms. If you calculate the difference between that one and the original sum there isn't much left. The calculation looks like this:

$$\begin{aligned}(1-k)\sum_{i=0}^{n-1} ak^i &= \sum_{i=0}^{n-1} ak^i - k\sum_{i=0}^{n-1} ak^i\\ &= (ak^0+ak^1+ak^2+\cdots+ak^{n-1})-(ak^1+ak^2+ak^3+\cdots+ak^n)\\ &= ak^0-ak^n\end{aligned}$$

from which we can get the desired relationship:

Theorem 4.2: ***The Sum of a Geometrical Sequence***

$$\sum_{i=0}^{n-1} ak^i = \frac{ak^0-ak^n}{1-k} = a\frac{1-k^n}{1-k}$$

■

The easiest way of remembering this theorem is probably to try to remember the derivation, with the help of which the theorem can be derived when needed.

Example 4.12 We use the theorem to calculate the second sum in example 4.8:

$$\sum_{k=0}^{3}(-2)^k = \sum_{k=0}^{4-1} 1\cdot(-2)^k = 1\cdot\frac{1-(-2)^4}{1-(-2)} = \frac{1-16}{1+2} = \frac{-15}{3} = -5 \qquad \blacksquare$$

Exercise 4.16 We have the octal number 3333333333. What is this in decimal?

4.2.3 Products

Besides the symbol Σ that is used to denote *sum*, you have to know about the symbol Π (capital *pi*) that is used to denote **product**. The definition is exactly the same as for sum, except that you use multiplication signs instead of plus. The definition can be written

$$\prod_{i=k}^{n} f_i = \underbrace{f_k \cdot f_{k+1} \cdot \cdots \cdot f_{n-1} \cdot f_n}_{n-k+1 \text{ factors}}$$

or recursively

$$\begin{cases} \prod_{i=k}^{n} f_i = 1 & n < k \\ \prod_{i=k}^{n} f_i = \left(\prod_{i=k}^{n-1} f_i\right)\cdot f_n & \text{otherwise} \end{cases}$$

Example 4.13 A bank account has the following terms: In the first year, the interest will be 1 %. Then the interest is increased by one percentage point yearly as long as you don't withdraw any money. Write an expression giving the amount of money that will available in the account after n years, if you started by depositing 100 crowns at the turn of the year.

The answer ought to be

$$100\cdot 1.01\cdot 1.02\cdot 1.03\cdot\cdots\cdot(1+0.01n) = 100\prod_{k=1}^{n}(1+0.01k) \qquad \blacksquare$$

Exercise 4.17 Calculate

(a) $\displaystyle\prod_{k=1}^{4}\frac{10-k}{k}$ (b) $\displaystyle\prod_{n=-100}^{100} n$

Exercise 4.18 Express the following using the product symbol: $100\cdot 99\cdot 98\cdot 97\cdot 96\cdot 95\cdot 94\cdot 93\cdot 92\cdot 91$

Exercise 4.19 Write a program in your favourite language to compute $\prod_{i=k}^{n} f(i)$, given that the function f is defined. *

4.3 Proof by Induction

The first time one is confronted with proof by induction, one usually thinks "What?". The next reaction is usually "Can you do that?!". The final reaction is: "Yes, of course you can. How clever!" The time passing between these reactions depends on the individual, and varies between 10 seconds and a couple of years. Sooner or later one gets unstuck. Don't lose courage if it seems fishy the first time around.

4.3.1 Introductory Example

Let's say that we want to prove the following statement:

$$5^n \equiv 5 \pmod{10} \qquad n \in \mathbb{Z}_+$$

The statement is actually an infinite number of statements: one for each possible value of n. And an infinite number of statements ought to need an infinite number of proofs. In other words we'll never finish, however long we work. Well, let's disregard that, and prove the statements one at a time, and see what happens:

$n = 1$: $5^1 = 5 \equiv 5 \pmod{10}$

$n = 2$: $5^2 = 5\cdot 5^1 \equiv 5\cdot 5 = 25 = 2\cdot 10 + 5 \equiv 5 \pmod{10}$
According to the previous proof

$n = 3$: $5^3 = 5\cdot 5^2 \equiv 5\cdot 5 = 25 = 2\cdot 10 + 5 \equiv 5 \pmod{10}$
According to the previous proof

$n = 4$: $5^4 = 5\cdot 5^3 \equiv 5\cdot 5 = 25 = 2\cdot 10 + 5 \equiv 5 \pmod{10}$
According to the previous proof

$n = 5$: $5^5 = 5\cdot 5^4 \equiv 5\cdot 5 = 25 = 2\cdot 10 + 5 \equiv 5 \pmod{10}$
According to the previous proof

$n = 6$: $5^6 = 5\cdot 5^5 \equiv 5\cdot 5 = 25 = 2\cdot 10 + 5 \equiv 5 \pmod{10}$
According to the previous proof

$n = 7$: ...

Well, we can keep on like this forever! But we can observe that with the exception of the first one, all the proofs look identical; the only difference is the number used. Furthermore, every proof uses the fact that we have already proved the statement for the previous number.

Based on this, we feel fairly sure that the statement actually holds for any possible value of n; we just have to keep going on like this for long enough. Because of this, we put together a re-usable proof, based on the idea "run this calculation a relevant number of times, and put in the current value in each step". The whole thing looks like this:

Statement

$5^n \equiv 5 \pmod{10} \qquad n \in \mathbb{Z}_+$

Proof
Start The statement is true for the simplest possible value of n, since

$5^1 = 5 \equiv 5 \pmod{10}$

Continuation Assume that we've already proved the statement for the number $n = p$ (that is: assume that we've proved that $5^p \equiv 5 \pmod{10}$). Now we use this to prove that the statement *in that case* is true for the next number, $n = p + 1$, as well.

$5^{p+1} = 5 \cdot 5^p \equiv 5 \cdot 5 = 25 = 2 \cdot 10 + 5 \equiv 5 \pmod{10}$

According to the assumption

Yes, that works.

Summary These two sub-proofs *combined* show that it's possible to make up proofs for any value of n. (Just do the start, and then run the continuation step enough number of times.) And since it's possible to make a proof for every value of n, the statement has to be true for every value of n. Which was to be proved.

4.3.2 The Induction Principle

What we did in the previous section can be summarised like this:

We want to prove that a certain statement holds in all possible cases. Then we organise our reasoning in the following way:

1. We start by proving that the statement holds in the simplest possible case. This is called the **induction basis** or **base step**.
2. Then we prove that *if* the statement holds in a certain – not specified – case *then* it will hold in the next case as well. This is called the **inductive step**. The assumption is called the **induction hypothesis**.

Since the statement according to (1) holds in the first case it must, according to (2), hold in the second case. Since it holds in the second case it has, again according to (2), to hold in the third case. And if it holds in the third case then according to (2) it holds in the fourth case, etc. Then it holds in every case, which was to be proved.

For all this to work, firstly there has to be something that can be identified as the "simplest" case, and secondly it has to be possible to determine what the "next" case is. If the statement concerns all natural numbers the "simplest possible case" is the one about 0, and the "next case" after p is of course $p+1$.

Example 4.14: *The Domino Effect* This is an example of how the induction principle can affect real life.

We have an exhibition hall, with high screens covered with photographs and drawings. If

1. the first screen is knocked over (base step)
2. the screens are placed so close to each other that a falling screen knocks over the one in front (inductive step)

then these two things *together* mean that every screen in the exhibition hall is knocked over. (If the first one hadn't been knocked over, nothing would have happened. If the screens had been placed further apart, only the first one would have fallen. It is the base step and the inductive step combined that give the total effect.) ■

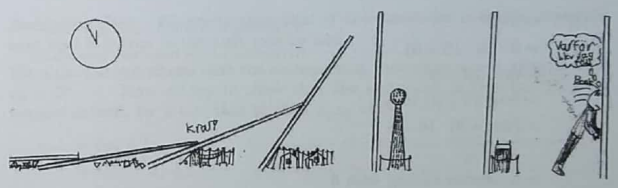


Figure 4.1: *The phenomenon described here is also called the* ***domino effect****. One difference, though, is that when a row of domino tiles is knocked over, it's usually done intentionally. (The text in Swedish means "Why was I born?".)*

4.3.3 Application

To demonstrate how inductive proofs can be done in practice, a number of different examples will follow here:

Example 4.15: *Last Digits*

Statement All positive integer powers of 6 end with the digit 6 (when using decimal notation).

Introductory Investigation Before starting the proof we check that the statement seems reasonable, and a sensible way of doing this is by working out a number of powers of 6 on a piece of scrap paper to see how they look.

$6^1 = 6, \quad 6^2 = 36, \quad 6^3 = 216, \quad 6^4 = 1296$

Yes, the statement seems to be true. Now the proof:

Proof

Base Step The "simplest possible case" is 6^1, a number that undoubtedly ends with 6. Thus, the statement is true in the first case.

Inductive Step We *assume* that the statement is true in a certain case, and then try to show that it *if this is so* also is true in the next case.

This means that we assume that the last digit of 6^p is 6, and try to show that in that case the last digit of 6^{p+1} is 6 as well.

An integer ending in 6 can be written as $m \cdot 10 + 6$, where m is an integer. So we make the induction hypothesis that

$$6^p = m \cdot 10 + 6$$

and see what consequences this will have.

Now we are to show, based on this hypothesis, that 6^{p+1} ends with 6.

$$\begin{aligned}
6^{p+1} &= 6 \cdot 6^p && \text{According to normal rules of powers} \\
&= 6 \cdot (m \cdot 10 + 6) && \text{According to the induction hypothesis} \\
&= 6m \cdot 10 + 36 \\
&= 6m \cdot 10 + 3 \cdot 10 + 6 \\
&= \underbrace{(6m + 3)}_{\text{an integer}} \cdot 10 + 6 \\
&= \text{a number ending with 6}
\end{aligned}$$

Summary The statement is true in the simplest case, 6^1, and if it is true in a certain case, say 6^p, then it is true in the next case, 6^{p+1}, as well. Then it is true in every case, which was to be proved.

Remark Note that the inductive step is done for the unspecified number p, that we never say what it is. That is necessary, since this part of the proof has to work no matter what value is put in p's place. ∎

Included in this example was a phrase that should *always* be present in an inductive proof: "according to the hypothesis". One starts the inductive step by making a hypothesis, and this hypothesis must in some way be used. (If you *don't* use it, what is the point of making it?)

Example 4.16: *Recursive Equation* We are to show that the **recursive equation** $a_{n+1} = 2a_n + 1$ with the starting value $a_0 = 0$ has the **solution** $a_n = 2^n - 1$.

Statement The numbers in the recursive sequence $a_0 = 0$, $a_{n+1} = 2a_n + 1$ can be written $a_n = 2^n - 1$ when $n \in \mathbb{N}$.

Investigation We calculate the first numbers in the sequence, and check that they seem to fit the formula.

$$\begin{aligned}
a_0 &= 0 = 1 - 1 = 2^0 - 1 \\
a_1 &= 2a_0 + 1 = 2 \cdot 0 + 1 = 1 = 2 - 1 = 2^1 - 1 \\
a_2 &= 2a_1 + 1 = 2 \cdot 1 + 1 = 3 = 4 - 1 = 2^2 - 1 \\
a_3 &= 2a_2 + 1 = 2 \cdot 3 + 1 = 7 = 8 - 1 = 2^3 - 1 \\
a_4 &= 2a_3 + 1 = 2 \cdot 7 + 1 = 15 = 16 - 1 = 2^4 - 1
\end{aligned}$$

Yes, they seem to fit well!

Proof

Base Step We are to show that the statement is true in the simplest possible case, which has to be for $n = 0$. We copy the first line in the investigation:

$$a_0 = 0 = 1 - 1 = 2^0 - 1$$

Inductive Step We are to show that *if* the statement is true in a certain case *then* it is true in the next case as well.

We make the hypothesis that the statement is true when $n = p$, that is: that $a_p = 2^p - 1$. Now we try to show that the statement is true for the next number as well, for $p + 1$, that is: that $a_{p+1} = 2^{p+1} - 1$.

$$\begin{aligned}
a_{p+1} &= 2a_p + 1 && \text{According to the definition of the sequence} \\
&= 2(2^p - 1) + 1 && \text{According to the induction hypothesis} \\
&= 2 \cdot 2^p - 2 + 1 \\
&= 2^{p+1} - 1
\end{aligned}$$

Remark This gives us a direct formula to use to calculate the numbers in the sequence in example 4.4 on page 75. If you just want to find number n:o 100 it's rather handy *not* to have to find all the numbers before that in the sequence, so formulas of this kind are nice to have. ∎

Example 4.17: *The Cardinality of the Power Set*

Statement A set with n elements has 2^n subsets.

Comment "One set" in this context means "all sets".

Investigation We check whether the statement seems to be true, by verifying it for a couple of sets of different sizes.

(0) The empty set $\{\}$ (with zero elements) has only one subset: $\{\}$.

(1) The set $\{a\}$ (with one element) has two subsets $\{\}$ and $\{a\}$.

(2) The set $\{a, b\}$ (with two elements) has firstly the two previous subsets $\{\}$ and $\{a\}$, and secondly two new subsets also containing b: $\{b\}$, and $\{a, b\}$, four subsets in total.

(3) The set $\{a, b, c\}$ has firstly the four previous subsets $\{\}$, $\{a\}$, $\{b\}$, and $\{a, b\}$, and secondly four new subsets also containing c: $\{c\}$, $\{a, c\}$, $\{b, c\}$, and $\{a, b, c\}$, eight subsets in total.

The statement that a set with n elements has 2^n subsets holds this far. This investigation also gives us some ideas about how to organise the argument. The fact that the subsets can be partitioned into two equally big groups seems useful.

Proof

Base Step The simplest case is a set with zero elements, that is the empty set (there is only one), which according to the investigation has $2^0 = 1$ subset.

Inductive Step Assume that the statement is true for *all* sets with p elements, that is, assume that every set with p elements has 2^p subsets. Show that the statement in that case is true for sets with $p + 1$ elements as well, that is, show that every set with $p + 1$ elements has 2^{p+1} subsets.

We study an arbitrary set with $p+1$ elements. The subsets of the set can be partitioned into two groups:

(1) *Subsets that don't include element n:o* $p + 1$: Such a subset is a subset of the set containing the first p elements, and the number of subsets of a set with p elements is (according to the hypothesis) 2^p.

(2) *Subsets that do include element n:o* $p + 1$: Such a subset can be created by taking a subset not containing element n:o $p + 1$ and adding the said element. Since there are, as already mentioned, 2^p subsets *without* element n:o $p + 1$ it's possible to create 2^p subsets *with* this element.

Summation of the two cases shows that there are $2^p+2^p = 2 \cdot 2^p = 2^{p+1}$ subsets of the set we studied.

Remark Note that this proof almost in its entirety consists of words; neither symbols nor complicated calculations are included. ∎

Example 4.18: ***Calculation of a Sum***

Statement Show that

$$\sum_{i=1}^{n}(10 + 8i) = (14 + 4n)n$$

(This is the formula for arithmetical sequences, see page 82, applied to the sum based on "The Twelve Days of Christmas", see page 81, but we don't have to admit that we know this at the moment.)

Investigation We start by verifying on a piece of paper that the statement is true for some values of n. Since there is no point in having an upper limit that is smaller than the lower one, the first meaningful value is $n = 1$, and

since the statement is otherwise rather tangled, the best way of unravelling the proofs seems to be to start at both the right- and left-hand sides, hoping to meet somewhere in the middle.

$$n = 1: \begin{cases} \text{LHS}_1 = \sum_{i=1}^{1}(10 + 8i) = 10 + 8 \cdot 1 = 18 \\ \text{RHS}_1 = (14 + 4 \cdot 1) \cdot 1 = 18 \end{cases}$$

$$n = 2: \begin{cases} \text{LHS}_2 = \sum_{i=1}^{2}(10 + 8i) \\ \quad = \underbrace{(10 + 8 \cdot 1)}_{\text{LHS}_1} + (10 + 8 \cdot 2) = 18 + 26 = 44 \\ \text{RHS}_2 = (14 + 4 \cdot 2) \cdot 2 = 44 \end{cases}$$

$$n = 3: \begin{cases} \text{LHS}_3 = \sum_{i=1}^{3}(10 + 8i) \\ \quad = \underbrace{(10 + 8 \cdot 1) + (10 + 8 \cdot 2)}_{\text{LHS}_2} + (10 + 8 \cdot 3) \\ \quad = 18 + 26 + 34 = 78 \\ \text{RHS}_3 = (14 + 4 \cdot 3) \cdot 3 = 78 \end{cases}$$

Well, it seems to hold this far, at least. And it should be possible to make use of the fact that every sum is the previous sum augmented with another term. (This is an application of the recursive sum definition on page 79.)

Proof

Base Step Show that the statement is true in the first case, for $n = 1$:

$$\text{LHS}_1 = \sum_{i=1}^{1}(10 + 8i) = 10 + 8 \cdot 1 = 18 = 14 + 4 \cdot 1 = \text{RHS}_1$$

Inductive Step Assume that the statement is true in a certain case, let's say for $n = p$, that is, assume that

$$\sum_{i=1}^{p}(10 + 8i) = (14 + 4p)p$$

Now show that the statement will then be true for the next number, the number $n = p + 1$, as well; that is, show that

$$\sum_{i=1}^{p+1}(10 + 8i) = \big(14 + 4(p + 1)\big)(p + 1)$$

We proceed in the same way as in the investigation: start with the left-hand side simplifying it as far as possible, and then restart with the right-hand

side, trying to simplify it into the same thing. (It's usually a lot easier to simplify an expression than to make it more complicated in the right way.)

$$\begin{aligned}
\mathrm{LHS}_{p+1} &= \sum_{i=1}^{p+1}(10+8i) \\
&= \underbrace{\sum_{i=1}^{p}(10+8i)}_{\text{the first } p \text{ terms}} + \overbrace{\big(10+8(p+1)\big)}^{\text{the last term}} && \text{According to the sum definition} \\
&= \mathrm{LHS}_p + \big(10+8(p+1)\big) \\
&= \mathrm{RHS}_p + \big(10+8(p+1)\big) && \text{According to the induction hypothesis} \\
&= (14+4p)p + (10+8p+8) \\
&= 14p + 4p^2 + 10 + 8p + 8 \\
&= 4p^2 + 22p + 18
\end{aligned}$$

$$\begin{aligned}
\mathrm{RHS}_{p+1} &= \big(14+4(p+1)\big)(p+1) \\
&= (14+4p+4)(p+1) \\
&= (18+4p)(p+1) \\
&= 18p + 18 + 4p^2 + 4p \\
&= 4p^2 + 22p + 18
\end{aligned}$$

The left-hand side and the right-hand side are equal to the same thing, so they have to be equal to each other. ∎

Usually, proofs by induction aren't written quite as wordily as in these examples. Among other things, the investigation isn't someting that's included in the proof; it's something done on a piece of scrap paper. But still it's a good thing to make a check of this kind. One: you verify that the statement is true and that you have fully understood the meaning of the statement. Two: the calculations can give inpiration for finding a suitable strategy for the proof, as in example 4.17 and example 4.18.

There are a lot of different conventions concerning how proofs by induction are to be written, and many different kinds of stenographic notations. Often great sins are committed against reasonable expectations of clearness, even in textbooks in mathematics. It's important that it's completely clear what one wants to prove, what is the simplest case, what is assumed and how the assumption is used. Write as much explanatory text as needed.

Here comes another example of proof by induction, with a somewhat more concise text.

Example 4.19: ***Recursive Equation***

Statement The recursive equation

$$\begin{cases} a_0 = 1 \\ a_1 = 1 \\ a_{n+2} = a_{n+1} + 2a_n \quad n \geq 0 \end{cases}$$

has the solution

$$a_n = \tfrac{1}{3}\big(2^{n+1} + (-1)^n\big), \qquad n \geq 0$$

Comment In this example, every number depends on the *two* previous ones. That means firstly that in the base step we have to show that the statement is true for the *first two* numbers, secondly that as an induction hypothesis we have to make the assumption that the statement is true for *two consecutive* numbers.

Proof

Base Step Show that the statement is true for $n = 0$ and for $n = 1$:

$$n = 0: \quad \begin{cases} \mathrm{LHS}_0 = a_0 = 1 \\ \mathrm{RHS}_0 = \tfrac{1}{3}\big(2^{0+1} + (-1)^0\big) = \tfrac{1}{3}(2+1) = 1 \end{cases} \qquad \text{OK!}$$

$$n = 1: \quad \begin{cases} \mathrm{LHS}_1 = a_1 = 1 \\ \mathrm{RHS}_1 = \tfrac{1}{3}\big(2^{1+1} + (-1)^1\big) = \tfrac{1}{3}(4-1) = 1 \end{cases} \qquad \text{OK!}$$

Inductive Step Assume that the statement is true for two consecutive numbers $n = p-1$ and $n = p$. Show that it is then true for the next number, $n = p+1$. That is, assume that

$$a_{p-1} = \tfrac{1}{3}\big(2^{(p-1)+1} + (-1)^{p-1}\big) \quad \text{and that} \quad a_p = \tfrac{1}{3}\big(2^{p+1} + (-1)^p\big)$$

and show that

$$a_{p+1} = \tfrac{1}{3}\big(2^{(p+1)+1} + (-1)^{p+1}\big)$$

We get going:

$$\begin{aligned}
\mathrm{LHS}_{p+1} &= a_{p+1} \\
&= a_p + 2a_{p-1} && \text{According to definition} \\
&= \tfrac{1}{3}\big(2^{p+1} + (-1)^p\big) + 2 \cdot \tfrac{1}{3}\big(2^p + (-1)^{p-1}\big) && \text{According to hypothesis} \\
&= \tfrac{1}{3}\big(2^{p+1} + 2 \cdot 2^p + (-1)^p + 2(-1)^{p-1}\big) \\
&= \tfrac{1}{3}\big(2^{p+1} + 2^{p+1} + (-1)(-1)^{p-1} + 2(-1)^{p-1}\big) \\
&= \tfrac{1}{3}\big(2 \cdot 2^{p+1} + (-1)^{p-1}\big) \\
&= \tfrac{1}{3}\big(2^{p+2} + (-1)^{p+1}\big) && \text{since } (-1)^{p-1} = (-1)^{p+1} \\
&= \mathrm{RHS}_{p+1} \qquad \text{OK!}
\end{aligned}$$

Remark Note that this is the recursive equation that we derived in example 4.5 on page 75, with the child who was buying sweets. ∎

Exercise 4.20 A number sequence is defined as $a_0 = 1$, $a_1 = 9$, $a_n = 6a_{n-1} - 9a_{n-2}$ when $n \geq 2$. Show that the numbers in the sequence can be calculated using the formula $a_n = (1+2n)3^n$.

Exercise 4.21 Show that $\sum_{i=1}^{n} \frac{1}{i(i+1)} = \frac{n}{n+1}$

Exercise 4.22 You surely know that the sum of the angles in a triangle is 180°. This relationship can be expanded to the following:

> The sum of the angles in an polygon with n corners is $(n-2) \cdot 180°$ $(n \geq 3)$.

Prove this using induction!

You may assume that the theorem about the sum of the angles in a triangle has been proved, and that the polygon is **convex** (which means that all inner angles are less than 180°).

Exercise 4.23 Show that $4^{2n} - 1$ is divisible by 15 for all integers $n \geq 1$. (Hint: A number that is divisible by 15 can be written as $m \cdot 15$, where m is an integer.)

Exercise 4.24: *Important!* For a proof method to be worth anything it's not enough that you can use it to prove true things – it must also be impossible to prove untrue things! (As a matter of fact, that is an absolute requirement for the method to be worth anything.) Check how good the inductive method seems to be from this aspect, by trying to prove the following completely untrue statements:

(a) All positive integer powers of 4 end with 4.

(b) All positive integer powers of 6 end with 4.

What happens?

4.3.4 Proving Inequalities

The relationships that we have proved this far have all been *equalities*. But not all the things that one wants to prove are equalities; one may also want to show that two expressions are unequal in such a way that one of the expressions is always *greater than* the other one.

Before starting to study how to prove inequalities we'll run through in which circumstances you can proceed exactly as in proofs of equalities, and where it differs.

A proof of an equality is usually a whole chain of equalities, such as $a = b = c = d = e$, from which it is possible to conclude that $a = e$. Inequalities can be concatenated like this as well, at least if they are of the same kind. If you have $a < b = c \leq d < e$ it is possible to conclude that $a < e$.

Exercise 4.25 Convince yourself that it is true that you can concatenate inequalities in this way. *

Exercise 4.26 A somewhat too creative student has ended up showing that $a < b > c$. What conclusions can be made concerning the relationship between a and c?

Furthermore, for equalities it is true that if you treat both sides identically they will, if they were alike before the treatment, also be alike after the treatment. It's not quite as simple as that when you are working with inequalities. The calculation rules look like this:

Theorem 4.3: ***Some Calculation Rules for Inequalities***

If $0 < m < n$ the following is true:

(a) $m + a < n + a$ irrespective of the value of a.

(b) $\frac{1}{m} > \frac{1}{n}$

(c) $a \cdot m \begin{cases} < a \cdot n & \text{if } a > 0 \\ = a \cdot n & \text{if } a = 0 \\ > a \cdot n & \text{if } a < 0 \end{cases}$ ∎

Rule (c) means that you really have to keep your eyes open when multiplying an inequality by a number, since completely different things happen depending on the number. In particular, that means that you have to be very careful when it comes to multiplying an inequality by something *unknown*, since you can't be sure then which of the three cases that applies. Note that these warnings apply to *division* as well, since dividing an inequality by the number b is the same thing as multiplying it by the number $1/b$.

Exercise 4.27: *Important!* Study the rules and convince yourself that they hold. *

Exercise 4.28

(a) How will the rules look if instead $m < 0 < n$ (that is, if the signs of the numbers are different instead of both numbers being positive)?

(b) How will the rules look if $m < n < 0$?

Here follow some proofs by induction concerning inequalities:

Example 4.20

Statement

$$\sum_{i=0}^{n} i^2 \leq n^3 \qquad n \in \mathbb{N}$$

Proof

Base Step Show this for the smallest natural number $n = 0$:

$$\left.\begin{aligned} \mathrm{LHS}_0 &= \sum_{i=0}^{n} i^2 = 0^2 = 0 \\ \mathrm{RHS}_0 &= 0^3 = 0 \end{aligned}\right\} \quad \Rightarrow \quad \mathrm{LHS}_0 \leq \mathrm{RHS}_0 \qquad \text{OK!}$$

Inductive Step Assume that $\mathrm{LHS}_p \leq \mathrm{RHS}_p$. Show that $\mathrm{LHS}_{p+1} \leq \mathrm{RHS}_{p+1}$.

$$\begin{aligned} \mathrm{LHS}_{p+1} &= \sum_{i=0}^{p+1} i^2 \\ &= \sum_{i=0}^{p} i^2 + (p+1)^2 && \text{Unhook the last term} \\ &= \mathrm{LHS}_p + (p+1)^2 \\ &\leq \mathrm{RHS}_p + (p+1)^2 && \text{According to hypothesis} \\ &= p^3 + p^2 + 2p + 1 \\ &\leq (p^3 + p^2 + 2p + 1) + (2p^2 + p) && \text{since } 2p^2 + p \geq 0 \text{ if } p \geq 0 \\ &= p^3 + 3p^2 + 3p + 1 \\ &= (p+1)^3 \\ &= \mathrm{RHS}_{p+1} \end{aligned}$$

so $\mathrm{LHS}_{p+1} \leq \mathrm{RHS}_{p+1}$.

Remark The fact that we realised that we had to add exactly $(2p^2 + p)$ is because we wrote down what we had $(p^3 + p^2 + 2p + 1)$ and what we wanted to have $((p+1)^3)$ on a piece of paper, and checked how much they differed. ■

Example 4.21: ***Polynomial Contra Exponential Function***

Statement $n^2 < 2^n$ if $n \in \mathbb{Z}$, $n \geq 5$.

Proof

Base Step Verifying that the statement holds if $n = 5$:

$$\left.\begin{aligned} \mathrm{LHS}_5 &= 5^2 = 25 \\ \mathrm{RHS}_5 &= 2^5 = 32 \end{aligned}\right\} \quad \Rightarrow \quad \mathrm{LHS}_5 < \mathrm{RHS}_5 \qquad \text{OK!}$$

Inductive Step Assume that the statement holds for a number $n = p$, that is, assume that $\mathrm{LHS}_p < \mathrm{RHS}_p$. Show that the statement then holds for the number $n = p + 1$ as well, that is, show that $\mathrm{LHS}_{p+1} < \mathrm{RHS}_{p+1}$.

$$\begin{aligned} \mathrm{LHS}_{p+1} &= (p+1)^2 \\ &= p^2 + 2p + 1 \\ &= \mathrm{LHS}_p + 2p + 1 \\ &< \mathrm{RHS}_p + 2p + 1 && \text{According to hypothesis} \\ &= 2^p + 2p + 1 \\ &< 2^p + 2^p && \text{According to lemma below} \\ &= 2 \cdot 2^p \\ &= 2^{p+1} \\ &= \mathrm{RHS}_{p+1} \end{aligned}$$

Lemma $2n + 1 < 2^n$ when $n \in \mathbb{Z}$, $n \geq 5$.

$\mathrm{LHS}_5 = 2 \cdot 5 + 1 = 11 < 32 = 2^5 = \mathrm{RHS}_5$, so the statement holds for $n = 5$. (This was the base step.)

Now suppose that $\mathrm{LHS}_p < \mathrm{RHS}_p$ and show that $\mathrm{LHS}_{p+1} < \mathrm{RHS}_{p+1}$.

$$\begin{aligned} \mathrm{LHS}_{p+1} &= 2(p+1) + 1 \\ &= 2p + 2 + 1 \\ &= 2p + 1 + 2 \\ &= \mathrm{LHS}_p + 2 \\ &< \mathrm{RHS}_p + 2 && \text{According to hypothesis} \\ &= 2^p + 2 \\ &< 2^p + 2^p && \text{since } 2 < 2^p \text{ if } p > 1, \text{ which was the case here} \\ &= 2 \cdot 2^p \\ &= 2^{p+1} \\ &= \mathrm{RHS}_{p+1} \end{aligned}$$

Remark 1 What occurred here, that we discovered in the middle of a proof that we needed a lemma that we hadn't proved, is something that happens quite often. When transcribing the proof you usually place the lemmas first, and then the reader tries in vain to figure out how you could anticipate having to need them.

Remark 2 In calculus you occasionally meet the expression "the exponential function grows faster than every power", and this statement was a demonstration of this. This is by the way something that is of great practical use when studying the complexity of computations, because it tells us that an algorithm with exponential complexity (that is: where the run time is an exponential function of the size of the in-data set) for large enough amounts of in-data will be slower than an algorithm with polynomial complexity. ■

Exercise 4.29 Which ones of the calculation rules for inequalities have we used in the calculation? *

Exercise 4.30 Have you any idea about how to proceed to prove the same thing in calculus?

Exercise 4.31 Does the statement in example 4.21 hold when $n < 5$?

Exercise 4.32 Prove that $n^3 < 2^n$ starting at a number that you have to determine yourself. In the proof you may refer to the already proved statement $n^2 < 2^n$, if needed. *

Exercise 4.33 Show that the numbers in the recursive sequence $a_0 = 2$, $a_{n+1} = 3a_n + 1$ satisfies the relationship $a_n > 3^n$.

Exercise 4.34: *Important!* The **factorial function** is defined as $n! = \prod_{k=1}^{n} k$. ("$n!$" is pronounced "n-factorial".) Find an n such that $n! > 2^n$, and prove that the inequality holds for every number from this point. (So here we have found a function that grows even faster than the exponential function! More about the factorial function in the next chapter.) *

4.4 More Exercises

4.4.1 Routine Work

Recursion

Exercise 4.35 Define the greatest common divisor of two positive integers m and n recursively. (If you feel extra ambitious you can remove the restriction that the integers should be positive.)

Exercise 4.36 Write a recursive algorithm for generating all binary strings (sequences of zeros and ones) of length n so that they come in order of size, interpreted as binary numbers.

Exercise 4.37 A person has to move n petrol cans, and can manage to carry two at once, but one is easier. If you take just one you'll have to walk the distance more times, though. Write a recursive expression for the number of ways a moving sequence can look. (1–1–2–1–2–2 is an example of a moving sequence for $n = 9$.)

Exercise 4.38 Write a recursive expression giving the number of binary strings of length n that don't include any consecutive zeros.

Exercise 4.39 A number sequence is defined as

$$\begin{cases} a_1 = 32 \\ a_2 = 38 \\ a_3 = 206 \\ a_n = 2a_{n-1} + 5a_{n-2} - 6a_{n-3} \quad n > 3 \end{cases}$$

Calculate the numbers up to a_6.

Exercise 4.40 A number sequence is calculated according to

$$\begin{cases} b_0 = 1 \\ b_1 = 9 \\ b_{k+2} = 6b_{k+1} - 9b_k \quad k \geq 0 \end{cases}$$

Calculate the numbers up to b_5.

Sums and Products

Exercise 4.41 Write the following sums using the summation symbol:

(a) $1 - {}^2/_3 + {}^3/_9 - {}^4/_{27} + {}^5/_{81}$ **(b)** $10 + 9.5 + 9 + 8.5 + 8 + 7.5 + 7$

Exercise 4.42 Write the following products using the product sign:

(a) $1 \cdot 1 \cdot 2 \cdot 2 \cdot 3 \cdot 3 \cdot 4 \cdot 4$ **(b)** $\dfrac{10 \cdot 9 \cdot 8 \cdot 7 \cdot 6}{1 \cdot 2 \cdot 3 \cdot 4 \cdot 5}$

Exercise 4.43 Calculate $\prod_{j=2}^{n} \frac{2}{j}$ for $n = 2$, 3, 4, and 5.

Exercise 4.44 Calculate $\sum_{k=1}^{m} (-k)^3$ for $m = 1$, 2, 3, and 4.

Exercise 4.45 Calculate

(a) $\sum_{i=0}^{4} \frac{1}{2^i}$ **(b)** $\sum_{n=3}^{10} 3n$

Induction

Exercise 4.46 Show that $\sum_{k=1}^{m} (-k)^3 = -\dfrac{m^2(m+1)^2}{4}$

Exercise 4.47 Show that $\sum_{i=0}^{n-1} 2^i = 2^n - 1$

Do not use the formula for geometrical sums! Are you able to come up with some interesting consequence of the statement as well?

Exercise 4.48 Show that the principal remainder when dividing 4^n by 12 will always be 4 when n is a positive integer.

Exercise 4.49 Show that $6^n \equiv 0 \pmod 9$ for all integers $n \geq 2$.

Exercise 4.50 Prove that the numbers in the sequence in exercise 4.39 can be calculated using the formula $a_n = 4 - 5(-2)^n + 6 \cdot 3^n$.

Exercise 4.51 Prove that the numbers in the sequence in exercise 4.40 can be calculated using the formula $b_k = (2k+1)3^k$.

Exercise 4.52 Show that the area of an An-paper is $1/2^n$ m^2. (For the definition of the A-formats, see example 4.1 on page 71.)

Exercise 4.53 Show that the algorithm for gray code generation (see example 4.7 on page 78) really generates a gray code, that is 2^n different code words, ordered so that that every word differs from the previous one at exactly one place.

4.4.2 To Ponder About

Exercise 4.54: *Pyramid* If you want to build a "pyramid" (actually a tetrahedron) out of marbles, the top layer usually consists of one single marble. The layer below consists of three marbles, in a triangle, so a two-layer pyramid contains four marbles. The picture shows a three-layer pyramid.

Photo: *Vanda Gavel*

(a) Write an expression giving the number of marbles in a pyramid consisting of n layers.

(b) Show by using your expression that the number of marbles in a pyramid consisting of n layers is

$$\frac{n}{3} + \frac{n^2}{2} + \frac{n^3}{6}$$

Exercise 4.55 A number sequence is defined as follows:

$$\begin{cases} a_3 = 4 \\ a_{n+1} = a_3 + a_4 + \cdots + a_n \quad n \geq 3 \end{cases}$$

Statement: All numbers in the sequence are divisible by 4.

(a) If you want to prove this, what does the induction hypothesis have to look like?

(b) Prove the statement.

Exercise 4.56 Circular proofs are "proofs" where That Which Is To Be Proved is used in the proof itself. The only thing a circular proof proves is "if this is true, then it is true". Explain why proofs by induction *aren't* circular proofs. *

Exercise 4.57 Here comes an interesting proof:
Proof that all marbles you take from a bag of marbles are of the same colour:
Base step: Show that it is true for a fistful containing one marble. In such a fistful it is clear that all marbles have the same colour.
Induction: Assume that the statement is true for fistfuls containing $n-1$ marbles. Show that it is then true for fistfuls with n marbles as well.

Remove a fistful with $n-1$ marbles. All those marbles have, according to the hypothesis, the same colour. Now remove another marble from the bag, and put back one of the others. Now you have a different fistful with $n-1$ marbles, that (according to the hypothesis) is unicoloured as well. The union of these fistfuls contains n marbles, and has to be unicoloured as well (because if you have a unicoloured set and replace one element with a different one and the set remains unicoloured, then the new element must be of the same colour as the one it replaces.)

The statement is clearly untrue, but *where* in the reasoning is the fault to be found?

Exercise 4.58 Show that the number of ways of writing the positive number n as a sum of positive integers is 2^{n-1} (if we accept sums consisting of only one term, and regard two sums containing the same numbers in a different order as different). (Hint: Partition the sums into two categories: those that end with $+1$ and those that don't.) *

Exercise 4.59: *Subdivision of the Plane* If you draw a line on an infinite plane it's divided into two parts. If you draw another line, not parallel to the first one, the plane is divided into four parts. Let n be the number of lines and a_n the number of parts into which n non-parallel lines divide the plane.

(a) Write a recursive expression giving a_n. (You may presume that three lines never intersect one another in the same place.)

(b) Show, using your recursive expression, that $a_n = \frac{1}{2}(n^2 + n) + 1$.

(c) Show that it is possible to colour the subdivided plane using just two colours, without parts sharing a common border getting the same colour.

Exercise 4.60: *Monthly Allowance* Kimmo's parents wanted to encourage savings. His monthly allowance was therefore determined like this: every month he got for a start the same sum as last month. But then there was the possiblity of a bonus: if Kimmo had saved the allowance from two months back he got the difference between the last month and the month before that.

(a) Let a_n be the allowance that Kimmo gets in month n. Explain why either $a_n = a_{n-1}$ or $a_n = 2a_{n-1} - a_{n-2}$.

(b) It started with Kimmo getting one crown in January and three crowns in February, that is $a_1 = 1$ and $a_2 = 3$. The maximal savings will thus give $a_3 = 2a_2 - a_1 = 2 \cdot 3 - 1 = 5$ and $a_4 = 2a_3 - a_2 = 2 \cdot 5 - 3 = 7$. The maximal spending will give $a_4 = a_3 = a_2 = 3$. Prove (using induction) that no matter how Kimmo alternates between saving and not saving it will be the case that $a_n \leq 2n - 1$ for all $n \geq 1$.

Exercise 4.61: *Venn Diagrams* If you are to analyse a set-algebraic problem using Venn diagrams you have to draw one where all possible combinations of sets are included. The diagram on page 14 is for instance unusable to analyse a problem that may contain intersecting sets – it represents disjoint sets and nothing else.

Here come recursive instructions for drawing a Venn diagram containing n sets that is useful in the general case.

- If $n = 0$ the diagram consists of just a frame.

- Otherwise: Draw a diagram with $n-1$ sets, and then draw a simple closed curve that passes through all areas in the diagram once and only once.

(a) Draw a Venn diagram with four sets using the instructions. Check the number of subdivisions; if the diagram is drawn correctly there should be 16.

(b) Explain why you are guaranteed to get 2^n subareas when using this algorithm for n sets.

(c) Draw a diagram containing four circular sets, and count the number of subareas.

(d) Show that n circles partition the plane into $n^2 - n + 2$ parts if every circle intersects every other circle in exactly two places, and no more than two circles cross in the same place. ($n \geq 1$.)

(e) Show that $n^2 - n + 2 < 2^n$ for $n \geq 4$.

(Here we see why it isn't possible to draw a Venn diagram representing more than three sets using circles.)

Exercise 4.62 Show that the Fibonacci numbers (see page 76) can be calculated using the formula

$$f_n = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n \quad *$$

Exercise 4.63 Let f_i be the i:th Fibonacci number. Show that $\sum_{i=1}^{2n} f_i f_{i-1} = (f_{2n})^2$. (Hint: Start at both ends, meeting in the middle.) *

Exercise 4.64 Show that $\sum_{i=1}^{n} (-1)^i i^2 = (-1)^n \sum_{i=1}^{n} i$. *

Exercise 4.65 A number sequence $\{S_n\}_1^\infty$ is defined according to the formula $S_n = \sum_{k=1}^{n} k^3$.

Calculate some of the first numbers in the sequence, make a hypothesis about their values, and prove the hypothesis.

Exercise 4.66 A number sequence $\{a_n\}_1^\infty$ is defined as $a_n = n^5 - n$.

Calculate some numbers in the sequence, make a hypothesis about some property they have in common, and prove the hypothesis.

5 Combinatorics and Probability

In how many ways can the cards be ordered in a pack of 52 cards? How long will it take a certain computer program to sort a database if every sorting operation takes one millisecond? If you are going by train from Lund to Kiruna, how great is the risk that the train will derail both on the way out and on the way home?

Combinatorics is about *counting*, for instance the number of ways a certain event can happen. Since the goal of a combinatorial argument is often (but far from always) to calculate the probability of an event we will start the chapter with basic probability. Probability theory, in turn, is based on set theory from chapter 2.

Highlights from this chapter.

- The different meanings of the concept of probability.
- The *addition principle* (for disjoint cases) and *multiplication principle* (for combined cases) in probability and combinatorics.
- The number of *permutations* of n elements, which is $n!$.
- The number of *selections* of k elements among n ones, which is $\binom{n}{k}$.
- Applications of the simple *pigeonhole principle*, which says that if you have more letters than pigeonholes one hole has to get more than one letter.

5.1 Basic Probability

Probability theory concerns statements of the following kind:

1. "The probability that a mobile phone of this model breaks during its first year is less than ten percent."
2. "The probability that a tossed coin will show tails is $1/2$."

3. "The probability that a rolled die will show four is $1/6$."
4. "The probability that you'll get tails three times running when tossing a coin is $1/2^3 = 12.5$ %."
5. "The probability that two rolled dice will show the sum of eleven is $2/6^2 = 1/18 \approx 5.6$ %."
6. "The probability that the Green Party will win the election is at most one in ten"
7. "The probability that there will be a nuclear meltdown during the lifetime of this power station is zero."
8. "The probability that the human being should have been developed by evolution is less than the one that a hurricane blowing through a scrap yard should happen to assemble a working aircraft."

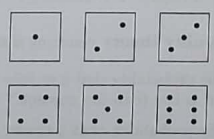
What all statements about probability have in common is that they can be stated as a number between zero and one. An impossible event has the probability of zero. An event that is guaranteed to happen (for instance that the sun will go out sooner or later) has the probability of one (= 100 %).

A closer analysis of the statements above will show that apart from that, there are several completely different kinds of probabilities. The first probability is a measurement that the product department in the factory is able to calculate easily:

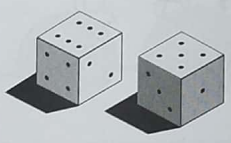
$$\frac{\text{number of phones of this model reported as broken during their first year}}{\text{total number of phones of this model sold}}$$

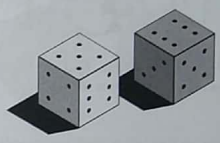
The probability in this case is simply a measurement of the proportion (the measured frequency). Then we assume that this proportion won't change all that much in the nearest future and therefore we base our forecastes on this.

For the tossing of dice and coins in statements 2 and 3 it's a bit different. No one has measured the share of coin throws that have shown tails recently. Instead we expect that the symmetrical design of coins and dice will mean that every side will come up with approximately the same frequency if we throw them a lot of times – since there is no reason why any side should be favoured. Two sides of the same coin will then mean a probability of $1/2$ for tails, while six sides on a die mean a probability of $1/6$ that four spots will be on top.



These basic probabilities are then used in statements 4 and 5 to determine probabilities for combined events. If the coin tosses are independent there will one chance in two in every toss to get tails, and thus one chance in two times two times two to get tails three times running (according to the multiplication principle below). If two dice are to show eleven there are two possibilities, *six and five* and *five and six*. Each of these possibilities has one chance in six times six to occur, and thus the total probability is $\frac{2}{6 \cdot 6}$ (according to the addition principle below).





In statement 6 an estimate of the probability of a certain outcome in an election is made. Every parliamentary election is a unique situation so frequency caculations aren't possible to make. This is called a **subjective probability** based on individual experience and judgement. Since the experiment can't be repeated under the same conditions no probability is "right" interpreted as a frequency.

Figure 5.1: *After the election the probability of defeat proved to be one hundred percent.*

In statement 7 the probability of a complicated accident is estimated. Again there is no possibility of testing the totality of the probability estimation with experiments, but in this case it's probably a total estimation of a number of safety systems where all of them have to break down at the same time. Maybe it's possible to make experiments on all parts of the safety system separately and get objective probability estimations which are afterwards put together in a model giving a probability estimation for a total catastrophe. (Or the person making the statement means that the lifetime of the nuclear power station by definition is over when a meltdown happens!)

Statement 8 is pure nonsense.

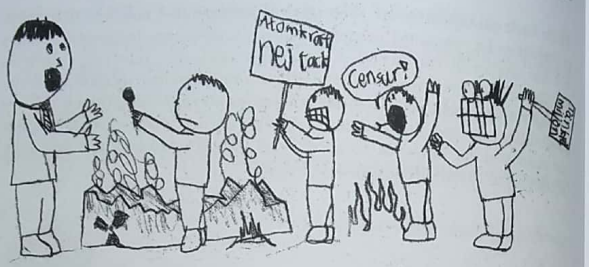


Figure 5.2: *"Nono, just as I said the lifetime of the nuclear power station ended when the meltdown occured!"*

Exercise 5.1 What kind of probability estimations are most likely the case in the following statements:

(a) The probability for smokers of gettin lungcancer is...

(b) The probability of winning 10,000 crowns in this lottery is...

(c) The probability that the duration of the recession will be long is...

5.1.1 Uniformly Distributed Probability

The calculation of probabilites of various complicated events is only possible when the probabilities of all basic events are known. These basic probabilities may be the result of symmetry considerations (as was the case with dice and coins), be measured in experiments, or just be subjectively estimated probabilities. The connection with combinatorics is strongest when we study the kind of situations where all outcomes have the same probability. This is called **uniformly distributed probability** over the set of all possible outcomes, the so-called **sample space**. We can denote the sample space Ω.

Probability is usually denoted P. The probability of a certain **event** A (which is a subset of all the possible outcomes in Ω) when the probability is uniformly distributed will be

$$P(A) = \frac{\text{number of outcomes that lead to } A}{\text{total number of possible outcomes}} = \frac{|A|}{|\Omega|}$$

If we for instance toss a coin twice the probability of tails coming up at least once is

$$P(\text{tails at least once in two tosses}) = \frac{3}{4} = 75\ \%$$

since there are four possible outcomes out of which three give tails at least once:

By the **complement** of an event A is meant the event that A *doesn't* occur. The probability of the complement of event A is always $1 - P(A)$ since the total probability that A either happens or doesn't happen always is $1 = 100\ \%$. The probability of *not* getting tails at least once when tossing twice is thus $1 - {}^3/_4 = {}^1/_4 = 25\ \%$.

Exercise 5.2 If you toss a drawing pin it can either land pin-up or lying on its side. Does it seem correct by analogy with the example of the coins to claim that the probability to get pin-up at least once when tossing twice is ${}^3/_4$?

Exercise 5.3 If you roll two dice, calculate the probability of getting a sum:

(a) equal to eight **(b)** larger than eight **(c)** smaller than eight.

(Hint: Draw all the possible outcomes.)

Exercise 5.4 When tossing a coin three times, calculate the chance of getting:

(a) tails every time

(b) tails only when the previous toss gave heads (but heads several times is allowed)

(c) tails at least twice running.

(Hint: Draw all the possible outcomes.)

Example 5.1: ***The Car and the Goats*** A notoriously difficult probability problem is *The Car and the Goats* (also known as the **Monty Hall problem**).

You are a participant in a TV-programme where it's possible to win a car. The presenter shows you three doors and explains that there is a car behind one of the doors, but that behind each of the other two doors there are only goats. You choose a door. When you have chosen, but before the doors are opened, you are given some more information from the presenter: he opens one of the doors that you haven't chosen and shows that there is a goat. Now you are given the opportunity to switch doors (to the other unopened door) if you want to. Do you?

The answer is that you should switch doors! Because that increases the probability of winning the car from $1/3$ to $2/3$.

If you think in the right way it is easy to find this probability. There are three possible outcomes when choosing a door at first, all equally probable:

In the first case you win if you don't switch doors. In the other two outcomes you win if you switch. The probability of winning when not switching is then equal to the probability that the first outcome is chosen, that is $1/3$. The probability of winning when switching is equal to the probability of having chosen one of the other two outcomes, that is $2/3$. ■

Exercise 5.5 Assume that there are five doors in *The Car and the Goats*, behind which one car and four goats are hidden. Assume that the presenter opens three doors hiding goats. How much does your chance of winning increase if you switch to the remaining door?

5.1.2 The Addition and Multiplication Principles in Probability

At the stock exchange investors can try to decrease the risk in their investments by buying and selling so-called options. Options are a kind of contract that means that you get the right at a future date to buy or sell a certain security at a certain price. By combining different options you can get a risk profile that fits your expectations of how the stock market will develop. A common combination of options means that you gain money if the stock prices don't move more than within a certain interval. You lose money if the prices go down or up a lot.

Assume that we are able to estimate the probability of these events separately. Experience might tell us that $P(\text{the prices go down a lot}) = 2\ \%$ and that

$P(\text{the prices go up a lot}) = 1\ \%$. What is the probability that one of these events occur, either the first *or* the second one? Since the events are **disjoint**, that is, can't occur at the same time, you just have to add their individual probabilities! $P(\text{the prices go down a lot or up a lot}) = 3\ \%$. This is called the **addition principle** and applies every time the problem includes the word "or".

Theorem 5.1: ***The Addition Principle*** If the events A and B are disjoint then $P(A \text{ or } B) = P(A) + P(B)$. ■

Using set theory we can express this as follows: If two sets A and B are disjoint they don't intersect, that is $A \cap B = \emptyset$. The relationship between the sizes of the sets and their union is then as simple as $|A \cup B| = |A| + |B|$, and thereby the addition principle is proved directly:

$$P(A \text{ or } B) = \frac{|A \cup B|}{|\Omega|} = \frac{|A| + |B|}{|\Omega|} = \frac{|A|}{|\Omega|} + \frac{|B|}{|\Omega|} = P(A) + P(B) \qquad (1)$$

Example 5.2 The probability of getting only heads when tossing a coin three times is $1/8$. The probability of getting only tails is $1/8$ as well. The probability of getting the same result every time is then $1/8 + 1/8 = 2/8 = 1/4$. ■

Exercise 5.6

(a) Does the relationship in equation 1 hold even if the probability isn't uniformly distributed?

(b) Does the relationship hold even if the universe is infinite?

Exercise 5.7 What is the probability that the sum of two dice will be at most three or at least ten?

Exercise 5.8 What is the probability that a randomly drawn playing card will be a five or lower or a jack, queen, king, or ace?

Exercise 5.9: *Important!* Derive the formula $P(A \text{ or } B) = P(A) + P(B) - P(A \text{ and } B)$ for non-disjoint events.

Exercise 5.10 What is the probability that a randomly drawn playing card will be clubs or a king?

Exercise 5.11 A bookie at a shady casino tells you: "Stake 100 crowns and roll two dice. If the sum is at most three or at least ten or odd you lose, otherwise you'll get 150 crowns back." What is the chance of winning? How much do you win then? What is the risk of losing? How much do you lose then? Does it seem wise to play the game?

When parachuting, two parachutes are usually used, one ordinary and one extra. For it to end in a really bad way neither of the parachutes would have to unfold. Assume that we can determine the risk of each of these events separately. Let's say that $P(\text{ordinary parachute broken}) = 0.01 = 1\ \%$ and $P(\text{extra parachute broken}) = 0.02 = 2\ \%$. What is the probability that both these events occur, both the first *and* the second one? We assume that the risk of one parachute being broken isn't affected by the fact that the other one is.

We say that two events are **independent** if the probability of one of them isn't dependent on whether the other one occurs or not. For independent events we get the probability that both will occur by multiplying the probabilities of the two separate events: $P(\text{ordinary parachute broken and extra parachute broken}) = 0.01 \cdot 0.02 = 0.0002 = 0.02\ \%$. This principle is called the **multiplication principle** and applies every time the problem includes the word "and".

Theorem 5.2: ***The Multiplication Principle*** If A and B are independent events then $P(A \text{ and } B) = P(A) \cdot P(B)$. ∎

Using set theory we can explain the principle like this: if two events A and B are independent the proportion of A-occurrencies among the B-occurrencies is the same as the proportion of A-occurrencies in the total sample space:

$$\frac{|A \cap B|}{|B|} = \frac{|A|}{|\Omega|} \quad \text{if the events are independent.}$$

If we multiply both sides with $|B|/|\Omega|$ we get the relationship:

$$\frac{|A \cap B|}{|B|} \cdot \frac{|B|}{|\Omega|} = \frac{|A|}{|\Omega|} \cdot \frac{|B|}{|\Omega|}$$

The left-hand side is equal to $\frac{|A \cap B|}{|\Omega|}$, which by definition is the probability that both A and B will occur. The right-hand side is by definition the product of the probability of A and the probability of B. Thus we have proved the multiplication principle for the probability of independent events. (Or, if we want to be very particular: we have shown it for uniformly distributed probabilities in a finite sample space.... The principle holds otherwise as well, but the proof is different.)

Exercise 5.12 Give an intuitive explanation of your own to the relationship above. *

Exercise 5.13 The events A, B, C, and D are independent, with $P(A) = P(B) = 0.5$ and $P(C) = P(D) = 0.8$. Calculate the probability of the following events:

(a) A and B. **(b)** A and C. **(c)** A, B, C, and D.

(d) A but not D. **(e)** Neither A, B, C, nor D.

Exercise 5.14 An alarm system is maintained yearly. The system consists of two parts that we can call I and II. Both parts have to be functioning for the system to work. The risk of part I breaking during the year is 10 percent while the risk of part II breaking is 20 percent. These risks are independent.

(a) What is the risk that the system will stop functioning during the year?

(b) To increase the reliability of the alarm system three copies of part I are installed, and two copies of part II. It's enough that there is one functioning copy of each part for the system to function. Now, what is the risk that the system will stop working during the year?

Exercise 5.15 In the twenty-five largest listed companies there is no female CEO. Assume that the appointments of the CEOs are independent and that the chance that the CEO will be a woman is p every time an appointment is made. Explain why the probability of getting a male CEO in all twenty-five of the companies is $(1-p)^{25}$. How small does p have to be for this probability to be greater than 50 percent?

Exercise 5.16 The Car and the Goats again, as in exercise 5.5 on page 110.

(a) We have five doors, four goats and one car. Now the presenter opens just one door hiding a goat when you have made your choice. How much are your chances increased if you switch to one of the remaining doors?

(b) The same situation again, but now with two cars (a Volvo and a Saab) and three goats. The presenter opens a door hiding a goat. How much do your chances of winning a car increase if you switch to one of the remaining doors? (Hint: You can calculate the chance of winning the Volvo and the chance of winning the Saab separately.)

5.1.3 Conditioned Probability

Of course not all events are independent. The risk of some miscreant sabotaging one of your parachutes is hopefully rather small, but if somebody sabotages one parachute anyway (event B) the risk of her taking the opportunity of making the other one unusable as well (event A) is probably frighteningly high. In other words there is a high proportion of the outcomes where event B occurs in which *both* events A and B occur. This proportion is called the **conditioned probability** for the event A *given* that event B occurs and is usually denoted $P(A \mid B)$.

Theorem 5.3: ***Conditioned Probability*** The conditioned probability of event A given event B is

$$P(A \mid B) = \frac{|A \cap B|}{|B|}$$

■

This theorem follows directly from the definition, since the right-hand side is precisely the proportion of the outcomes where B occurs that consist of outcomes where A occurs as well.

Exercise 5.17 Two cards are to be drawn from a pack of 52 cards. Let A be the event that the first card is the two of spades. Let B be the event that the second card is the two of spades. Calculate the four conditioned probabilities $P(B \mid A)$, $P(A \mid B)$, $P(\text{not } A \mid B)$, and $P(B \mid \text{not } A)$.

Exercise 5.18 Prove that the events A and B are independent exactly when $P(A \mid B) = P(A)$.

Exercise 5.19: *Important!* Derive the formula

$$P(A \text{ and } B) = P(A \mid B) \cdot P(B)$$

for independent events.

5.2 Basic Combinatorics

Combinatorics concerns counting the *number* of items. We have already seen applications of this in the section about uniformly distributed probability, where we for instance counted the number of possible outcomes when rolling two dice. But combinatorics has many other uses; we'll see some of them in the remainder of the text.

5.2.1 The Addition and Multiplication Principles in Combinatorics

In combinatorics we thus calculate numbers instead of probabilities. By just keeping the numerators in the probability quotients we get the combinatorial correspondences to the addition and multiplication principles:

- Assume that there are x ways of processing a case (say X) and y ways of processing another case (say Y) and that X and Y are disjoint cases, that is, they exclude each other. The **addition principle** then says that there will be $x + y$ ways of processing X *or* Y.

- Assume instead that X and Y are independent steps that are to be carried out and that there are x and y ways repectively of doing them. The **multiplication principle** in combinatorics then says that there will be $x \cdot y$ ways of doing X *and* Y.

Example 5.3 The supplement importer BANT-IT is carrying out a customer survey by telephone. The interviewers are told to categorise the respondents according to sex. The men are then to be subdivided into two classes depending on whether they weigh less than or more than 100 kg, while the women are to be subdivided inte three weight classes: less than 50 kg, between 50 and 100 kg, and over 100 kg. How many categories is that in total?

The answer is five $(2 + 3)$, according to the addition principle.

After the subdivision into weight classes everyone has to answer the same battery of questions consisting of two multiple choice questions, both having eight alternative answers. In how many ways can a person answer these two multiple choice questions?

The answer is sixty-four $(8 \cdot 8)$, according to the multiplication principle. ■

Exercise 5.20 The restaurant "Tree of Health" has on its menu 6 different kinds of fruit juice and 11 kinds of herbal tea.

(a) For lunch you may choose either fruit juice or herbal tea. How many choices of beverages are possible?

(b) Dinner, on the other hand, includes both fruit juice and herbal tea. How many choices of beverages are possible?

Here comes a somewhat more extensive example, where both the principles are involved at the same time:

Example 5.4: ***Clothing Choices*** A small child has been given winter clothes from a number of relatives. The child owns

- An olive-green one-piece snowsuit with a hood.
- A pink jacket without a hood and a pair of pink snowpants.
- A mustard-coloured jacket with a hood and a pair of mustard-yellow snowpants.
- A purple hat, a red hat and a grass-green hat.

It the child takes the one-piece suit it should not take pants as well. If it takes a garment with a hood it doesn't need a hat, but may wear one. In how many ways is it possible to dress the child warmly using this set of clothes?

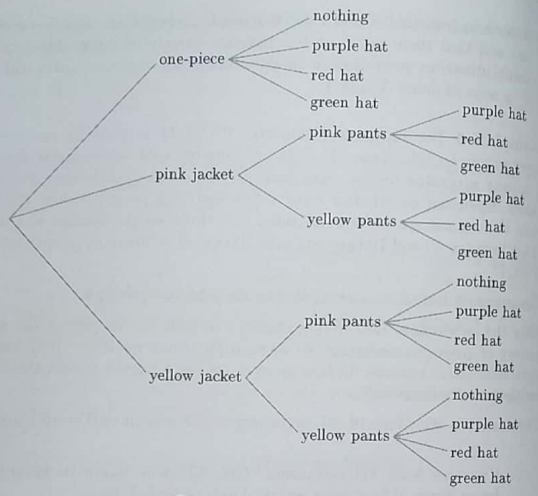


Figure 5.3: *Clothes tree*

One way of illustrating this is in a **decision tree**, according to figure 5.3. The number of possible clothing combinations is equal to the number of end points in the tree, here 18.

The same answer could be found via calculations, as "one-piece and one out of four head-pieces *or* pink jacket and one out of two pairs of pants and one out of three head-pieces *or* yellow jacket and one out of two pairs of pants and one out of four head-pieces", which according to the addition and multiplication principles give $1 \cdot 4 + 1 \cdot 2 \cdot 3 + 1 \cdot 2 \cdot 4 = 18$ possibilities.

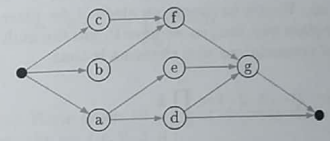
Usually one can't be bothered to draw the decision trees, except for very small problems, but they are a good aid just the same. You can always see them "in front of your inner eye". (More about trees in the next chapter.) ■

Exercise 5.21 Write down all the 18 clothes combinations in plain text.

Exercise 5.22 In a society, a chairperson and a treasurer are to be elected. Running for chairperson are Svea, Alfred, Rut, and Hanna. Running for treasurer are Alfred, Rut, Melker, Kasper, Baltsar, and Livia. The same person can't be both chairperson and treasurer. In how many ways can the two posts be appointed? (Hint: Divide into two cases – one where Svea or Hanna gets to be chairperson and one where Alfred or Rut gets to be chairperson.)

Exercise 5.23 Explain why there are 2^n subsets (the empty set and the set itself included) to a set with n elements. (Hint: We can assemble a subset by choosing, for each of the n possible elements, whether it is to belong to the subset or not.) Then compare this proof and the one in example 4.17 on page 89. Which one of the proofs do you find more easy to understand?

Exercise 5.24 In how many different ways can a process pass through the following flow chart? (The goal is to get from one end to the other somehow, without going against the directions indicated by the arrows.)



Exercise 5.25 In a free magazine on a train you can read an article about the influence of chemicals. According to the article, an average Swedish person smears about 30 different products on themselves daily. (Make-up, soap, lotion, ...) On average the products contain 13 different substances. So, the author says, we daily smear on average 390 different substances on ourselves.

(a) Is this answer calculated using the addition or the multiplication principle?

(b) Is it correct?

5.2.2 Permutations and Ordered Selections

A common type of optimisation problem in an industry is to choose an optimal order in which to perform a number of operations. Let's say that there are ten operations that are to be performed and that in principle it's possible to do them in any order but that different orders are differently good. How many orders are there to try out?

The answer is amazingly large: 3,628,800. We'll soon see how this number is calculated. (In courses in combinatorial optimisation algorithms one studies methods of finding a good order *without* having to try out all of them.)

An ordering of the elements in a set is called a **permutation** of the elements in the set. As an example, we can write down all permutations of the elements in the set $\{1, 2, 3\}$:

123, 132, 213, 231, 312, 321.

So there were six different permutations of a set with three elements.

Exercise 5.26 Draw the decision tree giving the six permutations. *

Exercise 5.27 Write down all the twenty-four permutations of {1, 2, 3, 4}. Be methodical. For instance, begin with all those starting with 1, then all those starting with 2, etc. Can other methods be imagined? *

To find the number of permutations of n elements, it's possible to reason like this: for place one we can choose between n elements. Then we can choose between $n-1$ elements for place two, because one element is already used up. For the same reason, we can then choose between $n-2$ elements for place three, and so on. We are to choose an element for place one *and* to choose an element for place two, etc., so we have to use the multiplication principle. The number of permutations of n elements is thus

$$n(n-1)(n-2)\cdots 3\cdot 2\cdot 1 = \prod_{k=1}^{n} k$$

that is to say: the product of all integers between 1 and n. This number is called ***n*-factorial** and is denoted $n!$. Furthermore we define $0! = 1$ since there is exactly one way of ordering zero elements (do nothing at all). This is so important that we repeat it as a theorem:

Theorem 5.4: ***Permutations*** The number of permutations of n elements is $n!$. ∎

Exercise 5.28: *Important!* Prove that $n!$ satisfies the recursion $n! = n\cdot(n-1)!$ for $n \geq 1$, with the starting value $0! = 1$.

Exercise 5.29 To get an idea about how swiftly factorials grow, calculate $n!$ on your calculator for all n from one and upwards until the precision of your calculator isn't enough. How far did you get? *

Exercise 5.30 Why must there be exactly $n!$ bijections between two sets with n elements each? (Se section 2.6.2 on page 28.)

Exercise 5.31 In how many more ways is it possible to shuffle a pack of cards containing two jokers compared to one without jokers?

Exercise 5.32 In a large grocery store around six p.m. there are a lot of people queueing at four tills.

(a) Suddenly a fifth till is opened. Eight persons dash towards it. In how many different orders is it possible for them to end up?

(b) The same situation as above, but two new tills are opened at the same time. Eight persons dash towards them and distribute themselves so that four persons are queueing at each new till. In how many ways can they end up?

We can also solve problems concerning **ordered selections** in the same way. Assume that there are ten operations to do and that we are to choose three of them to perform in some order. The first operation can be chosen from the ten, the second one from the nine remaining operations, and the third one from the eight operations now remaining. Thus there are $10\cdot 9\cdot 8$ ways of performing three operations out of ten in some order. The reasoning gives the following general result:

Theorem 5.5: ***Ordered Selection*** The number of ways of ordering k elements choosen among a totality of n is

$$n(n-1)(n-2)\cdots(n-k+1) = \prod_{i=0}^{k-1}(n-i)$$ ∎

Exercise 5.33 Write down all ways of choosing three elements, one at a time, from the set {1, 2, 3, 4}.

Exercise 5.34 Explain why

$$n(n-1)(n-2)\cdots(n-k+1) = \frac{n!}{(n-k)!}$$

Which version of the formula seems more practical to use, by the way?

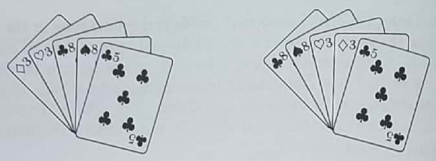
Exercise 5.35 Twenty persons are queueing.

(a) A new till is opened and eight persons dash over. In how many ways can this queue consisting of eight persons be formed from the total number of twenty customers?

(b) Assume instead that two new tills are opened and that eight persons dash over. At least two end up in each new queue. In how many ways can these queues consisting of a total of eight persons be formed from the total number of twenty cusomers?

5.2.3 Unordered Selections and Binomial Numbers

When playing card games, for instance poker or bridge, the only important thing in the deal is *which* cards you get – you don't care about the *order* in which you get the cards. The card hand is thus an **unordered selection** of cards from the pack. How many five-card hands can be put together from a pack containing 52 cards? The answer is that there are 2,598,960 different five-card hands, so there is no great risk of getting the same one several times running.



To find the number of five-card hands it's possible to think like this: drawing five cards in order can be done according to the previous sectionin $52!/47! = 52 \cdot 51 \cdot 50 \cdot 49 \cdot 48$ ways. But now we have counted each card hand 120 times, since there are $120 = 5! = 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$ ways of ordering a hand contaning five cards. Thus the number of unordered five-card hands is

$$\frac{52 \cdot 51 \cdot 50 \cdot 49 \cdot 48}{5 \cdot 4 \cdot 3 \cdot 2 \cdot 1} = 2{,}598{,}960$$

In general, this method shows that the number of ways of choosing an unordered subset with k elements from a set with n elements is

$$\frac{n(n-1)(n-2)\cdots(n-k+1)}{k!}$$

This is an expression that is used so frequently in combinatorics that it has been given a symbol of its own: $\binom{n}{k}$ which is read as n **choose** k. These numbers are called **binomial numbers**.

Theorem 5.6: ***Unordered Selection*** The number of ways of choosing k elements from a set of n is the binomial number

$$\binom{n}{k} = \frac{n(n-1)(n-2)\cdots(n-k+1)}{k(k-1)(k-2)\cdots 1} = \frac{n!}{k!(n-k)!} \qquad \blacksquare$$

Exercise 5.36 Calculate $\binom{7}{2}$, $\binom{7}{3}$, and $\binom{7}{4}$.

Exercise 5.37 Express the formula using the product symbol.

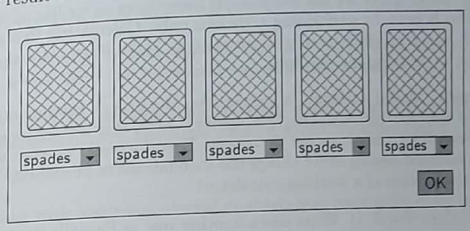
Exercise 5.38 Look through the ordered selections in exercise 5.33 on the previous page, and check that it's true that they correspond to $\binom{4}{3} = 4$ unordered selections. *

Exercise 5.39 In a computer program consisting of a thousand rows, twenty are to be choosen and checked.

(a) In how many ways can this be done?

(b) What is the probability that the last row won't be chosen?

Exercise 5.40 Somebody publishes a prediction experiment on the web: "Below you see five playing cards face down. Try to predict for each card which suit it is: spades, clubs, diamonds or hearts. Click OK when you are finished. Then the cards are turned up so that you can see how many you correct answers you got. The probability of getting all the answers correct by chance is almost non-existent. Please report your result via email to `stupid.experiment@fraudster.se`.



(a) Calculate the "almost non-existent" probability.

(b) The leaders of the experiment get twenty answers. Everyone has five or at least four correct answers! The conclusion made is that divination forces exist. Explain why this conclusion isn't based on facts. (Hint: which persons have done the experiment without reporting their results?)

(c) Suggest a better way of designing the test. *

An efficient way of tabulating the numbers $\binom{n}{k}$ for $n \geq 0$ and $k \geq 0$ is in **Pascal's triangle**:

$$\begin{array}{ccccccccccc}
 & & & & \binom{0}{0} & & & & & & \\
 & & & \binom{1}{0} & & \binom{1}{1} & & & & & \\
 & & \binom{2}{0} & & \binom{2}{1} & & \binom{2}{2} & & & & \\
 & \binom{3}{0} & & \binom{3}{1} & & \binom{3}{2} & & \binom{3}{3} & & & \\
\binom{4}{0} & & \binom{4}{1} & & \binom{4}{2} & & \binom{4}{3} & & \binom{4}{4} & & \\
\vdots & & & & \vdots & & & & & & \vdots
\end{array}$$

If we number the rows so that the top row is row zero then in the n:th row of Pascal's triangle we have the numbers $\binom{n}{0}, \binom{n}{1}, \ldots, \binom{n}{n}$. With the numbers worked out the beginning of the triangle looks like this:

$$\begin{array}{ccccccccccc}
 & & & & 1 & & & & & & \\
 & & & 1 & & 1 & & & & & \\
 & & 1 & & 2 & & 1 & & & & \\
 & 1 & & 3 & & 3 & & 1 & & & \\
1 & & 4 & & 6 & & 4 & & 1 & & \\
 & 5 & & 10 & & 10 & & 5 & & 1 &
\end{array}$$

Note that the sides of the triangle consist only of ones and that all the other numbers are the sum of the two numbers above in the triangle. This property can be described as a recursion for the binomial numbers, **Pascal's recursion**:

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}.$$

Using this property we can calculate as many rows as we like of the triangle, and this is done by performing additions only. Calculating Pascal's triangle in this way is an efficient way of calculating $\binom{n}{k}$ for large n:s and k:s without having to perform any divisions.

Remark Pascal's triangle is named after the French mathematician Blaise Pascal who was active in the 17th century. Still, this triangle was known by mathematicians much earlier than that. Pascal's really pioneering contributions included among other things the foundations of probability theory and the construction of a working calculator!

Exercise 5.41 Write down another row in Pascal's triangle.

Exercise 5.42 Prove the statement that the sides of Pascal's triangle consist of ones and nothing else. Start by phrasing the statement in a more mathematical way. Explain it based on the meaning of the numbers as well.

Exercise 5.43 Prove that every number in the triangle can be obtained by summing the two numbers above. (Hint: Use the recursive version of the definition of factorials, see excercise 5.28 on page 118.)

Exercise 5.44: *Important!* Give an alternative proof of Pascal's recursion by explaining why the number of subsets with k elements taken from $\{1, 2, \ldots, n\}$ equals the number of subsets with k elements from $\{1, 2, \ldots, n-1\}$ plus the number of subsets with $k-1$ elements from $\{1, 2, \ldots, n-1\}$. (Hint: divide the first group of subsets into two cases depending on whether the element n belongs to the subset or not.)

Exercise 5.45 You surely know to perform a quadratic expansion $(x+y)^2 = x^2 + 2xy + y^2$ and perhaps a cubic expansion $(x+y)^3 = x^3 + 3x^2y + 3xy^2 + y^3$ as well. The coefficients 1, 2, 1 and 1, 3, 3, 1 respectively are found as row two and row three in Pascal's triangle. Explain why this isn't pure chance. (Hint: in how many ways can you get a term of type x^2y by choosing factors from the parentheses in $(x+y)(x+y)(x+y)$?) *

Exercise 5.46: *Important!* Derive in the same way the general **binomial theorem**:

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^{n-k} y^k.$$

Since the binomial numbers appear as coefficients in this expression they are often called **binomial coefficients**.

Exercise 5.47 Calculate using the binomial formula $(x+y)^4$ and $(x-y)^4$.

5.2.4 Permutations of Multisets

A set includes one copy of each element. But it is also possible to work with **multisets**, where there can be several copies of some elements.

Exercise 5.48 Mention some situation where it seems reasonable to work with multisets, and some where it definitely *doesn't* seem reasonable.

When working with multisets, several methods may have to be combined.

Example 5.5: *Flag Signal* We have six signal flags: three blue ones, two red ones, and one white one, and are to hoist them in a row to make a signal. How many different signals can we make?

Solution, version 1: There are six flags, so they can be ordered in $6! = 720$ different ways. But not all versions *look* different. Irrespecitve of how the three blue flags are ordered mutually they still give the same signal. That means that we have counted every signal $3! = 6$ times. The same reasoning holds for the red flags; they can be ordered in $2! = 2$ ways. When we compensate for the "double billing" we find that there are

$$\frac{6!}{3!\,2!} = \frac{720}{12} = 60 \text{ different signals}$$

Solution, version 2: There are six possible positions on the flagpole. We are to choose three of these for the blue flags, and that can be done in $\binom{6}{3}$ ways. Among the remaining three positions we have to choose two for the red flags, which can be done in $\binom{3}{2}$ ways, and the last position has to be taken by the white flag. In total

$$\binom{6}{3}\binom{3}{2} \cdot 1 = \frac{6!}{3!\,3!} \cdot \frac{3!}{2!\,1!} = \frac{6!}{3!\,2!}$$

A completely different calculation, but the same answer! ■

Exercise 5.49 Which of the solutions do you think is easier to understand? Which one do you think that you yourself would have thought of first? *

Exercise 5.50 How many different signals would it be possible to make using four blue, three red, and three white flags?

The kind of problem described in this section occurs often enough to merit a name of it's own:

Theorem 5.7: ***Multinomial Number*** The number of ways of ordering n objects, where there are k_1 of the first kind, k_2 of the second kind, and so on up to k_m of the last kind (where $k_1 + k_2 + \cdots + k_m = n$) is

$$\binom{n}{k_1, k_2, \ldots, k_m} = \frac{n!}{k_1!\, k_2! \cdots k_m!}$$

The expression on the left side of the equal sign is called a **multinomial coefficient**. ∎

Exercise 5.51 Write the answer to example 5.5 using a multinomial coefficient.

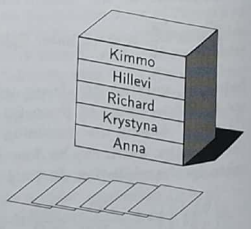
Exercise 5.52 How many different sequences can be created from the letters in the words "multinomial coefficient"?

Exercise 5.53 Binomial coefficients are really a special case of multinomial coefficients. Explain the connection!

Exercise 5.54 Try to calculate $(a+b+c)^3$ using multinomial coefficients. (Generalise the binomial theorem!)

5.3 The Pigeonhole Principle

Assume that the postwoman arrives with six letters that are to be delivered in a post room containing five pigeonholes. Is it possible that she'll end up putting at most one letter in each pigeonhole? No, of course not! Since there are five pigeonholes she can at most put in five letters in total if there is to be at most one letter in each hole. The conclusion we can make is that *there must be some hole containing at least two letters.*



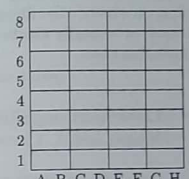
This simple principle is called the **pigeonhole principle**. It can be expressed as a general theorem:

Theorem 5.8: ***The Pigeonhole Principle*** If more than n letters are to be distributed into n pigeonholes some hole will get at least two letters. ∎

The pigeonhole principle is, in spite of its simplicity, surprisingly useful when you want to prove that some situations will never occur.

Example 5.6: ***Chess*** Prove that it isn't possible to put 33 playing pieces on a chess board (8×8 squares, at most one piece on each square) without two pieces ending up side by side in some row.

This problem positively begs to be solved using the pigeonhole principle. The 33 pieces correspond to the letters. We now have to partition the chess board into 32 pigeonholes in such a way that two pieces in the same hole mean that two pieces are standing side by side in a row. A simple way of doing that is this:



If two pieces end up in the same 1×2-rectangle they are standing side by side in the row. Every piece has to end up in one of the rectangles, since they cover the whole board. If 33 pieces are to be put on the board, then according to the pigeonhole principle at least two will end up in the same 1×2-rectangle and thus be standing side by side. ∎

Example 5.7: ***Shaking hands*** At a party n persons meet. Some of them shake hands. Show that there has to be at least two persons who have shaken hands the same number of times.

The maximal number of hand-shakings a person can have participated in are $n-1$ (since people don't shake hands with themselves). The minimal number is zero. There are thus n possible values, to be distributed to n persons. How are we to apply the pigeonhole principle here?

We realise that some of the values can't exist at the same time. If there is somebody with zero hand-shakes, that is, who has shaken hands with nobody, then there can't *at the same time* exist somebody who has shaken $n-1$ hands, that is, who has shaken hands with everybody, and vice versa. Thus there are only $n-1$ *simultaneously* possible values, and then at least two persons have to have the same number. ∎

Exercise 5.55 Show that it isn't possible to place 17 pieces on a chess board without two pieces ending up side by side in some row, column or diagonally. *

Exercise 5.56 In an all-meets-all-tournament every participant plays a game against every other participant. A party of n persons play tennis in such a tournament. Nobody loses all the games they participate in. Show that there has to be two persons with the same score. *

Exercise 5.57 A person has the habit of pairing socks when they are removed from the washing machine.

(a) If there are 20 pairs of socks in the machine, how many in the worst case do you have to remove before finding two that match?

(b) How great is the probability that you really have to remove that many (if you just fish randomly in the machine)?

5.4 Partitioning of Sets

The human being is mad about dividing things into smaller parts. Things can usually be modelled using sets. The mathematics gets different, though, depending on whether the things to be divided are different individuals (such as students) or basically identical (like gingerbread biscuits). Here we'll look at two possible cases.

5.4.1 Distribution of Different Objects – Stirling Numbers

A common problem is to divide a set into a number of *disjoint* subsets. Perhaps you want to subdivide the students in a class in school into a number of groups, for group work. Everyone is to belong to one and only one group. A division of this kind is called a **partitioning** of the set.

If you have decided on the number of subsets you want (which is often the case in group work, where there is a set number of subjects) you can formulate the problem as "divide n different objects into k piles". (It's possible to formulate it in lots of other ways as well, but "piles" is easy to visualise, easier than for instance "unlabelled non-empty containers", which is a common variant.)

The number of ways of doing this is easily expressed recursively:

- Putting n different objects into a single pile can be done in *one* way (put them all into the same pile).
- Dividing n different objects into n piles can be done in *one* way (they have to make up a pile each).
- If we are to divide n objects into k piles we can either divide the first $n-1$ objects into $k-1$ piles and let the last object make up a pile of its own, *or* we divide the first $n-1$ objects into k piles, and throw the last object onto one of them.

This gives us the following recursive equation:

Theorem 5.9: ***The Stirling Numbers*** The number of ways of dividing n different objects into k piles is $S(n,k)$ where

1. $S(n,1) = S(n,n) = 1$
2. $S(n,k) = S(n-1,k-1) + k \cdot S(n-1,k)$ if $1 < k < n$.
3. $S(n,k) = 0$ otherwise. ∎

These numbers are called **Stirling numbers of the second kind**, after the Scottish mathematician James Stirling (1692–1770). (There are Stirling numbers of the first kind as well, but they are outside the scope of this book.)

The Stirling numbers can be described in several ways, but none is actually simpler than the recursive formula. The first Stirling numbers are

	1	2	3	4
1	1			
2	1	1		
3	1	3	1	
4	1	7	6	1

Exercise 5.58 Add another row to the table.

Exercise 5.59 Check that the whole thing seems to work by writing down all the ways of partitioning the set $\{1, 2, 3, 4\}$ into three parts.

Exercise 5.60 How can you calculate the number of ways of giving n different objects to k persons?

5.4.2 Distribution of Identical Objects

Assume that four (bad-mannered[1]) mathematicians are sitting around a conference table, on which a plate with ten gingerbread biscuits is placed. When the meeting is finished the biscuits are finished as well. In how many ways can the biscuits have been distributed among the mathematicians, if the only thing of interest is *how many* biscuits each person has eaten?

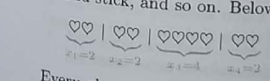
The problem can be rewritten as "How many are the solutions to the equation

$$x_1 + x_2 + x_3 + x_4 = 10$$

if x_1 and so on are natural numbers?" (x_1 then represents the number of biscuits eaten by mathematician n:o 1.)

The funny thing with this problem is that it can easily be rewritten as a completely different problem, one that on top of it is easily solved.

Assume that we place the ten biscuits in a row. In the row we then place three sticks. The first mathematician gets all the biscuits to the left of the first stick, the second mathematician the biscuits between the first and the second stick, and so on. Below we see *one* possible placement.



Every placement of this kind corresponds to one (and only one) biscuit distribution, so we can count the number of placements instead of the number of distributions.

[1] Well-mannered mathematicians eat approximately the same number of biscuits each. Bad-mannered matematicians allow for more solutions.

A placement consists of ten biscuits and three boundary sticks, that is: thirteen objects in total. If we choose three out of the thirteen positions for the sticks, and put biscuits in the remaining ones, we then have a placement. The number of different choices (which is equal to the number of placements, which is equal to the number of distributions) is

$$\binom{13}{3}$$

In this way we are able to solve all problems of this kind! (This may be an example of how a problem is solved using a "flash of genius".)

You can, if you want to, write down a nice formula for this. But formulas are easily forgotten. It is better to think about the problem when you have to solve it.

Exercise 5.61 Write down a formula anyway.

Exercise 5.62 We have five mathematicians and 20 biscuits.

(a) In how many ways can the biscuits be distributed among the mathematicians?

(b) Same as above, with the constraint that we know that everyone ate at least two biscuits.

(c) Same as (a), but with the constraint that it's possible that the mathematicians aren't able to eat all the biscuits.

Exercise 5.63 In how many ways can the integer n be written as a sum of positive integers (if we allow sums consisting of just one term, and sums consisting of the same numbers in a different order are to be counted as different)? (We have, by the way, already solved this problem, in excercise 4.58 on page 102, but in a completely different way!)

Exercise 5.64 We have by now studied a number of different versions of the problem "choosing things". Here we'll run through all of them:

In how many ways is it possible to choose 4 things among 10 different ones if...

(a) You may take the same one several times and the order matters?

(b) You may not take the same one twice and the order matters?

(c) You may not take the same one twice and the order doesn't matter?

(d) You may take the same one several times and the order doesn't matter?

For each of the problems, state as well a realistic example of a situation where such a selection is made.

In these "biscuit-problems" we could also see an application of bijections, that were mentioned in section 2.6.2 on page 28. The different ways of eating the biscuits can be paired with the different solutions to the equation $x_1 + x_2 + \cdots + x_k = n$, and the solutions can be paired with the different ways of placing biscuits and sticks. And the biscuit-and-stick-problem is easily solved, and thereby we have solved the remaining two problems as well! Lots of combinatorial work consists in principle of finding a bijection between the problem one wants to solve and another one that has already been solved.

5.5 Combinatorial Problem-solving

The world is full of combinatorial problems. Some examples:

- Is there any way of seating three couples at a dinner party around a round table so that people sitting next to each other are of different sexes and nobody is sitting next to their partner? In that case, in how many ways is it possible?
- What is the chance of getting at least five correct numbers at Lotto?
- Is it true that you can beat the bank at Black Jack if you are good at keeping track of the cards that have been used?
- How many operations will a program that is sorting n numbers using the algorithm MERGESORT perform before returning a result?

To solve a combinatorial problem sometimes a flash of genius is needed (as in the biscuit-problem in the previous section), but it's possible to get far using a few rules of thumb.

1. Start by solving some small examples where the answers are easily found. That can give an insight into how a larger problem can be solved.
2. Partition the problem into different disjoint cases.
3. Divide the problem into a sequence of independent operations.
4. For some standard cases there are standard solutions: factorials (order problems) and binomial numbers (unordered selections).
5. Be careful in the steps above! It happens so easily that a case is neglected or counted twice. Make an estimate to see if the answer is reasonable.
6. If you aren't able to find the answer at all, trying to simulate the problem on a computer is a possibility.

Exercise 5.65 Is there any way of seating three couples at a dinner party around a round table so that people sitting next to each other are of different sexes and nobody is sitting next to their partner? In that case, in how many ways is it possible?

Exercise 5.66 What is the chance of getting at least five correct numbers at Lotto? (A Lotto row consists of seven numbers picked among thirty-five possible ones.)

Exercise 5.67 Is it true that you can beat the bank at Black Jack if you are good at keeping track of the cards that have been used?

Exercise 5.68 Find out how Mergesort works, and how to proceed to find the number of operations. *

Problems are seldom so simple that you just have to calculate a factorial or a binomial coefficient. Usually several methods have to be combined. Here are some solved examples to show how it may be possible to proceed.

Example 5.8: ***Chemicals*** In how many ways can 12 different chemical substances be ordered on a shelf, if substance A and substance B unfortunately react with each other and therefore can't be placed side by side?

Version 1: It seems a lot simpler to calculate the number of arrangements where the substances *do* stand side by side. Therefore, we'll do that instead, and subtract that number from the total number of possible arrangements, which is very easy to calculate: 12 different objects can be arranged in 12! ways.

If we want to make sure that A and B are placed side by side we can glue them together. Then we'll have 11 objects (10 normal ones and one double) to arrange, which can be done in 11! ways. Furthermore, the glued package can be turned in two ways: as AB and as BA. Because of this there are $2 \cdot 11!$ arrangements of this kind.

This means that there are

$$12! - 2 \cdot 11! = 399{,}168{,}000$$

ways of arranging the substances without strange consequences.

Version 2: It's also possible to reason like this:

There are 10! ways of arranging the harmless substances. Then there are 11 possible positions in which to put A and B: on the left side of the row, on the right side of the row, or in one of the 9 spaces in between. Let A take one place and B one of the remaining 10; we are guaranteed that there will be somehing else in between. This can be done in

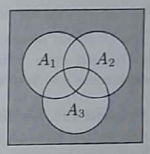
$$10! \cdot 11 \cdot 10 = 399{,}168{,}000 \text{ ways.}$$

The same answer, different reasoning! ■

Very often it is possible, as in this example, to solve one problem in several completely different ways. If you can think up several ways it's usually a good idea to try all of them – if they give the same answer you have probably been thinking in the right way; if you get two different answers at least one of them has to be wrong.

Example 5.9: ***Keyboard Complications*** The key for digit 8 is broken on the keyboard. How many three-digit numbers (integers between 100 and 999) can still be written?

It seems simple to determine the number of three-digit numbers having an eight in a certain position. This can be used for solving the problem, by drawing the situation as a Venn diagram.



The universe consists of the three-digit numbers. A_1 is the set of three-digit numbers where the first digit is eight. A_2 those where the second digit is, and A_3 those where the third digit is. These sets are *not* disjoint.

We are looking for the number of integers in the coloured set, that is in $\mathcal{U} \setminus (A_1 \cup A_2 \cup A_3)$. We have: $|\mathcal{U}| = 900$ (the total number of three-digit integers). $|A_1| = 100$ (the first digit eight, the second and third digits have ten possible values each). $|A_2| = 90$ (the second digit eight, the first and third digits have nine and ten possible values respectively). $|A_3| = 90$ (the third digit eight, the first and second digits have nine and ten possible values respectively). $|A_1 \cap A_2| = 10$ (the first and second digits are eight, the third digit has ten possible values). $|A_1 \cap A_3| = 10$ (the first and third digits are eight, the second digit has ten possible values). $|A_2 \cap A_3| = 9$ (the second and third digits are eight, the first digit has nine possible values). $|A_1 \cap A_2 \cap A_3| = 1$ (all digits are eight).

It follows from exercise 2.28 on page 23 that the number of three-digit integers containing at least one eight is

$$\begin{aligned} |A_1 \cup A_2 \cup A_3| &= |A_1| + |A_2| + |A_3| \\ &\quad - (|A_1 \cap A_2| + |A_1 \cap A_3| + |A_2 \cap A_3|) \\ &\quad + |A_1 \cap A_2 \cap A_3| \\ &= 100 + 90 + 90 - (10 + 10 + 9) + 1 = 252 \end{aligned}$$

The remaining ones, $900 - 252 = 648$ integers, do not include the digit eight and can still be written.

This method is called the **principle of inclusion/exclusion** since one alternates between adding and subtracting.It can be generalised to an arbitrary

number of sets $A_1, \ldots, A_n$:

$$|A_1 \cup \cdots \cup A_n| = \sum_i |A_i| - \sum_{i_1<i_2} |A_{i_1} \cap A_{i_2}| + \sum_{i_1<i_2<i_3} |A_{i_1} \cap A_{i_2} \cap A_{i_3}| - \ldots + (-1)^{n-1} \sum_{i_1<\cdots<i_n} |A_{i_1} \cap \cdots \cap A_{i_n}|$$

All the sums are for indices between 1 and n. ∎

Example 5.10: ***Bookcase*** A person has 20 different books and a bookcase with two shelves. In how many ways can the books be put into the bookcase if both shelves are to be used?

The problem seems to consist of two parts: deciding which books that are to be put onto which shelf, and putting the books in order. It's hard to determine which of these things we should start with, so we try two versions:

Version 1: We divide first and arrange afterwards. If we want to use both shelves we have to put at least *one* book on each shelf. We can pick out a book for the upper shelf (can be done in $\binom{20}{1}$ ways), and put the remaining 19 books on the other shelf. The single book can be arrange in $1!$ way, the remaining ones in $19!$ ways. Or we pick two books for the upper shelf, and so on. In total we get

$$\binom{20}{1} 1!\,19! + \binom{20}{2} 2!\,18! + \binom{20}{3} 3!\,17! + \cdots + \binom{20}{19} 19!\,1!$$

This looks rather discouraging, but if we write down the values of the binomial coefficients we get

$$\frac{20!}{1!\,19!}\,1!\,19! + \frac{20!}{2!\,18!}\,2!\,18! + \frac{20!}{3!\,17!}\,3!\,17! + \cdots + \frac{20!}{19!\,1!}\,19!\,1! = \underbrace{20! + 20! + 20! + \cdots + 20!}_{19 \cdot 20!'s} = 19 \cdot 20! \approx 2.4 \cdot 10^{18}$$

Version 2: We arrang first and divide afterwards. We put the 20 books in a row, which can be done in 20! ways, and then we divide the row into two parts, where the first part is to be placed on the upper shelf and the second one on the lower one. A row consisting of 20 items can be divided in 19 places, so in total we get

$$19 \cdot 20!$$

This was a lot simpler!

Remark If you get an answer as simple as the one in version 1 you may suspect that there is also a simple way of finding it. ∎

The "To Ponder About"-exercises at the end of the chapter can be seen as belonging to this section.

5.6 More Exercises

5.6.1 Routine Work

Probability

Exercise 5.69 What is the probability that the sum of two dice will be odd?

Exercise 5.70 What is the probability that a randomly drawn playing card from a pack will be clubs or diamonds?

Exercise 5.71 One year Kimmo gambled on LOTTO – a game where a winning row containing seven numbers taken from thirty-five are drawn. Every month Kimmo sent the same row: (2, 3, 5, 7, 11, 13, 17).

(a) What is the special mathematical property of Kimmo's row?

(b) What was the probability that Kimmo wouldn't get all the numbers correct at least once during the year?

Combinatorics

Exercise 5.72 If you are going to shoe a horse you have to hammer in four nails at each side of the shoe. To avoid the shoe getting askew you should see to it that the difference between the number of nails on the two sides never exceeds one. In how many ways can you choose the order in which you place the nails?

Exercise 5.73 We have $2n$ work groups in a class and are to arrang the groups in pairs, so that they can criticise each other's work.

(a) In how many ways can this be done?

(b) Write down all possible pairings of the groups a, b, c, d, e, and f, to check that your reasoning was correct.

Exercise 5.74 A **palindrome** is a word that looks the same forwards and backwards. How many five-letter palindromes can be made from the letters {a, b, c, d}

(a) if you may use the same letter several times?

(b) if you mustn't use a letter more than twice?

(c) if you *have* to use at least one letter more than twice?

(d) How many more do the palindromes get if they are to consist of six letters instead?

Exercise 5.75 An ordinary Swedish registration number for a car consists of three letters between A and Z, W included but without I, Q, and V, followed by three digits.

(a) How many registration numbers can be made using this specification?

(b) In how many of them are all the symbols different?

(c) How many end with an even digit?

(d) Do you think that all the combinations are used? *

Exercise 5.76 A juggler owns three yellow, four black, six red, two green, and six purple balls.

(a) For her circus number (juggling with five balls) she has to choose one ball of each colour. In how many ways can that be done?

(b) For the number this evening the juggler wants five balls of the *same* colour. In how many ways can that be done?

Exercise 5.77 Show the equality

$$\binom{n+2}{k} = \binom{n}{k} + 2\binom{n}{k-1} + \binom{n}{k-2}$$

where n and k are at least 2

(a) using the formula for binomial numbers (algebraic proof).

(b) by analysing the meaning of the expressions (combinatorial proof).

Exercise 5.78 Show the equality

$$\binom{r}{m}\binom{m}{k} = \binom{r}{k}\binom{r-k}{m-k}$$

(a) using the formula for binomial numbers (algebraic proof).

(b) by analysing the meaning of the expressions (combinatorial proof).

Exercise 5.79 A 7-person committee is to be chosen in a workplace where 16 men and 12 women work. In how many ways can that be done

(a) if we don't have any special restrictions?

(b) if the sexes are to be represented in a proportional way?

(c) if there has to be at least one representative of each sex?

(d) if the committee is to be single-sex?

Exercise 5.80 A **binary string** of length n is a sequence of n zeros and/or ones.

(a) How many binary strings of length 8 are there?

(b) How many of those include exactly two ones?

(c) How many include an even number of ones?

Exercise 5.81 Insert $x = y = 1$ into the binomial theorem and derive that the sum of the binomial coefficients on row n in Pascal's triangle is 2^n. Explain as well why this means that there are 2^n subsets of a set with n elements. (We already knew that – see exercise 5.23 on page 117.)

Exercise 5.82 Insert $x = 1$ and $y = -1$ in the binomial theorem and derive that the alternating sign sum of the binomial coefficients in row k of Pascal's triangle is 0 for $k \geq 1$. Explain as well why this means that the numbers of subsets with an odd number of elements and with an even number of elements are equal. *

Exercise 5.83 5 cards are drawn from an ordinary pack of card (52 cards; 13 ranks in 4 suits). Calculate the number of ways you can get:

(a) A flush, that is: five cards of the same suit.

(b) Four of a kind, that is: four cards of the same rank.

(c) Full house, that is: three cards of one rank and and two of another rank.

(d) Two pairs, that is: two different pairs and a fifth card of a third rank.

(e) (For programmers:) Write a program that finds all the five-card hands and counts these cases to check that the reasoning was correct.

Exercise 5.84 A bridge hand consists of 13 cards from a normal pack. When you assess a bridge hand you count points for the cards according to the following rules: each ace gives 4 points, each king gives 3 points, each queen gives 2 points and each jack gives 1 point.

(a) How many different bridge hands will give exactly one point?

(b) What is the probability that a randomly chosen bridge hand will give exactly one point?

Exercise 5.85 We have a regular octagon.

(a) How many triangles, where the corners are corners in the octagon as well, are there? (One of them is shown in the picture.)

(b) How many of those don't have any side that belongs to the octagon?

(c) Answer the first question for a regular n-gon.

(d) Answer the second question for a regular n-gon.

(Here, by the way, we have the explanation why *trigonometry* – measuring triangles – is so useful. All polygons can be subdivided into triangles, so if you are able to work with triangles you are able to work with any kind of figure.)

Exercise 5.86 An exam includes the question

> How many 4-digit codes (using decimal digits) are there that include exactly 3 different digits?

50 % of the students answer $10 \cdot 9 \cdot 8 \cdot 3 = 2160$. This answer is *wrong*.

(a) In what way have the persons giving this answer been reasoning?

(b) Wherein lies the error?

(c) What is the correct answer?

Exercise 5.87 Calculate, using the principle of inclusion and exclusion, the number of four-digit integers that can be written without the digit zero (see example 5.9 on page 131). *

Exercise 5.88 How many of the integers between 1 and 100 are coprime to both 5, 7, and 9?

5.6.2 To Ponder About

Exercise 5.89: ***Number of Divisors***

(a) Can you figure out some method that will make it possible, based on the prime factorisation of a number, to determine the number of positive divisors of the number? (You can start by studying the number 40, which was analysed in exercise 3.6 on page 39.)

(b) What does the prime factorisation have to look like if a number is to have exactly six positive divisors?

Exercise 5.90 Show that a company consisting of six persons must include either a group of three persons that don't know one another or a group of three persons that do know one another.

Exercise 5.91 Show that if we take 51 integers between one and one hundred one of the numbers in the set has to divide some other one.

Exercise 5.92: ***Showing Pictures*** The researchers KJ and LK are doing an experiment. Test subjects are to look at a succession of pictures through 3D glasses, and state how far away they think the depicted objects are. The researchers have a certain number (let's say n) pictures, but to get more data they have decided to show every picture twice. Then they can see if people make consistent estimates as well. Since it's possible that the opinion about a picture is affected by the way the previous picture looked, the pictures are shown in a randomised order. But now problems arise: if the two copies of a certain picture are shown consecutively, the subject thinks "but I did indicate picture change, didn't I?" and changes the picture again, without stating anything about copy 2. This messes up the data. Furthermore, it seems to happen every time the experiment is run. The researchers decide, out of pure curiosity, to figure out how large the probability of this problem is. After staring at the problem statement for a while they give up, and LK says resignedly that this must be very difficult. KJ states in a dead certain way that this can be solved by a certain HG in ten minutes.

Well, how great is the probability? Start by calculating it for a given value of n (2 and 3 are realistic). Then try to design a formula covering the general case. In addition, suggest a way of generating a random order where this problem can't arise. *

Exercise 5.93 Prove the principle of inclusion and exclusion (see example 5.9 on page 131) using induction. *

Exercise 5.94: ***Hats*** Use the principle of inclusion/exclusion to solve the following classical problem: a number of absent-minded professors arrive at a meeting. They all put their hats on the hat shelf. After the meeting all of them take a hat from the shelf, without considering whether it's the right hat. What is the probability of no professor getting the right hat? *

Exercise 5.95 Show that $\sum_{k=0}^{n} \binom{n}{k}^2 = \binom{2n}{n}$.

Exercise 5.96: ***Trigonometry*** In trigonometry we find the two addition formulas

$$\cos(x+y) = \cos x \cos y - \sin x \sin y \qquad \sin(x+y) = \sin x \cos y + \cos x \sin y$$

Using these it is possible to find formulas for "the double angle", according to

$$\cos 2x = \cos(x+x) = \cos x \cos x - \sin x \sin x = \cos^2 x - \sin^2 x$$
$$\sin 2x = \sin(x+x) = \sin x \cos x + \cos x \sin x = 2 \cos x \sin x$$

In the same way it is possible to find formulas for $\cos 3x$, $\sin 4x$, and so on. Then the following can be noted:

$$\cos nx = \binom{n}{0} \cos^n x \sin^0 x - \binom{n}{2} \cos^{n-2} x \sin^2 x + \binom{n}{4} \cos^{n-4} x \sin^4 x - \dots$$

$$\sin nx = \binom{n}{1} \cos^{n-1} x \sin^1 x - \binom{n}{3} \cos^{n-3} x \sin^3 x + \binom{n}{5} \cos^{n-5} x \sin^5 x - \dots$$

Prove this using induction!

6 Graph Theory

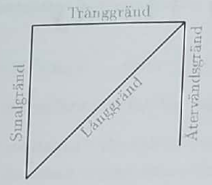
By **graph** in discrete mathematics is meant a structure consisting of dots connected by lines. Graphs are useful when one wants to model how for instance things, persons, or connection points are related to one another. In graph theory there are many useful models, concepts, and results ready for use in different situations. Graph theory is one of a discrete mathematician's most important tools.

Highlights from this chapter.

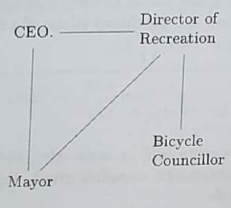
- Graph theory concepts such as *node, edge, degree, adjacent, path, cycle, connected component, complete graph, subgraph, bipartite graph,* and *directed graph.*
- Two particularly useful concepts: *Hamiltonian cycles* and *Eulerian circuits.*
- Similarity between graphs, so called *isomorphism.*
- Two ways of representing graphs when using computers: *adjacency matrix* and *incidence matrix.*
- The absolutely most common special case of graphs: *trees* (connected graph without cycles).
- Searching in rooted trees according to two methods: *breadth-first* and *depth-first.*
- Storage of data in *binary trees* and output in *in-order, pre-order,* and *post-order.*
- Modelling using graphs.

6.1 Basic Concepts in Graph Theory

Example 6.1: ***Bastumåla Tourist Office*** The tourist office in the little town of Bastumåla is distributing boasting brochures showing the town's extensive network of bicycle lanes:



On the back of the brochure the members of the municipal council are presented, and a description of who has regular meetings with whom:



A structure consisting of dots and lines, or junctions and connections, can be used to represent several different things. The same graph, , was a bicycle map when the connections were named, and an organisation chart when the junctions were named. ■

A connection point is called a **node** or **vertex** and a connection is called an **edge** or **arc**. The graphs above have four nodes and four edges. A **graph** is simply a set of nodes among which some pairs are connected by edges. Two nodes that have an edge between them are **adjacent** in the graph.

Exercise 6.1 Rudolph knows Donner and Santa. Both Donner and Elf know Santa as well. Draw a graph representing these relationships. *

Exercise 6.2 Do you know of some other mathematical concept that is called a *graph* as well?

The set of nodes or verticis in a graph is often called V and the set of edges E. A graph G can be defined as $G = (V, E)$, since the whole graph structure is determined by what the nodes are and which the edges are.

Graph theory contains lots of concepts and terms; here comes a first, very basic, concept:

Definition 6.1: ***Degree*** The **degree** of a node is the number of edges starting at the node.

The nodes a, b, e, and g in this graph have degree 3, the nodes c and f have degree 4, and nodes d and h have degree 2. ■

If we calculate the sum of the degrees in the graph above, we get $3+3+4+2+3+4+3+2 = 24$, an even number. Here comes the first theorem in this chapter:

Theorem 6.1: ***The Sum of the Degrees*** The sum of the degrees in a graph is always an even number. ■

Exercise 6.3 State the degees of the nodes in the graph showing the municipal council in Bastumåla, and check as well that the sum is even.

Exercise 6.4: *Important!* Prove theorem 6.1. (Hint: all the edges have two ends.)

Exercise 6.5: *Important!* Explain why the exercise above implies the so called **hand-shaking lemma**, which states that during a cocktail party there will always be an even number of guests that have shaken hands an odd number of times. (Hint: Let the guests be nodes and draw an edge between two persons that have shaken hands.)

Exercise 6.6 Prove the hand-shaking lemma using induction (over the number of handshakes) instead. (Hint: When a new handshake takes place there are three cases. Either both persons have shaken hands an even number of times, or both an odd number of times, or one an even number and the other one an odd one. What will happen to the number of persons having shaken hands an odd number of times in these three cases?) *

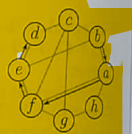
Graph theory contains, to begin with, a number of concepts that have been developed since they are natural and mirror relevant phenomena in the real world. An example: in the network of bicycle lanes in Bastumåla bikes drive around. In the municipal council the persons leave messages to each other. Both these things mean that something (bikes or messages) moves between the nodes in the graph.

Definition 6.2: ***To Move Around in a Graph***

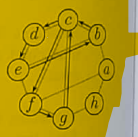
- A **walk** in a graph goes from node to node along edges. A walk can be described by listing the nodes in the order they are passed. This is the walk $e-b-c-b-a$.

- **A trail** is a walk that doesn't use the same edge twice. Here is the trail $d-c-g-f-c-b$.

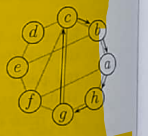
- **A path** is a trail that doesn't pass the same node twice. Here is the path $b-a-f-e-d$:



- **A circuit** is a trail that is **closed**, that is, that starts and ends at the same node. Here is the circuit $b-c-f-g-c-d-e-b$. In a circuit one doesn't care about where the start and finish is, only about the edges used. This circuit could just as well have been written as $f-g-c-d-e-b-c-f$. On the other hand, one does care about the *direction*; $b-e-d-c-g-f-c-b$ is a different circuit.



- **A cycle** is a closed path. $g-c-b-a-h-g$:



- A graph is **connected** if there is a path between every pair of nodes in the graph. A graph that isn't connected consists of a number of **connected components**, that is, parts that are connected but with no connections to other parts.

The graph that has illustrated the previous concepts is connected. The graph shown here is not connected, but consists of three components. For instance, there is no path between node e and node f. Note that this still counts as *one* graph, in spite of it consisting of several parts!

(We warn you that this terminology isn't standardised – in any language. In different books the words circuit, cycle, and so on can mean different things.) ∎

These concepts (no matter what they are called) are fundamental in graph-theoretical analysis. There exist for instance efficient algorithms for finding the shortest walk between two points, for checking whether a graph contains a cycle, and for checking whether a graph is connected. These algorithms, though, are outside the scope of this book.

Exercise 6.7 A traffic network is given. Consider what the graph-theoretical concepts walk, trail, path, circuit, and cycle could correspond to in the world of a bus driver. *

Exercise 6.8 Draw a Venn diagram showing the sets $W = \{\text{possible walks}\}$, $T = \{\text{possible trails}\}$, $P = \{\text{possible paths}\}$, $K = \{\text{possible circuits}\}$, and $C = \{\text{possible cycles}\}$ so that it's clear what is a subset of what.

Exercise 6.9 The shortest walk between two points is always a path. Explain why! *

Exercise 6.10

(a) Figure out some problem that can be illustrated using a graph that ought to be connected.

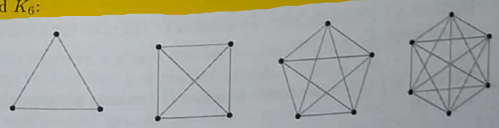
(b) Figure out some problem where the graph probably won't be connected.

Exercise 6.11 Explain why the number of connected components has to be less than or equal to the number of vertices, but the number of edges can be both greater and smaller than both these numbers. *

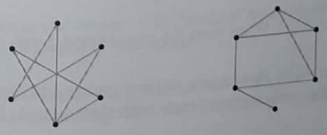
Some graphs contain lots of edges, others just a few.

Definition 6.3: ***Complete Graphs, Bipartite Graphs, and Related Concepts***

- A graph is **complete** if every pair of nodes is connected via an edge. The complete graph with n nodes is denoted K_n. Here we have K_3, K_4, K_5, and K_6:



- For each graph G there is a unique **complement graph** $\overline{G}$ which has the same node set as G but which has edges between precisely those pairs of nodes that don't have an edge in G. Here we see a graph and its complement.



- A **subgraph** of a graph G is a graph G' that consists of a subset of the nodes and edges of G. If $G = (V, E)$ and $G' = (V', E')$ then $V' \subseteq V$ and $E' \subseteq E$ has to hold.

The graph on the right is a subgraph of the graph on the left. If a graph has n nodes both the graph and its complement are subgraphs of K_n.

- A graph is **bipartite** if the node set can be partitioned into two disjoint subsets (let's say left and right) so that all the edges go between the left and the right node set.

Here is an example of a bipartite graph.

- A graph is a **complete bipartite graph** if it is bipartite and every node in the left node set is connected to every node in the right one. The complete bipartite graph that has m nodes to the left and n nodes to the right is denoted $K_{m,n}$.

Here's a picture showing $K_{4,3}$. ■

Exercise 6.12 Draw K_1, K_2, and $K_{1,2}$. *

Exercise 6.13 Explain why the number of edges in the complete graph K_n is $\binom{n}{2}$.

Exercise 6.14 Explain why the number of edges in the complete bipartite graph $K_{m,n}$ is mn.

Exercise 6.15 Here are three graphs. Determine whether they are bipartite!

(a) 

(b) 

(c) 

Exercise 6.16 What does the complement graph of $K_{4,3}$ look like?

Exercise 6.17 How many different subgraphs can be made from a graph that looks like a triangle with the nodes A, B, and C?

Exercise 6.18 If a graph describes a social network, that is to say: which people know which, what does the complement graph then describe?

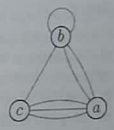
Exercise 6.19 What does it mean if a social network can be described with a complete graph? When do such situations occur?

Exercise 6.20 Bipartite graphs can be used to model the happenings at a certain kind of dance event, but not at all kinds. What do we refer to with this cryptic statement?

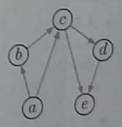
There are situations where more than one edge between two nodes is needed to model the real world (for instance when modelling two different roads between the same two cities). There are also situations where the edges can only be traversed in one direction (for instance a one-way street). Since this gives new possibilities for the ways a graph can look, a number of new concepts are needed.

Definition 6.4: ***Different Kinds of Edges***

- Two edges or more between the same pair of nodes are called **multiple edges**. The graph here has multiple edges between a and b and also between a and c.



- An edge that starts and ends at the same node is called a **loop**. This graph has a loop at node b.
- If one wants to stress that multiple edges and loops are allowed one says that one works with **multigraphs**.
- If instead one wants to make clear that multiple edges and loops mustn't occur in some context, one says that one works with **simple graphs**. Complete graphs and bipartite graphs, for instance, are simple graphs.
- If the edges in a graph can be traversed in only one direction one has a **directed graph**. That is usually indicated with an arrowhead. Otherwise, one has an **undirected graph**. An edge in a directed simple graph can be represented with an (ordered) pair of nodes: (a, b); an edge in an undirected simple graph as a two-element set: $\{a, b\}$. All the graphs we have studied this far have been undirected; here is a directed one.



Usually it's clear from the context if by "graphs" is meant directed or undirected graphs, simple graphs or multigraphs. ■

Exercise 6.21 Try to think of some contexts where it's natural to use simple graphs and multigraphs, respectively, for modelling. *

Exercise 6.22 Try to think of some contexts where it's natural to use undirected graphs and and directed graphs, respectively, for modelling.

Exercise 6.23 Contemplate on the concept of "degree".

(a) Does the concept work with multiple edges as well?

(b) Does it work with loops?

(c) Does it work with directed edges?

Exercise 6.24 Does it hold for multigraphs as well that the sum of the degrees is even (see theorem 6.1 on page 141)?

6.2 Euler and Hamilton – two Classical Graph Problems

Hamiltonian cycles and *Eulerian circuits* are two classical concepts in graph theory.

The concepts are easiest understood when the graph models a road network. A postwoman who carries post has to walk through all the roads in her district to pass all the addresses, but she doesn't want to go along the same little street several times since this is walking unnecessarily. The ideal route of the postwoman is thus along each *edge* (street) exactly once – and this is what is meant by an Eulerian circuit.

For a salesman travelling along the main roads between different cities it is unnecessary to get back to the same city several times during the same tour, if it can be avoided. The ideal route of the travelling salesman thus visits each *node* (city) exactly once – and this is what is meant by a Hamiltonian cycle.

Definition 6.5: ***Euler and Hamilton***

- A **Hamiltonian path** is a path that visits every vertex in the graph exactly once. $a-b-c-d-e$ is a Hamiltonian path in the "envelope graph" to the right.

- A **Hamiltonian cycle** is a closed Hamiltonian path. $a-b-c-d-e-a$ is a Hamiltonian cycle in K_5 to the right.

- An **Eulerian trail** is a trail that follows every edge in the graph exactly once. $a-b-c-d-e-a-d-b-e$ is an Eulerian trail in the envelope graph.

- An **Eulerian circuit** is a closed Eulerian trail. $a-b-c-d-e-a-c-e-b-d-a$ is an Eulerian circuit in K_5. ■

Exercise 6.25 Explain why a graph has an Eulerian trail if and only if it's possible to draw all the edges in one move without lifting the pen.

Exercise 6.26 Explain why a graph has an Hamiltonian cycle if and only if the graph can be drawn as an pearl necklace with some extra threads.

6.2.1 The Background of the Problems

The classical concepts Hamiltonian cycles and Eulerian trails have been given several modern applications and played among other things a major part at the gigantic data processing that had to be made during the charting of the human genome (the so called Hugo project). The concepts are named after their inventors, the Irishman William Rowan Hamilton (1805–1865) and the Swiss Leonhard Euler (1707–1783). Both concepts originated as recreational mathematics! (See below.) Afterwards, they have been of great practical importance.

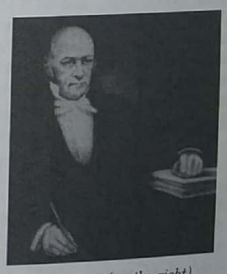


Figure 6.1: *Euler (on the left) and Hamilton (on the right).*

Hamilton was appointed professor of astronomy at the age oftwenty-two, but did also make ground-breaking contributions to mathematics, mechanics and optics. He was an infant prodigy such as seldom seen as well, and at the age of fourteen spoke thirteen languages.

Exercise 6.27 How many languages did Hamilton speak at the age of sixty-four?

In year 1856 Hamilton marketed a game consisting of a dodecahedron (a solid where the twelve sides are regular pentagons) that represented a globe. On each of the corners the name of a big city was printed, and roads between these cities were placed along the edges on the solid. The point of the game was to find a route that visited all the cities but never returned to any city before being back at the starting point.

Exercise 6.28 How many cities must there have been on the solid?

Exercise 6.29

(a) Draw the cities on the dodecahedron as a graph.

(b) Explain why the task can be formulated as finding a Hamiltonian cycle in this graph. Find such a cycle as well.

Exercise 6.30 An important task in modern molecular biology is to read so called DNA sequences. A DNA sequence can be described as a sequence of symbols from { A, C, G, T }.

It's hard to read long DNA sequences in one go. Because of this, one usually reads short subwords, and tries to piece them together to the original sequence. Let's say that we can read subwords of three letters in length. Then in the sequence AGAGCT we can read four subwords: AGA, GAG, AGC, and GCT. The problem when we want to piece them together is that we aren't told in which order the subwords are to be placed. Because of this, a combinatorial trick is used: A directed edge is drawn between the word XYZ to the word YZW if the first word ends with the same two letters that the second word begins with: YZ. For instance, the words AGA, GAG, AGC, and GCT generate the directed graph

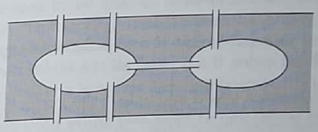
We will get the original sequence if we are able to find a unique directed Hamitonian path in this graph; in that case we put the words together in that order. The example has the Hamiltonian path AGA→GAG→AGC→GCT which gives the sequence AGAGCT.

(a) Find, using this method, the DNA sequence that gives the subwords ATG, CAT, CGC, GCA, GCG, TCA, TGC.

(b) Explain as clearly as possible why this method works, that is, what a Hamiltonian path in this graph has to do with the original sequence.

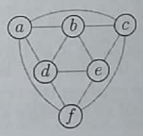
(c) Show, by finding a set of subwords such that several different sequences can be pieced together, that the sequence can't always be uniquely determined.

In the year 1736 Leonard Euler solved the problem of "the bridges of Königsberg". Like Riddarholmen and Gamla Stan in Stockholm parts of the city of Königsberg were situated on two islands. There were bridges both between the two islands and between the islands and the mainland on each side, as shown in the figure below.



The citizens of Königsberg were said to amuse themselves during their Sunday walks (that of course both start and end up at home somewhere in the city) by trying to cross each bridge exactly once. No one had ever succeded, and Euler managed to prove that it was impossible.

Exercise 6.31 Find at least two different Eulerian circuits in the graph below:



*

Exercise 6.32 Model the bridges of Königsberg using a multigraph and explain why the problem is equivalent to finding an Eulerian circuit in this graph.

Exercise 6.33 Assume that we walk around a circuit in a graph. Explain why we have to arrive at each node the same number of times as we leave it.

Exercise 6.34 Explain why the previous exercise means that every node has to have an even degree if an Eulerian circuit is to be possible.*

Exercise 6.35 Consider the previous three exercises again and write a complete and easily understood proof showing that it is impossible to walk along the briges of Königsberg starting and stopping at the same place and crossing each bridge exactly once. *

Exercise 6.36 If you may start and stop at different places (for instance if you are going to have Sunday dinner at your mother-in-law's), is it then possible to walk across all the bridges exactly once? Explain why this would mean that there is an Eulerian trail in the multigraph.*

Euler carried out the line of argument in the exercises and also found that the conditions concerning the degrees are not just necessary but also sufficient (if only all the edges are connected at all).

Theorem 6.2: ***Eulerian Graphs***

- A graph has an Eulerian circuit if and only if the graph is connected (except for isolated nodes, if any) and all the nodes have an even degree.
- A graph has an Eulerian trail if and only if the graph is connected (except for any isolated nodes) and at most two nodes have an odd degree. ∎

Exercise 6.37 Explain how the "only if"-parts of the theorem follow from the exercices above.

Exercise 6.38 Prove the "if"-part of the theorem by stating an algorithm that finds an Eulerian circuit or trail if the conditions are met. (Hint: Sketch a graph that meets the conditions. Walk around randomly until stopped at a node where all the edges have been used. Which node does this have to be? If some edge alongside the walk is unused, start walking along it and go on walking until it's again impossible to continue. In which node does that have to happen? Realise that it is possible to combine the two walks to one continuous walk. Keep on like this until all the edges are used up!)

Remark Königsberg was an East Prussian university city in which the leading philosopher of the 18th century, Immanuel Kant, was active. It must have been a very attractive city, since Kant never left Königsberg and its environs during his whole life! In 1946 Stalin changed the name of the city to Kaliningrad, which from then on has been a Soviet or post-Soviet military naval base.

6.2.2 The Complexity of the Problems

Let's say that a graph is **Eulerian** if it has an Eulerian circuit and that a graph is **Hamiltonian** if it has a Hamiltonan cycle.

Exercise 6.39 Find four graphs, all having eight nodes, so that:

(a) one graph is both Hamiltonian and Eulerian,

(b) one graph is Hamiltonian but not Eulerian,

(c) one graph is Eulerian but not Hamiltonian,

(d) one graph is neither Hamiltonian nor Eulerian.

Exercise 6.40 Think of some context where it can be of interest to determine whether a graph is Hamiltonian or Eulerian.

So a graph can be Hamiltonian but not Eulerian and vice versa. In practical applications one wants to be able to determine whether a graph has either of these properties. The strange thing is that the difficulties in determining the two properties are so different.

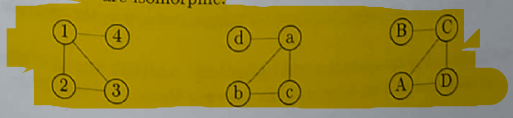
- To determine whether a graph is Eulerian is extremely easy. According to Euler's theorem we just have to check whether it's connected and whether all the nodes have even degree. Both these things can be determined in **linear time** (which means that the time needed is directly proportional to the number of nodes).
- To determine whether a graph is Hamiltonian is extremely hard. (NP-complete, the kind is called.) In principle one has to investigate all the possible paths, and they are usually exponentially many.

Exercise 6.41 Show that the complete graph K_n has $(n-1)!$ Hamiltonian cycles, for all $n \geq 3$. Remember that since a cycle is closed it doesn't matter in which node it starts; it's still the same cycle. Remember also that the nodes in a cycle are ordered, so it counts as a different cycle if we take the nodes in the opposite order.

Exercise 6.42 Show that the complete bipartite graph $K_{2,n}$ has $2(n-1)!$ Eulerian circuits if n is even, and zero Eulerian circuits if n is odd. *

6.3 Isomorphism and Representation of Graphs

A graph is an abstraction of a relation between certain objects that we interprete as nodes, where the relation is shown as edges beween certain nodes. How the nodes are to be placed is thus not specified, nor if the edges are to be straight or curved or squiggly. Two graphs that may look different but that can be drawn in the same way are called **isomorphic**. As an example, the graphs below are isomorphic:



One way to uncover the graph structure itself without introducing random drawing is to represent the graph using its **adjacency matrix**. In the adjacency matrix every node is given a row and a column. In position (i, j) 0 is written if there isn't an edge from node i to node j, and 1 if there is such an edge (or a higher number if there are multiple edges).

	1	2	3	4
1	0	1	1	1
2	1	0	1	0
3	1	1	0	0
4	1	0	0	0

	a	b	c	d
a	0	1	1	1
b	1	0	1	0
c	1	1	0	0
d	1	0	0	0

	A	B	C	D
A	0	0	1	1
B	0	0	1	0
C	1	1	0	1
D	1	0	1	0

Above we see the adjacency matrices for the three isomorphic graphs in the example. But the adjacency matrices don't all look alike!

Exercise 6.43: *Important!* What is the explanation as to why the adjacency matrices don't look alike for the isomorphic graphs?

Exercise 6.44 Write down the adjacency matrix for the envelope graph in definition 6.5 on page 146.

Exercise 6.45

(a) Which graphs don't have symmetrical adjacency matrices?

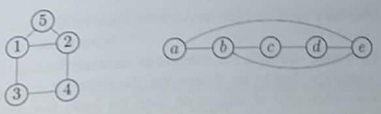
(b) Which graphs don't have only zeros along the main diagonal in the adjacency matrix?

Exercise 6.46 In how many different ways can an adjacency matrix for a graph with four nodes be written? (The columns are always given in the same order as the rows.)

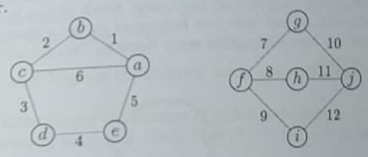
A strict definition of two graphs being isomorphic is that there is a bijection ϕ (see section 2.6.2 on page 28) between the nodes in the two graphs such that the elements in position (i, j) and $(\phi(i), \phi(j))$ respectively in the adjacency matrices are always alike. Such a bijection is called an **isomorphism**. If two nodes are adjacent in one of the graphs the corresponding nodes (according to the isomorphism) are adjacent in the other graph, and vice versa. In the example on the previous page $\phi(1) = \text{C}$, $\phi(2) = \text{A}$, $\phi(3) = \text{D}$, and $\phi(4) = \text{B}$ is an isomorphism.

Exercise 6.47 Find an alternative isomorphism in the example.

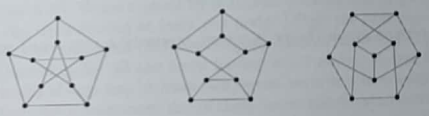
Exercise 6.48 Describe the adjacency matrices for the following graphs and find an isomorphism between them.



Exercise 6.49 Explain why the graphs below can't be isomorphic to each other.

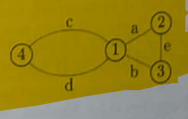


Exercise 6.50 Two of the graphs below are isomorphic. Which two?



However sparse the graph may be (that is however few the edges are) the adjacency matrix always has the same size for a certain number of nodes. If the number of nodes is n then n^2 elements are required. A graph can be represented in a more dynamic way that is more efficient when the graph is sparse. An **incidency matrix** for a graph is a matrix with one row for each node and one column for each edge.

	a	b	c	d	e
1	1	1	1	1	0
2	1	0	0	0	1
3	0	1	0	0	1
4	0	0	1	1	0



The incidency matrix above describes the multigraph on the right.

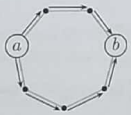
Exercise 6.51 Write down the incidency matrices of the graphs in exercise 6.49.

Exercise 6.52 How many edges can a graph at most have if the incidency matrix is to be smaller than the adjacency matrix?

Exercise 6.53 How can the incidency matrices be used to define isomorphism?

6.4 Trees

When a local area network for computer communication is to be built, a so-called "topology" has to be chosen for the network – should there be one cable between each pair of servers or shall we form the network into a ring or a star? If we use a graph to model the network, of course we want this graph to be connected, but there can be a point in not having alternative ways for the communication so that data won't arrive in the wrong order because information sent later has happened to use a faster route. We note that if there are two alternative ways between two nodes in a graph there must exist a cycle.



By a **tree** in graph theory is meant precisely a connected graph without cycles.

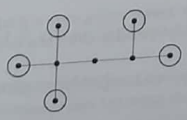
Exercise 6.54 Why is tree a suitable name for this concept? *

Exercise 6.55 Suggest some useful areas for modelling using trees. *

Exercise 6.56 Explain why a cycle will always be formed if an edge is added between two nodes in a tree.

Exercise 6.57 Explain why a disconnected graph will always be formed if an edge is removed from a tree.

At the outer end of the branches in a tree (with at least two nodes) we find **leaves**, by which is meant the nodes that have only one edge attached.



Exercise 6.58 Why does every tree with at least two nodes have to have at least two leaves?

In a tree having at least one node there is always a simple relationship between the number of nodes and the number of edges:

Theorem 6.3: ***Nodes and Edges in a Tree*** For all trees having at least one node the relationship $|V| = |E| + 1$, where V denotes the set of nodes and E denotes the set of edges, holds. ■

Exercise 6.59 Express the theorem using ordinary language.

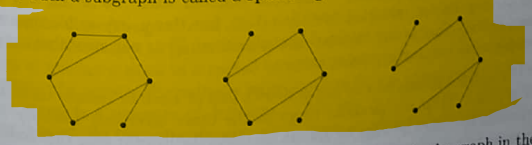
Exercise 6.60 Count the number of nodes and edges in the examples on the preceding page and check that $|V| = |E| + 1$ in each of the examples. *

Exercise 6.61 Write a proof by induction showing that $|V| = |E| + 1$ for all the trees having at least one node. (Hint: the base case is the tree with only one node (and zero edges). Assume that we have proved the relationship for all the trees having $n - 1$ nodes. A tree having $n > 1$ nodes always has at least two leaves, according to exercise 6.58. What happens if we remove one of these leaves and the attached edge?)

Exercise 6.62 Explain why there can't be any other kinds of connected graphs where the relationship $|V| = |E| + 1$ holds. *

6.4.1 Spanning Trees

A natural way of solving the last exercise in the previous section is to start by establishing that in each connected graph G that isn't a tree it's possible to find a subgraph that is a tree consisting of all the nodes in G but only some of the edges. As long as the graph has some cycle it's possible to remove one of the edges from the cycle; the graph remains connected. At last we get a connected graph without cycles, that is a tree, that reaches all the nodes in G. Such a subgraph is called a **spanning tree** for the graph G.



Exercise 6.63 Try to find all the spanning trees in the graph in the figure above.

Exercise 6.64 Find all the spanning trees in the complete graphs K_2 (one), K_3 (three) och K_4 (sixteen). *

Exercise 6.65 How many spanning trees are there in a cycle with n nodes?

Assume that a graph models the possible links in a communication network. Every link has a certain cost of usage (which can be affected by many different things, such as length, quality, capacity, or demand). If the costs are given, of course we want to use the spanning tree that has the lowest total cost. Such a tree is called a **minimal spanning tree**. At every moment minimal spanning trees are being calculated in computer networks all over the world!

We can model the costs of the links by marking each edge in the graph with a positive number, often called the **weight** of the edge. How can we find the spanning tree with the minimal total weight? An obvious idea is to try to do as before, that is to remove an edge (from a cycle) at a time – and in each step choose to remove the heaviest of the available edges.

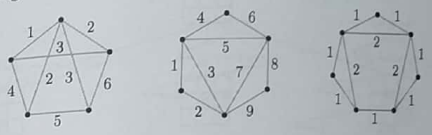
But how do we know for sure that the tree we have found using this algorithm really is the minimal spanning tree? Otherwise it would have been possible to get the minimal spanning tree $T_{\min}$ by letting the heaviest edge available, e_{heavy}, in some step remain and from this point onwards only removing edges lighter than e_{heavy}. If we remove e_{heavy} from $T_{\min}$ the graph separates into two components. Some other edge in the cycle to which e_{heavy} belonged could clearly be used to join the components to a tree again, and that edge must be lighter than e_{heavy}, since it has been removed later. So in this way we get an even lighter spanning tree than $T_{\min}$, but this is impossible since we have defined $T_{\min}$ as the minimal spanning tree! Thus our algorithm has to generate the minimal spanning tree.

If the graph contains lots of edges there's lots of work to remove them one at a time. Then it's possible to go in the other direction: assemble the spanning tree edge by edge by always choosing the lightest edge that doesn't form a cycle combined with the edges already chosen. This method is called **Kruskal's algorithm** for finding minimal spanning trees.

Exercise 6.66 Modify the argument above to prove that Kruskal's algorithm always gives the minimal spanning tree.

Exercise 6.67 How many edges does a graph with n nodes have to have for it to pay to use Kruskal's algorithm instead of edge-removal?

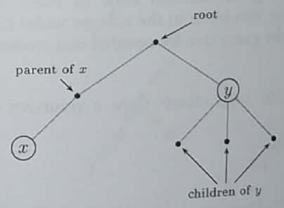
Exercise 6.68 Find minimal spanning trees for the following graphs by using either of the algorithms above:



Exercise 6.69 Five points in the plane are given, having the coordinates (0, 0), (3, 1), (1, 4), (2, 2), and (5, 4). Find the shortest network of ways between the points that tie them all together. (Hint: the length of a way between two points can be calculated using the Pythagorean theorem.)

6.5 Rooted Trees

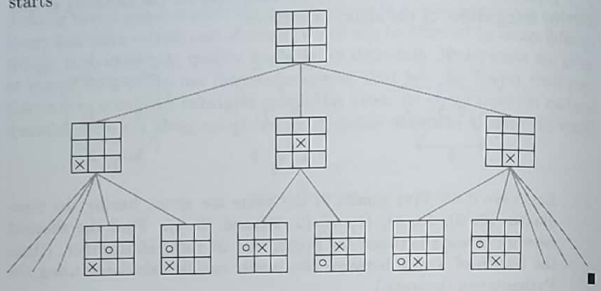
In a **rooted tree** one of the nodes has been appointed the **root** of the tree. The root is almost always drawn at the top! For each node in a rooted tree the adjacent node closer to the root is called its **parent** while the adjacent nodes further away from the root are called its **children**.



Example 6.2: ***Noughts and Crosses*** Here we give a foretaste of combinatorial game theory. A **game tree** is a model used for organising all possible ways of playing a game in a rooted tree. (Actually, it doesn't have to be a question of a game in the ordinary sense, but can be any situation whatsoever that demands a sequence of decisions. Then it's usually called a **decision tree** instead. We have already seen one of those, in example 5.4 on page 115.)

Let's for instance draw the top of the game tree for noughts and crosses. (For those unfamiliar with the game: you take turns drawing your symbol on the board. The first one to get three in a row wins.) In the root we have the starting position, in this case the empty 3×3-board. Let's say that the player using crosses starts. There are (symmetries discounted) three

different first moves: in the corner, in the middle, or on the edge. After that, the player using noughts has a number of different moves (five, two, and five, respectively, symmetries discounted) to choose from. The game tree thus starts



Exercise 6.70 Draw the complete second level below the root. Explain why there are five, two, and five, respecively, possible positions if symmetries are discounted. *

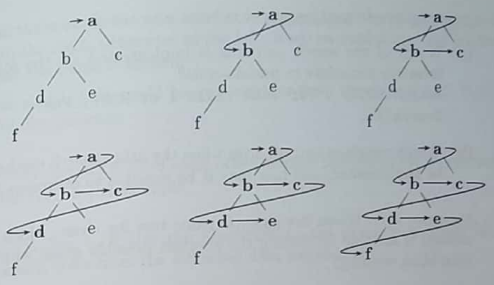
Exercise 6.71 Under each node in a rooted tree a rooted **subtree** (having the selected node as root) is hanging. Sketch the first two levels in the subtree under the position shown from the game tree for noughts and crosses. *

Exercise 6.72: *Important!* Give a recursive definition of rooted trees!

6.5.1 Breadth-first and Depth-first Search

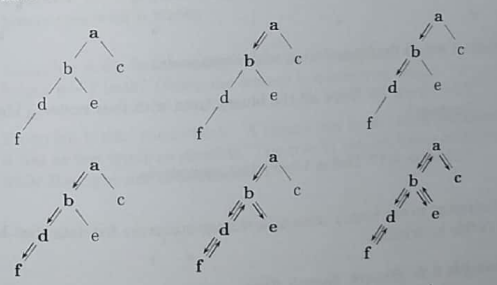
To determine in an intelligent way whether a move in a game is good or not, the game tree can be searched downwards to see what results can be reached. There are two common methods for searching a tree: breadth-first and depth-first.

In a **breadth-first search** one starts at the root of the tree (or subtree) that is to be searched. One takes all the nodes on level 1 under the root, that is, that are adjacent to the root, in turn. When there are no more such nodes one continues with the nodes on level 2, that is, those that are adjacent to the ones on level 1. Continue downwards level by level untill the whole tree has been searched (or the thing one was looking for has been found).



In the depicted tree we pass the nodes in the order a – b – c – d – e – f.

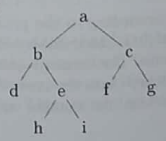
In a **depth-first search** one starts at the root and then goes to the first adjacent node on level 1. From there one continues to the first adjacent node on level 2, and so on downwards in the tree until a leaf is reached. Then one back-tracks upwards until one finds a way downwards that has not already been visited, and follows this path in the same way to the bottom, back-tracks upwards again, etc. One keeps on like this until all the nodes in the graph have been visited (or the thing one was looking for has been found).



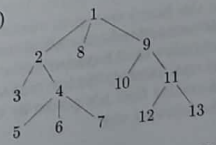
Now we pass the nodes in the order a – b – d – f – e – c instead.

Exercise 6.73 For the following trees, determine in which order the nodes are visited when searching breadth-first and depth-first, respectively.

(a)



(b)



Exercise 6.74: *Important!*

(a) Which of the search methods is implemented by the following recursive procedure in pseudo-code?
`Search(ROOT) : For each child X of ROOT, run Search(X).`

(b) Which complication will arise when the other search method is to be implemented, and how may it be possible to overcome it?

Exercise 6.75 Given the gigantic game tree for chess, which search method is suitable for finding the shortest possible game which ends with black winning?

6.5.2 Binary Trees

A **binary tree** is a rooted tree where each node has room for exactly two children, one to the left and one to the right.

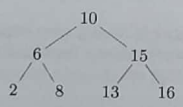
Above we see all the binary trees with three nodes.

Exercise 6.76 Draw all the binary trees with four nodes. (There are fourteen.) *

Exercise 6.77 Define binary tree recursively.

It's common to use binary trees as a storage structure for data that have to be swiftly searchable.

Example 6.3: ***Binary Search Tree***



In this binary tree seven numbers are stored according to the principle that all the numbers in the left subtree are smaller than the numbers in the right subtree, with the number in the root in between. The same property applies to each subtree. Such a tree is called a **binary search tree**. It can be assembled from a sequence by expanding the tree one number at a time in the following way:

Start at the root. If the new number is greater than the number in the root we go to the right, otherwise to the left. Keep on until a new leaf is created and store the number therein.

The tree shown here could be put together from the sequence 10, 6, 8, 15, 13, 2, 16, for instance. ■

Exercise 6.78 Find some other number sequence that also generates the tree in the example.

Exercise 6.79 Use the algorithm in the example to build binary search trees from the following data sequences.

(a) 101, 27, 17, 203, 156, 52, 240 **(b)** 240, 101, 27, 17, 203, 156, 52

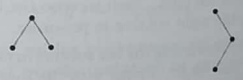
(c) 17, 27, 52, 101, 156, 203, 240

By the **height of a rooted tree** is meant the number of levels under the root.

Exercise 6.80 Explain why $n-1$ is the largest possible height of a binary tree with n nodes.

Exercise 6.81 In how many ways can a binary tree with n nodes and height $n-1$ look? (Hint: the answer is a power of two.)

Exercise 6.82: *Important!* A binary tree is said to be **balanced** if it has as few levels as possible. The tree to the left below is balanced while the right one is unbalanced.

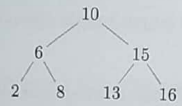


Of course searching is faster in balanced search trees. Is it a good thing or a bad thing if the sequence is sorted from the start when a search tree is built according to the algorithm?

6.5.3 In-order, Pre-order and Post-order

To print the data that have been stored in a binary search tree according to the method described, the nodes have to be **traversed** in-so called **in-order**. This order is specified recursively: start by traversing the left subtree in in-order. Then print the root. Finish by traversing the right subtree in in-order.

Example 6.4: ***In-order*** We use the same example of a binary tree as before:



We start with the left subtree:

In this subtree we first go down to the left, then the root and at last to the right: 2, 6, 8. Then we go up to the root of the tree itself: 10. After that the right subtree, which in the same way gives: 13, 15, 16. Thus in-order gives: 2, 6, 8, 10, 13, 15, 16. ∎

Exercise 6.83 Print, in in-order, data from the three trees you built in exercise 6.79 on the previous page.

Exercise 6.84 Implement in-order using pseudo-code.

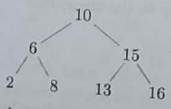
Exercise 6.85 Explain using induction why a binary tree built according to the method shown will always be delivered in increasing order when printed in in-order.

There are two more common ways of traversing (that is, go through the nodes in) a binary tree. Both ways are defined recursively, and the difference from in-order is simply when the root is handled compared to the two subtrees.

Pre-order: Start with the root. Then traverse the left subtree in pre-order. Finish by traversing the right subtree in pre-order.

Post-order: Start by traversing the left subtree in post-order. Then traverse the right subtree in post-order. Finish with the root.

Example 6.5: ***Pre-order and Post-order*** The same example again:

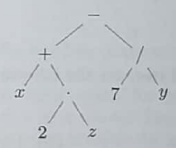


In pre-order we start with the root: 10. Then we take the left subtree:

There we take the root first, and then the child on the left and on the right: 6, 2, 8. Lastly comes the root of the right subtree followed by the child on the left and on the right: 15, 13, 16. Pre-order thus gives: 10, 6, 2, 8, 15, 13, 16.

In post-order we start instead with the left subtree. There we go first down to the left, then to the right and lastly the root: 2, 8, 6. Then in the same way we take the right subtree in post-order: 13, 16, 15. Finally we take the root of the whole tree: 10. Post-order thus gives: 2, 8, 6, 13, 16, 15, 10. ∎

Typical applications of pre-order and post-order are the handling of **binary expression trees**. That is trees that store expressions that are built from **binary operators** (such as plus and times), for instance $(x+(2\cdot z))-(7/y)$ which gives the tree:



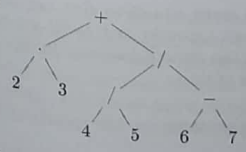
As the root of the tree we have the main operation in the calculation, that is to say: that which is done last, in this case the subtraction. The subtrees contain the calculations of the numbers that the main operation is going to operate on, and these trees are built according to the same principle. Each expression inside parentheses thus gives a subtree in the expression tree. Thereby the parentheses don't have to be stored! The order in which the expression is to be calculated is given by the structure of the tree. Expression trees are used in compilers and interpreters.

In pre-order and post-order the binary expression tree above is printed as $-\ +\ x\ \cdot\ 2\ z\ /\ 7\ y$ and $x\ 2\ z\ \cdot\ +\ 7\ y\ /\ -$ respectively.

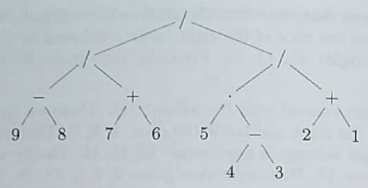
Remark These ways of writing mathematics are called **prefix notation** or **polish notation** and **postfix notation** or **reverse polish notation** respectively. "Normal" notation is also called **infix notation**. The programming language Lisp uses prefix notation and the graphics language PostScript uses postfix notation.

Exercise 6.86 Print the following three binary expression trees in both pre-order and post-order, and in the "normal" way as well.

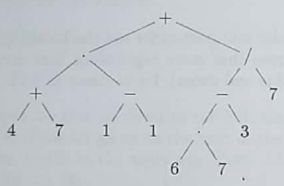
(a)



(b)



(c)



Exercise 6.87 Build trees for the following two expressions given in post-order:

(a) 6 2 3 · − 5 1 − + **(b)** 4 3 + 2 9 − + 600 6 / ·

(c) Calculate the value of the two expressions above!

Exercise 6.88: *Important!* *Plus* is an example of a **binary operation**, that is an operation which acts on two numbers (such as $3+5$). Binary operations are very common, but there are operations that only act on a single number as well, for instance the *square root of*: $\sqrt{3}$. Such operations are called **unary**.

(a) Which common symbol denotes both a binary operation and a unary operation?

(b) How is it possible, using binary trees model, to expression containing unary operations as well?

6.6 Three Examples Showing Modelling Using Graphs

As we have already seen there are many situations that can obviously be modelled using graphs, such as road networks and social networks. But graphs can be useful models even in less obvious contexts. In this concluding section we will show three examples of this:

- *Scheduling*, for instance building projects.

- *Line breaking* in word processors.
- Solving *Instant Insanity*, a well-known puzzle.

At first glance, these problems don't appear to have much to do with graphs, but when analysed more in depth can be handled using graph-theoretical methods.

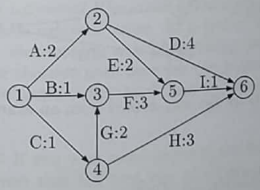
Remember that the key point when something is to be modelled using graphs is to choose what is to be represented by *nodes* and what is to be represented by *edges*.

6.6.1 Scheduling

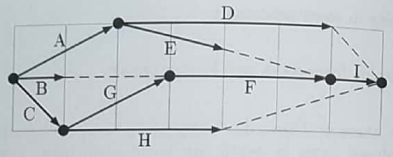
Making working schedules for building projects has always been problematic. A building project usually consists of a number of different tasks. It's possible to make a fair estimate of the time needed for each task, but this doesn't tell us anything about the project as a whole! The problem is that some tasks can be carried out in parallel (it's possible for instance to paint the ceilings at the same time as the facade is plastered) while others have to be done in a certain order (it's not possible to paint the ceiling before building it).

This problem is very suitable to model with the help of graphs. The tasks that are extended in time will be the edges, and the points in time when they are started or finished will be the nodes.

Here is a graph showing a small project. The labels on the edges are the name of the task and the number of weeks that it's estimated that the task will take.



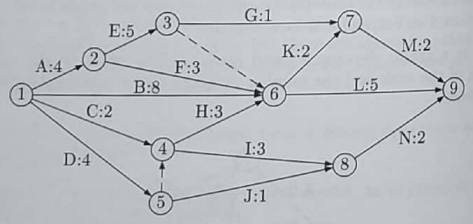
Now, if we want to know how much time the project will take, we can draw the whole thing on graph paper, with a time scale horisontally. One square can represent one week. Some arrows will then be stretched over a longer period of time than the task demands. That we indicate by denoting the task itself using a fat arrow and waiting time, if any, with a dashed line at the end. This project will get the following time graph:



From the graph we are able for instance to see that the whole project ought to take 7 weeks; it's the tasks C, G, F, and I – that have to be done in exactly that order – that determine the time, while delays in the other tasks probably don't matter. (The tasks C, G, F, and I are said to be *time critical*, which means that a delay in them delays the whole building project.)

The method for scheduling described here is said to have been invented when a way to computerise the calculations was sought, and proved to be good enough (at least in smaller projects) to make it unnecessary to involve computers in the calculations!

Exercise 6.89 Draw a scheduling graph for the building project below. (The dashed arrow between time 3 and time 6 indicates that time 6 mustn't take place before time 3, since the tasks that are to start at time 6 need task E to be finished. The same reasoning applies to the dashed arrow between time 5 and time 4.) State as well how long the project will take and which the time-critical tasks are. (The times indicated are in days.)



(This exercise is by the way copied from an exam in building production.)

6.6.2 Line Breaking in TEX

A paragraph in a text is in reality a long, continuous unit that has been subdivided into lines on the paper, since otherwise book pages several metres wide would be needed. In most books the lines have afterwards been manipulated so that all of them have the same length. Previously this has been something that the typesetters at the printers have been doing using their own judgement, but when the computerised typesetting systems arrived it

became necessary to automate the process. There are several ways of doing this. The one described here was invented by the American professor Donald E. Knuth, who got mad when he saw how ugly his latest book was. (By the it's way his program, TEX, that we have used for this book.)

When a line in the text is to be adjusted to the right length, the spaces between the words have to be used. They can if nessecary be squeezed together a bit, or be stretched. It's not possible to squeeze them too much, but there are no limits on how much they can be stretched, apart from the fact that it looks pretty stupid. Furthermore, it's possible to break words that only in parts can be fitted on the line, by hyphenating them. That lessens the readability a bit, but can be better than word spaces several centimetres long.

When we are to break our paragraph into lines we may consider several alternative ways of doing it. In this paragraph we might end the first line after "several" or after "alternative" or perhaps hyphenate "alternative". Then we make the same considerations for the next line. The possible break points of that line partly depend on where we broke the first line; if we had broken the first line of this paragraph after "several" it would not be possible to break the second one after "after", while it is possible when we have broken the first line by hyphenating "alternative".

All the possible ways of breaking the paragraph can be modelled using a directed graph. As nodes we have the possible break points. Two points are connected with an edge if it is possible to print a line that starts at the first point and ends at the second one. A breaking of the paragraph corresponds to a path from the node that represents the start of the paragraph to the node that represents its end.

What we want to do now is to choose the path that corresponds to the most beautiful (or least ugly) alternative. We do this by assigning every possible line an ugliness value. A line with spaces that have exacly the right length and no hyphenation has the ugliness value of 0. The more the spaces deviate from the ideal, the higher will the ugliness value of the line be. A hyphenation will also increase the ugliness. The most beautiful alternative is then the path with the least total ugliness value. In practice, we have rewritten the problem of finding a nice-looking way of breaking the paragraph into the problem of finding the shortest path between two nodes. (How *that* problem is solved is outside the scope of this book, but is covered in part 2.)

Exercise 6.90 If you scan some pages of the book you will find that it's quite common that the first line of a paragraph ends with a hyphenated word. What causes this?

Exercise 6.91 Most word processors and typesetting systems don't use these sophisticated methods. Examine some program to which you have access, and try to analyse how that program decides where the line breaks are to be placed.

6.6.3 Instant Insanity

Instant Insanity is a puzzle consisting of four cubes. Each of the six sides on each cube is painted in one out of four colours (let's say blue–B, black–K, gray–G, or white–W). The goal is to build a tower using the blocks, so that all the colours are present on all the sides of the tower.

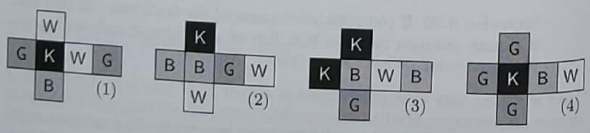
There are 41,472 different towers that can be built using four different blocks (see exercise 6.95 on page 170). So it's not practicable to simply build all the possible towers, and check whether one of them fulfills the specifikation.

If we start by trying to make the front of the tower nice we find that we design the back at the same time , since the opposite sides of the blocks will be visible there. Because of this, it seems wise to try in some way *at the same time* to place the colours on the front and on the back. For this we need to document which colours are painted on the opposite sides on the blocks. We can do this by drawing a graph (a multigraph) with one node representing each colour, and an edge between two colours if they are found on opposite sides of a block. In addition we note on the edge which block it describes. (See the picture in the example if this explanation was hard to understand.)

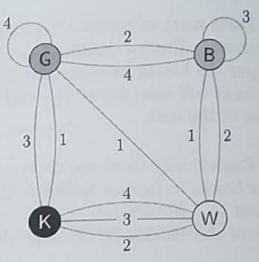
Now we want to place the blocks so that all the colours are represented on both the front and the back of the tower. If we select one of the edges belonging to block 1, this represents that block 1 is placed so that the colour at one end of the edge is on the front of the tower and the colour at the other end on the back. If we pick 4 edges, one for each block, and do it in such a way that each colour is picked twice, we have a description of how the blocks can be placed so that each colour is picked twice; once for the front and once for the back.

If we are then able to pick another 4 connections according to the same principle, we have additional instructions for how to fix it so that the left side and right side of the towers are multicoloured as well. And if we have fixed that the tower is finished!

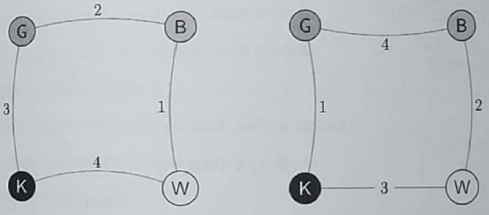
Example 6.6: ***Instant insanity*** We have the blocks below. (You have to imagine that you cut them out and fold them.)



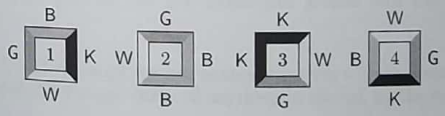
We draw the graph showing how the colours on opposite sides are placed in relation to each other;



Now we try to pick four edges from the graph, one for each block, so that each colour is connected to two of the edges. Then we have fixed the front and back. Then we try to find four different edges according to the same principle, to arrange the left and right sides. It's possible, according to



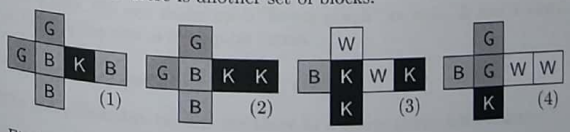
We can build a tower where the four blocks are placed like this:



■

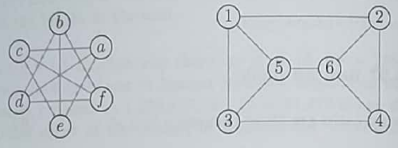
Exercise 6.92 Verify that the large graph really corresponds to the given blocks, that the suggested tower really corresponds to the subgraphs, that the tower is possible to build using the given blocks, and that it satisfies the requirements. *

Exercise 6.93 Here is another set of blocks.



Find a solution using these.

Exercise 6.103 Determine whether the following two graphs are isomorphic. Write down a bijection between the nodes if they are isomorphic; otherwise explain why they can't be.



Exercise 6.104 Determine whether the following two graphs are isomorphic. Write down a bijection between the nodes if they are isomorphic; otherwise explain why they can't be.

Exercise 6.105

(a) Write down the adjacency matrix of the graph on the left in the previous exercise.

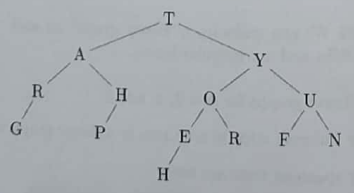
(b) Write down the adjacency matix of the complement of the graph.

(c) In general: how do you find the adjacency matrix of the complement of a graph, the adjacency matrix of the graph being given?

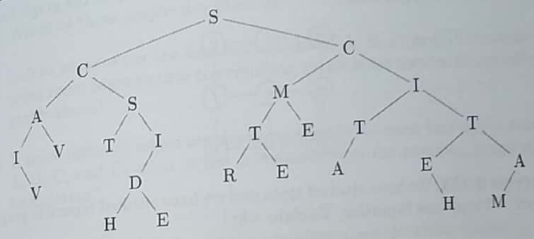
(d) What will the solution of this problem look like if you have the incidence matrix instead?

Trees

Exercise 6.106 Traverse the tree below breadth-first, depth-first, and in in-order, pre-order, and post-order.



Exercise 6.107 Traverse the tree below breadth-first, depth-first, and in in-order, pre-order, and post-order.



Exercise 6.108 Make up a tree of the type above with the property that we get two different correctly spelled English words when the tree is traversed in post-order and pre-order. *

Exercise 6.109 Make up a tree of the type above with the property that we get two different correctly spelled English words when the tree is traversed breadth-first and depth-first. *

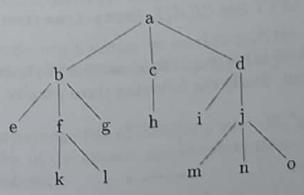
Exercise 6.110 The file system in a computer can (on the whole) be seen as a tree.

(a) What do the inner nodes consist of, and what are the leaves?

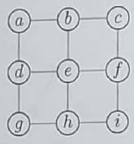
(b) Why do we write "on the whole"? Is there something that doesn't quite fit?

(c) If you are looking for a certain file, which search method is used by the computer?

Exercise 6.111 We have defined the concepts *in-order*, *pre-order*, and *post-order* for binary trees. A **ternary tree** is a rooted tree where every node has at most *three* children. Do the concepts work here as well? If they work, then traverse the tree below in the different orders!



Exercise 6.112 We have only been using breadth-first and depth-first search on trees. Is there anything that stops us from using the search methods on other kinds of graphs? If it's possible, then traverse the graph below breadth-first and depth-first!



Exercise 6.113 We have studied trees and we have studied bipartite graphs. In fact, all trees are bipartite. Explain why! *

6.7.2 To Ponder About

Exercise 6.114 In the Andes are found, isolated from the surrounding world, ten towns connected by a network of roads where it's possible to get between every pair of towns. From each town, *three* roads exit to other towns.

A certain kind of *donkey* refuses to visit a town more than once. A certain kind of *mule* is nicer – it's prepared to visit the same town several times but refuses to walk along the same road more than once.

(a) Is it certain that there is a route so that the *mule* can walk along all the *roads*? Or is it certain that there isn't? Explain!

(b) Is it certain that there is a route so that the *donkey* can visit all the towns? Or is it certain that there isn't? Explain!

Exercise 6.115 Implement the construction of binary search trees from a sequence of data. Your task is to write the function
`Add-to-tree(X, ROOT)`.
`ROOT` referes to the root of the tree. `ROOT.value`, `ROOT.left`, `ROOT.right` give the value in the root and the left and right subtrees. You may use the function
`Build-tree(LEFT, ROOTVALUE, RIGHT)`
that creates a tree where the root has the value ROOTVALUE and with left and right subtrees LEFT and RIGHT. `Empty-tree` creates an empty tree.

Exercise 6.116 We'll call a graph *Eulerian* if it has either an Eulerian trail or an Eulerian circiut. Study the following three graphs:

(a) Explain why every graph with three edges and four nodes has to be isomorphic to one of the graphs above!

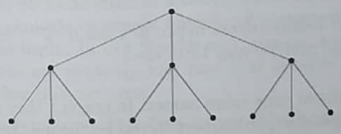
(b) Which of these graphs are Eulerian?

(c) In how many ways can four nodes marked A, B, C, and D be connected using three edges so that the resulting graph is isomorphic to the leftmost graph above?

(d) If three different edges are drawn randomly between four nodes marked A, B, C, and D, what is the probability that the resulting graph won't be Eulerian?

Exercise 6.117 Let $\lceil x \rceil$ denote the rounding upwards of the number x. Explain why $\lceil \log_2(n+1) \rceil - 1$ is the lowest possible height of a binary tree having n nodes.

Exercise 6.118: ***Ternary Trees*** By **complete ternary tree** is meant a rooted tree where every node either is a leaf (that is, has no children) or is an inner node having exactly three children. In the figure a complete ternary tree having 4 inner nodes and 9 leaves is shown.



Prove by induction that the number of leaves in a complete ternary tree with n inner nodes will be $2n+1$ if $n \geq 1$.

Exercise 6.119: ***A Table Game*** Ten squares where the edges are marked with digits (0 to 9) are given. The number sequence 7352 means that the numbers on the edges, read clockwise, are 7, 3, 5, and 2. There is a notch on each square to show which one of the edges is the first one in the sequence of digits. The squares are always placed with the same side upwards, but can be turned in four ways, so that the edge with the notch is placed upwards, downwards, to the right, or to the left.

(a) In how many ways can a square be marked if all the edges have to have different numbers? The same question if the edges may have the same number.

(b) Now we are to arrange the squares in a row. We are allowed both to change the order of the squares and to rotate them. In how many ways can the row be built?

(c) The goal is to arrange the squares in a row so that all the ten digits are present on both the long sides of the row. The sides of the squares have the numbers 0062, 4101, 6207, 3302, 3444, 9253, 6866, 7636, 2486, 4599. Solve the problem by modelling the squares with a graph with ten nodes and twenty edges, where there is an edge marked A from digit x to digit y if square A has the digits x and y on opposite edges. Search the graph for ten edges marked differently so that each node touches two of these edges.

Exercise 6.120: ***Hypercubes*** Here we define a type of graph, that we call a **hypercube**:

- A zero-dimensional hypercube consists of a node and nothing more.
- You'll get an $n+1$-dimensional hypercube by drawing two copies of an n-dimensional hypercube, and then connecting corresponding nodes in the two graphs to each other. (Node 1 in one graph is to be connected to node 1 in the other graph, and so on.)

(a) Draw the zero-dimensional, the one-dimensional, the two-dimensional, and the three-dimensional hypercube. (If it's done correctly the three-dimensional one will be the figure we in normal speech call a *cube*.)

(b) Prove that an n-dimensional hypercube has 2^n nodes.

(c) Prove that an n-dimensional hypercube has $n \cdot 2^{n-1}$ edges. (The result in the previous exercise may be of use.)

Now, give the nodes in your cubes names. In graph 1 they are to be called 0 and 1. In each expansion you connect the nodes having the same names and then append a 0 to the names in one subgraph and a 1 in the other.

(d) If you now draw a Hamiltonian cycle in the graph the names of the nodes will form a Gray code (se example 4.7 on page 78). Explain why!

7 Logic and Boolean Algebra

An engineer, a physicist, and a mathematician were sitting on a train passing through Scotland. The engineer looked out through the window and saw a black sheep. "Look, there are black sheep in Scotland", he said. The physicist thought for a while and then said: "There is *at least one* black sheep in Scotland." The mathematician thought for a while and then said: "In Scotland there is a sheep which is black on one side."

In their professional role mathematicians speak something that on the surface is the same language as other people's, but under closer analysis it can be found that they use some words and phrases in a different way. Mathematical statements have to be formulated in a totally exact way and the arguments watertight. The analysis of exacly formulated statements and watertight arguments is called **logic**. In this chapter we'll analyse, among other things, the way statements are made up and how it is possible to proceed when determining whether they are true.

Logic is at the same time one of the foundations of both computer science and digital electronics, and we'll therefore mention a little about the connection to programming and hardware construction.

Highlights from this chapter.

- The fundamental logical connecting words *and*, *or*, and *not* and different ways of expressing them:

	and	or	not
English:	AND	OR	NOT
C-program:	&&	||	!
logical symbol:	$\wedge$	$\vee$	$\neg$
Boolean symbol:	$\cdot$	$+$	$^-$
set symbol:	$\cap$	$\cup$	^{c}
gate:			

- *Implication*, $A \to B$: if the statement A holds then the statement B holds as well.
- *Equivalence*, $A \leftrightarrow B$: the statement A holds if and only if the statement B holds.
- Usage of *truth-tables* for logical expressions.
- The rules of propositonal logic.
- Calculations in *Boolean algebra* using *Boolean functions*.
- Different ways of writing Boolean functions, such as *disjunctive normal form* and *conjunctive normal form*.
- Implementation of Boolean functions using so-called *gate networks*.
- Statements $P(x)$ about general objects x, which are called *predicates*. For instance: "x is a dog".
- The so-called *quantifiers* $\forall x P(x)$: for all x $P(x)$ holds, and $\exists x P(x)$: there exists some x such that $P(x)$ holds.

7.1 Reflections on the Language and Meaning of Mathematics

If a normal person says "a sheep which is black on one side" you take it for granted that the sentence has an unsaid continuation: "... and the other side has a different colour". The mathematician simply means: "... and the other side I haven't seen, so I don't want to say anything about it. It may be black, it may not be". Because of this, you should never assume that mathematical sentences include anything unsaid. They are to be interpreted exactly as given, and not in any other way. (On the other hand, you can usually assume that everything said is *relevant*, since mathematicians are of the opinion that the more brief you can be, the better.)

Furthermore, a normal person saying *there is a...* means "one" and nothing else. A mathematician always means *at least one*. If the mathematician actually means *one* the said person says *there is one and only one...*.

If you take a closer look on how mathematicians express themselves, you find that the way the sentences are formulated is directly connected to the way proofs are organised. It's for instance usually fairly easy to prove that there is *at least* one object that satisfies some condition. (You can hunt one down and point at it. That proves that it exists, but not that there aren't others as well.) Proving that there is *exactly* one object satisfying the condition takes a second subproof that shows that there isn't more than one such object. It seems reasonable that the statement that is easier to explain should be easier

to express as well.

Why are mathematicians so hot on proving things, then? Well, why do you want to calculate at all? Some of us calculate because it's fun, but most people make calculations because they want to find out something, whether it is how many rolls of wallpaper they need for the living room or how a space shuttle is to be dimensioned. What you then demand is that the calculation methods you have been taught actually work, that they are guaranteed to give the right answer to the question. (In addition, the question has to be correctly put. "Garbage in, garbage out", is a saying among programmers.)

The proofs, they are what guarantees that the calculation methods work. Textbooks in mathematics contain proofs so that the reader may convince herself that the book is telling the truth.

7.2 Propositional Logic

In **propositional logic** one works, as hinted by the name, with **propositions**. By propositions is in logic meant simple or complex statements that can be either true or false.

Example 7.1: ***Some Simple Propositions***

- "Stockholm is the capital of Sweden" (true)
- "4711 is a prime number" (false)
- "The battle of Lützen took place on 16 November" (true according to the German calender, false according to the Swedish one)
- "The square of the hypotenuse is equal to the sum of the squares of the two other sides" (true) ■

That statements are either true or false is admittedly a rather simplified model of reality, where things can be more or less true, or true from one point of view and false from another, or simply undefined. (What is the truth-value of the statement "pffrrrth!"?)

There are a number of competing ways of denoting true and false. Here are some variants:

true:	T	t	$\top$	1
false:	F	f	$\bot$	0

In this text we'll use 0 and 1, since they are easy to write and easy to tell apart. If you prefer som other variant, use that one instead. Propositions, both simple and more complicated, we'll denote by p (as in *proposition*) and the letters after that in the alphabet.

7.2.1 Putting Propositions Together

Most interesting propositions consist, when investigated more closely, of several subpropositions that are connected in some way or another. Propositions that can't be further subdivided are sometimes called **atomic**. The connecting words are called **connectives**.

Example 7.2: ***A Composite Proposition*** The proposition "if you are under the age of 26 or a student, then you can buy a reduced-price ticket" firstly consists of the parts "you are under the age of 26 or a student" and "you can buy a reduced-price ticket". These parts are connected by "if... then...". The first part is made up from the parts "you are under the age of 26" and "you are a student" and these sub-parts are connected using the word "or". The proposition thus consists of three atomic propositions and two connectives. ∎

Exercise 7.1 Take these propositions apart into their components. How many atomic propositions and how many connectives are there?

(a) "If there are two or more ways to do something, and one of those ways can result in a catastrophe, then someone will do it."

(Stated by an engineer named Murphy, when somebody had placed all the sensors that were to be used in an experiment backwards and glued them into place.)

(b) "A graph has an Eulerian cicuit if and only if the graph is connected and all the nodes have an even degree."

(Quoted from this book.)

The **truth-value** of a composite proposition is determined by the truth-values of the parts it consists of, that is by whether they are true or false, and also by the way they are connected. When one wants to analyse a composite proposition it's therefore possible to set up a **truth-table**, where the atomic propositions are assigned all possible combinations of values, after which one checks what consequences this will have for the whole. Since each atomic proposition has two possible values (true or false) the number of combinations is $2^{\text{number of atoms}}$. A number of truth-tables are included in the next section, which, perhaps better than this explanation, will show what is meant.

7.2.2 Connectives

Here the definitions of the most common connectives will follow. The presentation is fairly detailed; the intention is that it shouldn't be necessary to read it more than once or twice.

Definition 7.1: ***Negation*** "It's not true that I blew the exam!"

This proposition seems to consist of the atom "I blew the exam" with the addition "it's not true that". Adding "it's not true that" to a proposition is called to **negate** it or to **invert** it, and the operation is called **negation**. Often the shorter name *not* is used.

The negation of the proposition p will in this book be denoted $\neg p$. There are a number of alternative ways of writing, such as $\sim p$, p', and $\overline{p}$.

The truth-table for $\neg p$ looks like this:

p	$\neg p$
0	1
1	0

When p is false $\neg p$ is true, and vice versa. ∎

Exercise 7.2 Find one everyday and one mathematical statement that are true negations.

Definition 7.2: ***Conjunction*** "I am female and I like logic".

This proposition consists of the two atomic propositions "I am female" and "I like logic", connected by the word "and". A connection made using an "and" is called a **conjunction**. The conjunction of two propositions is counted as true only if both the propositions are true; in all other cases it is classified as false.

Conjunction is usually denoted $\wedge$, but &, multiplication signs, and juxaposition are used as well.

The truth-table for conjunction looks like this:

p	q	$p \wedge q$
0	0	0
0	1	0
1	0	0
1	1	1

Since there are two subpropositions there are $2^2 = 4$ rows in the table. Note how the different combinations of truth-values are ordered. If they are interpreted as numbers given in binary they represent the numbers 0–3, in order of size. You *may* write the values in any order you like, but this one has several advantages, since it's very systematical. The risk of missing some combination isn't high, which in a problem with lots of variables is otherwise easily done. ∎

Exercise 7.3 Find one everyday and one mathematical statement that are true conjunctions.

Exercise 7.4 Find some other systematical way of ordering the combinations in the table.

Definition 7.3 "I'm slow to understand or hard of hearing!"

This proposition consists of the parts "I'm slow to understand" and "I'm hard of hearing", connected by the word "or". This is called a **disjunction**.

The word "or" in logic always means "and/or". A disjunction is thus classified as true as soon as at least one subproposition is true. It is false only if all the parts are false.

Disjunction is usually denoted $\vee$, but $+$ is sometimes used. The truth-table looks like

p	q	$p \vee q$
0	0	0
0	1	1
1	0	1
1	1	1

∎

Exercise 7.5 Find a true everyday statement and a true mathematical statement that are both disjunctions.

Exercise 7.6 Are the expressions you have written "and/or-expressions" or "either/or-expressions", that is to say, can both halves of the expressions be true *at the same time* or not?

Exercise 7.7 In some context one really wants to use *either/or*, **exclusive or**, **XOR**. That is often denoted $\oplus$.

(a) Write the truth-table for XOR.

(b) Express XOR using the connectives already defined.

(c) Why do you think that *and/or* is the standard interpretation of the word *or*, and not *either/or*?

The connectives given here (negation, conjunction, and disjunction) tend to be defined in most programming languages, and are used when writing conditional clauses.

Exercise 7.8 How are these connectives denoted in your favourite language? *

Actually, it's possible to get by with these connectives, since all expressions in propositional logic can be reformulated in a way that is possible to express using them. But the formulations can be incredibly cumbersome! Because of this, there are a couple of connectives that are adapted to the way one usually wants to express oneself.

Definition 7.4: *Implication* "If it's Sunday, then the bank is closed".

This proposition consists of the parts "it's Sunday" and "the bank is closed", connected by "if... then...". This is called an **implication**.

Implication is in propositional logic usually denoted by $\rightarrow$, but $\Rightarrow$ and $\supset$ are used as well.

The truth-table for implication looks like

p	q	$p \rightarrow q$
0	0	1
0	1	1
1	0	0
1	1	1

This means that the statement "If it's Sunday then the bank is closed" is false only if it is Sunday and the bank is open in spite of that. If it isn't Sunday the bank may be either open or closed without the statement being false, and because of this it counts as true.

It's possible, if so desired, to pronounce the implication as "it's Sunday only if the bank is closed", or as "the bank is closed if it's Sunday". If one wants to show off one's great vocabulary one might say "it being Sunday *implies* that the bank is closed"; the English version of that formulation is "it being Sunday *leads to* the bank being closed".

It may be noted that almost all theorems in mathematics are implications of some kind. Some examples:

- *If* a linear equation system has fewer equations than unknowns *then* it won't have a unique solution.
- *If* a function is differentiable *then* it is continuous.
- *If* set A is a subset of set B and set B is a subset of set A *then* the sets are identical.

Most people have at some point participated in a conversation like this: "What day is it?" "Well, the bank isn't closed, so at least it can't be Sunday."

It's apparently possible to rephrase "if it's Sunday then the bank is closed" as "if the bank isn't closed then it isn't Sunday". $p \rightarrow q$ and $\neg q \rightarrow \neg p$ mean the same thing.

On the other hand, one has to be careful so that one *doesn't* confuse $p \rightarrow q$ with the **reversal** $q \rightarrow p$. "If it's Sunday then the bank is closed" holds at least in Sweden, but "if the bank is closed then it's Sunday" doesn't hold at all. (It may be Saturday, a power failure, or midnight, for instance.)

Lastly, we may note that the truth-tables for implication and disjunction resemble each other closely; the only difference is where the zero is placed. On investigation one finds that the truth-table for $(\neg p) \vee q$ is identical to the one for $p \rightarrow q$. Because of this, these expressions can be exchanged freely for each other in analysis and calculations. ∎

Exercise 7.9 Find a true everyday statement and a true mathematical statement that are both implications.

Exercise 7.10 Does the reversal of your statements hold as well?

Exercise 7.11 What kind of reasoning is the basis of statements like "If you win the game then I want to eat my hat!"?

Definition 7.5: ***Equivalence*** "Saying that you didn't have time is the same thing as saying that you didn't think it was as important as the other things you had to do!"

After removing some extraneous words one finds that this sentence consists of the two statements "you didn't have time" and "you didn't think it was as important as the other things you had to do", connected by "is the same thing as".

"Is the same thing as" is called **equivalence**. ("equi" – alike; "valence" – value.) The expression **if and only if** is often used. The operation is denoted $\leftrightarrow$ (or sometimes $\Leftrightarrow$ or $\equiv$). The truth-table is

p	q	$p \leftrightarrow q$
0	0	1
0	1	0
1	0	0
1	1	1

We can see that the expression is true if p and q are both true or are both false. That means that $p \leftrightarrow q$ can be written as $(p \wedge q) \vee ((\neg p) \wedge (\neg q))$ as well.

The pronunciation *if and only if* hints that the expression can be written as $(p \rightarrow q) \wedge (q \rightarrow p)$ too. That reformulation is the one used in proofs. If you have to prove that two statements are equivalent you start by showing that the first one leads to the second, and then that the second one leads to the first.

In spoken language it can be difficult to tell implications and equivalences apart. If somebody says "if you keep on singing then I'll kill you" you assume that you'll escape being killed if you stop singing. Strictly speaking, the speaker never said that, but it has to be taken as a convention to assume that such an unsaid second part is included in the sentence. A mathematician, on the other hand, doesn't assume anything that isn't clearly stated, and thinks that the correct way of expressing the request is "I'll kill you if and only if you keep on singing". ∎

Exercise 7.12 Find a true everyday statement and a true mathematical statement that are both equivalences.

Exercise 7.13 Rephrase one of the statements you made up according to the two alternative ways of expressing an equivalence, $(p \rightarrow q) \wedge (q \rightarrow p)$ and $(p \wedge q) \vee ((\neg p) \wedge (\neg q))$.

Exercise 7.14 Write down the logical structure of the composite propositions in exercise 7.1 on page 180 by replacing the atomic propositions with letters and the connectives with symbols. (Don't forget parentheses!)

7.2.3 Rules of Syntax and Calculations in Propositional Logic

A common logical problem is determining whether a composite expression is true or not, or under which conditions it is true. The elegant way of doing this is to get to the bottom of the meaning of the expression, but if it's very complicated (let's say like an election promise in politics or a paragraph in fine print in an insurance contract) that can be inpracticable. Then it can be a good idea to set up a truth-table, so that it's possible to see if/when the expression is true.

Example 7.3: ***A Truth-table*** We study the expression $(p \vee (\neg q)) \rightarrow ((\neg r) \wedge q)$. It consists of three **propositional variables** and five connectives. That gives us eight columns in the table, one for each variable and one for each connective. That there are three variables also gives us the number of rows in the table: eight $(= 2^3)$. We set up the table:

p	q	r	$\neg q$	$p \vee (\neg q)$	$\neg r$	$(\neg r) \wedge q$	$(p \vee (\neg q)) \rightarrow ((\neg r) \wedge q)$
0	0	0	1	1	1	0	0
0	0	1	1	1	0	0	0
0	1	0	0	0	1	1	1
0	1	1	0	0	0	0	1
1	0	0	1	1	1	0	0
1	0	1	1	1	0	0	0
1	1	0	0	1	1	1	1
1	1	1	0	1	0	0	0

∎

Exercise 7.15 Look through at least two rows of the table and check that you understand all the subcalculations in them. *

In the same way as when using the four rules of arithmetic, the result when calculating the value of a composite logical expression depends on the order in which the operations are carried out. In the same way, the desired order can be indicated using parentheses. And here as well, the expressions may end up completely unreadable if you actually do this! Study for instance the expression

$$(\neg(((\neg p) \wedge q) \wedge r)) \rightarrow ((s \vee (\neg t)) \leftrightarrow \neg((\neg u) \vee ((\neg v) \vee x)))$$

Because of this, conventions for the order of precedence between the operations have been developed (just like the "times is done before plus"-rule in arithmetic).

1. First of all comes negation. $\neg p \wedge q$ means $(\neg p) \wedge q$.

2. After that come conjunction and disjunction, which have the same precedence. If you have an expression including several operations of this kind, the order has to be indicated using parentheses, since $p \wedge (q \vee r)$ and $(p \wedge q) \vee r$ don't mean the same thing at all.

3. Last of all come implication and equivalence. $p \wedge q \to r \vee s$ is to be interpreted as $(p \wedge q) \to (r \vee s)$.

Furthermore, you get exactly the same result when you calculate $(p \wedge q) \wedge r$ as when you calculate $p \wedge (q \wedge r)$. Because of this, expressions of this kind that are one long chain of conjunctions are usually rationalised and written just as $p \wedge q \wedge r$ without any parentheses. The same thing holds for disjunctions. But note that no such rule exists for *implications*! Because of this, writing $p \to q \to r$ is not allowed, since the meaning of the expression isn't unambiguous.

Using these rules, the unreadable expression can be written as

$$\neg(\neg p \wedge q \wedge r) \to \big(s \vee \neg t \leftrightarrow \neg(\neg u \vee \neg v \vee x)\big)$$

(To increase the readability even more, we have been manipulating the spaces between the symbols and the sizes of the parentheses as well. It's usually a good idea to do this, at least when writing by hand.)

Exercise 7.16 Check that it's true that $p \wedge (q \vee r)$ and $(p \wedge q) \vee r$ don't mean the same thing. *

Exercise 7.17 Write the truth-tables for the two possible interpretations of $p \to q \to r$ and compare them. *

Exercise 7.18 Does it seem necessary to include parentheses in the expression $p \leftrightarrow q \leftrightarrow r$?

Exercise 7.19 Remove the unnecessary parentheses in the expression $(p \vee (\neg q)) \to ((\neg r) \wedge q)$.

Exercise 7.20 Draw a "binary expression tree" (see page 163) for $p \wedge \neg q \to \neg(r \vee s)$.

Besides the rules of precedence, in arithmetic there is also the very important symbol "=", which as you know is used between expressions that have the same value, as in $(a+b)(a-b) = a^2 - b^2$. The corresponding concept in logic is called **equivalence between expressions** and will be denoted here by $\Leftrightarrow$ ($\equiv$ and simply $=$ are used as well). Two expressions are equivalent if they mean the same thing, which means that they have the same truth-tables. It's possible for instance to write

$$(p \wedge q) \wedge r \;\Leftrightarrow\; p \wedge (q \wedge r)$$

Rules for the Connectives $\wedge$, $\vee$, and $\neg$

associative laws

$$(p \vee q) \vee r \;\Leftrightarrow\; p \vee (q \vee r) \qquad (p \wedge q) \wedge r \;\Leftrightarrow\; p \wedge (q \wedge r)$$

commutative laws

$$p \vee q \;\Leftrightarrow\; q \vee p \qquad p \wedge q \;\Leftrightarrow\; q \wedge p$$

distributive laws

$$p \vee (q \wedge r) \;\Leftrightarrow\; (p \vee q) \wedge (p \vee r) \qquad p \wedge (q \vee r) \;\Leftrightarrow\; (p \wedge q) \vee (p \wedge r)$$

De Morgan's laws

$$\neg(p \vee q) \;\Leftrightarrow\; \neg p \wedge \neg q \qquad \neg(p \wedge q) \;\Leftrightarrow\; \neg p \vee \neg q$$

idempotence laws

$$p \vee p \;\Leftrightarrow\; p \qquad p \wedge p \;\Leftrightarrow\; p$$

absorption laws

$$p \vee (p \wedge q) \;\Leftrightarrow\; p \qquad p \wedge (p \vee q) \;\Leftrightarrow\; p$$

double negation

$$\neg(\neg p) \;\Leftrightarrow\; p$$

inverse laws

$$p \vee \neg p \;\Leftrightarrow\; 1 \qquad p \wedge \neg p \;\Leftrightarrow\; 0$$

identity laws

$$p \wedge 1 \;\Leftrightarrow\; p \qquad p \vee 0 \;\Leftrightarrow\; p$$

dominance laws

$$p \wedge 0 \;\Leftrightarrow\; 0 \qquad p \vee 1 \;\Leftrightarrow\; 1$$

Table 7.1: *Rules of calculations in propositional logic.*

We thus have *two* operations, $\leftrightarrow$ and $\Leftrightarrow$, that are both called "equivalence". What's the difference? Area of use, is a summary. $\leftrightarrow$ is used *inside* logical expressions (in the same way for instance as division in arithmetic), $\Leftrightarrow$ *between* them (like equal to). If it's true that "composite expression" $\Leftrightarrow$ "different expression", then "composite expression" $\leftrightarrow$ "different expression" is a *tautology*. More about those in the next section.

In normal mathematics there are a great number of rules that are used when rewriting expressions to a form one likes better. There exists a corresponding collection of rules for propositional logic. Table 7.1 above contains the most important rules.

By the way, it's not necessary to learn the *names* of the rules; the important thing is their *meaning*. If you want to learn some names it's first of all the *distributive laws* and *De Morgan's laws* that are important to be able to refer to, since they are the most complicated rules.

Rules for the Connectives $\rightarrow$ and $\leftrightarrow$
Rewriting $\rightarrow$
$p \rightarrow q \;\Leftrightarrow\; \neg p \vee q$
$p \vee q \;\Leftrightarrow\; \neg p \rightarrow q$
Rewriting $\leftrightarrow$
$p \leftrightarrow q \;\Leftrightarrow\; (p \rightarrow q) \wedge (q \rightarrow p)$
$p \leftrightarrow q \;\Leftrightarrow\; (p \wedge q) \vee (\neg p \wedge \neg q)$

Table 7.2: *Rules for the rewriting of implication and equivalence.*

Exercise 7.21 Study the rules, and contemplate their meaning and why they hold. *

Exercise 7.22 Write truth-tables for the rules you didn't understand, and see if you can convince yourself that way that they hold. (If you understood all of them, write truth-tables for a couple of rules anyway!) *

Exercise 7.23 Does anything strike you if you compare the right- and left-hand sides of the table with each other? *

So there are a lot of rules to use for rewriting expressions containing the three connectives negation, conjunction, and disjunction. But there are not as many for implication and equivalence. What is usually done with those expressions is to rewrite them into conjunctions and/or disjunctions, after which the rules for those connectives are applied. Table 7.2 above contains these rules.

If you want to determine whether two logical expressions mean the same thing but aren't able to analyse them you can, just as before, write truth-tables and compare, or using the established rules for rewriting try to get from one expression to the other. If you succeed you have proved that the expressions are equivalent. (A failure, on the other hand, doesn't prove that they aren't equivalent – perhaps you just didn't realise what you should do.)

Exercise 7.24 Show that $p \rightarrow q \vee r$ is equivalent to $\neg q \rightarrow \neg p \vee r$, both by writing the truth-tables and by using the rules for rewriting to get from one expression to the other. Try as well to imagine a sensible sentence having this structure, and check how it sounds reformulated.

7.2.4 Satisfiability in Propositional Logic

When you write the truth-table for an expression it usually contains a mixture of *true* and *false*. But sometimes you get only *true* or only *false*.

A proposition that is *always* true, irrespective of the values of its parts, is called a **tautology**. It can also be called a **valid statement**. $p \vee \neg p$ is an example of a tautology. (Election speeches are often tautologies. "It is necessary to take the measures that are necessary" is a good example. Can be rewritten as "if a measure is necessary then it is necessary", which is undoubtedly true.)

A proposition that is true at least sometimes is said to be a **satisfiable proposition**. All tautologies are satisfiable, but not all satisfiable propositions are tautologies. $p \vee q$, for instance, is satisfiable but not a tautology.

A proposition that is *never* true is called a **contradiction**. A contradiction isn't satisfiable. $p \wedge \neg p$ is a typical contradiction.

The truth-value of most propositions depends on their *contents*, on the values of the atomic propositions. The truth-value of tautologies and contradicitons, on the other hand, depends on their *form*. They are made up in such a way that the contents don't matter.

If you want to determine whether a proposition is a contradiction, satisfiable without being valid, or a tautology, there are several ways to proceed. The least profound one is to write the truth-table. If the expression is long or contains many propositional variables this method entails lots of work, though.

Another way is to see whether it's possible, using the rules for rewriting, to condense the expression to 0 (contradiction) or 1 (tautology). That entails even more work, if possible, and is nothing you want to start if you don't *know* that it will succeed.

The most elegant way is to contemplate the actual meaning of the expression and whether what is stated seems to hold. One version of this method is to check whether it's possible to assign values to the variables so that the expression becomes true, alternatively false. If the expression can be made true it isn't a contradiction; if it can be made false it isn't a tautology. We demonstrate using an example:

Example 7.4: *Analysis of an Expression* Is this expression a tautology, a contradiction, or neither?

$$(p \vee q) \wedge (q \vee r) \wedge (r \vee s) \wedge (s \vee t) \quad \rightarrow \quad (p \rightarrow q) \vee s$$

The expression contains 5 propositional variables, which means that the truth-table vill contain $2^5 = 32$ rows. Furthermore 10 connectives are included, which gives $5 + 10 = 15$ columns. Since there are more interesting things in life than filling in $32 \cdot 15 = 480$ values in a table, we check whether we can find the answer to the question by reasoning.

The main operation in this expression is *implication*. Implications are usually true; it's only when the second part is false at the same time as the first part is true that they are false. We check whether it's possible to assign values to the variables so that this becomes the case.

Since the second part is simpler, we start there. We want to make it false, and since the expression is a disjunction we have to make both parts of it, $p \to q$ and s, false. This they will be only if we assign the values $p = 1$, $q = 0$, and $s = 0$. Any other combination of values makes the second part true, and then the whole implication is classified as true. We can thus already see that the expression is *satisfiable*.

Now we have the values of three out of the five variables. We insert these values in the first clause, and then try to assign values to the remaining 2 variables so that the first part becomes true.

$$\begin{array}{c} (\underset{p}{1} \vee \underset{q}{0}) \wedge (\underset{q}{0} \vee r) \wedge (r \vee \underset{s}{0}) \wedge (\underset{s}{0} \vee t) \\ \Leftrightarrow \\ 1 \wedge r \wedge r \wedge t \\ \Leftrightarrow \\ r \wedge t \end{array}$$

We want the first part, which can be simplified into $r \wedge t$, to be true, and that it will be only if $r = t = 1$.

So we have found that the expression is false for the value combination $p = 1$, $q = 0$, $r = 1$, $s = 1$, $t = 1$, and *only* for that. For all other combinations it is true. It's thus neither a tautology nor a contradiction. ∎

Exercise 7.25 Determine whether $p \wedge \neg q \wedge r \wedge \neg s \ \to \ q \vee s \vee t$ is a tautology.

Studying tautologies and contradictions is intreresting for several reasons. If you have written a tautology in a conditional clause in a program (so that the test always gives true) or a contradiction (so that it always gives false) your reasoning has almost certainly been faulty. A test that always gives the same result doesn't really test anything, and the program will be faster without it.

Correctly made arguments translated into symbols will be tautologies:

Example 7.5: *Modus Ponens* "If it is Sunday the bank will be closed, and it *is* Sunday. So the bank is closed" is a correctly made argument. Translated into symbols it will become

$$(p \to q) \wedge p \ \to \ q$$

This expression is a tautology, and the line of argument is called **modus ponens**. ∎

Exercise 7.26

(a) Check that $(p \to q) \wedge p \ \to \ q$ really is a tautology by writing the truth-table.

(b) Convert the expression to 1 using the calculation rules.

Exercise 7.27 **Modus tollens**, $(p \to q) \wedge \neg q \ \to \ \neg p$, is another standard line of argument.

(a) Express in words an argument having this structure.

(b) Check that the expression is a tautology by writing the truth-table.

(c) Convert the expression to 1 using calculation rules.

7.3 Boolean Algebra

7.3.1 General Boolean Algebra

If you look at the table giving the caculation rules in propositional logic, on page 187, and the table giving the calculation rules in set theory, on page 20, it may strike you that they look very alike. If you take the set table, and exchange all $\cup$ for $\vee$, all $\cap$ for $\wedge$, all ^{c} for $\neg$, and $\emptyset$ for 0 and $\mathcal{U}$ for 1, you get the propositional table. The operations in set theory and propositional logic apparently have something in common.

If you then look at the graph of subsets on page 24 and the divisor graph on page 41, they are also strikingly alike. Apparently these operations have something in common as well. The union of two subsets is found above the sets, the intersection below. The least common multiple of two numbers is found above the numbers, the greatest common divisor below. So in this case the operations union and lcm have the same function, and the same thing applies to intersection and gcd. 1 seems to play the same part as the empty set, and the starting number and the starting set fills the same role.

That two different operations on the face of it have as a matter of fact striking similarities is something often used in higher mathematics. By using the properties they have in common it's possible to save work, since arguments made in one situation can be transformed to the other.

It's reasoning in this way that is *algebra*, in the real sense of the word. All the operations we have been discussing here – set theory, propositional logic, and working with divisors in numbers without repeated prime factors – are examples of what is called **Boolean algebras**, after the English mathematician George Boole (1815–1864).

We will not go into these matters in more depth, since they are a bit too abstract for this book, but we would like to mention that anything that satisfies a collection of rules like this is called a Boolean algebra.

Exercise 7.28 Express the concepts of *complement*, *union*, and *intersection* using the logical symbols and the set builder.

7.3.2 2-valued Boolean Algebra

Usually, people mentioning "Boolean algebra" mean **two-valued Boolean algebra**, which is simply the same thing as propositional logic (where the two values *true* and *false* are used). The difference between those working with "propositional logic" and those working with "Boolean algebra" is the notation. Logicians are usually mathematicans and use the weird symboles introduced in the previous section. Those working with "Boolean algebra" tend to be digital technicians, who are about to design circuits that are to be used in various products (anything from washing machines to computers). People working in production are forced to write their technical reports using standardised text editors (or in the Stone Age: typewriters), and those are usually not equipped with any keys for $\wedge$ and $\vee$. Lacking those symbols a multiplication sign is used for conjunction, plus for disjunction, and a prim-sign (p') or overline ($\overline{p}$) for negation. (When using symbols in this way with an established order of precedence, that order is borrowed as well. Here conjunction has higher precedence than disjunction.) The table showing the rules now looks like table 7.3 on the next page.

Exercise 7.29 Why do you think that $+$ was chosen for disjunction and $\cdot$ for conjunction, and not the other way around? *

Exercise 7.30 Is there any rule that you think looks actually "weird"?

Exercise 7.31

(a) Write this expression using the notation we used in the previous section:

$$\overline{x + \overline{y}z}$$

(b) Write this expression using the notation we introduced here:

$$\neg\big((x \wedge \neg y) \vee \neg(z \wedge w)\big)$$

7.3.3 Boolean Functions

A **Boolean function** is a function that takes a number of **Boolean variables** as input and gives a **Boolean value** as output.

Boolean functions usually appear when somebody is trying to develop a digital circuit that is to give different signals out for different combinations of signals in.

Example 7.6: ***A Boolean Function*** Here we have a Boolean function of three variables:

$$f(x, y, z) = (xy + \overline{z} + \overline{x}y)\overline{x\overline{y}\,\overline{z}}$$

Rules for the Operations $\cdot$, $+$ and $^-$
associative laws
$(p+q)+r = p+(q+r) \quad (p\cdot q)\cdot r = p\cdot(q\cdot r)$
commutative laws
$p+q = q+p \quad p\cdot q = q\cdot p$
distributive laws
$p+q\cdot r = (p+q)\cdot(p+r) \quad p\cdot(q+r) = p\cdot q + p\cdot r$
De Morgan's laws
$\overline{p+q} = \overline{p}\cdot\overline{q} \quad \overline{p\cdot q} = \overline{p}+\overline{q}$
idempotence laws
$p+p = p \quad p\cdot p = p$
absorption laws
$p+p\cdot q = p \quad p\cdot(p+q) = p$
double negation
$\overline{\overline{p}} = p$
inverse laws
$p+\overline{p} = 1 \quad p\cdot\overline{p} = 0$
identity laws
$p\cdot 1 = p \quad p+0 = p$
dominance laws
$p\cdot 0 = 0 \quad p+1 = 1$

Table 7.3: *Rules of calculations in Boolean algebra.*

It has the following table of values:

x	y	z	$f(x,y,z)$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

■

Exercise 7.32 Check at least two rows in the table of the function in example 7.6 for correctness. *

Exercise 7.33 Write the table of values for the Boolean function $f(x,y,z,w) = (x+\overline{y})\overline{z} + \overline{wz}y$.

Exercise 7.34 How many different Boolean functions of n variables are there? (Functions giving the same output for the same combinations of input are counted as identical, even if they use completely different methods to calculate the said output. More about functions in the next chapter.)

The fact that one usually writes the expression defining the function when analysing what the circuit is to do tends to lead to a rather strange end result from a mathematical point of view. Since there is only a finite number of Boolean functions, but on the other hand an infinite number of ways of writing the expression for the computation of the function, it's of interest to define some simple standard ways of expressing a function.

The most important standard ways are firstly **disjunctive form**, where the function is written as a "disjunction of conjunctions" (a sum of products). The function in example 7.6 on page 192 can for instance be written as

$$f(x,y,z) = \overline{x}\,\overline{z} + yz + xy\overline{z}$$

(There are lots of other ways of writing the function that fulfils the conditions for being a disjunctive form. This is just one variant.)

Then we have **conjunctive form**, where the function instead is written as a conjunction of disjunctions (a product of sums). The same function in conjunctive form:

$$f(x,y,z) = (\overline{x}+y)(x+y+\overline{z})$$

There are many conjunctive forms as well.

If we want to standardise even further there are **disjunctive normal form**, **d n f**, and **conjunctive normal form**, **c n f**. There we demand as well that every subexpression must include all the variables, perhaps negated. The function in example 7.6 has the disjunctive normal form

$$f(x,y,z) = \overline{x}\,\overline{y}\,\overline{z} + \overline{x}y\overline{z} + \overline{x}yz + xy\overline{z} + xyz$$

and the conjunctive normal form

$$f(x,y,z) = (x+y+\overline{z})(\overline{x}+y+z)(\overline{x}+y+\overline{z})$$

Exercise 7.35 Verify that the given expressions, both those in normal form and the others, really correspond to the same function as in example 7.6.

The point about the normal forms is that there is only *one* way of writing a function in normal form (order excepted), and that can be rather practical.

How do you then go about it to rewrite functions to disjunctive and conjunctive (normal) form? Usually one has to start by moving all the negations into the centre of the expressions using De Morgan's laws, after which one applies one distributive law or the other, until one ends up with the form one wants. If one wants to go as far as to the normal form the expressions have to be augmented with some "$+\,x\overline{x}$" or "$\cdot\,(x+\overline{x})$" at suitable places, after which the distributive laws are used again.

Example 7.7: ***Normal Form from Calculations*** We rewrite the function from example 7.6 on page 192 to disjunctive normal form using the calculation rules.

$$\begin{aligned}
f(x,y,z) &= (xy+\overline{z}+\overline{x}y)\overline{xy\,\overline{z}} \\
&= (xy+\overline{z}+\overline{x}y)(\overline{x}+y+z) \\
&= xy\overline{x} + xyy + xyz + \overline{z}\,\overline{x} + \overline{z}y + \overline{z}z + \overline{x}y\overline{x} + \overline{x}yy + \overline{x}yz \\
&= x\overline{x}y + xy + xyz + \overline{x}\,\overline{z} + y\overline{z} + \overline{z}z + \overline{x}\,\overline{x}y + \overline{x}yy + \overline{x}yz \\
&= xy + xyz + \overline{x}\,\overline{z} + y\overline{z} + \overline{x}y + \overline{x}yz \quad \longleftarrow \text{disjunctive form} \\
&= xy(z+\overline{z}) + xyz + \overline{x}(y+\overline{y})\overline{z} + (x+\overline{x})y\overline{z} \\
&\quad + \overline{x}y(z+\overline{z}) + \overline{x}yz \\
&= xyz + xy\overline{z} + xyz + \overline{x}y\overline{z} + \overline{x}\,\overline{y}\,\overline{z} + xy\overline{z} + \overline{x}y\overline{z} \\
&\quad + \overline{x}yz + \overline{x}y\overline{z} + \overline{x}yz \\
&= xyz + xy\overline{z} + \overline{x}y\overline{z} + \overline{x}\,\overline{y}\,\overline{z} + \overline{x}yz \quad \longleftarrow \text{disjunctive } \textit{normal} \text{ form}
\end{aligned}$$

We can also get it to conjunctive form this way:

$$\begin{aligned}
f(x,y,z) &= (xy+\overline{z}+\overline{x}y)\overline{xy\,\overline{z}} \\
&= (xy+\overline{x}y+\overline{z})(\overline{x}+y+z) \\
&= \big((x+\overline{x})y+\overline{z}\big)(\overline{x}+y+z) \\
&= (y+\overline{z})(\overline{x}+y+z) \quad \longleftarrow \text{conjunctive form} \\
&= (x\overline{x}+y+\overline{z})(\overline{x}+y+z) \\
&= (x+y+\overline{z})(\overline{x}+y+\overline{z})(\overline{x}+y+z) \quad \longleftarrow \text{conjunctive } \textit{normal} \text{ form} \quad \blacksquare
\end{aligned}$$

Exercise 7.36 What rules have been used in the different steps in the calculations in the example? *

Exercise 7.37 Rewrite the function in excercise 7.33 on page 193 to disjunctive normal form using the rules, and note in every step which rule you use.

In general, it tends to be easier to rewrite to disjunctive normal form, since you then mainly use the distributive law that looks like the one you are used to. Multiplying things into parentheses feels rather more familiar than adding into them.

One way of solving this problem is to add two negations on the outside of the expression, "feed" one of them into the expression using De Morgan's laws while letting the other one remain outside, rewrite the inner expression to *disjunctive* normal form, after which one moves the other negation, which will turn the disjunctive expression to a conjunctive one. Then only the distributive law that looks "sensible" has to be used.

> Exercise 7.38 Try to rewrite the function in example 7.6 on page 192 to conjunctive normal form using the suggested method with the double negation. *

It's also possible to completely avoid the calculation rules by using the table of values as a starting point instead (something that's especially useful in the not uncommon situation when one only has a table and no formula).

Example 7.8: ***Normal Form from Table*** We extend the table of values in example 7.6 on page 192 with two columns:

x	y	z	$f(x,y,z)$	minterms	maxfactors
0	0	0	1	$\overline{x}\,\overline{y}\,\overline{z}$	
0	0	1	0		$x+y+\overline{z}$
0	1	0	1	$\overline{x}y\overline{z}$	
0	1	1	1	$\overline{x}yz$	
1	0	0	0		$\overline{x}+y+z$
1	0	1	0		$\overline{x}+y+\overline{z}$
1	1	0	1	$xy\overline{z}$	
1	1	1	1	xyz	

The first expression in the column **minterms** has the value one for the value combination in that row, but not for any other combination of values. The same thing applies to the other expressions in that column. By adding these expressions we get a sum of terms, where some term will be one for those value combinations for which the function is to be one. For the remaining combinations of indata all the terms are zero, and thereby the whole expression will be zero. If we look at the terms, we find that it's exactly the same terms that we got when rewriting the expression to disjunctive normal form using calculation rules.

The expressions in the **maxfactors** column will be zero for the value combination in the row where they are written, and one for all other combinations of indata. If we multiply the expressions we get something that is zero for the indata where the function is supposed to be zero, and one in other cases. ∎

> Exercise 7.39 Write the function in exercise 7.33 on page 193 in disjunctive and disjunctive normal form using the table of values.

What is "better", conjunctive or disjunctive normal form? That's a matter of taste, but we can note that if the function mostly consists of zeros the

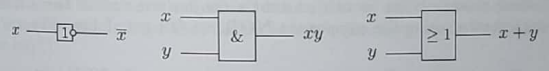
disjunctive form is *simpler*, while the conjunctive form is shorter when there are a lot of ones.

Which one of the two methods to get to the normal form is better, then? Generally speaking, the method using the table of values is guaranteed to work, while you can be sitting forever when trying to rewrite, if you don't realise which steps will take you to the goal. If the expression is very close to normal form, it's usually faster to start there, otherwise the the method using the table is probably quicker. If you already *have* a table it's of course simpler to start there.

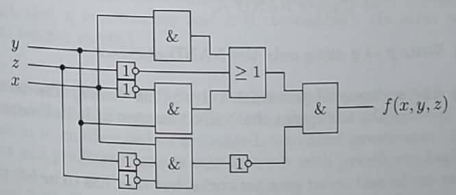
7.3.4 Some Words about Gate Networks

A Boolean function is then to be realised in the form of a **circuit** that carries out the computation and transmits the result. The components in one of these are called **gates**.

The most common gates are firstly **inverters**, also called **NOT-gates**. They "invert" the input signal, so that one in gives zero out, and vice versa. Then there are **AND-gates**. They can usually take an arbitrary number of input signals, and send out one if all the signals are one. Otherwise, they send out zero. Lastly, there are **OR-gates**, that send out zero if all the input signals are zero, and one otherwise. The gates are, according to ISO-standard, denoted by the symbols



The function in example 7.6 on page 192 can be implemented as



If you start with a formula given in disjunctive form you get a somewhat more structured circuit. First there is a layer of inverters, then a layer of AND-gates, and at the end an OR-gate. The conjunctive form is the same, although with OR-gates first and AND-gate last.

> Exercise 7.40 Look at the circuit drawn here and mark on all the connections which signal, (x, y, and so on) is carried in them, and check that the end result really is $f(x,y,z)$. *

Exercise 7.41 Draw the circuit diagram of the disjunctive and conjunctive normal forms of the function in example 7.6.

It is possible to standardise even more: if you imagine a circuit that implements a disjunctive form, it is possible to add two inverters on each connection between the AND-gates and the OR-gate. (Two inversions cancel out, so that doesn't affect the output from the circuit.) Inverting the input signals to an OR-gate corresponds, according to De Morgan's law, to inverting the output from an AND-gate, so we can move one of the layers of inverters to the outside of the circuit if we exchange the OR-gate for an AND-gate.

If you build a kind of gate that corresponds to *and* followed by *not* (which put concretely means that it sends out 0 only if all input signals are 1) it can be used in place of both the AND-gates and the OR-gate. That means that the whole circuit can be implemented using just one kind of component. This component is called an **NAND-gate**.

It's practical that you can use the same component everywhere, since that reduces the risk of putting in the wrong thing. But even from a theoretical point of view this is interesting: all logical expressions can be written in disjunctive form. An expression given in disjunctive form can be calculated using just one operation. Thus the totality of propositional logic can, if desired, be expressed using just one operation: NAND. The three connectives we started with were as a matter of fact an unnecessarily large set.

The same reasoning can be carried out on conjunctive normal form. It can be implemented using the component **NOR**, an OR-gate followed by not.

Exercise 7.42 Describe in words the function of a NOR-gate.

Exercise 7.43 Denote NAND $\overline{\wedge}$.

(a) Write $p \to q$ using only the NAND-operator.

(b) The sentence "If you wash I dry" (the dishes) has the structure $p \to q$. Try to express the NAND-version of it in English.

Much more can be said about gate networks, but that has to be left to courses in digital technology.

7.4 Predicate Logic

Propositional logic, irrespective of whether the logical or the digital-technological symbols are used, is useful. But it's not quite powerful enough to handle all the things you may want to say.

7.4.1 Quantifiers and Predicates

We start by studying a number of (more or less true) statements:

- "All students are lazy."
- "All integers greater than one can be factorised into primes."
- "No reasonable teacher exists."
- "A number with the square of one exists."

When studied in detail, the statements seem to belong to two basic kinds: "All" and "Exists", respectively. On closer examination it can be found that almost all mathematical theorems and a great part of everything else that is stated has one of these two basic structures (even if sometimes expressed a bit differently). Because of this, two **quantifiers** have been introduced, $\forall$ that represents "for all" or "for each", and $\exists$ that represents "there is" or "there exists". Furthermore one needs, just as when describing sets using the set builder, to use some name, like x, in the expression.

Example 7.9: ***Usage of Quantifiers*** Some statements, expressed using quantifiers:

$$\forall x(x \text{ is a student} \to x \text{ is lazy})$$

which is read as "for each individual in the universe, call the said individual x, it's true that if x is a student, then x is lazy". (In other words: "All students are lazy".)

$$\neg\exists x(x \text{ is a teacher} \wedge x \text{ is reasonable})$$

reads "it is not true that there exists some individual, call the said individual x, such that x is a teacher and x is reasonable". (In other words: No reasonable teacher exists.) ■

In the composite statements there are substatements such as "x is a student" and "x is reasonable". Expressions like this, where the properties of an object or its relation to another object are stated, are called **predicates**. If one uses quantifiers and predicates one is said to work with **predicate logic**.

Definition 7.6: ***Predicates*** A **monadic predicate** is a function from the universe to the two truth-values true and false. (What's meant by *function* will be explained in more detail in the next chapter.)

As an example we can take "is a student". We'll denote "x is a student" by $S(x)$. For instance S(president of the student union) is a true statement, while S(vice-chancellor of the university) (probably) is false. Another monadic predicate is "x is invertible" (useful when discussing modular arithmetic). If we denote this concept by $I(x)$ we find that $I(1)$ is true, while $I(0)$ is false.

A dyadic predicate is a function from the set of ordered pairs in the universe to the truth-values. One example is "have written a book together with". If we denote "x has written a book together with y" by $B(x, y)$ we find that B(Kimmo, Hillevi) is true, while B(Kimmo, Strindberg) isn't. (Dyadic predicates are also called *binary relations*, a concept that a large part of the next chapter will cover.)

In the same way, triadic and so on predicates can be defined.

Predicates are usually denoted by capital letters or a descriptive name, and for the rest written as the functions that they are.

Note that since undetermined values don't exist in normal logic the predicates have to deliver values for all possible indata, not just sensible ones. ∎

Exercise 7.44 Invent a monadic, a dyadic, and a triadic predicate, applicable on the set of humans.

Exercise 7.45 Invent a monadic, a dyadic, and a triadic predicate, applicable to the set of natural numbers.

Example 7.10: ***Using Predicates*** "All students are lazy" can be written

$$\forall x\big(\mathit{Student}(x) \to \mathit{Lazy}(x)\big)$$

and "no reasonable teacher exists"

$$\neg\exists x\big(\mathit{Teacher}(x) \wedge \mathit{Reasonable}(x)\big)$$

If we prefer shorter names on the predicats we can introduce the notation $S(x)$ for "x is a student", $L(x)$ for "x is lazy", $T(x)$ for "x is a teacher", and $R(x)$ for "x is reasonable". The sentences will then be written as

$$\forall x(S(x) \to L(x)) \qquad \text{and} \qquad \neg\exists x(T(x) \wedge R(x))$$ ∎

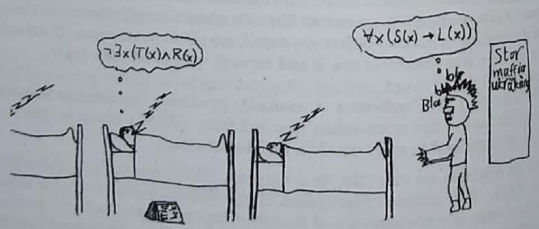


Figure 7.1: *The question is who isn't inspiring whom in this situation...*

In the example, we could just as well have used the letter y instead of x, or whichever symbol we like that isn't already in use in some other sense; it doesn't matter which one we take. (It would for instance not be a good idea to use 7, since people definitely have preconceived ideas about the meaning of this symbol. But we are welcome to use ξ, if we find that letter easy to write.)

Exercise 7.46 Indicate which parts of the plain-text expressions ("for each individual...") in the example that correspond to which symbols in the mathematical expressions.

There is a dyadic predicate that is so unique that it isn't possible to manage without it. It's **equal to**, usually denoted by the symbol "=". $x = y$ means "x and y is the same individual in the universe".

Example 7.11: ***One and Only One*** Express "there is one and only one object in the universe having the property P" using predicate-logical symbols.

The sentence can be reformulated as "there is one object in the universe having the property P, and if one finds two objects having the property P they are, as a matter of fact, identical". (The proof of the fundamental theorem of arithmetic on page 48 is organised according to this model.)

Converted to symbols the sentence becomes

$$\exists x\, P(x) \;\wedge\; \forall x \forall y\big(P(x) \wedge P(y) \to x = y\big)$$ ∎

Exercise 7.47 Write "There is more than one object having property P" using predicate-logical symbols.

7.4.2 Truth-Values, and Rules for Syntax and Calculations in Predicate Logic

As may be apparent from the examples given, propositional logic is part of predicate logic. An expression like $P(a)$ has a truth-value, and is thus a proposition. The quantifier-decorated expressions are propositions as well, and are treated as such. Now we are going to take a closer look at how they are put together.

A for-all statement has the form

$$\forall x\,\{\text{proposition}_1 \text{ (that probably but not necessarily contains } x)\}$$

(The variable doesn't have to be named x, but we have to print something.) The whole thing is read as "for each element, call it x, in the universe, proposition_1 is true". Proposition_1 may be a for-all expression or an existence expression, or made up of smaller propositions, bound together by connectives, and those may in their turn be predicate-logical.

Example 7.12

$$\forall x \forall y \big(P(x,y) \wedge Q(y,x)\big)$$

is a correctly formed for-all expression. Proposition$_1$ is in this case

$$\forall y \big(P(x,y) \wedge Q(y,x)\big)$$

which is a for-all expression, the attached proposition being the conjunction

$$P(x,y) \wedge Q(y,x)$$

∎

The for-all expression is classified as true if proposition$_1$ is true irrespective of the value put into the place of x.

Example 7.13: *For All* If our universe consists of the numbers 2, 3, 5, and 7, and $P(x)$ means "x is a prime number" then $\forall x\, P(x)$ ("all objects in the universe are prime numbers") is true, since $P(2)$, $P(3)$, $P(5)$, and $P(7)$ are true, all of them.

If $O(x)$ means "x is odd" then $\forall x\, O(x)$ *isn't* true, since $O(2)$ isn't true. (That $O(x)$ is true for all other values of x doesn't help.) ∎

The same principles hold for existence statements. An existence statement has the form

$$\exists x \{\text{proposition}_2 \text{ (that probably but not necessarily contains } x)\}$$

which is read as "There exists an element, call it x, in the universe, for which proposition$_2$ is true".

Existence expressions are classified as true if there exists some value that put into x's place makes proposition$_2$ true.

Example 7.14: *There Exists* If we use the same universe and predicates as in the previous example, then $\exists x \big(P(x) \wedge \neg O(x)\big)$ ("there exists an object in the universe that is prime and not odd") is true, since $P(2) \wedge \neg O(2)$ is true. $\exists x \neg P(x)$ on the other hand isn't true, since there is no number in our universe that isn't prime. ∎

Exercise 7.48 We have a universe consisting of the letters a–f. $C(x)$ represents "x is a consonant" and $V(x)$ "x is a vowel". Which of the following statements are true?

(a) $C(a)$ **(b)** $V(a)$ **(c)** $C(b)$ **(d)** $V(b)$
(e) $\forall x\, C(x)$ **(f)** $\exists y\, C(y)$ **(g)** $\forall z\, V(z)$ **(h)** $\exists t\, V(t)$
(i) $\forall u \big(C(u) \vee V(u)\big)$ **(j)** $\exists v \big(C(v) \wedge V(v)\big)$

If a proposition includes a predicate with a variable (like $P(x)$) then that variable must have been introduced at some point. The quantifier expressions act in part as variable declarations, and they are necessary if the truth-value is to be determined. (If you don't know if "x" belongs to a "for-all" or an "exists" it's impossible to determine whether the expression is true.) If we introduce a variable x using a $\forall x$ or an $\exists x$ then this variable is defined in the expression immediately to the right. To write for instance

$$\forall x\, P(x) \wedge Q(x)$$

is *incorrect*, since the "expression immediately to the right" is $P(x)$ and nothing more. The x in $Q(x)$ is arriving "out of nowhere". What the writer presumably meant was

$$\forall x \big(P(x) \wedge Q(x)\big)$$

The outer pair of parentheses makes the expression into one entitety.

Exercise 7.49 Mark in the expression below in which part of the expression each of the variables is defined.

$$\forall x \big\{P(x) \wedge \exists y\, Q(y,x) \rightarrow \forall z \big(R(x,z) \vee \exists w\, S(z,w)\big)\big\}$$

(We have used $\{\}$ as the outermost backets instead of $()$ to make it easier for the reader to match the parentheses. It has no deeper meaning apart from this.)

Nothing prohibits the reusage of a name that has been released; it's totally acceptable to write things like

$$\forall x\, P(x) \;\wedge\; \exists x\, Q(x)$$

The x in $P(x)$ belongs to the all-quantifier while the one in $Q(x)$ belongs to the existence-quantifier.

It's even allowed (but perhaps not to be recommended) to write things like

$$\forall x \big(P(x) \wedge \exists x\, Q(x)\big)$$

Each x belongs to the *innermost* of the surrounding quantifiers, so the x in $Q(x)$ belongs to the existence-quantifier, while the x in $P(x)$ is the one declared by the all-quantifier. (Those who have programmed in for instance the language Pascal ought to recognise the principle.)

Exercise 7.50 Determine to which quantifier the different y's in this expression belong:

$$\exists y \big(P(y) \vee \forall y\, Q(y) \vee R(y)\big)$$

All the rules of propositional logic can be used in predicate logic as well. Besides, som new rules are added. First two rules concerning negation:

$$\neg\forall x\, P(x) \;\Leftrightarrow\; \exists x \neg P(x) \quad \text{and} \quad \neg\exists x\, P(x) \;\Leftrightarrow\; \forall x \neg P(x)$$

Examples of the application of the rules:

"Not all teachers of logic are men" is the same thing as "There exists at least one teacher of logic who isn't a man".

"It's not true that there exists an examiner who is nice" is the same thing as "All examiners are non-nice".

These rules are important. (Who hasn't heard arguments like "I once met a foreigner who was unpleasant, so therefore all foreigners are crooks" or "my grandmother can't count, so therefore no women understand mathematics"? Things like that are logically incorrect.)

Example 7.15: ***Reformulation*** We rewrite the statement "no reasonable teacher exists" several times, mostly to check in how many different ways it is possible to express the same thing:

$$\begin{array}{ll}
\neg\exists x\big(T(x) \wedge R(x)\big) & \neg\exists x\big(\mathit{Teacher}(x) \wedge \mathit{Reasonable}(x)\big) \\
\Leftrightarrow & \Leftrightarrow \\
\forall x \neg\big(T(x) \wedge R(x)\big) & \forall x \neg\big(\mathit{Teacher}(x) \wedge \mathit{Reasonable}(x)\big) \\
\Leftrightarrow & \Leftrightarrow \\
\forall x\big(\neg T(x) \vee \neg R(x)\big) & \forall x\big(\neg \mathit{Teacher}(x) \vee \neg \mathit{Reasonable}(x)\big) \\
\Leftrightarrow & \Leftrightarrow \\
\forall x\big(T(x) \rightarrow \neg R(x)\big) & \forall x\big(\mathit{Teacher}(x) \rightarrow \neg \mathit{Reasonable}(x)\big)
\end{array}$$

The three new versions can be read as "for all beings in the universe it's untrue that they are both teachers and reasonable", "for all beings in the universe it's true that they aren't teachers or aren't reasonable", and "for all beings in the universe it's true that if the said being is a teacher the said being isn't reasonable", or put more concisely "teachers aren't reasonable". ■

Exercise 7.51 What rules have been used in the different steps in the calculation?

Exercise 7.52 Can you reformulate the statement in even more ways?

Furthermore you can change the names of variables, as long as you don't create any clashes of names. For instance it's fully acceptable to rename y as x in $\forall x\, P(x) \;\wedge\; \exists y\, Q(y)$ while it can't be done in $\forall x \exists y\, P(x, y)$.

Exercise 7.53 What exactly would happen if you changed the name in the latter case?

Calculation Rules in Predicate Logic

Negation

$$\neg\forall x\, P(x) \;\Leftrightarrow\; \exists x \neg P(x) \qquad \neg\exists x\, P(x) \;\Leftrightarrow\; \forall x \neg P(x)$$

Transposal of quantifiers of the same kind

$$\forall x \forall y\, P(x, y) \;\Leftrightarrow\; \forall y \forall x\, P(x, y) \qquad \exists x \exists y\, P(x, y) \;\Leftrightarrow\; \exists y \exists x\, P(x, y)$$

Name changes(*)

$$\forall x\, P(x) \;\Leftrightarrow\; \forall y\, P(y) \qquad \exists x\, P(x) \;\Leftrightarrow\; \exists y\, P(y)$$

Extension and restriction of the reach of quantifiers(*)

$$\begin{array}{ll}
\forall x\, P(x) \;\wedge Q \;\Leftrightarrow\; \forall x\big(P(x) \wedge Q\big) & \exists x\, P(x) \;\wedge Q \;\Leftrightarrow\; \exists x\big(P(x) \wedge Q\big) \\
\forall x\, P(x) \;\vee Q \;\Leftrightarrow\; \forall x\big(P(x) \vee Q\big) & \exists x\, P(x) \;\vee Q \;\Leftrightarrow\; \exists x\big(P(x) \vee Q\big)
\end{array}$$

Table 7.4: *Rules of predicate logic. The rules marked with (*) holds under the condition that the change doesn't lead to any variable being linked to another quantifier than previously. ("Clashes of names").*

Furthermore you can extend and limit the extension of a quantifier, still provided that it doesn't create any collisions of names.

$$\forall x P(x) \;\wedge\; Q \;\Leftrightarrow\; \forall x\,\big(P(x) \wedge Q\big)$$

provided that no x in the composite expression Q gets linked to the all-quantifier by this rewriting. It's fully allowed to change $\exists y\big(\forall x\, P(x) \;\wedge\; Q(y)\big)$ to $\exists y \forall x\big(P(x) \wedge Q(y)\big)$ while the same thing can't be done with $\exists x\big(\forall x\, P(x) \;\wedge\; Q(x)\big)$.

Exercise 7.54 What would happen if you changed the second expression according to the same pattern as the first one?

Corresponding rules hold for expressions of the types $\forall x\, P(x) \vee Q$, $\exists x\, P(x) \wedge Q$, and $\exists x\, P(x) \;\vee\; Q$.

Using these rules it's possible to rewrite all predicate-logical expressions so that all the quantifiers are placed at the beginning, if so desired. This is called **prenex form**.

Example 7.16: ***To Prenex Form*** Rewrite the expression $\neg\big(\exists x\, P(x) \;\wedge\; \forall x \neg Q(x)\big)$ so that all the quantifiers are placed at the beginning.

If there are to be quantifiers in the beginning there should not be a negation there. Our first move because of this will be to "feed" the negation into the expression. After this we try to extend the reach of the quantifiers, so that they cover the whole expression. To do this, we have to rename one of the variables to something else, because we can't have two different things both

named x in the same expression.

$$\begin{aligned}
&\neg\big(\exists x\, P(x) \ \wedge\ \forall x\, \neg Q(x)\big)\\
&\qquad\Leftrightarrow\\
&\neg\exists x\, P(x) \ \vee\ \neg\forall x\, \neg Q(x)\\
&\qquad\Leftrightarrow\\
&\forall x\, \neg P(x) \ \vee\ \exists x\, \neg\big(\neg Q(x)\big)\\
&\qquad\Leftrightarrow\\
&\forall x\, \neg P(x) \ \vee\ \exists x\, Q(x)\\
&\qquad\Leftrightarrow\\
&\forall x\big(\neg P(x) \ \vee\ \exists x\, Q(x)\big)\\
&\qquad\Leftrightarrow\\
&\forall x\big(\exists x\, Q(x) \ \vee\ \neg P(x)\big)\\
&\qquad\Leftrightarrow\\
&\forall x\big(\exists y\, Q(y) \ \vee\ \neg P(x)\big)\\
&\qquad\Leftrightarrow\\
&\forall x \exists y\big(Q(y) \vee \neg P(x)\big)
\end{aligned}$$

If rewriting the expression in this way was a *good* idea is a different question. Some like to declare all the variables involved from the start, others think that the logical structure is clearer if the variables are introduced at the point where they are needed. (On this point programmers as well are of different opinions; some want to declare all the variables of a function from the start, others think that it's better to define temporary variables such as loop counters in the block where they are used.) ∎

Exercise 7.55 What rules have been used in the different steps in the calculation? *

Exercise 7.56 Rewrite $\exists x\, P(x) \to \forall x\, Q(x)$ so that all the quantifiers end up in front.

If you have an expression beginning with several quantifiers *of the same kind* you can let them change places. $\forall x \forall y\, P(x, y)$ and $\forall y \forall x\, P(x, y)$ mean the same thing, because both expressions say that P holds for every pair. The same thing applies to existence-quantifiers.

On the other hand you can normally *not* let quantifiers of different kinds swap places, since $\forall x \exists y\, P(x, y)$ and $\exists y \forall x\, P(x, y)$ usually don't mean the same thing. If we for instance say that $P(x, y)$ stands for "x is named y" then the first expression means "everybody is named something" while the other means "there is something that everybody is named". And those sentences aren't quite the same. (The first one is fairly true, while the second one definitely isn't.)

Exercise 7.57 What is actually the great difference between "everybody is named something" and "there is something that everybody is named"?

Exercise 7.58 Find some more examples where $\forall x \exists y\, P(x, y)$ is true while $\exists y \forall x\, P(x, y)$ is false.

Exercise 7.59 Find some examples where $\forall x \exists y\, P(x, y)$ and $\exists y \forall x\, P(x, y)$ actually mean the same thing.

Exercise 7.60 Is it possible to find an example where $\forall x \exists y\, P(x, y)$ is false while $\exists y \forall x\, P(x, y)$ is true?

7.4.3 Translating to Predicate-logical Notation

Before applying the predicate-logical rules on a statement you have to express it using logical notation.

It's usually not possible to translate a statement in English (or some other language) directly into predicate-logical notation. Normal languages are a lot more brief than the language of logic, and it's usually necessary to fill in lots of details before starting to introduce symbols. Often a number of reformulations are needed, where the form of the sentence is adapted step by step to the logical grammar.

Example 7.17: ***Translation***

A student who has failed an exam hates the exam designers.

It says "a student" but on reflection one realises that in spite of this the sentence is supposed to apply to *all* students. (When speaking about *one* totally unspecified object the statement can usually be applied to all objects of that kind.) So a first reformulation is

For everyone that is a student it is true that if the said person has failed an exam then the said person hates the exam designers.

"An exam" is in the same way to be interpreted as "all exams". Furthermore the "exam designer" are all the persons that have made the exam that we are discussing at the moment.

For all students and for all exams it is true that if the first-mentioned has failed the last-mentioned then the first-mentioned hates everyone who has designed the last-mentioned.

This can almost be turned into symbols, we just have to tidy up the end a bit.

For all students and for all exams it is true that if the first-mentioned has failed the last-mentioned then it is true for all individuals that if they have designed the last-mentioned then the first-mentioned hates them.

Turned into predicate-logical notation this becomes

$$\forall x \forall y \{ \mathit{Student}(x) \wedge \mathit{Exam}(y) \wedge \mathit{Failed}(x,y) \rightarrow \forall z \big(\mathit{Designed}(z,y) \rightarrow \mathit{Hates}(x,z) \big) \}$$

If we prefer one-letter names on the predicates we can introduce the notation $S(x)$ for "x is a student", $E(x)$ for "x is an exam", $F(x,y)$ for "x has failed y", $D(x,y)$ for "x has designed y" and $H(x,y)$ for "x hate s y". Then we get

$$\forall x \forall y \{ S(x) \wedge E(y) \wedge F(x,y) \rightarrow \forall z \big(D(z,y) \rightarrow H(x,z) \big) \}$$

Which is to be preferred, long names or short ones, depends somewhat on what you are going to use the expression for. The long ones are more clear but take more work to write; the short ones have the advantage of not hiding the stucture of the sentence.

If one prefers prenex form, the expression can be rewritten as

$$\forall x \forall y \forall z \{ S(x) \wedge E(y) \wedge F(x,y) \rightarrow \big(D(z,y) \rightarrow H(x,z) \big) \}$$ ∎

Exercise 7.61 How does the sentence

$$\exists x \exists y \{ E(y) \wedge D(x,y) \wedge \forall z \big(S(z) \rightarrow H(x,z) \big) \}$$

read in English? (The same predicates as in the example.)

Translation Rules of Thumb for-all expressions are almost always implications, or can be rewritten as implications using rules from propositional logic. The reason is that there are very few statements that really apply to *everything in the whole universe*. At least you tend to have to add some disclaimer to the effect that the statement applies to objects of a certain type. (A statement discussing inversion ends up somewhat without meaning if you feed a tadpole into it. Are tadpoles invertible?)

Existence statements are hardly ever implications. The reason for that is that an implication is automatically classified as true if the first clause is false, and it's almost always possible to find some ridiculous value that put into the first clause gives the result false, and then "there is something that makes this proposition true" is fullfilled. That makes an existence statement that is an implication true in almost all possible universes, and that's seldom the intention.

(Of course there are expressions that contain both "exists" and implications. But the implication is seldom the *main operation* in the existence part of the proposition.)

Usually existence expressions are conjunctions, on the line of "there is an object that fullfils this and this and this requirement". ∎

Example 7.18 We try to translate the sentences "every student is hard-working" and "there is a student who is hardworking" to logical notation.

We start by introducing notation. $S(x)$ can represent x is a "student" and $H(x)$ "x is hard-working".

The first sentence becomes "For each individual it's true that if it's a student then it's hard-working",

$$\forall x \big(S(x) \rightarrow H(x) \big)$$

and the second one becomes "there is somebody that is firstly a student and secondly is hard-working",

$$\exists x \big(S(x) \wedge H(x) \big)$$ ∎

Exercise 7.62

(a) One student tries to get the following translation of the first sentence accepted: $\forall x \big(S(x) \wedge H(x) \big)$. What does the thing the student has written mean?

(b) Another student tries with the following translation of the second sentence: $\exists x \big(S(x) \rightarrow H(x) \big)$. What does this mean?

Exercise 7.63 Translate into good English the following (more or less true) sentences, where the predicates represent

$E(x)$:	x is an exam
$S(x)$:	x is a student
$H(x)$:	x becomes happy
$D(x,y)$:	x has designed y.
$P(x,y)$:	x has passed y.
$G(x,y)$:	x thinks that y is good.
$T(x,y)$:	x har been tested on y.

(a) $\forall x \forall y \big(E(x) \wedge D(y,x) \rightarrow G(y,x) \big)$

(b) $\forall x \{ E(x) \wedge \neg \exists y \big(T(y,x) \wedge P(y,x) \big) \rightarrow \forall z \big(D(z,x) \rightarrow \neg H(z) \big) \}$

(c) $\forall x \forall y \big(E(x) \wedge S(y) \wedge T(y,x) \wedge P(y,x) \rightarrow H(y) \wedge B(y,x) \big)$

Exercise 7.64 Translate the following sentences to predicate-logical notation (the same predicates as in the previous exercise).

(a) "A student who hasn't passed an exam doesn't think that the exam was good".

(b) "There is an exam designer who gets happy when somebody has passed the exam."

(c) "If everyone who is tested on the exam passes it as well there is no-one who thinks that the exam was bad".

7.4.4 Satisfiability in Predicate Logic

To determine whether a predicate-logical expression is a tautology, a contradiction, or neither is in general *a lot* more difficult than the corresponding problem in propositional logic.

When working with propositions, it's always possible to test all the possible combinations of values. When working with predicates you would have to test all the possible universes. To make it possible to determine the truth of a predicate-logical expression an **interpretation** is usually needed, which means that one decides which the individuals of the universe are and what truth-values the predicates will have for the individuals. And the set of all possible interpretations is infinite, to say the least!

It's always possible to get some information by using an easily managed universe (with for instance one individual), and checking the truth-value of the expression there. If it's true there then it isn't a contradiction; if it's false it's not a tautology. On the other hand, you can't *confirm* that an expression is a contradiction/tautology using this method; that takes an argument that shows that it's the same case in *all* interpretations. (There are, by the way, expressions that are true in all finite universes, but not in infinite ones. And who has time to make a trial run through an infinite universe?)

There are systematic methods for finding a universe that makes an expression true/false, but they are outside the scope of this book.

Of course, there are predicate-logical expressions where the truth-value is easily determined. That $\forall x\big(P(x) \vee \neg P(x)\big)$ is a tautology is for instance fairly obvious.

> Exercise 7.65 Express in words what the following statement says, and exclude one of the alternatives "tautology" and "contradiction" by inventing an interpretation that makes the statement false or alternatively true.
>
> $$\forall x(P(x) \to \exists y\, P(y))$$

7.5 Proof Technique

Why does one take the trouble to translate easily understood sentences to hard-to-understand symbols in this way?

This question has several answers. The happy logician answers: "Because it's fun!", but normal people may not share that opinion.

A more sensible reason is that it helps in the analysis of how statements are fundamentally constructed. A prerequisite for proving something is that one has understood what it actually is that is to be proved. The nicely disassembled predicate-logical form can often give hints on how the proof is to be organised.

The logical rules for rewriting can then also be of help. It might for instance be the case that it's a lot easier to prove the statement "if a graph has an Eulerian circuit then all nodes have an even degree" than to prove "no graph exists that has an Eulerian circuit and nodes of an odd degree", and since the statements are equivalent you can choose to prove the version that seems easier. Very often the first step in a proof consists of rewriting the statement into an equivalent but more easily proved form, after which the rewritten statement is proved.

7.5.1 Direct and Indirect Proofs

There are two main types of proofs: **direct proofs** and **indirect proofs**. In a direct proof you explain "it's like this, because of...". In an indirect proof, also called **proof by contradiction**, you explain "it's like this, because it can't be in any other way". An example of an indirect proof: "It can't have been this Sunday that we were there, since if it had been Sunday the bank would have been closed, and it wasn't!"

Direct proofs are usually to be preferred, but it's not always possible to make them. The nice thing about indirect proofs is that you are more at liberty about the conclusion you want to reach. In a direct proof, you are supposed to conclude That Which Was To Be Proved, and nothing else. In an indirect proof you want to conclude something absurd, never mind what.

7.5.2 Proof Strategies

Here follows a short summary of suitable ways of organising proofs, based on the structure of the statements.

Most statements are more or less composite. If for instance you want to prove a for-all statement, where the main operation is implication, with a first part that is a disjunction and a second part that is a conjunction, you have to use almost all the tricks here one at a time. You choose your strategy based on the *main operation* in the part of the statement that is on the table at the moment, which usually results in having to prove some smaller part of the statement, and then you have to choose a strategy for that part.

Implication Since implication is the most common structure of all we take it first.

An implication is always classified as true if the first clause is false. Because of this, there is no reason to investigate that case, and it's what happens if the first clause is *true* that is of interest. So you start by assuming that the first clause is true, and then try to prove the second part, and in this proof you are at liberty to use everything included in the first clause. (As a matter of fact, you usually *have* to use it, since nobody would have bothered to write it down if it wasn't relevant.)

For-All Statements One method: prove the statement for some object chosen completely at random, where we don't presume any special properties except for the ones given in the statement. If the statement can be proved for this object, then the same line of argument must be applicable to all other objects as well, and the statement thereby holds for all the objects, which was to be proved.

Alternatively, you can resort to proof by contradiction, and assume that there is an object to which the statement doesn't apply, after which you prove that the consequences are absurd.

> Exercise 7.66 Find some place in this or some other book where something is proved according to the first method, and one where something is proved according to the second one. *

Existence statements One method: find an object that fits into the statement, and point to that. Then it's obvious that it exists. "$x = 2$ is a solution of the equation $x^3 - 6x^2 + 8x = 0$. Then a solution of the equation exists!" (This is called a **constructive proof**.)

Another method: explain by reasoning why an object that fits into the statement has to exist. "$x^3 - 6x^2 + 8x$ is -48 when $x = -1$ and 15 when $x = 5$ and then the expression has to be zero for some number inbetween, and thus the equation $x^3 - 6x^2 + 8x = 0$ has to have a solution!"

A third method is proof by contradiction, where you assume that the statement is a lie for all the objects, after which you show that the consequences are absurd.

> Exercise 7.67 Find some place where something is proved according to the first method, some place where something is proved according to the second method, and some place where something is proved according to the third one. *

Conjunctions Simple. Prove the two parts one at a time, independently.

Disjunctions Use that $p \vee q \Leftrightarrow \neg p \to q$, and prove the implication instead, using the tricks for implications.

Alternatively, use proof by contradiction and De Morgan's law, and show that $\neg p \wedge \neg q$ leads to something absurd.

> Exercise 7.68 Find some place where something is proved according to the first method and one where something is proved according to the second one. *

Equivalences Use $p \leftrightarrow q \Leftrightarrow (p \to q) \wedge (q \to p)$, and prove the two implications one at a time.

> Exercise 7.69 Find some place where an equivalence is proved. *

7.6 More Exercises

7.6.1 Routine Work

Propositional Logic

Exercise 7.70 Determine for each of the following expressions whether it's a tautology, a contradiction, or neither.

(a) $(\neg q \wedge p) \vee (p \to q)$

(b) $(p \wedge q) \vee (\neg q \to q) \vee p$

(c) $(p \wedge q) \wedge \neg\big((p \leftrightarrow q) \vee (p \vee q)\big)$

(d) $\big((p \vee q) \leftrightarrow q\big) \wedge \big(\neg q \wedge (p \to q)\big)$

(e) Convert the expression that is a tautology to 1, using the rules for rewriting, and the one that is a contradiction to 0.

Exercise 7.71 Determine for each of the following expressions whether it's a tautology, a contradiction, or neither.

(a) $(p \vee q) \wedge p \wedge \big((p \wedge q) \vee q\big)$

(b) $p \wedge (q \vee p) \wedge q \;\leftrightarrow\; \neg p \vee \neg q$

(c) $\neg(p \wedge \neg q) \;\to\; (\neg q \to \neg p)$

(d) $(p \vee q) \wedge \neg q \;\to\; \neg(p \vee \neg q)$

(e) Convert the expression that is a tautology to 1, using the rules for rewriting, and the one that is a contradiction to 0.

Boolean Algebra

Exercise 7.72 Study $f_1(x, y, z) = (x + \overline{yz})\big(x(\overline{x} + z) + y\big)$.

(a) Write down the table of values for f_1.

(b) Rewrite f_1 to conjuntive and disjunctive form, without using the table.

(c) Write f_1 in conjunctive and disjunctive normal form in any way you like.

(d) If you want to: draw a circuit that computes the function.

Exercise 7.73 Do the same things as in the previous exercise, but for $f_2(x, y, z) = xy + xz + (x + y)\overline{x}\,\overline{y}$.

Predicate Logic

Exercise 7.74 Our universe consists of the integers between 0 and 9. We have the predicates $P_2(x)$, that means "x is divisible by 2", and $P_3(x)$, that means "x is divisible by 3". Determine whether the following statements are true or false:

(a) $P_2(0)$ **(b)** $P_3(1)$ **(c)** $\exists x\, P_2(x)$ **(d)** $\forall y\, P_3(y)$

(e) $\exists z(P_2(z) \wedge P_3(z))$ **(f)** $\forall w(P_2(w) \vee P_3(w))$

Exercise 7.75 We have the following predicates: $Z(x)$: "x is a number in $\mathbb{Z}_7$"; $N(x)$: "x is the number zero (nought)", and $I(x, y)$: "x is the inverse of y". Translate the following (to top it all: true) sentences to predicate-logical notation:

(a) "All numbers in $\mathbb{Z}_7$ except for zero have an inverse."

(b) "There is a number in $\mathbb{Z}_7$ that is its own inverse".

(c) "There are numbers in $\mathbb{Z}_7$ that have an inverse but there are numbers in $\mathbb{Z}_7$ that don't have an inverse as well."

(d) "Zero is the only number in $\mathbb{Z}_7$ that doesn't have an inverse".

(Hint: Use longer names on the predicates while working.)

Exercise 7.76 We have the following predicates: $M(x)$: "x is a set". $S(x, y)$: "x is a subset of y". $E(x, y)$: "x is an element in y", and $D(x, y)$: "x and y are disjoint". Translate the following sentences to predicate-logical notation.

(a) "Every set is a subset of itself".

(b) "There is a set that is a subset of all sets". (Which one, by the way?)

(c) "If a set is a subset of a set, then all the elements in the first one also belong to the second one".

(d) "If two sets are disjoint then there is no element that belongs to both sets".

Exercise 7.77 We have the predicates $G(x)$: "x is a graph", $E(x)$ "x has an Eulerian circuit", $C(x)$: x is connected. What is

$$\forall x(G(x) \wedge E(x) \to C(x))$$

in normal language? Is the statement true, by the way?

Exercise 7.78 What is

$$\forall x(G(x) \wedge \neg C(x) \to \neg E(x))$$

in normal language? Is it true? (The same predicates as in the previous exercise.)

Exercise 7.79 Rewrite the following expression to prenex form (not that we think that prenex form is especially good, but is a good way to practise the rules):

$$\neg\{\forall x\, P(x) \ \wedge\ \exists x(Q(x) \wedge R(x))\}$$

Exercise 7.80 Rewrite the following expression to prenex form:

$$\neg(\forall x\, P(x) \ \to\ \exists x\, Q(x))$$

Proof Technique

Exercise 7.81 Study the following logical equivalence:

$$p \vee q \to r \ \Leftrightarrow\ (p \to r) \wedge (q \to r)$$

(a) Check that it's true.

(b) Invent a statement that has the logical structure of the left-hand side, and check how it sounds reformulated according to the right-hand side.

(c) What consequences can this logical equivalence have when working out proofs?

Exercise 7.82 Study the following logical equivalence:

$$p \to (q \to r) \ \Leftrightarrow\ p \wedge q \to r$$

(a) Check that it's true.

(b) Invent a statement that has the logical structure of the left-hand side, and check how it sounds reformulated according to the right-hand side.

(c) What consequences can this logical equivalence have when carrying out proofs?

7.6.2 To Ponder About

Exercise 7.83: ***Application to Programming*** In this chapter we have claimed that two logical expressions are equivalent if they have the same truth-table. Is that really true in practice? Can you think of some context connected to computers where not just the truth-table but also the form of the expression matters?

Exercise 7.84 In exercise 2.20 on page 19 a fairly extensive argument is carried out. Write down this argument using logical symbols instead of words.*

Exercise 7.85 In exercise 6.98 on page 171 a trail that isn't a path is to be found. In pedagogical circles a common saying is that you should express yourself positively; you should say what somehing should be and not what it mustn't be. Express the meaning of a trail that isn't a path using logical symbols, and rewrite the expression into something without negations. Include all the steps!

Exercise 7.86 The expression $(p \vee q) \wedge \neg p \rightarrow q$ is a tautology.

(a) Check that it really is a tautology in some way.

(b) Invent a *sensible* sentence having this structure.

(c) The line of argument here has a name. Which? (The only place where this name is included in this book is in the answer to this exercise, so there is no point in searching for it.)

Exercise 7.87 The expression $\exists x(P(x) \wedge Q(x)) \rightarrow \exists x\, P(x) \wedge \exists y\, Q(y)$ is a tautology.

(a) What does the expression actually say?

(b) Explain why it's true.

(c) Does the reversal of the expression hold as well?

Exercise 7.88

(a) From De Morgan's laws the generalised De Morgan's laws can be derived:

$$\neg(p_1 \wedge p_2 \wedge \cdots \wedge p_n) \Leftrightarrow \neg p_1 \vee \neg p_2 \vee \cdots \vee \neg p_n$$

and

$$\neg(p_1 \vee p_2 \vee \cdots \vee p_n) \Leftrightarrow \neg p_1 \wedge \neg p_2 \wedge \cdots \wedge \neg p_n$$

Prove these using induction! (You can presume that the ordinary De Morgans laws are proved.)

(b) The rules for negation of quantifier expressions (se page 204) can be seen as an extension of these arguments. Explain how!

Exercise 7.89: *The Principle of Induction* The principle of induction is used precisely to prove that a statement (say $P(x)$) holds in all cases. Let $S(x)$ mean "x is the simplest case" and $N(y, x)$ "y is the case next after x". Express the principle of induction using predicate-logical symbols.

8 Relations and Functions

Mathematical **functions** are probably well known from high school. Functions are the mathematical way of modelling the relationship between one quantity (in-value, "x-value") and another (out-value, "y-value"). We will here make a thorough going through of the concept of function, which is a special case of mathematical **relations** where one x-value may be related to several different y-values.

Highlights from this chapter.

- Examples of modelling using relations.
- Representation of a relation $\mathcal{R}$ using graphs and matrices.
- *Composition* $\mathcal{R}_1 \circ \mathcal{R}_2$ of two relations.
- Examples of relations that are *reflexive* (where $x\,\mathcal{R}\,x$ always holds), *symmetric* (where $x\,\mathcal{R}\,y$ leads to $y\,\mathcal{R}\,x$), *anti-symmetric* (where $x\,\mathcal{R}\,y$ and $y\,\mathcal{R}\,x$ only hold at the same time if $x = y$), or *transitive* (where $x\,\mathcal{R}\,y$ and $y\,\mathcal{R}\,z$ combined lead to $x\,\mathcal{R}\,z$).
- Relations that are both reflexive, symmetric, and transitive, which are called *equivalence relations*, such as $=$, $\equiv$, and "has the same length as".
- Relations that are both reflexive, anti-symmetric, and transitive, which are called *partial orders*, such as $<$, $\subseteq$, and "boss of".
- Functions from one set to another, $f : A \rightarrow B$, where the set A of possible in-values is called the *domain* of f and the set B of possible out-values is called the *codomain* of f.
- The *composition* $f_1 \circ f_2$ of two functions, defined as $f_1 \circ f_2(x) = f_1(f_2(x))$.
- Examples of functions that are *surjective* (where each value in B is used as a function value at least once), *injective* (where each value in B is used as a function value at most once), or *bijective* (where each value in B is used as a function value exactly once).
- Bijective functions are *invertible*. If $f : A \rightarrow B$ is a bijective function its inverse $f^{-1} : B \rightarrow A$ can be defined by $f^{-1}(f(x)) = x$.

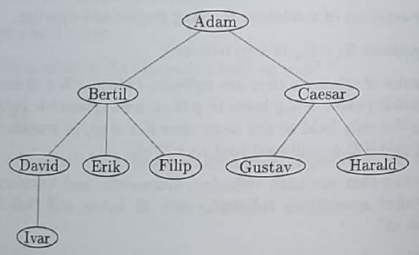
8.1 Relations

The word "relation" most people probably know – on the other hand it may feel a bit surprising to encounter it in mathematics. (It's usually seen as some form of sociological concept.)

There are many kinds of relations; it's for instance possible to talk about "sibling relations" and "love relations". If you are speaking about sibling relations you can pick a pair of persons, and then answer "yes" or "no" to the question if they have such a relationship.

This is exactly what is meant by relation in mathematics as well. If you have a **binary relation on a set**, you can for each pair of elements/individuals (a, b) from the set say "Yes, a has that relation to b" or "No, a doesn't have that relation to b". (Usually the word "binary" is omitted and just "relation" used, and that's what we will do henceforwards. But it is possible to imagine relations among for example triples instead of pairs.)

Example 8.1: ***Family Tree*** Here we have the family tree of the male individuals in the Adamson family:



Using this family tree we can define lots of different relations on this set of persons: *son-of*, *cousin-of*, *grandfather-of* are just some examples.

We will return to this family repeatedly in this chapter. ∎

Even if the *word* "relation" may feel new in mathematics, the *concept* is in fact well known. Here is a list of some mathematical relations that you ought to know: *equal-to*, *parallel-to*, *greater-than*, *perpendicular-to*, *not-equal-to*, *divisor-of*, ... All these concepts have the property that if you take two elements from the set on which they are defined, either the first of them is related to the second one, or it *isn't*. 2 is a divisor of 4, the floor is parallel to the ceiling, 3 isn't greater than 17, and so on.

Exercise 8.1 State three everyday and three mathematical relations.

Most relations used in mathematics have symbols of their own. The given examples have the symbols $=$, $\parallel$, $\subseteq$, $>$, $\perp$, $\neq$, and $|$. All the symbols except maybe the ones representing *parallel-with* and *perpendicular-to* ought to be well known. When henceforwards we are talking about an arbitrary relation we will call it $\mathcal{R}$. "a is related to b" is written $a \mathcal{R} b$. "a is *not* related to b" can then be written as $a \not\mathcal{R} b$ or $\neg(a \mathcal{R} b)$. (We may note that $a \mathcal{R} b$ can be treated as a proposition. From a logical point of view, a relation is the same thing as a dyadic predicate.)

8.1.1 Different Ways of Representing Relations

There are different ways of representing a relation. One method is to reel off the pairs that are related. If we look at the Adamsson family in example 8.1 on the facing page, the relation *son-of* can be described as the pairs $\{ (b, a), (c, a), (d, b), (e, b), (f, b), (g, c), (h, c), (i, d) \}$. (b, a) then represents "Bertil is the son of Adam".

If the set is infinite, writing the whole list isn't possible, and one has to resort to the set-builder. The divisor-relation on the set of natural numbers can be described as the set

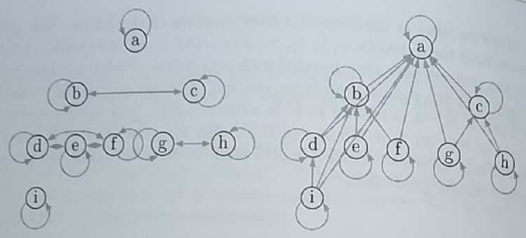
$$\{ (a, b) \mid a, b \in \mathbb{N} \ \wedge \ \exists c (c \in \mathbb{N} \ \wedge \ b = a \cdot c) \}$$

We may note that the pairs of which the relation consists make up a subset of the ordered pairs from the set. Some books even define *relation* as "a subset of the ordered pairs", but we think that this mostly seems like an effort to make a simple and well-known concept appear very mysterious.

Another way of representing the relation is by drawing a **relation graph**. Then there is one node for each element in the set, and an arrow from a to b is a is related to b. This will thus be a directed graph, perhaps including loops. The family tree in the example is such a graph, for the relation *son-of* (if we decide that all the lines have an implied arrowhead at the upper end. If we decide that the arrowhead is to be placed at the lower end, the graph shows *father-of*).

A third way of representing the relation is to write its **matrix**, which simply means using the matrix representation of the graph (see page 152). That is the normal way of representing relations when using computers. Here the same complication as when writing the matrix for a graph arises, namely that you have to decide in which order the elements are to be given. In the case of the Adamson family, alphabetical order seems natural, but in other cases it can be more complicated. The matrix may look completely different depending on how one chooses to proceed.

Example 8.2: ***Relationships*** We draw the graphs showing the relations *have-the-same-father-as* and *is-a-descendant-of* in the Adamson family. Here we have a problem: are you a descendant of yourself or not? That's not clear, but we choose to answer "yes" to the question.



The pictures are a good illustration of a problem with graphs: The pictures are often messy. And then we have actually rationalised, and drawn arrows with heads at both ends instead of one arrow in each direction for the pairs that were mutually related!

The matrices of the two relations are (if we take the persons in alphabetical order)

$$\mathsf{M}_1 = \begin{pmatrix} 1&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0 \\ 0&0&0&1&1&1&0&0&0 \\ 0&0&0&1&1&1&0&0&0 \\ 0&0&0&1&1&1&0&0&0 \\ 0&0&0&0&0&0&1&1&0 \\ 0&0&0&0&0&0&1&1&0 \\ 0&0&0&0&0&0&0&0&1 \end{pmatrix} \qquad \mathsf{M}_2 = \begin{pmatrix} 1&0&0&0&0&0&0&0&0 \\ 1&1&0&0&0&0&0&0&0 \\ 1&0&1&0&0&0&0&0&0 \\ 1&1&0&1&0&0&0&0&0 \\ 1&1&0&0&1&0&0&0&0 \\ 1&1&0&0&0&1&0&0&0 \\ 1&0&1&0&0&0&1&0&0 \\ 1&0&1&0&0&0&0&1&0 \\ 1&1&0&1&0&0&0&0&1 \end{pmatrix}$$

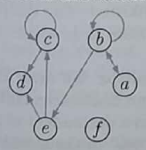
The 1 in the first column of the first row states for instance that the first person (Adam) is related to himself. The 0:s after that state that he isn't related to anyone else. The bottom row of the second matrix states that the last person (Ivar) is related to the first, second, and fourth person, and to himself. ∎

Exercise 8.2 Draw the graph and the matrix of the relations *cousin-of* and *grandfather-of* in the Adamson family. *

Exercise 8.3 How many edges will there be in the graph showing the relation *related-to* in the Adamson family, and what does the matrix look like? (You don't have to draw them.)

Exercise 8.4 Draw the graph and the matrix for the relation $\mathcal{R} = \{(a,c), (a,d), (b,b), (c,a), (c,c), (d,b), (d,e)\}$.

Exercise 8.5 Which pairs does this relation consist of?



8.1.2 Relations Between Sets

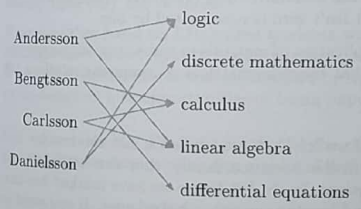
Up until now, we have been talking about relations *on a set*. It's also possible to study **relations between sets**. We can for instance say that set A is all the students at the university, and set B all the courses in mathematics given at the university. A relation between A and B is *is-taking-the-course*, another is *has-passed-the-course*.

A relation between two sets has a *bipartite* graph.

Example 8.3: ***Distribution of Courses*** The teachers Andersson, Bengtsson, Carlsson, and Danielsson have been asked to say what courses they want to teach next year. Each teacher was allowed to choose two courses. The list looked like this:

Andersson:	logic, linear algebra
Bengtsson:	calculus, differential equations
Carlsson:	calculus, linear algebra
Danielsson:	logic, discrete mathematics

The graph showing the relation *wants-to-teach* can be drawn as



∎

Exercise 8.6

(a) Write down the matrix of the relation *wants-to-teach* in the example.

(b) Can you draw the graph in a somewhat more elegant way?

(c) What does the matrix look like if you take the nodes in the order given in your drawing?

8.1.3 Composite Relations

If we have two relations we can link them. If we link the relation *has-the-brother* to the relation *has-the-son* we get the relation *has-the-nephew*. That is called a **composite relation**. If the original relations are called $\mathcal{R}_1$ and $\mathcal{R}_2$ the composite relation is denoted $\mathcal{R}_1 \circ \mathcal{R}_2$. Here is the formal definition of composite relation:

Definition 8.1: ***Composite Relation***

$$\forall x \forall y \big(x \, \mathcal{R}_1 \circ \mathcal{R}_2 \, y \;\leftrightarrow\; \exists z (x \, \mathcal{R}_1 \, z \;\wedge\; z \, \mathcal{R}_2 \, y) \big)$$

That means that x and y are related according to $\mathcal{R}_1 \circ \mathcal{R}_2$ if and only if there exists an intermediate element z, such that you can get from x to z using $\mathcal{R}_1$ and then onwards from z to y using $\mathcal{R}_2$.

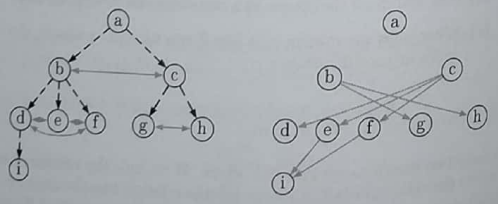
(This was an example of a case where the predicate-logical notation may be more easily read than the corresponding expression in normal language!) ∎

Another example of a composite relation is if you have the relation *studies-the-course* from students to courses and another relation *uses-the-book* from courses to books. If we put them together we get the relation *studies-the-book* from students to books.

Theorem 8.1: ***The Matrix of a Composite Relation*** You get the matrix $\mathsf{M}_{\mathcal{R}_1 \circ \mathcal{R}_2}$ of the composite relation $\mathcal{R}_1 \circ \mathcal{R}_2$ by multiplying the matrices $\mathsf{M}_{\mathcal{R}_1}$ and $\mathsf{M}_{\mathcal{R}_2}$ according to the standard rules of matrix algebra, with the addition that everything that isn't zero is represented by one.

(If you haven't studied multiplication of matrices in some previous course you have better skip everything that concerns matrices of composite relations.) ∎

Example 8.4: ***Nephews*** If we let $\mathcal{R}_1$ be the relation *has-the-brother* and $\mathcal{R}_2$ the relation *has-the-son* in the Adamson family, they can be linked and form the relation *has-the-nephew*. In the left figure we have marked *has-the-brother* using blue arrows, and *has-the-son* using dashed ones. If you want to get from a person to his nephew, you start by following a blue *brother*-arrow, after which you continue along a *son*-arrow. In the figure on the right the *nephew*-arrows are drawn.



We get the matrix of *has-the-nephew* by multiplying the matrices of brother

and son according to

$$\mathsf{M}_{\mathcal{R}_1 \circ \mathcal{R}_2} = \mathsf{M}_{\mathcal{R}_1} \mathsf{M}_{\mathcal{R}_2}$$

$$= \underset{\textit{has-the-brother}}{\begin{pmatrix} 0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&1&0&0&0&0 \\ 0&0&0&1&0&1&0&0&0&0 \\ 0&0&0&1&1&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \end{pmatrix}} \underset{\textit{has-the-son}}{\begin{pmatrix} 0&1&1&0&0&0&0&0&0&0 \\ 0&0&0&1&1&1&0&0&0&0 \\ 0&0&0&0&0&0&1&1&0&0 \\ 0&0&0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \end{pmatrix}} = \underset{\textit{has-the-nephew}}{\begin{pmatrix} 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&1&1&0 \\ 0&0&0&1&1&1&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \end{pmatrix}}$$

∎

Exercise 8.7 Check that the matrix we got by the multiplication corresponds to the graph showing the *nephew*-relation. *

When multipliying the matrices in the example we got only zeros and ones in the composite matrix, without having to resort to "everything except zero is represented by one". That's because there is only *one* way of being the *nephew* of somebody. (The connection has to go via a father, and people usually have only one of those.)

If on the other hand we had paired students with exam-designers via exams we might have ended up with higher numbers, since it's entirely possible that a student writes several exams all designed by the same designer. (That gives exam-designers nice opportunities of being impopular several times over.)

Exercise 8.8 Find the matrix of the relation *grandfather-of* in the Adamson family using the matrix of *father-of*. *

Exercise 8.9 We have three sets: $\{a, b, c, d\}$, $\{A, B, C, D, E\}$ and $\{\alpha, \beta, \gamma, \delta\}$. In addition, we have the relation $\mathcal{R}_1 = \{(a, A), (a, C), (b, D), (c, C), (d, B)\}$ and the relation $\mathcal{R}_2 = \{(A, \alpha), (B, \gamma), (C, \alpha), (E, \delta)\}$. Find the composite relation $\mathcal{R}_1 \circ \mathcal{R}_2$, both by drawing a graph and picking out two-step connections, and by using the matrices. Check that you get the same answer.

8.1.4 Interesting Properties of Relations

Some hypothetical exchanges:

"Do you know her?" "Yes, she's my sister", "Oh, but if you are her sister you may...".

"How is it, is Pelle older or younger than Olle?" "Let's see, Pelle is younger than me, and Olle older, and then Pelle has to be younger than Olle."

Everyone has participated in conversations of this kind. But what lines of thought are they based on?

There are properties that some relations have, and if you know that a relation has a property that makes it possible to draw certain conclusions from given data (such as since a is a sibling of b then b has to be a sibling of a, or since O is older than and P younger than a certain person then P has to be younger than O).

Here definitions of some properties that relations may have will follow:

Definition 8.2: ***Reflexivity*** If the relation $\mathcal{R}$ has the property

$$\forall x\, x\, \mathcal{R}\, x$$

the relation is called **reflexive**.

Translated into normal language the definition means that all the elements are related to themselves. Examples of reflexive relations are *related-to*, *same-age-as*, *parallel-to*, and *divisor-of*. Examples of relations that *aren't* reflexive are *younger-than*, *son-of*, *perpendicular-to* and *greater-than*.

If the relation is reflexive, there will be a loop on every node in the graph, and the matrix has ones along the whole main diagonal. ∎

Exercise 8.10 Convince yourself that the examples given really have the properties stated. *

Exercise 8.11 Invent one everyday and one mathematical example of relexive relations, and another everyday and mathematical example of relations that *aren't* reflexive.

Exercise 8.12 Give one everyday and one mathematical example of situations where the fact that a relation is reflexive is used. (Not just a situation where a reflexive relation is used, but it has to be relevant that it is reflexive.)

Definition 8.3: ***Symmetry*** If the relation $\mathcal{R}$ has the property

$$\forall x \forall y (x\, \mathcal{R}\, y \to y\, \mathcal{R}\, x)$$

the relation is said to be **symmetric**.

That simply means that the relationship in all cases is mutual. *Sibling-of*, *in-the-same-form-in-school*, *parallel-to*, and *not-equal-to* are examples of symmetric relations. *father-of*, *shorter-than*, *proper-subset-of*, and *greater-than* aren't.

If the relation is symmetric all connections in the graph are bidirected, and the matrix of the relation is symmetric (that is, the same value is found in position (i, j) as in position (j, i)). ∎

Exercise 8.13 Convince yourself that the examples given really have the stated properties. *

Exercise 8.14 Invent one everyday and one mathematical example of symmetric relations, and another everyday and mathematical example of relations that *aren't* symmetric.

Exercise 8.15 Give one everyday and one mathematical example of situations where the fact that a relation is symmetric is used.

Definition 8.4: ***Anti-symmetry*** If the relation $\mathcal{R}$ has the property

$$\forall x \forall y (x\, \mathcal{R}\, y \wedge y\, \mathcal{R}\, x \;\to\; x = y)$$

the relation is said to be **anti-symmetric**. The definition given here is the standard one, but the following logically equivalent expression may be easier to understand:

$$\forall x \forall y (\neg (x\, \mathcal{R}\, y \wedge y\, \mathcal{R}\, x) \;\vee\; x = y)$$

Translated into English the second version of the definition means "the relationship is never mutual, except perhaps if the related objects are identical".

The first version can be read as "if the relationship is mutual then the only explanation is that the objects are identical". This is something that we, as a matter of fact, have already been using several times in this book (at which places you'll have to find out in exercise 8.20).

The relations *bosses-over* and *son-of* are anti-symmetric. (In the case of *son-of* we may note that it's possible to say "the relationship is never mutual and that's it", without having to use the opt-out "except perhaps if the objects are identical". But the opt-out doesn't do any harm, even if it is unnecessary in this case.) The relations *subset-of* and *greater-than* are anti-symmetric as well.

On the other hand, *brother-of*, *in-the-same-form-in-school*, *parallel-to* and *not-equal-to* are not anti-symmetric.

In a graph of an anti-symmetric relation there are no bidirected connections (if we don't count loops as such). If there is a one in position (i, j) in the matrix there will for sure be a zero on the other side of the main diagonal, in position (j, i). (This rule doesn't apply to ones on the main diagonal, where $i = j$. There can't be both a one and a zero in this position!) ∎

Exercise 8.16 Rephrase the predicate-logical definition in at least two more different ways, and contemplate how these variants of the definition are to be phrased in English.

Exercise 8.17 Convince yourself that the examples given really have the stated properties. *

Exercise 8.18 Invent one everyday and one mathematical example of anti-symmetric relations, and another everyday and mathematical example of relations that *aren't* anti-symmetric.

Exercise 8.19 Is the relation *divisor-of* anti-symmetric?

Exercise 8.20

(a) Find some place in this book where we use the fact that *greater-than-or-equal-to* and *subset-in*, respectively, are anti-symmetric.

(b) Give as well an everyday example of a situation where the fact that a relation is anti-symmetric is used.

Definition 8.5: ***Transitivity*** If the following holds for the relation $\mathcal{R}$

$$\forall x \forall y \forall z (x \mathcal{R} y \wedge y \mathcal{R} z \rightarrow x \mathcal{R} z)$$

the relation is said to be **transitive**. (Compare the concept "transit", which means to transport something from one country to another via a third.)

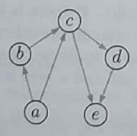
This means that if two objects are related *indirectly* (via another object) then they are related *directly*. It's that property of the *equal-to*-relation that is used when writing things like

$$2 + 3 \cdot 4 = 2 + 12 = 14$$

which is actually *two* equalities: $2 + 3 \cdot 4 = 2 + 12$ and $2 + 12 = 14$, from which the conclusion $2 + 3 \cdot 4 = 14$ is drawn.

Other examples of transitive relations are *descendant-of*, *younger-than*, *divisor-of*, and *parallel-to*. *Son-of*, *in-love-with*, *not-equal-to*, and *perpendicular-to* aren't.

If a relation is transitive all objects that are connected to each other via some other object are directly connected as well. (It's usually fairly difficult to verify this from a graph, though, since there tend to be lots of connections to check.)



The relation depicted here is for instance *not* transitive, since it's possible to get from b to d via c, while there is no direct connection. (That all other two-step connections have corresponding one-step connections doesn't help – if there aren't direct connections in *all* cases the relation isn't transitive.)

To determine whether a relation is transitive using its matrix you need the matrix showing the two-step connections, that is to say: the matrix of $\mathcal{R} \circ \mathcal{R}$, and then you check that in every place where it has a one there is a one in the matrix for $\mathcal{R}$ as well. (It doesn't matter if $\mathsf{M}_{\mathcal{R}}$ has ones in a number of extra places besides.) ■

Exercise 8.21 Convince yourself that the examples given have the stated properties. *

Exercise 8.22 Invent one everyday and one mathematical example of transitive relations, and another everyday and mathematical example of relations that *aren't* transitive.

Exercise 8.23 Give one everyday and one mathematical example of situations where the fact that a relation is transitive is used.

Exercise 8.24 Determine whether $\mathcal{R}_1 = \{ (a,a), (a,b), (a,c), (a,e), (b,a), (b,b), (b,c), (b,e), (d,c), (d,d) \}$ and $\mathcal{R}_2 = \{ (a,b), (a,c), (a,e), (b,a), (b,b), (b,c), (b,e), (d,c), (d,d), (e,e) \}$ are transitive.

Exercise 8.25 Write in English what it would mean if the relation *in-love-with* were

(a) reflexive

(b) symmetric

(c) anti-symmetric

(d) transitive.

Exercise 8.26 Write a table, where you check which of these relations are reflexive, symmetric, anti-symmetric, and transitive, respectively: *not-equal-to*, *parallel-to*, *subset-of*, *perpendicular-to*, *less-than*, *equal-to*, *divisor-of*, *equivalent-to* (modular arithmetic), *approximately-equal-to*, *greater-than-or-equal-to*, *in-love-with*.

Exercise 8.27 In all the examples where we have analysed the properties of relations we have been looking at relations on a set. Are these concepts (reflexivity and so on) at all meaningful for relations between sets?

8.1.5 Special Kinds of Relations

Many important relations combine several of the properties that we covered in the previous section. Two combinations are important enough to have been given names of their own.

Definition 8.6: ***Equivalence Relation*** An **equivalence relation**, a "same-value-as-relation", is a relation that is

1. reflexive
2. symmetric
3. transitive

A prime example of such a relation is *equal-to*. Another example is *in-the-same-form-in-school* on the set of students in an elementary school. A third example is the relation *has-the-same-father-as* on the Adamson family. ∎

If you look at the graph of an equivalence relation you find that it will consist of a number of separate components, where each component is a complete graph (see page 143), with loops on all the nodes as well. As an example we can take a look at the *has-the-same-father-as*-graph for the Adamson family, on page 219. That graph contains two K_1 (Adam and Ivar), two K_2 (Bertil–Caesar and Gustav-Harald) and one K_3 (David–Erik–Filip).

The elements belonging to a component form an **equivalence class**. All the elements in an equivalence class are related to all the elements in the same class, but not to any element outside it.

Exercise 8.28 If we study the students in an elementary school and the relation *in-the-same-form-in-school*, we have an equivalence relation. What do the equivalence classes look like?

Exercise 8.29 Modular arithmetic in actual fact consists of calculations using equivalence classes in $\mathbb{Z}$. Describe, using the set builder, the equivalence classes into which calculations modulo 3 partition the integers.

The second frequently used type is

Definition 8.7: ***Partial Order*** A **partial order** is a relation that is

1. reflexive
2. anti-symmetric
3. transitive

Examples of partial orders are the mathematical concept of *subset-of* and the relation *descendant-of* in the Adamson family. ∎

That this type of relation is called a partial *order* is because it can be used to order the elements in a set. If we look for instance at a graph showing subsets

(see page 24), we find the empty set at the bottom and the "full set" on top, and the remaining subsets placed according to the number of elements.

That it's called a *partial* order is because not all the objects are related at all. If we work with subsets we find for instance that $\{1,2\} \not\subseteq \{1,3\}$ but also $\{1,3\} \not\subseteq \{1,2\}$.

If it really is possible to compare *all* the objects in the set the order is called a **total order**. *Greater-than-or-equal-to*, $\geq$, is an example of such a one.

If we look at the graph of *descendant-of* on page 219 it looks very messy; it's hardly possible to see anything because of all the lines. Because of this, a modified version of the graph of a partial order has been created, where – for reasons of readability – the loops are excluded (they have to be imagined), and direct connections between nodes that are connected by a path are excluded as well. Futhermore the arrowheads are left out, and have to be imagined at the upper end. If we expose the graph of *descendant-of* to this treatment we simply get the family tree that we started with!

Other examples of this type of diagram are subset-graphs and divisior-graphs. Diagrams of this kind are called **Hasse diagrams**.

Exercise 8.30 Here we have a partial order of the set of all permutations of the numbers 1, 2, and 3.

$$\begin{aligned}\mathcal{R} = \{&(123,123),(123,132),(123,213),(123,231),(123,312),\\ &(123,321),(132,132),(132,231),(132,312),(213,213),\\ &(213,231),(213,312),(231,231),(312,312),(321,231),\\ &(321,312),(321,321)\}\end{aligned}$$

Draw the Hasse diagram of this partial order. Can you tell according to which principle it is made, by the way?

Exercise 8.31 *Subgraph-of* is a partial order. In exercise 6.17 on page 144 the number of subgraphs of a certain graph was analysed. Draw the Hasse diagram of the set of these subgraphs.

Exercise 8.32 If we take a closer look at the *equal-to*-relation, we find that it's both symmetric and anti-symmetric. This means that it's classified as a partial order, besides being an equivalence relation!

(a) What does the relation graph look like for the *equal-to*-relation?

(b) How is it possible that a relation can be both symmetric and anti-symmetric?

(c) Is it possible to find more relations that are both equivalence relations and partial orders?

8.2 Functions

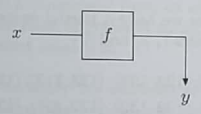
Everyone knows what a function is: it's somehing like "$y = f(x)$". You think until you get into a situation where you have to explain exactly what is meant.

A programmer usually thinks that functions are things of this kind:

```
float f(float x)
{
    float y;
    y = 17*x-8;
    return y;
}
```

But what is the *formal* definition?

If we look at the C-function that we defined, the way it works is that you put in an x-value of the real number type ("float"), and then a y-value is put out, also of the real number type. Which y that is put out depends on which x it was that was put in.



If we want to be even more particular, we may note that we always get the *same* y out for a certain x; it's not the case that $f(3)$ is sometimes 43 and sometimes something else, it's *always* 43. Furthermore, there belong y:s to *all* real x:s; none of them has to do without. (If we want to think strictly like programmers, we realise that sooner or later overflow will occur in the calculations, but that complication we'll pretend to be ignorant about.)

These observations combined give the formal definition of the concept of function:

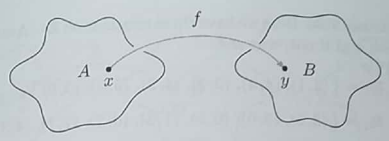
Definition 8.8: ***Function*** **A function** f from the set A to the set B is a rule that connects every element in A (that is, every possible x-value) to *one and only one* element in the set B (that is, one and only one y-value). ■

This is usually written as

$$f : A \longrightarrow B$$

The "rule" that connects the elements of the sets to each other is often but far from always a calculation formula.

The set A (from which the x-values are taken) is called a **domain**. The set B (that contains potential y-values) is called a **codomain**.



Quite often no-one has bothered to indicate either the domain or the codomain. Then the reader has to use their own judgement. In mathematics, where calculation formulas is the most common method for describing functions, it can be taken for granted that the domain consists of all the x-values for which the calculation can be carried out, and the codomain of all the y-values that can be the result.

Example 8.5 What seems to be a reasonable domain and codomain for this function?

$$f(x) = \sqrt{x}$$

Square roots are only defined for non-negative real numbers, and can be calculated for all of them. It thus seems reasonable to assume that the domain is $\mathbb{R}_+ \cup \{0\}$.

The square root of a positive real number is a positive real number, the square root of zero is zero, so it seems sensible to say that the codomain is $\mathbb{R}_+ \cup \{0\}$ as well. But it would be completely correct to choose the codomain $\mathbb{R}$; it doesn't matter that some potential y-values are "left over". ■

> Exercise 8.33 Suggest a domain and codomain for the functions sine and tangent.

> Exercise 8.34 Suggest a domain and codomain for the functions *has-the-mother* and *costs*.

If we look at the function $f : \mathbb{R} \longrightarrow \mathbb{R}$, $f(x) = x^2$, we find that we don't use all the values in the codomain $\mathbb{R}$. The negative numbers are left over, since the square of a real number is never negative. The part of the codomain that we actually *use*, in this case $\mathbb{R}_+ \cup \{0\}$, is called the **range** of the function.

> Exercise 8.35 What are the ranges of the functions in the two previous exercises?

A function $f : A \longrightarrow B$ can be described by a set of pairs (a, b) from $A \times B$, where each element $a \in A$ is found exactly once. (A table of values, in other words.) A relation from the set A to the set B can also be described using a set of pairs, but without any restrictions on how many times the elements may be found. That makes it possible to regard functions as a *special case* of relations.

Exercise 8.36 Here we have three relations on $\mathbb{Z}_7$. Are they functions or not, and if not, why not?

(a) $\mathcal{R}_1 = \{(2,1), (4,3), (0,2), (1,3), (5,0), (3,6)\}$

(b) $\mathcal{R}_2 = \{(3,6), (5,0), (6,5), (1,3), (0,2), (2,1), (4,3)\}$

(c) $\mathcal{R}_3 = \{(0,2), (5,0), (3,6), (2,1), (4,3), (1,3), (6,5), (2,4)\}$

Exercise 8.37: *Important!* What does the matrix for a relation that is a function look like?

Exercise 8.38 The gcd and lcm are two functions that we have been looking at earlier in this book. What are their domains?

8.2.1 Composite Functions

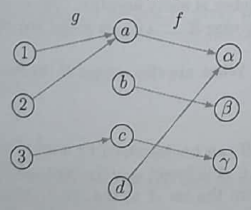
Functions are thus a special case of relations. But it took some time before anyone realised that, and during this time the notations used for functions and for relations had been developed independently. In the world of functions one writes $y = f(x)$ while in the world of relations the same thing is written as $x\,f\,y$ or $(x,y) \in f$. The end result is rather unfortunate when it comes to **composite functions**, which are otherwise just a special case of composite relations.

Example 8.6: ***Composite Function*** We have the set $A = \{1,2,3\}$, the set $B = \{a,b,c,d\}$, and the set $C = \{\alpha,\beta,\gamma\}$. Furthermore we have the function $g : A \longrightarrow B$ and the function $f : B \longrightarrow C$. The tables of values of the functions are

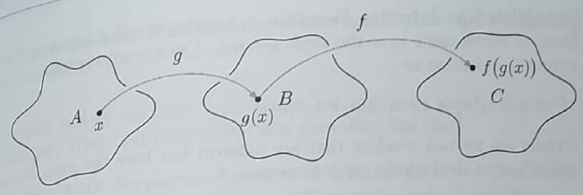
x	$g(x)$
1	a
2	a
3	c

x	$f(x)$
a	α
b	β
c	γ
d	α

The relationships can be illustrated in a graphical way as well:

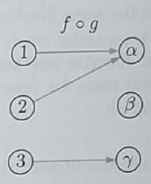


We can now form the composite function $f(g(x))$, which is read as "f of g of x".



That function has the table of values and graph below:

x	$f(g(x))$
1	α
2	α
3	γ



This can also be written as $(f \circ g)(x)$. So here $f \circ g$ means "first g, then f", the complete opposite of the meaning of the corresponding operation on relations! ■

Exercise 8.39 We have the functions $f_1(x) = x^2$ and $f_2(x) = 2x$. What is

(a) $(f_1 \circ f_2)(x)$ **(b)** $(f_2 \circ f_1)(x)$

(c) Does it seem to matter in which order the functions are taken?

8.2.2 Interesting Properties of Functions

Some hypothetical discussions:

"How many variables were there in that Boolean function?" "Let's see, the table had 8 rows, so the number of variables ought to have been 3."

"Who is it really that owns this apartment?" "Don't know, but it has to be *somebody*!"

How does one know that the question about the variables has only *one* answer, and that the second question can be answered at all?

Apparently function can have certain practical properties that are often used without thinking. (And that are sometimes used even in situations where they don't apply!) Let's go through some common properties of this kind.

We start with the case with the variables:

Definition 8.9: ***Injective Function*** A function is called **injective** if different x values can't have the same y-value. Using predicate logic one can write the definition as

$$\neg\exists x_1 \exists x_2 \big(x_1 \neq x_2 \ \wedge \ f(x_1) = f(x_2)\big)$$

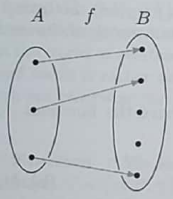
("there are no two x-values that are different but have the same function value") or on the logically equivalent way

$$\forall x_1 \forall x_2 \big(f(x_1) = f(x_2) \ \rightarrow \ x_1 = x_2\big)$$

("If two x-values have the same function values they have to be equal.") ∎

Exercise 8.40 Rewrite one version of the definition into the other using the logical rules for rewriting, and translate the intermediate results into English.

The concept can be illustrated with the following figure:



An injective function is "nicely combed"; the arrows never converge into one point. From a practical point of view, this means that the equation $y = f(x)$ will never have several different solutions. (On the other hand, nothing stops it from being unsolvable.)

Sometimes injective functions are called **injections**, but not very often, since that name sounds rather more medical than mathematical.

Exercise 8.41

(a) Is $f(x) = x^3$ injective when working in $\mathbb{Z}_7$?

(b) Is "cube root" a meaningful concept in $\mathbb{Z}_7$?

Exercise 8.42 Is the function *has-the-personal-identity-number* from the set of Swedes to the set of numbers injective? Is the function *is-named* from the set of Swedes to the set of letter-combinations injective?

Exercise 8.43 Do you have any idea about how to check whether a function is injective in calculus?

Exercise 8.44 What does the relation matrix of an injective function look like?

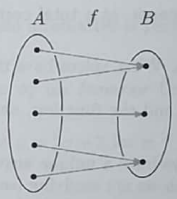
Now we'll look at the other case, where the argument was "it has to be something". When is it possible to claim something like that?

Definition 8.10: ***Surjective Function*** A function $f : A \longrightarrow B$ is called **surjective** if all the available y values in B are really used. In symbols:

$$\forall y \big\{ y \in B \ \rightarrow \ \exists x \big(x \in A \ \wedge \ f(x) = y\big)\big\}$$

("For every y-value in the codomain there exists an x-value in the domain that has the said y as its function value.") ∎

Here this figure fits:



Surjective functions are sometimes said to be "onto", since things land onto all the elements in B. (Compare the French preposition "sur", which means precisely "on".) The concrete consequence of this is that the equation $y = f(x)$ is solvable for all possible values of y.

Surjective functions are also called **surjections**.

Exercise 8.45 Is the function *has-the-personal-identity-number* surjective? Is the function *is-named* surjective?

Exercise 8.46 Determine whether the following functions are injective and/or surjective. Explain! (Skip (d) if you haven't studied complex numbers.)

(a) $f_1(x) = x^4$, $f_1 : \mathbb{R}_+ \longrightarrow \mathbb{R}_+$ (b) $f_2(x) = x^4$, $f_2 : \mathbb{Z}_+ \longrightarrow \mathbb{Z}_+$

(c) $f_3(x) = x^4$, $f_3 : \mathbb{Z} \longrightarrow \mathbb{Z}$ (d) $f_4(x) = x^4$, $f_4 : \mathbb{C} \longrightarrow \mathbb{C}$

Exercise 8.47 Do you have any idea about how to check whether a function is surjective in calculus?

Exercise 8.48 What does the relation matrix of a surjective function look like?

Being injective or surjective are useful properties of a function. If a function is *both* injective and surjective it's thus twice as useful. Such a function is called **bijective**, or a **bijection**. (We have met this name before in this book!) A bijective function has the property that the equation $y = f(x)$ has one and only one solution for each value of y. Concretely put, this means that the elements in A and B have been paired.

8.2.3 Inverses of Functions

Quite often you end up in a situation where you know the answer but need the question. Examples:

> The square of the hypotenuse is this. What is the hypotenuse?
>
> The student in question has this personal identity number. Which student is it?
>
> I bought 13 rolls of wallpaper, at a total cost of 1690 crowns. What does one roll cost?
>
> This dish tastes like this. What spices have they added?

Then you have the y-value and the function, and want to know which x it was that generated this y.

This question doesn't always have a unique answer, or for that matter, any answer at all. If the function we are studying isn't injective we can't be sure to find a unique x; there may be several. If the function isn't surjective we can't even be guaranteed that there is any x that fits. To be precise, it's only when the function is bijective that we can be sure that everything works out.

Exercise 8.49 We have the function $f(x) = \cos x$.

(a) Our y-value is 0.5. What problems occur when we want to find the matching x-value?

(b) We want instead to find the x-value that belongs to the y-value 2. What problems occur now?

(c) Is the function $f(x) = \cos x$ injective? Is it surjective? Is it bijective?

To find the x that belongs to a certain y we have to find something that does the exact opposite of the operation that the function f performs. The opposite of an operation is usually called the **inverse** of the operation. The **inverse function** to the function f is denoted f^{-1}. This inverse function f^{-1} is thus defined by the fact that it performs the opposite operation of f, that is to say that $f^{-1}(f(x)) = x$.

Exercise 8.50 Name some context where you have previously heard the word *inverse*.

Exercise 8.51 What would you say are the inverse operations of the following operations?

(a) addition (b) multiplication (c) exponentiation

Exercise 8.52: *Important!* What is the inverse of the inverse of a function?

Exercise 8.53 What do you get if you calculate the composite function $f(f^{-1}(x))$?

So only bijective functions have inverses. If you want to show that a function has an inverse there are then two ways to proceed.

1. Show that the function is both surjective and bijective.
2. Calculate the inverse. If that's possible it exists!

What we mean by the second alternative is perhaps best explained using an example.

Method 8.1: ***Calculation of an Inverse*** Does the function $f(x) = 2x + 3$ have an inverse?

We try to solve the equation $y = 2x + 3$ for x:

$$\begin{aligned} y &= 2x + 3 \\ y - 3 &= 2x + 3 - 3 \\ y - 3 &= 2x \\ \frac{y-3}{2} &= \frac{2x}{2} \\ \frac{y-3}{2} &= x \end{aligned}$$

(To make us able to feel sure that nothing wierd has taken place during the calculation we have printed *all* the details.)

Nowhere in the calculation have we had any problems with ambiguities, so the function is injective. The resulting expression can be calculated for every y (at least if we are working with real numbers – if we use integers, the whole thing gets more problematical), so the function is surjective. The function $f : \mathbb{R} \longrightarrow \mathbb{R}$ is thus bijective.

If we want to, we can write the end result as

$$f^{-1}(y) = \frac{y-3}{2}$$

We may add that some persons don't like calling the input of a function (in this case of the function f^{-1}) y – they prefer the name x. These persons would instead write

$$f^{-1}(x) = \frac{x-3}{2}$$

It doesn't really matter what you write – the variable is after all just a marker telling us where the actual value is to be inserted. ∎

Exercise 8.54 If the function f in the demonstration of the method had been defined for integers, which values of y would then have been problematical?

8.2.4 The Number of Functions of Different Kinds

This last section is very important. Furthermore, it is of a kind where you learn most if you answer the questions yourself. Because of this, the whole section consists of exercises. Anyone who wants to cram this material afterwards will have to peek into the answer section.

In all these exercises we are talking about two finite sets A and B, where the number of elements in A is $|A|$ and the number of elements in B is $|B|$.

Exercise 8.55: *Important!*

(a) How many functions from A to B are there? Explain!

(b) How many *injective* functions from A to B are there? Explain!

(c) How many *surjective* functions from A to B are there? Explain!

(d) What does it really take to make it possible to create a bijective function from A to B? Explain!

(e) How many different bijective functions are there from A to B? Explain!

(f) If a function $f : A \longrightarrow B$ is surjective, is it then bijective or not? The same question about injective functions.

(g) If $f : A \longrightarrow A$ is surjective, is it then injective? And if it's injective, is it then surjective?

(h) Will the previous question have the same answer if A is infinite?

This may be the place to reread section 2.6.2 on page 28 once again, and contemplate the contents and see how the concepts that this chapter has covered applies there.

8.3 More Exercises

8.3.1 Routine Work

Relations

Exercise 8.56 Here is a relation: $\mathcal{R} = \{(a,a), (a,b), (a,c), (a,f), (b,a), (b,b), (b,c), (b,f), (c,a), (c,b), (c,c), (d,d), (d,e), (e,d), (e,e), (f,a), (f,b), (f,f)\}$.

(a) Draw the graph of the relation.

(b) Write down the matrix of the relation.

(c) Is the relation reflexive? (Explain!)

(d) Is the relation symmetric? (Explain!)

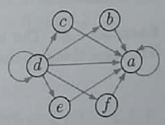
(e) Is the relation anti-symmetric? (Explain!)

(f) Is the relation transitive? (Explain!)

(g) Is the relation an equivalence relation? Is it a partial order?

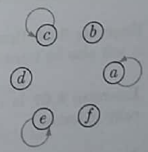
Exercise 8.57 Here is another relation: $\mathcal{R} = \{(a,a), (a,b), (b,b), (b,c), (c,c), (d,c), (d,c), (d,e), (e,e), (e,f), (f,a), (f,c), (f,f)\}$. Answer the same questions as in exercise 8.56.

Exercise 8.58 Here is the graph of a relation.



Answer the same questions as in exercise 8.56, but exchange the first questions for "list the pairs that the relation consists of".

Exercise 8.59 Here is the graph of a relation.



Answer the same questions as in exercise 8.56, but exchange the first question for "list the pairs that the relation consists of".

Exercise 8.60 Here is the matrix of a relation on the set $\{a,b,c,d,e,f\}$.

$$\begin{pmatrix} 1&1&0&0&0&1 \\ 0&1&0&0&0&0 \\ 0&0&1&0&0&0 \\ 0&0&1&1&0&0 \\ 0&0&1&1&1&0 \\ 1&1&0&0&0&1 \end{pmatrix}$$

Answer the same questions as in exercise 8.56, but exchange the second question for "list the pairs that the relation consists of".

Exercise 8.61 Here is the matrix of a relation on the set $\{a, b, c, d, e, f\}$.

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \end{pmatrix}$$

Answer the same questions as in exercise 8.56, but exchange the second question for "list the pairs that the relation consists of".

Functions

Exercise 8.62 All the common operations in arithmetic, such as addition, are functions. Explain clearly why addition is a function, and state the domain and range.

Exercise 8.63 State the domain and codomain of the function in example 7.6 on page 192.

Exercise 8.64 We are studying the function $f(x) = 2x$, $f : \mathbb{Z} \longrightarrow \mathbb{Z}$.

(a) Is f injective? **(b)** Is f surjective?

(c) What does the range look like?

Now assume that we are calculating modulo 9 instead, that is to say that f is defined as $f : \mathbb{Z}_9 \longrightarrow \mathbb{Z}_9$.

(d) Is f injective? **(e)** Is f surjective?

(f) What does the inverse of f look like?

(g) Does this have any connection to the concept of *inverse* in modular arithmetic?

Exercise 8.65 The composition of two surjective functions is surjective. The same thing applies to injective functions. Explain why!

8.3.2 To Ponder About

Exercise 8.66 Prove that if a relation is both symmetric and anti-symmetric it will also be transitive.

Exercise 8.67 Is a relation that is both symmetric and transitive atomatically reflexive?

Exercise 8.68 What would it mean if a *function* were

(a) reflexive? **(b)** symmetric?

(c) anti-symmetric? **(d)** transitive?

Exercise 8.69

(a) How many different relations are there in total on a set with n elements?

(b) How many out of these are reflexive?

(c) How many are symmetric?

(d) How many are anti-symmetric?

(e) How many are transitive?

(f) How many are equivalence relations?

(g) How many are partial orders?

Exercise 8.70 A set is (according to section 2.6.2 on page 28) *countably infinite* if it's possible to make a bijection from the set to $\mathbb{N}$. That means that it's just as infinite as $\mathbb{N}$. Does it seem possible for a set to be *less* infinite than $\mathbb{N}$? (That is to say, that the set is infinite but it's not possible to find a surjective function to $\mathbb{N}$?)

Exercise 8.71 The functiton NAND, which is described on page 198, what is the domain of that?

Exercise 8.72: *Total Order* A common situation which tends to result in a partial order is when one one looks through a set of tasks that have to be done for a whole job to be done. Some tasks have to be done before others (for instance you have to make the batter before baking the cake). Other tasks can be done in any order. (It doesn't matter if you grease the baking tin before breaking the eggs or afterwards.) If you want to sew a simple pair of trousers you may for instance get the partial order in figure 8.1.

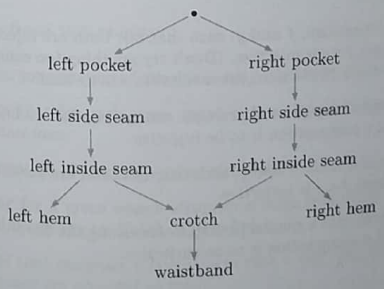


Figure 8.1: *Partial order for sewing trousers.*

When doing the job, one has to order the tasks in a total order (if one isn't able to do several things at the same time). This total order has to preserve the partial order, so that if a preceeds b in the partial order the same thing has to be true in the total order.

(a) Make a total order based on the partial order for sewing trousers.

(b) Describe an algorithm for solving the corresponding problem in general.

(c) Describe some real situation where a problem of this kind has to be solved.

(d) What is the difference between this situation and the one described in section 6.6.1 on page 165?

Exercise 8.73 Find four functions f_1, f_2, f_3, and f_4 from the natural numbers to the integers where

(a) f_1 is neither injective nor surjective.

(b) f_2 is injective but not surjective.

(c) f_3 is surjective but not injective.

(d) f_4 is both injective and surjective.

Explain clearly how it comes about that the functions have the properties they are supposed to have.

Exercise 8.74 When making composite functions it's true that the composition of two injective functions is injective, and the composition of two surjective functions is surjective, according to exercise 8.66 on page 240. But it is possible to get both injective and surjective functions *without* using two functions having these properties.

(a) Find two functions, f and g, such that not both are injective while the composition $f \circ g$ is injective. (Don't try anything too complicated! Sets with less than 10 elements are a suitable size to handle.)

(b) Try to formulate a general principle concerning the conditions that have to hold if a composition is to be injective.

(c) Find two functions, f and g, such that not both are surjective while the composition $f \circ g$ is surjective.

(d) Try to formulate a general principle concerning the conditions that have to hold if a composition is to be surjective.

9 Artificial Languages and Finite-State Machines

When we use devices (computer, telephone, dishwasher, coffee vending machine, etc.) we expect that they shall perform certain well-defined functions according to our instructions. Instructions for devices are often given by pressing buttons, which we can call an **artificial language**. This chapter describes how it's possible to model the function of a device (given its language of available instructions) using **finite-state machines**, which in their turn are modelled using graphs. In addition, we discuss how simple languages can be modelled in a mathematical way using so-called **regular expressions**.

Finite-state machines are in many situations a sufficient model, and can be implemented both in software and hardware, which you can learn more about in courses in computer science or electronics. To describe advanced calculating machines such as computers, finite-state machines are not adequate – then the theory has to be extended with so-called *Turing machines*, which are covered in courses in theoretical computer science.

The theory of languages that we will cover here has to be complemented with the theory for *grammars* to be directly applicable for compilers and word processors.

Highlights from this chapter.

- The basic construction of *machines* with *states* and *transitions*.
- Machines that for each *given input give a certain output*, so-called *Mealy machines*.
- Representation of machines using *state tables*.
- Searching for a given search string in a large text using a machine designed according to *Knuth-Morris-Pratt*.
- Machines that *recognise* a language in such a way that the words in the language are *accepted* as input.
- Natural languages and artificial languages.

- Concepts from the analysis of languages: *alphabet* (a set of symbols), *string* (a sequence of symbols from an alphabet), *language* (a set of accepted strings) and *words* (an accepted string in the language).
- *Concatenation* of words or of whole language. The concatenation of RABBIT and EARS becomes the new word RABBITEARS, the concatenation of the languages { RABBIT, DOG } and { EARS, FOOD } becomes the new language { RABBITEARS, RABBITFOOD, DOGEARS, DOGFOOD }.
- Operations on language: *multiplication* (by concatenation of languages as above) and *addition* (by union of the languages). For instance, { RABBIT, DOG } · ({ EARS } + { FOOD }) = { RABBITEARS, RABBITFOOD, DOGEARS, DOGFOOD }.
- The class of *regular languages*, which is defined as the languages that can be described using *regular expressions* containing symbols from an alphabet and the operations +, ·, and * (the so-called *Kleene star* that represents repetition an arbitrary number of times).
- A basic theorem: the regular languages are precisely the languages that can be recoginised by finite-state machines!

9.1 Finite-State Machines

Saying that a machine does someting **automatically** means that it performs some task on demand. (The word derives from the Greek word *automatos*, which means "that moves by itself".) The user does something, such as putting money in the coin slot. The machine does something, such delivering a soft drinks can.

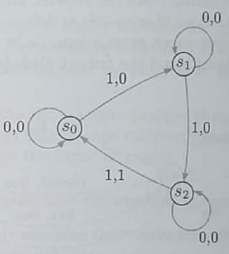
A **finite-state machine** or **automaton** can be described as a finite set of **states**, where in each state every permitted input generates a certain transition to some state. Non-permitted input (for instance a chewing gum in the coin slot) makes the machine stall forever. Some state is marked as the **initial state**, often using an arrow without a sender. In the machines described here, the state s_0 is always the initial state.

9.1.1 Mealy Machines

A **Mealy machine** is a kind of finite-state machine that gives output of an arbitrary kind. Such a machine (for instance a computer component or a soft drinks vending machine) has a number of states, a set of available **input signals** (for instance ones and zeros or coins and button-presses) that is called the **input alphabet** $\mathcal{I}$, and a set of available **output signals** (for

instance ones and zeros or change and soft drinks cans) that are called its **output alphabet** $\mathcal{O}$. When the machine gets an input signal it makes a transition to the **next state**, and in addition gives an output signal. The relationship between these things can either be illustrated using a graph or be organised in a table.

Example 9.1: ***A Mealy Machine*** A machine takes binary strings (sequences of 0:s and 1:s) as input, and is to send out a 1 for every third 1 in the input, and 0:s for the rest. The input and output alphabets are then both $\{0, 1\}$. The workings of the machine can be described by the following graph:



The nodes represent the states of the machine and the arrows the **transitions**. The pairs of numbers on the arrows mean that the arrow shall be followed if the first number is given as input, and that the second number then is to be given as output. If you are in state s_0 and get a 1 in you are thus to transit to state s_1 and give a 0 out.

The meaning of the first state s_0 (which is the initial state) is that we are waiting for the first 1 in a group. The second one, s_1, means that we are waiting for the second 1 in a group. The third one, s_2, means that we are waiting for the third and last 1 in a group.

If we want to describe the same thing using a table, then for each possible combination of state and input we have to note both what the next state ν is and which outsignal ω is to be given out.

	ν		ω	
	0	1	0	1
s_0	s_0	s_1	0	0
s_1	s_1	s_2	0	0
s_2	s_2	s_0	0	1

The Greek letters ν (*nu*) and ω (*omega*) are the initial letters in *next state* and *output*.

From the table we can for instance see that if we are at state s_1 and get 1 as insignal we shall transit to state s_2 and give 0 as outsignal. Exactly the same

information can be aquired from the graph, but the table takes up less space on the paper. However, it's usually easier to understand what the machine actually *does* when looking at the graph. ■

Exercise 9.1 Walk around in the machine in the example given the input 0110011011001. What is the output?

Example 9.2: ***Soft Drinks Machine*** A simple soft drinks vending machine accepts five-crown coins and ten-crown coins, and has furthermore a cancel button. The soft drinks costs ten crowns, and as soon as the required amount has been put in a can of coca-cola is delivered together with change, if any. If you put in a five-crown piece a lamp is lit. The indata alphabet is thus $\mathcal{I} = \{\text{five, ten, cancel}\}$ and the output alphabet $\mathcal{O} = \{\text{nothing, lamp, can, five, can+five}\}$.

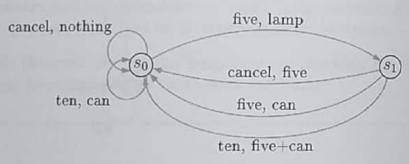


Figure 9.1: *"No, no violence. Money is the only language it understands!"*

The state table is

	ν			ω		
	five	ten	cancel	five	ten	cancel
s_0	s_1	s_0	s_0	lamp	can	nothing
s_1	s_0	s_0	s_0	can	can+five	five

Note that we have to specify what the machine is to do even for pointless input. There is no point in pushing *cancel* if you haven't done anything that can be cancelled, but that won't stop somebody from doing it and because of this the machine has to know what to do. ■

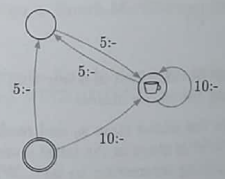
Exercise 9.2 Design a Mealy machine for an door code reader: if a combination of digits ending with the correct four-digit code (let's say 4711) is punched in, then the entry door will open.

Exercise 9.3 A change machine changes twenty-crown notes for ten- or five-crown coins. The change machine has two buttons marked 10 and 5 and a slot for the input of notes.

(a) Specify in English how the machine is supposed to work.

(b) Design a Mealy machine that works according to the specification.

Exercise 9.4: *Important!* In a so-called **Moore machine** the output doesn't happen during the transition between two states but at the arrival at the new state. In the diagram this is shown by specifying the output inside the rings and not beside the arrows:



The state with the double ring is the initial state. What is the function of the Moore machine in the diagram?

Exercise 9.5 Design a Moore machine for the soft drinks vending machine in example 9.2 on the facing page. *

Exercise 9.6 Ponder about whether it's always possible to design a Moore machine that performs the same thing as a given Mealy machine, and vice versa. Which type will normally give the smaller machine?

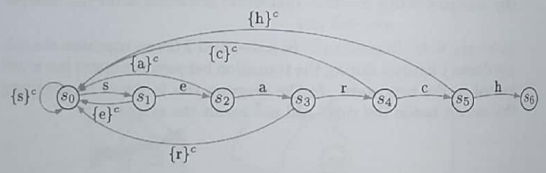
The same function specification for a machine can be fulfilled in several different ways, more or less elegant. Systematical methods for the optimisation of clumsy machines exist, but we won't delve into them here.

9.1.2 Automata for String Matching

The instructions fot a machine can be seen as a language, but finite state machines have also a role to play when *analysing* languages: spell checking, grammar control, text search, interpretation, pattern recognition, style analysis, and so on.

Let's limit ourselves to the example text search: We have a large amount of text at our disposal and have to find all the places that discuss the concept "search". If you just scroll through the text you may miss several instances of the word searched for. If we are to make a machine perform the search for us, what should the instructions look like? A suggestion is:

```
Read one letter at a time until the latest one is "s".
Check the next five letters.
If they are "e", "a", "r", "c", and "h", respectively, we have
found the word "search".
Otherwise, go on reading one letter at a time as before.
```



These instructions can be described in a way that is both more unambiguous and more simple using the graph above.

The machine starts in the initial state s_0 and reads letter after letter, and as long as the letter isn't "s" it stays in the intial state, which in principle means "we haven't read anything interesting so far". When it encounters an "s" a transition to the next state s_1 takes place, and that state means "we have read an 's'" If the next letter is an "e" it goes on to the next state, otherwise it returns to the intial state and restarts. In the same way it checks whether the letters "a", "r", "c", and "h" follow. When reaching s_6, the machine stops.

Exercise 9.7 Walk around in the machine according to the input * `this is the season for research`.

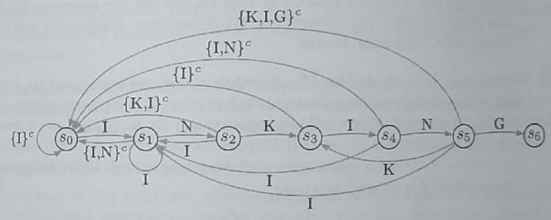
Exercise 9.8 Write down the meaning of all the states.

Exercise 9.9

(a) Walk around in the machine according to the input
`this uses inversesearch`
Why didn't the machine detect that this input did include the word "search"?

(b) Correct the machine by adding transitions that take into account that the word "search" may start anywhere.

A finite-state machine that searches for a certain string, as in the example above, we'll call a **Knuth-Morris-Pratt-machine** (after three computer scientists). If the sought string is such that the first letters appear in several places in the string, as in INKING, the Knuth-Morris-Pratt-machine becomes more complicated:



Exercise 9.10 Explain why the machine above looks the way it does.*

Exercise 9.11 Design a Knuth-Morris-Pratt-machine that searches for BABAR.

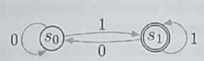
Exercise 9.12 Design a Knuth-Morris-Pratt-machine that searches for the Swedish word MINIMIMINNE (which means "minimal memory").

9.1.3 Acceptors

When spell-checking one wants to check whether a given string is included in a given dictionary. For this end it's of course possible to put together a finite-state machine à la Knuth-Morris-Pratt, but if the dictionary is large and irregular the machine will be horribly complex, so this isn't what is done in practice.

In an artificial language (more about those on page 251) on the other hand, the words may be formed in an extremely regular way, and then machines that check whether a word belongs to the language can be made very elegant. A simple language is the binary language that consists of all the words that

end with 1: $\{1, 01, 11, 001, 011, 101, 111, \ldots\}$. There are an infinite number of words in this language, but the machine needs only two states:



The initial state s_0 means "the last symbol read is not 1" and s_1 means "the last symbol read *is* 1". In the diagram s_1 is marked using an extra ring. The ring indicates that this is an **accepting state**. A machine is designed to recognise a certain language if all the words in the language lead to accepting states, while any other input leads to non-accepting states.

This machine, which is called an **acceptor**, ends up in the accepting state every time it reads a one, and in the non-accepting state every time it reads a zero. Thereby it will accept precisely the strings that end with one.

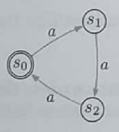
Exercise 9.13 Design a machine that recognises the language of all the binary strings that include exactly three ones. Explain as well in words what each state means.

Exercise 9.14 Design an acceptor that recognises the language of all the binary strings that end with at least three consecutive instances of the same symbol. (Hint: You can use several accepting states.)

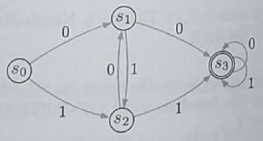
Exercise 9.15 Give some advice on how to proceed when designing an acceptor for a given language. *

Exercise 9.16 Which languages are recognised by the machines below?

(a)



(b)



Exercise 9.17 Explain in an intuitive way why it isn't possible to make a finite state machine that recognises the language consisting of all binary palindromes (strings that look the same backwards).

9.2 Languages

9.2.1 Natural Languages and Artificial Languages

What is a language? Usually we refer to languages such as Swedish, English, Chinese, and Swahili. Every such language is a complex system of components such as sound, written characters, words, grammar, intonation, and

more, and in addition meanings that are communicated using these tools. Swedish and English are instances of **natural languages** in the meaning that they have been developed – and keep on being developed – by the usage of the language by humans.

Exercise 9.18 A common statement is that there are appoximately 5000 spoken languages. What difficulties can arise when trying to determine the exact number?

There are many more languages than just the natural ones. In this book we have seen programming languages such as C (designed at the beginning of the 1970:s by Kernighan and Ritchie), written mathematical languages for set theory and logic (inventions about a hundred years old), and the kind of languages that the machines in the previous section identified. Languages designed in that way are usually called **artificial**. Esperanto (invented in 1887 by Ludwik Lazar Zamenhof) is an artificial language for international communication, designed to have clear grammatical rules and to be easy to pronounce.

Not all languages can be pronounced. **Machine code** is the name of the language consisting of ones and zeros in which instructions for a data processor is written. In all communications between human and machine (soft drinks vending machines, booking of tickets, mobile phones, and suchlike) some kind of language is required.

If we look at the structure of written English we see that the smallest unit is letters (and punctuation marks and other symbols). Letters are joined to words that are separated by spaces and other punctuation marks. Symbols and sequences of symbols are the basic components in all written languages, both natural and artificial.

Definition 9.1: ***Concepts from Language Theory***

- The set of permitted letters or symbols is called the **alphabet** of the language. In a number of previous examples and exercises we have been using the binary alphabet $\mathcal{B} = \{0, 1\}$.
- A sequence of symbols from the alphabet is called a **string**. 01001 is a binary string. XYZZY is a string over the Swedish alphabet.
- A new string can be formed by **concatenation** of two shorter strings. The concatenation of 01 and 001 gives 01001. The concatenation of MODE and STY gives MODESTY.
- The **empty string** that consists of zero letters we denote by λ. A string is thus not affected by concatenation with λ.
- A **language** is a collection of permitted strings that are called **words**. The Swedish language has a couple of hundred thousand words. The language $\mathcal{L} = \{00, 01, 10, 11\}$ is a language with four binary words. ■

Exercise 9.19 The Swedish alphabet is usually said to have 28 letters (or 29 if you count W). Ponder on how many symbols are really needed to write in different contexts, for example when chatting or when writing a book like this one.

Exercise 9.20 Why is an alphabet with two letters suitable for computers? Why is around thirty letters more suitable for Swedes? Are there natural "alphabets" of completely different sizes?

Exercise 9.21 Explain why $\lambda w = w\lambda$ for all strings w.

Exercise 9.22 Assume that the alphabet has at least two letters. Explain why the empty string is the only string v that satisfies $vw = wv$ for all strings w.

Exercise 9.23: *Important!* Let $\mathcal{L}_1 = \{00, 01, 10, 11\}$ and $\mathcal{L}_2 = \{0, 1\}$. We can **concatenate languages** by concatenating every word in one of the languages with every word in the other language. For instance $\mathcal{L}_2\mathcal{L}_2 = \mathcal{L}_1$.

(a) Write down the languages $\mathcal{L}_1\mathcal{L}_2$ and $\mathcal{L}_2\mathcal{L}_1$.

(b) If $\mathcal{L}_1$ and $\mathcal{L}_2$ are two languages, does it then always hold that $\mathcal{L}_1\mathcal{L}_2 = \mathcal{L}_2\mathcal{L}_1$?

Exercise 9.24 If $|\mathcal{L}_1| = n$ and $|\mathcal{L}_2| = m$, show that $|\mathcal{L}_1\mathcal{L}_2| \leq nm$ has to hold and give an example that shows that $|\mathcal{L}_1\mathcal{L}_2| = nm$ doesn't have to hold.

9.2.2 Regular Languages

In this section we will introduce a mathematical way of writing expressions – **regular expressions** – that can describe an important class of languages.

If a language has a finite number of words it can be described simply by listing the words. But many languages contain words of arbitrary length. For instance, the input language for a reverse vending machine consists of arbitrary long sequences of empty soft drinks cans. It's thus of practical value to be able to describe arbitrarily long sequences in a concise way. The tool is called **Kleene star**: *.

Definition 9.2: *Kleene Star* x^* means the language where all the words are repetitions of x an arbitrary number of times (zero or more). ∎

The language of the reverse vending machine can now be described simply as *can**! To put together expressions that describe languages, the Kleene star is supplemented with two other symbols, $+$ and $\cdot$, that are given new meanings in this context.

Definition 9.3: *+ and · for Languages* By $\mathcal{L}_1 + \mathcal{L}_2$ we mean the union of the languages $\mathcal{L}_1$ and $\mathcal{L}_2$, that is: the the language of all the words that are included in at least one of $\mathcal{L}_1$ and $\mathcal{L}_2$.

By $\mathcal{L}_1 \cdot \mathcal{L}_2 = \mathcal{L}_1\mathcal{L}_2$ we mean the concatenation of the languages $\mathcal{L}_1$ and $\mathcal{L}_2$.

By $\mathcal{L}^n$ we mean the concatenation of the language $\mathcal{L}$ with itself n times. ∎

Note that the meaning of $+$ and $\cdot$ are different when we are working with languages; it's not normal addition and multiplication of numbers that is meant.

Example 9.3 Define $\mathcal{L}_1 = \{\text{dead, oak}\}$, $\mathcal{L}_2 = \{\text{line}\}$, and $\mathcal{L}_3 = \{\text{wood}\}$. Then $\mathcal{L}_1 \cdot (\mathcal{L}_2 + \mathcal{L}_3) = \{\text{deadline, deadwood, oakline, oakwood}\}$. ∎

Example 9.4 Using $+$ and $\cdot$ we can describe the meaning of the Kleene star in an alternative way:

$$x^* = \lambda + x + x \cdot x + x \cdot x \cdot x + \cdots = \sum_{n=0}^{\infty} x^n$$

where the empty string $\lambda = x^0$, that is to say: zero repetitions of the word x. ∎

Example 9.5: *The Language of Natural Numbers* We are now to give a purely mathematical specification of the language $\mathcal{N}$ of all the natural numbers (that is to say $\{0, 1, 2, \ldots\}$ written in the base 10), expressed in the alphabet consisting of the digits from zero to nine.

In writing, natural numbers are given as sequences of digits where the first digit isn't zero (except when the number is zero). Because of this, we introduce the sets of digits $S = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ and $S_0 = S + \{0\}$. (We repeat that $+$ in this case is the union of languages, not addition of digits!) S_0 is thus the language of all one-digit numbers. To get a two-digit number we have to start with some digit that isn't zero, so SS_0 is the language of two-digit numbers. In the same way, SS_0^2 is the language of all three-digit numbers, and so on. The language of natural numbers can thus be described as

$$\mathcal{N} = S_0 + SS_0 + SS_0^2 + SS_0^3 + \ldots.$$

With the help of the Kleene star we can simplify this expression for the language of all natural numbers to

$$\mathcal{N} = S_0 + SS_0(S_0)^*.$$

∎

Exercise 9.25 Using the same notation as in example 9.5, what is the difference between the languages given by the the expressions $S_0 + SS_0(S_0)^*$ and $S(S_0)^*$?

Exercise 9.26 Let $\mathcal{R}$ denote the set $\{+, -, \cdot, /\}$ of symbols of the four rules of arithmetic. Explain why $\mathcal{N}(\mathcal{RN})^*$ gives the language of all arithmetical expressions of the type $5 + 41 \cdot 2 - 976$. *

Sensible expressions that can be put together using an alphabet $\mathcal{A}$ and the empty string λ and the operations $+$, $\cdot$, and * are called **regular expressions**. We will now give a precise recursive definition of the expressions that are "sensible".

Definition 9.4: ***Regular Expressions***

1. The empty string λ is a regular expression, as well as each separate symbol of the alphabet $\mathcal{A}$.
2. If R_1 and R_2 are regular expressions, then (R_1+R_2) is a regular expression as well.
3. If R_1 and R_2 are regular expressions, then $(R_1 \cdot R_2)$ is a regular expression as well.
4. If R_1 is a regular expression, then R_1^* is a regular expression as well.

Clearly unnecessary parentheses can be omitted. ∎

Example 9.6: ***Construction of Regular Expressions*** From the recursive definition above we can derive that the expression $\big(c^*(a+b)\big)^*$ is a regular expression over the alphabet $\mathcal{A} = \{a, b, c\}$:

- Since a, b, and c belong to the alphabet they are regular expressions, according to rule 1.
- Since c is a regular expression, c^* is a regular expression as well, according to rule 4.
- Since a and b are regular expressions, $(a+b)$ is a regular expression as well, according to rule 2.
- Since c^* and $(a+b)$ are regular expressions, $c^*(a+b)$ is a regular expression as well, according to rule 3.
- Since $c^*(a+b)$ is a regular expression, $\big(c^*(a+b)\big)^*$ is a regular expression as well, according to rule 4. ∎

Exercise 9.27 Derive that $(a+b)^*$ and $abb+abba$ are regular expressions according to the recursive definition. *

Exercise 9.28 Explain why $a+$, $(bc))$, and $^*c$ aren't regular expressions.

Example 9.7: ***Text-editing*** Powerful text-editors (for instance EMACS, that has been used when writing this book) can handle regular expressions, which can be very useful.

When breaking lines (see section 6.6.2 on page 166) it's not good if a digit ends up at the end of a line and the thing it's counting at the beginning of the next line; the digit and the unit should be treated as an unbreakable whole. That's something usually forgotten when writing the text. Being able to write the regular expression meaning "a digit followed by a space", and search for this and change all these spaces for unbreakable ones, can save an incredible amount of time. ∎

A language that can be described by a regular expression is a **regular language**. From the exercises above we can for instance conclude that the language of arithmetical expressions is a regular language.

Exercise 9.29 Show that the languge of all binary strings of odd length is a regular language. (Hint: which language is defined by $(\mathcal{B}^2)^*$?)

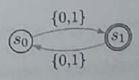
Exercise 9.30 Explain in an intuitive way why the language of all the palindromes over a certain alphabet can't be a regular language. (See exercise 9.17 on page 250.)

We have now introduced a class of particularily simple languages, the regular languages. One question is how they are related to the languages that we spent the last section upon – the languages that can be recognised by finite-state machines. It transpires that they are exactly the same languages in both cases!

Theorem 9.1 Every regular language can be recognised by a finite-state machine. ∎

The proof of theorem 9.1 is given as a sequence of exercises on the following page. (As a matter of fact, the reverse of the theorem is true as well, that is, that there is a regular expression for every language that can be recognised by a machine, but that's less interesting and more cumbersome to prove.)

Example 9.8 The language of all the binary strings of odd length is regular according to exercise 9.29. This language is recognised by the machine below:



∎

Exercise 9.31 Design finite-state machines that recognise the following regular languages:

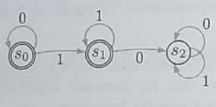
(a) $(10)^* + (01)^*$.

(b) $(1^*)0(1^*)0(1^*)$.

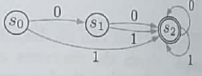
(c) $(0+1+2+3+4+5+6+7+8+9)^*4711$.

Exercise 9.32 Formulate regular expressions for the languages recognised by the following machines.

(a)



(b)



Anyone interested in the proof of theorem 9.1 on the preceding page is encouraged to try to carry it out on their own in the sequence of exercises below. Then it's necessary to understand what is meant by a **nondeterministic finite-state machine**, which is a machine where the same input (λ included) can lead to several different states. You then imagine that the machine tries all the ways indicated for the given input. A word is accepted if one of these possible ways leads to an accepting state.

Exercise 9.33 Prove that every regular language can be recognised by some nondeterministic finite-state machine with just one accepting state. (Hint! Start by showing how a machine that recognises a single symbol or λ can be made. Then show how nondeterministic machines having just one accepting state for $R_1 + R_2$, R_1R_2, and R_1^* can be put together if you already have machines for the regular expressions R_1 and R_2. Then the statement follows by induction.) *

Exercise 9.34 Show that every language that is recognised by some nondeterministic finite-state machine I can also be recognised by some normal (deterministic) finite-state machine A. (Hint: the states in A will correspond to sets of states in I.)

9.3 More Exercises

9.3.1 Routine Work

Exercise 9.35 Design a machine that takes twenty-, fifty- and one hundred-crown notes as input and outputs cinema tickets costing 70 crowns and change (ten crown coins and twenty crown notes.) Nothing more is needed.

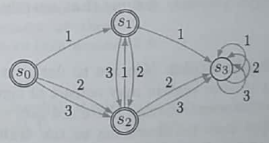
Exercise 9.36 Design a machine that takes strings of zeros and ones as input and outputs the number of ones on the form $3 + 3 + \ldots + 3$. (If the number of ones isn't divisible by three the extra ones don't count.) The output alphabet thus consists of the symbols 3. $+$, and λ (the empty string).

Exercise 9.37 Design a Knuth-Morris-Pratt-machine that looks for INFRINGE. *

Exercise 9.38 Design a machine that recognises the language of all the binary strings that contain an odd number of ones and an odd number of zeros.

Exercise 9.39 Design a machine that recognises the language of all the ternary strings (in the alphabet $\{A, B, C\}$) where the same symbol never appears twice in a row.

Exercise 9.40 What language is recognised by the machine below? Describe it both in words and using a regular expression!



Exercise 9.41 Let A, B, C be three languages defined by $A = \{\text{sept, oct, nov, dec}\}$, $B = \{\text{em, o}\}$, and $C = \{\text{ber}\}$.

(a) Write down all the words in the language ABC and in the language C^{10}.

(b) How many words are there in the language B^{10}?

Exercise 9.42

(a) Assume that the alphabet has only one letter. Show that concatenation commutes, that is to say that $w_1w_2 = w_2w_1$ for all the strings w_1 and w_2 over this alphabet. Example: if the alphabet is $\{x\}$ and the words are $w_1 = xx$ and $w_2 = xxx$, then $w_1w_2 = w_2w_1 = xxxxx$.

(b) Assume that the alphabet has at least two letters. Show that in this case there will always exist strings w_1 and w_2 so that $w_1 w_2 \neq w_2 w_1$.

Exercise 9.43 Write a regular expression for permitted Swedish car registration numbers. (For the rules about what registration numbers may look like, see exercise 5.75 on page 134.)

Exercise 9.44 Let p, q, and r be logical variables. Let $\mathcal{L}$ be the language of all logical expression containing p, q, and r that are given in disjunctive normal form.

(a) Find a regular expression over the alphabet $\{p, q, r, \vee, \wedge, \neg\}$ that describes the language $\mathcal{L}$. (Assume here that conjunction has higher priority than disjunction, since that makes it possible to manage without parentheses.)

(b) Design a machine that recognises the language $\mathcal{L}$.

9.3.2 To Ponder About

Exercise 9.45: *Relation* Design a machine that reads a 3×3 matrix for a relation as a string consisting of nine zeros/ones. The machine is to accept the string if and only if the relation is symmetric. You have to define for yourself a suitable order for the reading of the elements of the matrix (it doesn't have to be done row by row). A well-chosen input order simplifies the task considerably! You may assume that exactly nine symbols are put in.

Exercise 9.46: *Traffic Lights* You are to design a system for intelligent traffic lights. Input to your system may consist of signals from a number of timers and sensors that sense whether there are cars coming and pedestrians waiting. The output may be directions to the traffic lights for cars and pedestrians (don't forget the audible signal) and to reset the timers.

(a) Specify in English how the system is to work.

(b) Design a Mealy machine that works according to the specification.

Exercise 9.47 The set of all the regular expressions over an alphabet $\mathcal{A}$ is of course a language in itself, whose alphabet is $\mathcal{A}$ extended with the symbols $($, $)$, $+$, $\cdot$, and *. Is this language regular?

Answers

Chapter 1

1.3 Yes, stone-throwing and shot-putting follow the same physical principles.

1.4 No, discus-throwing is completely dependent on the interaction between the discus and the air that carries the discus, and thus demands a more complicated model.

1.6 $x^2 - y^2$. The fact that the numbers are positive is relevant when simplifying $\sqrt{x^2/y^2}$ to x/y. (In which way?)

Chapter 2

2.1 E.g., $\{n \mid n$ is an integer between -2 and $2\}$.

2.2 $\{N \mid N$ is a Swedish name that starts with P and consists of three letters$\}$. Or are there more such names than Per, Pål and Pia?

2.3 B is the set of all Swedish three-letter names beginning with P.

2.4 $\{1, 3, 5, 7, 9\}$

2.5 A and B are both subsets of C. (Furthermore all three of the sets are – by definition – subsets of themselves.)

2.6 B is a subset of A.

2.7 They are the same set. The first part of the information tells us that everything in A is also included in B. Thus, there is nothing in A that *isn't* included in B. And B contains nothing that A doesn't include as well. Then they have to contain the same things. And then they are the same set.

2.8 $A \cap B = \{x \mid x$ belongs to A and $B\}$

2.9

(a) $\{2, 3, 4, 6, 7, 8, 9, 10\}$ **(b)** $\{6\}$

2.12 The correct statements are $B \not\subseteq A$, $|B| = 3$, $B \cap A = \{5\}$, $B \cup A = \{0, 1, 2, 3, 4, 5, 7, 9\}$.

2.13 $|A|$ is a number. $\emptyset$, $\mathcal{U}$, $A \cap B$, $A \cup B$, $A \setminus B$, and A^c are sets. $A \subseteq B$, $A \not\subseteq B$, $x \in A$ och $x \notin A$ are statements.

2.14

(a) $M_1 \cap M_2 = \{\text{rock}\}$

(b) $M_1 \cap M_2 \cap M_3 = \{\text{rock}\}$

(c) $M_2 \cup M_3 = \{\text{rock, wood, punk, metal, jazz}\}$

(d) $M_1 \cap (M_2 \cup M_3) = \{\text{rock}\}$

(e) $(M_1 \cap M_2) \cup M_3 = \{\text{rock, punk, metal, jazz}\}$

(f) $M_1 \setminus M_2 = \{\text{paper, scissors}\}$

(g) $M_3 \setminus M_4 = \{\text{rock, punk, jazz}\}$

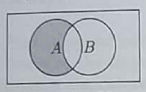
(h) $(M_1 \cup M_3) \setminus (M_2 \cup M_4) = \{\text{scissors, punk, jazz}\}$

(i) $(M_1 \setminus M_2) \cup (M_3 \setminus M_4) = \{\text{paper, scissors, rock, punk, jazz}\}$

2.16

(a) $A \setminus B = \{x \mid x \in A \text{ and } x \notin B\}$

(b)



(c) $A \setminus B = A \cap B^c$

2.17

$$(M_1 \cup M_3) \setminus (M_2 \cup M_4) =$$
$$= \big(M_1 \setminus (M_2 \cup M_4)\big) \cup \big(M_3 \setminus (M_2 \cup M_4)\big)$$
$$\subseteq (M_1 \setminus M_2) \cup (M_3 \setminus M_4)$$

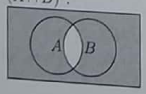
The equals sign holds since it doesn't matter if we remove all $(M_2 \cup M_4)$-elements from M_1 and M_3 separately and then join them or if we do it the other way around. The subset relation holds since we remove more elements on the right-hand side than in the middle step.

2.18 The symbols you can "cross out" are $\{\subseteq, \in\}$. That is the set of symbols that represent *statements*, expressed in a positive way.

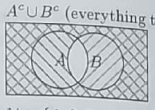
2.19 Since A and B have four elements each while $A \setminus B$ has three elements they must have exactly one element in common. Thus, $B \setminus A$ has three elements, $A \cap B$ one element and $A \cup B$ seven elements.

2.20

(a) $(A \cap B)^c$:



$A^c \cup B^c$ (everything that is marked)

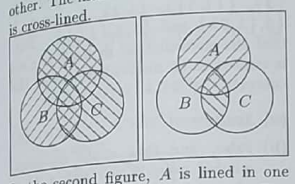


(b)
$$\mathcal{U} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$$
$$A = \{2, 4, 6, 8, 10\}$$
$$B = \{6, 7, 8, 9, 10\}$$
$$A \cap B = \{6, 8, 10\}$$
$$(A \cap B)^c = \{1, 2, 3, 4, 5, 7, 9\}$$
$$A^c = \{1, 3, 5, 7, 9\}$$
$$B^c = \{1, 2, 3, 4, 5\}$$
$$A^c \cup B^c = \{1, 2, 3, 4, 5, 7, 9\}$$

(c) This shows that the equality holds for *these* sets. But how can we be sure that it holds for other sets as well?

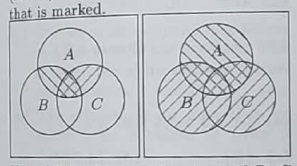
(d) Take an element x in $(A \cap B)^c$. x can't belong to *both* A and B, because then it would belong to the intersection of these sets (and thus not to the complement of the intersection). If x belongs to A then x *doesn't* belong to B, but to B^c. And then x belongs to whichever set $\cup B^c$. (The union contains all elements in the sets that are united.) And if x doesn't belong to A it belongs to A^c, and then to $A^c \cup$ whatever. In both cases x belongs to $A^c \cup B^c$. So $(A \cap B)^c \subseteq A^c \cup B^c$.
Now take an element y in $A^c \cup B^c$. If $y \in A^c$ then we have $y \notin A$. And then $y \notin A \cap B$ holds, since all elements in $A \cap B$ belong to A. And if $y \notin A^c$ then y has to belong to B^c instead. And using the same reasoning we see that y then can't belong to $A \cap B$. In both cases $y \notin A \cap B$ holds, which is the same thing as $y \in (A \cap B)^c$. So $A^c \cup B^c \subseteq (A \cap B)^c$.
Thus the sets are equal, Q.E.D.
This argument gets shorter if you can use logical symbols, and exercise 7.84 on page 215 will discuss this.

2.21 In the first figure we have lined $A \cup B$ in one direction and $A \cup C$ in the other. The intersection $(A \cup B) \cap (A \cup C)$ is cross-lined.



In the second figure, A is lined in one direction and $B \cap C$ in the other. The union is everything that is marked in some way. It is the same area in both cases.

Here $A \cap B$ is lined in one direction and $A \cap C$ in the other. The union $(A \cap B) \cup (A \cap C)$ consists of everything that is marked.



Here A is lined in one direction and $B \cup C$ in the other. The intersection $A \cap (B \cup C)$ is cross-lined.

The explanation using words we can't be bothered to write down!

2.23

(a)
$$(A^c \cup B)^c =$$
$$= (A^c)^c \cap B^c \quad \text{De Morgan}$$
$$= A \cap B^c \quad \text{Double complement}$$

(b)
$$(B \cap A) \cup (A^c \cap B) =$$
$$= (B \cap A) \cup (B \cap A^c) \quad \text{Com.}$$
$$= B \cap (A \cup A^c) \quad \text{Distr.}$$
$$= B \cap \mathcal{U} \quad \text{Inverse}$$
$$= B \quad \text{Identity}$$

2.24 One example: Databases contain lots of data. You will often want to find composite information (as shown in exercise 2.61 on page 33). Depending on the way the data is stored and which of it there is lots of and which there is little of, different methods can use quite different amounts of times. To be able to rewrite an expression into another can be of great practical help.

2.25 $|A| = 2$, $|B| = 3$, $|A \cup B| = |\{\text{apple, banana, muesli, yoghurt}\}| = 4$, $|A \cap B| = |\{\text{banana}\}| = 1$. $2+3 = 4+1$, which is correct.

2.26 $1812 + 1066 - 2001 = 877$

2.27 The audience consists of 60 males and 60 females. Of those, 80 are children, among them 48 girls. Women and children make up a total of $60+80-48 = 92$ persons. $120-92 = 28$ grown-up men remain.

2.29

$$|A \cup B| = |A| + |B| - |A \cap B|$$
$$|A \cup B \cup C| = |A| + |B| + |C|$$
$$- (|A \cap B| + |A \cap C| + |B \cap C|)$$
$$+ |A \cap B \cap C|$$

2.30

$$|A \cup B \cup C \cup D| =$$
$$= |A| + |B| + |C| + |D|$$
$$- (|A \cap B| + |A \cap C| + |A \cap D|$$
$$+ |B \cap C| + |B \cap D| + |C \cap D|)$$
$$+ |A \cap B \cap C| + |A \cap B \cap D|$$
$$+ |A \cap C \cap D| + |B \cap C \cap D|$$
$$- |A \cap B \cap C \cap D|$$

2.31 Larger sets are placed above smaller ones. Lines are drawn from every set to all its subsets on the level below. This way you can find every subset of a set by following the lines downwards in the diagram.

2.33 $\mathcal{P}(A) = \{X \mid X \subseteq A\}$

2.34 $\{17\}$ has just one proper subset: the empty set $\emptyset$.

2.35 {(milk, white), (egg, white), (milk, cup), (egg, cup)}

2.37 $|X \times Y| = |X| \cdot |Y|$.

2.40 $\mathbb{N} \cup \mathbb{Z}_+ = \mathbb{N}$, $\mathbb{N} \cap \mathbb{Z}_+ = \mathbb{Z}_+$, and $\mathbb{N} \setminus \mathbb{Z}_+ = \{0\}$. $\mathbb{Z} \setminus \mathbb{N}$ is the set of all negative integers. $\mathbb{N} \setminus \mathbb{Z} = \emptyset$. $\mathbb{N} \times \mathbb{N}$ is the set of integer points in the first quadrant. $\mathbb{Z} \times \mathbb{Z}$ is the set of integer points in the plane.

2.41 $2\mathbb{Z}$ is the set of all even integers. $\mathbb{Z} \setminus 2\mathbb{Z}$ is the set of all odd integers.

2.42 $\mathbb{Q} = \{x \mid \text{there are } a \in \mathbb{Z} \text{ and } b \in \mathbb{Z}_+ \text{ such that } x = a/b\}$

2.43 Every integer n can be written as a quotient $n/1$, and thus conforms to the requirements for belonging to $\mathbb{Q}$.

2.45 $\mathbb{R} \times \mathbb{R}$, also written as $\mathbb{R}^2$, is the set of all pairs (x, y) where both the numbers are real. These pairs of numbers are often used to specify the places of points in the plane, and are then called **coordinates**. This ought to be known from previous courses!

2.47 $\mathcal{P}(\{1,2,3\})$ is the power set of $\{1,2,3\}$, the set of all subsets: $\{\{\}, \{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}, \{1,2,3\}\}$. One bijection is

$\phi(000) = \{\}$ $\quad \phi(001) = \{3\}$
$\phi(010) = \{2\}$ $\quad \phi(011) = \{2,3\}$
$\phi(100) = \{1\}$ $\quad \phi(101) = \{1,3\}$
$\phi(110) = \{1,2\}$ $\quad \phi(111) = \{1,2,3\}$

A description of this bijection is that element n:o i should be included if the corresponding digit is one, otherwise not. (This bijection is often used, since binary strings are simpler both to generate and to manipulate than sets.) Lots of other bijections (or to be precise: 40,319 ones) exist, but this one is the most natural.

2.51 $f(x) = -\ln x$ works nicely.

2.52 The elements are, as can be seen from the definition of A, 1, 2, and $\{1,3\}$, three elements in total. The power set of A then contains the subsets $\{\}$, $\{1\}$, $\{2\}$, $\{\{1,3\}\}$, $\{1,2\}$, $\{1,\{1,3\}\}$, $\{2,\{1,3\}\}$ and $\{1,2,\{1,3\}\}$. Note that the element that in itself is a set is treated as an undivisible unit in this context.

(a) True.

(b) False, and that could be determinded even if you didn't know what A is. Only sets can be subsets.

(c) False. (That $\{1\}$ is a subset in one of the elements in A isn't enough.)

(d) True. (e) False. (f) True.

(g) True. (h) False.

2.53 $\{x \mid x = y^2, \text{ where } y \in \mathbb{Z}\}$

2.54

(a) $\{1,2,3,5,7,9\}$ (b) $\{3,5,7\}$

(c) $\{2\}$ (d) $\{1,9\}$

(e) $\mathbb{N}$ (f) $\emptyset$

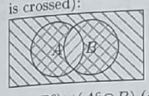
(g) $\{x \mid x \text{ is an even natural number}\}$

2.55

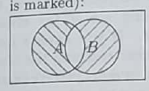
(a) Can be done in several ways; here is one with *all* the details included:

$$(A \cup B) \cap (A \cap B)^c =$$
De Morgan
$$= (A \cup B) \cap (A^c \cup B^c)$$
Distributive law
$$= \big((A \cup B) \cap A^c\big) \cup \big((A \cup B) \cap B^c\big)$$
Commutative law
$$= \big(A^c \cap (A \cup B)\big) \cup \big(B^c \cap (A \cup B)\big)$$
Distributive law
$$= \big((A^c \cap A) \cup (A^c \cap B)\big) \cup \big((B^c \cap A) \cup (B^c \cap B)\big)$$
Commutative law
$$= \big((A \cap A^c) \cup (A^c \cap B)\big) \cup \big((A \cap B^c) \cup (B \cap B^c)\big)$$
Inverse law
$$= (\emptyset \cup (A^c \cap B)) \cup ((A \cap B^c) \cup \emptyset)$$
Commutative law
$$= ((A^c \cap B) \cup \emptyset) \cup ((A \cap B^c) \cup \emptyset)$$
Identity law
$$= (A^c \cap B) \cup (A \cap B^c)$$

(b) $(A \cup B) \cap (A \cap B)^c$ (everything that is crossed):



$(A \cap B^c) \cup (A^c \cap B)$ (everything that is marked):



(c) The set of all elements that belong to either A or B but not both. This is called the **symmetrical difference** between A and B.

(d) The solution is so long that we couldn't be bothered to write it down!

2.56 $|A| = 10$, $|B| = 10$. There are 5 odd numbers in A, so $|C| = 5 \cdot 10 = 50$. There are 3 vowels in B, so $|D| = 10 \cdot 3 = 30$. $|C \cap D| = 5 \cdot 3 = 15$. $|C \cup D| = |C| + |D| - |C \cap D| = 50 + 30 - 15 = 65$. Alternatively, you can use the complement set instead: $|A \times B| = |A| \cdot |B| = 10 \cdot 10 = 100$, The pairs that *don't* belong to $C \cup D$ have to consist of an even number followed by a consonant. There are 5 even numbers and 7 consonants; $100 - 5 \cdot 7 = 65$.

2.57

(a) Yes, 1 can be found on the number line.

(b) Yes, $3.14 = {}^{314}/_{100}$ which is a quotient of integers.

(c) No, π can't be expressed exactly as a fraction.

(d) No, -5 isn't a possible number of items.

(e) Yes, ${}^{64}/_{8} = 8$ which is an integer.

2.58 "The set of all fractions that aren't integers" or "the set of all quotients that don't break even".

2.59 We won't include a picture, but will mention that for four sets there should be $2^4 = 16$ different areas, and for five sets $2^5 = 32$. More about this in exercise 4.61 on page 102.

2.60 We can order the sets in a list, and also order the elements in each set. Then we can write a matrix where the first row is the elements of the first set, and so on. By drawing a path in the same way as in example 2.8 on page 30 we get a list of the union of the sets. (If an element is included in several of the sets we number it the first time we pass it.)

2.61 We can start by constructing a number of basic sets, such as

$A = \{p \mid p.\text{number-of-visits} < 10\}$
$B = \{p \mid p.\text{age} < 35\}$
$C = \{p \mid p.\text{family-doctor} \neq 000\}$

The first question can then be written

$A \cap B \subseteq C$?

The second question becomes

$|A \cap B| =$?

The third question becomes

Kimmo $\in A \cap B$?

Chapter 3

3.1 Example: We have 11 wheels. How many cars will that supply? 2.75 cars doesn't make sense; "There are enough wheels for 2 cars, and there will be

3 wheels left over" is sensible.

3.2 $\{r \mid \text{there is } q \in \mathbb{Z} \text{ such that } 11 = 4q + r\}$

3.3 Quotient 8, remainder 3 and quotient -1, remainder 1.

3.4

(b) It makes sense to let the lowest possible quotient represent the lowest possible rank. Then it is a matter of taste if you want the aces to be lowest or highest. The same thing about the suits. If you want to analyse bridge, it can be a good thing to put the suits in the order they have in that game; if you are writing a patience program it may be more useful to set odd card-number = red, even = black.

(c) Don't start with zero but higher, so that cards with quotient 3 are threes, and so on. This simplifies the conversion a lot.

3.5 All numbers are divisors of zero, but the only number that has zero as a divisor is zero itself. All numbers have one as a divisor, but the only numbers that divide one are 1 and -1.

3.6 To make it come out even, the pattern length has to be a divisor of the total length. In this context, only positive answers are meaningful, so we'll say that the lengths 1, 2, 4, 5, 8, 10, 20, and 40 can be used.

3.7 2. This is by the way the only even prime number.

3.8 4

3.9 $60 = 2{\cdot}2{\cdot}3{\cdot}5$, 61 is a prime number, $62 = 2{\cdot}31$, $63 = 3{\cdot}3{\cdot}7$, $64 = 2{\cdot}2{\cdot}2{\cdot}2{\cdot}2{\cdot}2$.

3.10 One method: write down all the integers between 2 and 1000. 2 is a prime number. Then cross out every second number (that is: all even numbers except 2). The next uncrossed number is 3, which is prime. Cross out every third number from that point. The next uncrossed number is 5, cross out every fifth. And so on. This is called **sieve of Eratosthenes**, named after the first known inventor. The list will be: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, and 47.

3.11 One method: check by dividing the number by each number from two and upwards and see if the division comes out even. If you haven't found any divisors before reaching the square root of the given number you are guaranteed not to find any after that either. If a number has a hundered digits the square root has about 50 digits. 10^{50} nanoseconds is 10^{41} seconds $\approx 3 \cdot 10^{33}$ years. There are prime tests that are a lot better than this one, but the algorithms for prime factorisation are not all that better.

3.12 That there are the same number of each prime factor of 30, while 28 has one *repeated prime factor* and one that isn't repeated.

3.13 Like a vertical line. Written horizontally, the graph for 2^{500} looks like this: 2^{500}—2^{499}—2^{498}—. . . —2^2—2—1.

3.14 See figure 2 on the facing page.

3.15
$$100 = 1 \cdot 70 + 30$$
$$70 = 2 \cdot 30 + \boxed{10}$$
$$30 = 3 \cdot 10$$

3.16 $10 = 70 - 2 \cdot 30 = 70 - 2 \cdot (100 - 70) = 3 \cdot 70 - 2 \cdot 100$

3.17 $\operatorname{lcm}(m, n)$ is the smallest positive number that is divisible by both m and n.

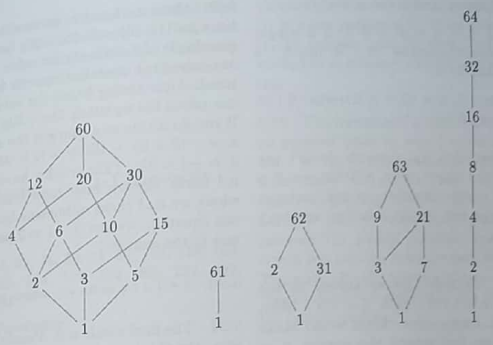


Figure 2: *Exercise 3.14*

3.18 A linear combination that gives the gcd, that is something that is written like $4711x + 777y = 7$. From another point of view, this is the equation of a straight line, which can also be written as

$$y = -\frac{4711}{777}x + \frac{7}{777} = -\frac{673}{111}x + \frac{1}{111}$$

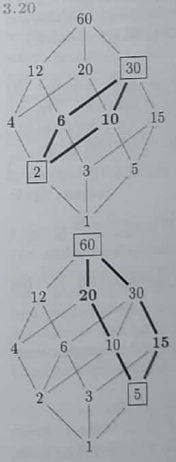
In the first try we did find an integer point on this line: $x = 16$, $y = -97$. If we want to get to another integer point we have to increase/decrease x by a multiple of 111, otherwise y will be a fraction and not an integer.

3.19 We refuse to give answers for the linear combinations; feed what you have written into your calculator and check that it is correct!

(a) $\gcd(408, 672) = 24$,
$\operatorname{lcm}(408, 672) = 11{,}424$

(b) $\gcd(527, 300) = 1$,
$\operatorname{lcm}(527, 300) = 158{,}100$

3.20



The gcd is the number we will find closest *below* the numbers in case; the lcm is closest *above*.

3.21 $\frac{11}{703}$

3.22

(a) $p = 6$, $a = 4$, $b = 9$, is a simple counterexample. $6 \mid 36$ but $6 \nmid 4$ and $6 \nmid 9$.

(b) $p = 6$, $a = 12$, $b = 3$ works. $6 \mid 36$ and $6 \mid 12$.

So note that the theorem doesn't say anything about what will happen if p isn't a prime number. p will perhaps, perhaps not, divide a factor; we don't know.

3.23 We would get an answer on the form $x = -110 - 245k$, $y = 155 - 345k$, which among other things would mean that we had missed the answer $x = -110 - 49$, $y = 155 + 69$.

3.24 $(135, -190)$, $(86, -121)$, $(37, -52)$, $(-12, 17)$, $(-61, 86)$, $(-110, 155)$, $(-159, 224)$, $(-208, 293)$, $(-257, 362)$, $(-306, 431)$, $(-355, 500)$

3.25 Yes, because both describe the same set of pairs of numbers. The important thing is which numbers you get, not the way used to describe them.

3.26

(a) $x = -(3 + 9k)$, $y = 3 + 8k$

(b) $x = -(6 + 13k)$, $y = 3 + 6k$

(c) Unsolvable. ($\gcd(84, 119) = 7$, but $7 \nmid 134$.)

3.27 For instance one might write down the function

$$y = -\frac{33.5}{14.5}x + \frac{1284.5}{14.5}$$

and then try $x = 1$, $x = 2$, and so on until one gets a y that is an integer. Using a programmable calculator, this way is probably faster.

3.28 None; when x is negativ y is positive, and vice versa.

3.29 One method is to realise that since $\gcd(15, 20) = 5$, $15x+20y$ has to be a multiple of 5, let's say $5w$, after which one solves the equation $5w + 9z = 1$ instead. After having found the value of w one solves the equation $15x + 20y = 5w$. If you do it this way, you get the answer $x = -(2 + 9k + 4l)$, $y = 2 + 9k + 3l$, $z = -1 - 5k$. (Those who have studied linear algebra may note that here, where we had three unknowns but only one equation, we got $3 - 1 = 2$ parameters in the answer.)

3.30 $x \in \{\, x \mid x = 8n+7, \text{where } n \in \mathbb{Z} \}$

3.31 The final result is 9. Purely workwise, the third version is a little simpler than the second one. The first one is most complicated, at least when calculating by hand.

3.33 Assign the days of the week numbers, and count modulo 7. Monday can be 1, and then those who think that Sunday is the first day of the week can use the numbers 0–6 and those who think that it's the last day can use 1–7. Tuesday is 2, $2 + 100 = 102 = 14 \cdot 7 + 4 \equiv 4 \pmod 7$. Thursday.

3.34 Many computer games have a modular coordinate system on the screen. If you exit to the right you enter again from the left. A more serious application is encryption, where the computations almost always contain modular arithmetic in some step. And computer programs without safety checks can include completly unintended modular arithmetic – if the result from computations consists of more digits than there is space reserved some vanish into thin air. (Programs *with* safety checks stop and give an error message in this situation.)

3.35 The weird rows have the numbers 2, 3, and 4. These numbers and the number 6 have factors in common, which 6 and 1 and 6 and 5, respectively, don't have.

3.36 6 corresponds to -1 in $\mathbb{Z}_7$. The last row corresponds to -1, -2, and so on – the numbers we get if we step around the number circle backwards.

3.38

(a) $x = 3$ or $x = 10$

(b) $x = 5$

(c) Unsolvable

3.39

(a) 14, 29, 44, 59, 74, 89, 104, 119, 134, 149, 164, 179, 194, 209, 224, 239, 254

(b) Unsolvable

(c) 33

3.41 If we call the number of words x, the information given tells that $x \equiv 1 \pmod 6$. This equation has the solution $x = 1 + 6k$, $k \in \mathbb{Z}$. Then we can add that x has to be positive, and that it is less likely that $x = 1$, so $x \in \{\, 7, 13, 19, \ldots \}$ seems reasonable.

3.42

(a) 1 and 5 have an inverse; both are their own inverses, since $1^2 = 1$ and $5^2 = 25 = 6 \cdot 4 + 1 \equiv 1 \pmod 6$.

(b) All numbers except 0 have an inverse. $1^{-1} \equiv 1$, $2^{-1} \equiv 4$, $3^{-1} \equiv 5$, $4^{-1} \equiv 2$, $5^{-1} \equiv 3$, $6^{-1} \equiv 6 \pmod 7$.

3.43

(a) A composite number can be factorised into two numbers smaller than itself. These numbers are used in $\mathbb{Z}_n$, and if you multiply them you get $n \equiv 0 \pmod n$.

(b) A prime number can't be the result of a multiplication using numbers smaller than itself, so there are no pairs of numbers in $\mathbb{Z}_p$ that gives p as the result.

3.44 That only some fractions can be expressed nicely using decimals. $1/2$ works well in our normal system, but $1/3$ is more problematic. If we switched to the number base 12, then $1/3$ would work, but in return $1/5$ would stop working.

3.45 The meaning of a digit is given by its *position* (that is: *place*) in the number. A one can for instance represent both "one", "one million", and "one trillionth", depending on where in the number it is placed. Other systems are for instance the **unary number system**, where you put one mark for each object. There are also the roman numerals. Neither of these two systems is very practical in calculations.

3.46 42 and 1100101, respectively.

3.47 The method usually used by persons who know the powers of two by heart is step by step to look for the closest power of two. $512 < 777 < 1024$, so 777 contains one 512. $777 - 512 = 265$, which contains one 256. $265 - 256 = 9$, and $9 = 8+1$, so $777 = 512+256+8+1$. How much work this method will be depends a lot on the number to be converted (and on how well you know the powers of two), while the work needed in method 3.7 depends only on the size of the number.

3.48 If you continue a couple of steps you add a number of leading zeros to the binary representation of the number. There are situations where a certain number of digits is wanted, and then you have to add zeros on numbers that are "too short". There are examples in Chapter 4 and Chapter 7.

3.49 69 in both cases! 105 and 69 are called "magical numbers" because they are connected in this way. The normal case is getting two completely different answers. (Try!)

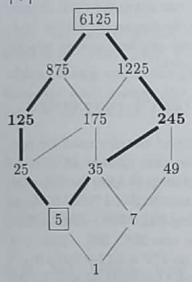
3.50 Go via decimal form; go via binary form; calculate in octal/hexadecimal in the conversion algorithms – it is *possible*, but you really

have to watch what you are doing! (Try it!)

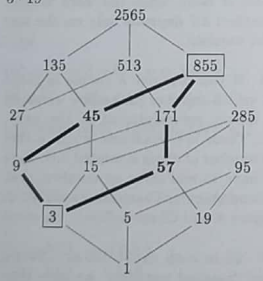
3.51 Quotient and principal remainder in colour:

(a) $34 = 5 \cdot 6 + 4$
(b) $34 = -5 \cdot (-6) + 4$
(c) $-34 = -6 \cdot 6 + 2$
(d) $-34 = 6 \cdot (-6) + 2$

3.52 The prime factorisation is $5 \cdot 5 \cdot 5 \cdot 7 \cdot 7$



3.53 The prime factorisation is $3 \cdot 3 \cdot 3 \cdot 5 \cdot 19$



Note that it would also have been possible to get to lcm(45,57) = 3 from 45 via 15 to 3, och to gcd(45, 57) = 855 from 57 via 285 to 855.

3.54 The answers given here are *not* the ones obtained by running the Euclidian algorithm backwards. You have to figure out how to determine whether what you have written is correct! (Hint: calculator.)

(a) $\gcd = 7$, $\operatorname{lcm} = 1092$,
$7 = 25 \cdot 91 - 27 \cdot 84$
(b) $\gcd = 17$, $\operatorname{lcm} = 22{,}525$,
$17 = 70 \cdot 425 - 33 \cdot 901$
(c) $\gcd = 1$, $\operatorname{lcm} = 328{,}861$,
$1 = 349 \cdot 617 - 404 \cdot 533$
(d) $\gcd = 36$, $\operatorname{lcm} = 83{,}160$,
$36 = 26 \cdot 2376 - 49 \cdot 1260$

3.55

(a) $\gcd = 1$, $\operatorname{lcm} = 6790$,
$1 = 83 \cdot 97 - 115 \cdot 70$
(b) $\gcd = 11$, $\operatorname{lcm} = 13{,}860$,
$11 = 47 \cdot 693 - 148 \cdot 220$
(c) $\gcd = 2$, $\operatorname{lcm} = 274{,}482$,
$2 = 284 \cdot 598 - 185 \cdot 918$
(d) $\gcd = 1$, $\operatorname{lcm} = 14{,}689{,}660$,
$1 = 4491 \cdot 2356 - 1697 \cdot 6235$

3.56 $\gcd(m, n) = \max\{\, d \mid d|m \text{ and } d|n \,\}$

3.57 $\gcd(m, n)$ is a positive number d having the following properties:

$$\begin{cases} d \mid m \\ d \mid n \\ \text{if } c \mid m \text{ and } c \mid n \text{ then } c \mid d \end{cases}$$

(In words: "d is a common divisor that furthermore has all common divisors as divisors".) This alternative definition is sometimes useful.

3.58 If your answer doesn't look like the one given here it doesn't have to be wrong. (The answers have been shifted down to the smallest possible numbers, which is fairly common.)

(a) $x = 1 + 37k$, $y = 2 - 27k$
(b) $y = 5 + 4k$, $z = -7 - 3k$
(c) Unsolvable.

3.59

(a) Unsolvable.

(b) $x = 2 + 17k$, $y = -1 - 11k$

(c) $x = 3 + 33k$, $y = 2 - 26k$

3.60 3 6-packs and 7 10-packs or 8 6-packs and 4 10-packs or 13 6-packs and 1 10-pack.

3.61

(a) 16 (b) 9

3.63

(a)

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	0
2	2	3	4	5	6	7	0	1
3	3	4	5	6	7	0	1	2
4	4	5	6	7	0	1	2	3
5	5	6	7	0	1	2	3	4
6	6	7	0	1	2	3	4	5
7	7	0	1	2	3	4	5	6

·	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	0	2	4	6
3	0	3	6	1	4	7	2	5
4	0	4	0	4	0	4	0	4
5	0	5	2	7	4	1	6	3
6	0	6	4	2	0	6	4	2
7	0	7	6	5	4	3	2	1

(b) 1, 3, 5, and 7 have an inverse. All of them are their own inverses.

(c) 4 solutions (1, 3, 5, and 7).

(d) Ordinary algebraic rules, like the one stating that a quadratic equation has exactly two solutions, clearly don't hold in modular arithmetic. If you've got any more comments, your teacher will probably be interested.

3.64

(a)

+	0	1	2	3	4	5	6	7	8
0	0	1	2	3	4	5	6	7	8
1	1	2	3	4	5	6	7	8	0
2	2	3	4	5	6	7	8	0	1
3	3	4	5	6	7	8	0	1	2
4	4	5	6	7	8	0	1	2	3
5	5	6	7	8	0	1	2	3	4
6	6	7	8	0	1	2	3	4	5
7	7	8	0	1	2	3	4	5	6
8	8	0	1	2	3	4	5	6	7

·	0	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8
2	0	2	4	6	8	1	3	5	7
3	0	3	6	0	3	6	0	3	6
4	0	4	8	3	7	2	6	1	5
5	0	5	1	6	2	7	3	8	4
6	0	6	3	0	6	3	0	6	3
7	0	7	5	3	1	8	6	4	2
8	0	8	7	6	5	4	3	2	1

(b) 0, 3, and 6.

(c) 3 solutions: $x = 0$, $x = 3$, and $x = 6$.

(d) $x^2 \equiv (x-3)(x-6)$ in $\mathbb{Z}_9$. The fundamental theorem of algebra, that a polynomial can be factorised in one and only one way, apparently doesn't hold in modular arithmetic.

3.65

(a) $x = 308$ (b) Unsolvable.

(c) $x \in \{\, 99, 227, 355, 483, 611, 739, 867, 995 \,\}$

3.66

(a) Insoluble.

(b) $x \in \{\, 3, 35, 67, 99, 131, 163, 195, 227, 259, 291, 323, 355, 387, 419, 451, 483, 515, 547 \,\} = \{\, x \mid x = 3 + 32k,\ k \in \mathbb{Z},\ 0 \le k \le 17 \,\}$

(c) $x = 18$

3.67

(a) 11111110, 376, FE
(b) 2989, 5655, BAD
(c) 1111000111010000, 61,904, F1D0

(d) 1011101011011010, 135,332, 47,834

3.68 A number a can only have one inverse, and this can be explaind like this: Suppose that we have found two inverses to a, let's say b and c. Then we can make the following calculation:

$$b = b \cdot 1 \equiv b \cdot (a \cdot c) = (b \cdot a) \cdot c \equiv 1 \cdot c = c$$

which tells that the two inverses were identical!

3.69 The original number. Two inversions cancel out.

3.71 $x = 3$, $y = 13$ and $x = 16$, $y = 2$ are the two possible solutions of the equation.

3.72 Both solutions fit when inserted into the equation, so both are correct – but not complete – solutions. Since the gcd $= 1$, the answers should be decorated with $+19k$ and $-11k$, respectively, and the acquaintances have failed to do this. One has succeded in getting only one half of the answers, the other one only one third.

3.73 $x + 1$

3.74

(a) It works if you switch to three dimensions, which this swimming ring illustrates:

If you take a closer look you'll see that it is divided into 16 squares. It can be regarded as a diagram showing $\mathbb{Z}_4 \times \mathbb{Z}_4$. If we appoint the ring around the middle as one of the coordinate rings and the ring in the other direction closest to us as the other one we have a working numbering system.

(b) To depict $\mathbb{Z}_n$ we need a number circle, which needs a two-dimensional paper for the drawing. $\mathbb{Z}_n \times \mathbb{Z}_n$ is found in the three-dimensional space. Thus, $\mathbb{Z}_n \times \mathbb{Z}_n \times \mathbb{Z}_n$ means that we have to work in four dimensions. It may be a bit hard to envisage, but nothing stops us from using it in theoretical arguments!

3.75

(a)
$$\begin{array}{r} {\scriptstyle 111} \\ 1011 \\ +\,1101 \\ \hline 11000 \end{array}$$

$11 + 13 = 24$ in decimal notation.

(b)
$$\begin{array}{r} {\scriptstyle 101010} \\ 4321 \\ -\;\;567 \\ \hline 3532 \end{array}$$

$2257 - 375 = 1882$ in decimal notation.

(c)
$$\begin{array}{r} 3\mathrm{D} \\ \cdot\;31 \quad {\scriptstyle 2} \\ \hline 3\mathrm{D} \\ +\mathrm{B}7 \\ \hline \mathrm{BAD} \end{array}$$

$61 \cdot 49 = 2989$ in decimal notation.

3.76

(a) Study a number ending in 5, such as $abcd5$. This number can be written as $a \cdot 10^4 + b \cdot 10^3 + c \cdot 10^2 + d \cdot 10^1 + 5 \cdot 10^0 = 10 \cdot (a \cdot 10^3 + b \cdot 10^2 + c \cdot 10^1 + e) + 5 = 5 \cdot (2 \cdot (a \cdot 10^3 + b \cdot 10^2 + c \cdot 10^1 + e) + 1) = 5 \cdot$ integer. The number is divisible by 5. (The same principle works even if the number of digits is different.)

(b) Study the number $abcde$. This number can be written as $a \cdot 10^4 + b \cdot 10^3 + c \cdot 10^2 + d \cdot 10^1 + e \cdot 10^0 \equiv a \cdot 1^4 + b \cdot 1^3 + c \cdot 1^2 + d \cdot 1^1 + e \cdot 1^0 = a + b + c + d + e \pmod 9$, since $10 \equiv 1 \pmod 9$.

(c) Divisibility by 2 and by 10 can be determined from the last digit, besides divisibility by 5. Last digit 0, 2, 4, 6, or 8: divisible by 2. Last digit 0 or 5: divisible by 5. Last digit 0: divisible by 10. 2, 5 and 10 are all divisiors in the number base.

(d) Divisibility by 3 can be checked in this way, since $10 \equiv 1 \pmod 3$. 3 is a divisor of 9.

(e) The number base 8 has the divisors 1, 2, 4, and 8. Last digit 0, 2, 4, or 6; divisible by 2. Last digit 0 or 4: divisible by 4. Last digit 0: divisible by 8.

(f) $16 \equiv 1 \pmod{15}$, so divisibility by 15 can be checked using the sum of the digits. 3 and 5 divides 15, so these divisibilities can also be checked using the sum.

Chapter 4

4.1 If we call the length of the shorter side on A0 x, then the longer side has length $\sqrt{2}x$. From the given area we get $x\sqrt{2}x = 1 \;\Rightarrow\; x = 1/\sqrt[4]{2} \approx 0.841$ m. The dimensions in mm for the formats are then the following: A0: 1189×841, A1: 841×595, A2: 595×420, A3: 420×297, A4: 297×210.

4.2

(a) $a^0 = 1$. (True for all positive bases a)

(b) $a^b = a \cdot a^{b-1}$

(c)
$$\begin{cases} a^0 = 1 \\ a^b = a \cdot a^{b-1} & \text{if } b > 0 \end{cases}$$

(d) $5^3 = 5 \cdot 5^2 = 5 \cdot 5 \cdot 5^1 = 5 \cdot 5 \cdot 5 \cdot 5^0 = 5 \cdot 5 \cdot 5 \cdot 1 = 5 \cdot 5 \cdot 5 = 5 \cdot 25 = 125$

4.3
$$\begin{cases} q(n,d) = 0 & \text{if } n < d \\ q(n,d) = 1 + q(n-d,d) & \text{otherwise} \end{cases}$$

$q(19,3) = 1 + q(16,3) = 2 + q(13,3) = 3 + q(10,3) = 4 + q(7,3) = 4 + q(4,3) = 6 + q(1,3) = 6 + 0 = 6$

4.4

(a) It starts $2 \cdot (-3) = 2 + 2 \cdot (-4) = 2 + 2 + 2 \cdot (-5)$, and can keep on like this for all eternity.

(b) There are several solutions that are equally good, and this is one:

$$\begin{cases} a \cdot b = 0 & b = 0 \\ a \cdot b = a + a \cdot (b-1) & b > 0 \\ a \cdot b = -(a \cdot (-b)) & b < 0 \end{cases}$$

(c) If we apply this definition on $2 \cdot (-3)$ we get

$$\begin{aligned} 2 \cdot (-3) &= -\big(2 \cdot (-(-3))\big) \\ &= -\big(2 \cdot 3\big) \\ &= -\big(2 + 2 \cdot 2\big) \\ &= -\big(2 + 2 + 2 \cdot 1\big) \\ &= -\big(2 + 2 + 2 + 2 \cdot 0\big) \\ &= -\big(2 + 2 + 2 + 0\big) \\ &= -6 \end{aligned}$$

4.5 There are several solutions that are equally good, and this is one:

$$\begin{cases} r(n,d) = r(n,-d) & d < 0 \\ r(n,d) = r(n-d,d) & d > 0, n \geq d \\ r(n,d) = r(n+d,d) & d > 0, n < 0 \\ r(n,d) = n & \text{otherwise} \end{cases}$$

4.6 1, 2, 4, 8, 16, 32, 64, 128, 256, 512. This sequence consists of all the powers of two.

4.7 $b_0 = 1$, $b_1 = 0$, $b_n = b_{n-1} + b_{n-2}$ otherwise. The sequence starts 1, 0, 1, 1, 2, 3, 5, 8, 13, 21. We see that it becomes the Fibonaccci sequence, shifted one step, so this was an example of "strange context where this sequence appears". Furthermore, we can realise that a somewhat smarter program will be content to find f_0 *once*.

4.8 Organise the calculations in exactly the same way as one does when calculating by hand.

4.9 { 000, 001, 011, 010, 110, 111, 101, 100 } and { 0000, 0001, 0011, 0010, 0110, 0111, 0101, 0100, 1100, 1101, 1111, 1110, 1010, 1011, 1001, 1000 }

4.10

(a) If you've only got one book it's guaranteed to be in the right order.

(b) Take book $n+1$ and find the place among the n sorted books where it should be, and put it there.

This sorting method is called **straight insertion**, and fairly suitable for bookcases. There are lots of other sorting algorithms.

4.11

(a) $(-3)^3 + (-4)^3 + (-5)^3 + (-6)^3 + (-7)^3 = -27 - 64 - 125 - 216 - 343$

(b) $(2\cdot0+1)+(2\cdot1+1)+(2\cdot2+1)+(2\cdot3+1)+(2\cdot4+1) = 1+3+5+7+9$

4.12 For instance

(a) $\sum_{i=1}^{4} \frac{1}{i}$ (b) $\sum_{k=0}^{5} 2(-1)^k$

4.13

(a) $\sum_{i=0}^{4}(i+3)^4$ (b) $\sum_{k=0}^{7}\cos(k-2)x$

4.14 $\sum_{k=1}^{n}(10+8k)$

4.15

(a) $\frac{1+9}{2}\cdot 5 = 25$

(b) $\sum_{k=1}^{10}(10+8k) =$

$= \frac{(10+8)+(10+8\cdot 10)}{2}\cdot 10$

$= 540$ seconds $= 9$ minutes

4.16 $3333333333_{\text{octal}} =$

$= 3\cdot 8^9 + 3\cdot 8^8 + \cdots + 3\cdot 8^1 + 3\cdot 8^0$

$= 3\cdot 8^0 + 3\cdot 8^1 + \cdots + 3\cdot 8^8 + 3\cdot 8^9$

$= \sum_{k=0}^{9} 3\cdot 8^k = 3\cdot\frac{1-8^{10}}{1-8}$

$= 460{,}175{,}067$

4.17

(a) $\frac{9}{1}\cdot\frac{8}{2}\cdot\frac{7}{3}\cdot\frac{6}{4} = 126$

(b) 0, since one of the factors is zero.

4.18 $\prod_{k=0}^{9}(100-k)$, or, if we turn the expression around before introducing the product sign: $\prod_{k=91}^{100} k$.

4.20 This answer, and some of the following ones, is mostly an outline showing the logical parts of the proof. A good solution includes many more steps and comments.

Base: $a_0 = 1 = (1+2\cdot 0)3^0$; $a_1 = 9 = (1+2\cdot 1)3^1$.

Induction: Assume that the formula works for a_p and a_{p+1}

$a_{p+2} = 6a_{p+1} - 9a_p$ acc. hyp.

$= 6(1+2(p+1))3^{p+1} - 9(1+2p)3^p$

$= 2\cdot 3(1+2p+2)3^{p+1} - 3^2(1+2p)3^p$

$= (2+4p+4-1-2p)3^{p+2}$

$= \big(1+2(p+2)\big)3^{p+2}$

4.21 (Outline) *Base:* $\text{LHS}_1 = {}^1/_2 = \text{RHS}_1$.

Induction: Assume that $\text{LHS}_p = \text{RHS}_p$. Then

$\text{LHS}_{p+1} = \text{LHS}_p + \frac{1}{(p+1)((p+1)+1)}$

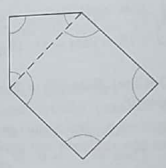
$= \text{RHS}_p + \frac{1}{(p+1)(p+2)}$ acc. hyp.

$= \frac{p(p+2)}{(p+1)(p+2)} + \frac{1}{(p+1)(p+2)}$

$= \frac{p+1}{(p+1)+1} = \text{RHS}_{p+1}$

4.22 *Base:* The statement holds for $n = 3$, according to the given theorem.

Induction: Assume that the statement holds for $n = p$, that is, assume that the sum of the angles in a "p-gon" is $(p-2)\cdot 180°$. Now study a $p+1$-gon. It has a protruding corner. Cut it off. (See figure.) The cut-off part is a triangle, where the sum of the angles is 180°. The remaining figure is a p-gon, and thus, according to the hypothesis, the sum of its angles is $(p-2)\cdot 180°$. The sum of the angles in the original figure equals the sum of the angles in the reduced figure plus the sum of the angles in the cut-off piece, that is $(p-2)\cdot 180° + 180° = ((p+1)-2)\cdot 180°$, Q.E.D.



4.23 (Outline) *Base:* $4^{2\cdot 1} - 1 = 15 = 1\cdot 15$.

Induction: Assume that $4^{2p} - 1 = m\cdot 15 \Leftrightarrow 4^{2p} = m\cdot 15 + 1$. Then

$4^{2(p+1)} = 4^2\cdot 4^{2p} = 16(m\cdot 15+1)$ acc. hyp.

$= 16m\cdot 15 + 15 + 1 = (16m+1)\cdot 15 + 1$

so $4^{2(p+1)} - 1 = \text{integer}\cdot 15$.

4.24

(a) The base step works excellently, but the inducive step is impossible to get to work.

(b) The inductive step works excellently, but you can't get a base step.

This shows clearly that both the base step and the inductive step are necessary components in the proof. Anyway, it was not possible to make up any proofs of these untrue statements.

4.26 None at all. For instance, $1 < 3 > 2$, and there we have $a < c$. But $1 < 3 > 0$ is a true statement with the same structure, and there $a > c$ holds. And for $1 < 2 > 1$ we have $a = c$!

4.28

(a) The rule in the middle will change to ${}^1/_m < {}^1/_n$, the others are unchanged.

(b) As written.

4.30 It is possible to make a function defined by the difference between the expressions. Then you find its lowest value in the interval. If the lowest value isn't negative then the difference will never be negative, and then the first term in the difference will always be larger.

4.31 Well, we have $0^2 = 0 < 1 = 2^1$, $1^2 = 1 < 2 = 2^1$, $2^2 = 4 = 2^2$, $3^2 = 9 > 8 = 2^3$, $4^2 = 16 = 2^4$. It varies a bit, we can say that the relationship oscillates for a while before becoming stable.

4.35 $\begin{cases} \gcd(m,n) = n & \text{if } n \mid m \\ \gcd(m,n) = \gcd(n,r) & \text{otherwise} \end{cases}$

(where r is the remainder from the division of m by n.)

4.36 For $n = 1$ the list looks like { 0, 1 }. For larger values than that generate the list for $n-1$. Write two copies of the list, one after the other. Add a zero first in the first copy, a one first in the second one.

4.37 Let a_n denote the number of moving sequences for n cans. $a_0 = 1$, $a_1 = 1$, $a_2 = 2$, is easily established. After that the moving sequences can be subdivided into two disjoint sets: those that end with 1 and those that end with 2. You can get a sequence ending with 1 by appending a 1 to a sequence of length $n-1$, and a sequence ending with 2 by appending a 2 to a sequence of length $n-2$.

That gives the recursive expression $a_0 = 1$, $a_1 = 1$, $a_n = a_{n-1} + a_{n-2}$, $n \geq 2$. (Note that we get here the Fibonacci sequence, shifted one step!)

4.38 There is one string of length zero: the empty string. There are two permitted strings of length one. If I then want to make a string of length $n-1$ I can either append a one to a any permitted string of length n or "10" to a string of length $n-1$. The recursive equation will be:

$$\begin{cases} a_0 = 1 \\ a_1 = 2 \\ a_{n+1} = a_n + a_{n-1} \end{cases}$$

This is the Fibonacci sequence shifted two steps!

4.39 32, 38, 206, 410, 1622, 4058

4.40 1, 9, 45, 189, 864, 3483

4.41 Several versions are possible; here is one:

(a) $\sum_{i=0}^{4} \frac{i+1}{(-3)^i}$ **(b)** $\sum_{j=0}^{6} (10 - 0.5j)$

4.42 Several versions are possible; here is one:

(a) $\prod_{i=1}^{4} i^2$ **(b)** $\prod_{k=1}^{5} \frac{10-k+1}{k}$

4.43 $2/2 = 1$, $4/6 = 2/3$, $8/24 = 1/3$, $16/120 = 2/15$

4.44 $-1, -9, -36, -100$

4.45 Geometrical and arithmetical:

(a) $31/16$ **(b)** 156

4.46 (Outline) *Base:* $\mathrm{LHS}_1 = (-1)^3 = -1$; $\mathrm{RHS}_1 = -\frac{1}{4}(1^2(1+1)^2) = -1$

Induction: Assume that $\mathrm{LHS}_p = \mathrm{RHS}_p$. Then

$$\begin{aligned} \mathrm{LHS}_{p+1} &= \mathrm{LHS}_p + (-(p+1))^3 \\ &= \mathrm{RHS}_p + (-(p+1))^3 \quad \text{acc. hyp.} \\ &= \tfrac{1}{4}(-p^4 - 6p^3 - 12p^2 - 12p - 1) \\ &= -\tfrac{1}{4}(p+1)^2(p+2)^2 = \mathrm{RHS}_{p+1} \end{aligned}$$

4.47 It says that the binary number $111\ldots11$ (with n ones) is just before the binary number $1000\ldots00$ (with n zeros), which is a well-known phenomenon.

4.48 (Outline) *Base:* $4^1 = 4 = 0 \cdot 12 + 4$

Induction: Assume that $4^p = m \cdot 12 + 4$. Then

$$\begin{aligned} 4^{p+1} &= 4 \cdot 4^p \overset{\text{acc. hyp.}}{=} 4 \cdot (m \cdot 12 + 4) \\ &= 4m \cdot 12 + 16 = (4m+1) \cdot 12 + 4 \end{aligned}$$

4.49 (Outline) *Base:* $6^2 = 36 = 4 \cdot 9 + 0 \equiv 0 \pmod 9$

Induction: Assume that $6^p \equiv 0 \pmod 9$. Then $6^{p+1} = 6 \cdot 6^p \overset{\text{acc. hyp.}}{\equiv} 6 \cdot 0 = 0 \pmod 9$.

4.50 Note that the base step has to include the first *three* numbers in the sequence, and the induction hypothesis has to be "assume that the statement holds for *three* consecutive numbers". (Outline)

Base: Show that the formula works for a_1, a_2, and a_3.

Induction: Assume that the formula works for a_p, a_{p+1}, and a_{p+2}. Then

$$\begin{aligned} a_{p+3} &= 2a_{p+2} + 5a_{p+1} - 6a_p \\ &= 2\big(4 - 5(-2)^{p+2} + 6 \cdot 3^{p+2}\big) \\ &\quad + 5\big(4 - 5(-2)^{p+1} + 6 \cdot 3^{p+1}\big) \\ &\quad - 6\big(4 - 5(-2)^p + 6 \cdot 3^p\big) \quad \text{acc. hyp.} \\ &= (2 + 5 - 6) \cdot 4 \\ &\quad + (-10 \cdot 2^2 + 25 \cdot 2 + 30)(-2)^p \\ &\quad + (12 \cdot 3^2 + 30 \cdot 3 - 36) \cdot 3^p \\ &= 4 + 5 \cdot 8 \cdot (-2)^p + 6 \cdot 27 \cdot 3^p \\ &= 4 - 5 \cdot (-2)^{p+3} + 6 \cdot 3^{p+3} \end{aligned}$$

4.51 (Outline) *Base:* Show that the formula works for b_0 and b_1.

Induction: Assume that the formula works for b_p and b_{p-1}.

$$\begin{aligned} b_{p+1} &= 6b_p - 9b_{p-1} \\ &= 6(2p+1)3^p - 9\big(2(p-1)+1\big)3^{p-1} \\ &= 2 \cdot 3(2p+1)3^p - 3^2(2p-1)3^{p-1} \\ &= (4p + 2 - 2p + 1)3^{p+1} \\ &= \big(2(p+1)+1\big)3^{p+1} \end{aligned}$$

4.52 *Base:* True for A0, which according to the definition has the area $1 = 1/1 = 1/2^0$ m^2.

Induction: Assume that format Ap has the area $1/2^p$ m^2. The next format, n:o $p+1$, is half that size, and thus has the area $(1/2^p)/2 = 1/(2^p \cdot 2) = 1/2^{p+1}$ m^2.

4.53 *Base:* Holds for $n = 1$, since $\{0,1\}$ undoubtedly consists of $2^1 = 2$ different code words, where each word differs from the previous one at exactly one place.

Induction: Assume that the algorithm for $n = p$ delivers 2^p different code words, ordered correctly. If we then run the algorithm for $n = p+1$ we get twice as many code words, that is to say $2 \cdot 2^p = 2^{p+1}$ ones. The code words within one "half" of the code are different, and consecutive ones differ at exactly one place, since the only thing we did was to add the same digit at the beginning. Code words from different halves are different, since the first digits are different. The code words at the "join" differ at exactly one place, since their first digits are different but the rest identical. So if the code for $n = p$ satisfies the conditions then the code for $n = p+1$ does as well.

4.54 We won't show the answer here, but would like to say that there is a surprisingly large number of completely different ways of solving this exercise. And the proof ends up completely different depending on the solution of (a).

4.55

(a) "Assume that the statement holds for *all* numbers between $n = 3$ and $n = p$, 3 and p included."

4.57 The fault lies in the conclusion that the replacing marble has the same colour as the replaced one, if the two fistfuls were unicoloured. This conclusion is correct in all cases except one, which is if the two fistfuls are disjoint. Then both fistfuls can be unicoloured without their union being so. And the fistfuls are disjoint if $n = 2$. Thereby the whole proof falls except for the base.

4.59

(a) Assume that we have drawn n lines dividing the plane into a_n parts. If we now draw another line it will initially divide one of the infinite outer areas into two. Every time it crosses one of the already present lines it arrives at a new area which is split as well. It crosses all the lines (since it isn't parallel to any one of them) one at a time (since three lines never intersect at the same place). That gives the recursive expression

$$\begin{cases} a_0 = 1 \\ a_{n+1} = a_n + n + 1 \end{cases}$$

(b) We haven't written down the proof, but note that in spite of starting 1, 2, 4, the sequence isn't 2^n.

(c) *Base:* The statement is clearly true when we have 0 lines, because then we can even manage using just one colour!

Induction: Assume that it is possible to colour a plane subdivided by n lines using two colours. What happens when we draw yet another line?

We can partition the areas into two groups: those on one side of the new line and those on the other. The ones on the first side can keep their colours. Since bordering areas there had different colours before, they will still have them (since

we haven't changed anything). On the ones on the other side we switch colours. Bordering areas will still be differently coloured after that. And areas that have the new line as a border will have different colours, since they consist of subdivisions of old areas, where we have changed the colour of one of the parts. So it is possible to colour a plane subdivided by $p+1$ lines as well.

4.60

(a) The new monthly allowance is the same as last month, that is $a_n = a_{n-1}$, if Kimmo hasn't been saving. If Kimmo has been saving he gets the difference $a_{n-1} - a_{n-2}$ as well, that is $a_n = 2a_{n-1} - a_{n-2}$ in total.

(b) If Kimmo hasn't saved during a certain month n he gets the allowance $a_n = a_{n-1}$. Then it won't help if he saves the next month; the next payment will be the same as last month: $a_{n+1} = 2a_n - a_{n-1} = a_n$ and so it continues. The only way to get an increase in the long run is thus to save every month. Then $a_n = 2a_{n-1} - a_{n-2}$ holds. We had $a_1 = 1 = 2 \cdot 1 - 1$ and $a_2 = 3 = 2 \cdot 2 - 1$. Let $k > 2$ be a month in the future and assume that $a_n = 2n - 1$ for all $n < k$. Then $a_k = 2a_{k-1} - a_{k-2} = 2(2k-3) - (2k-5) = 2k-1$. According to the induction principle then $a_n = 2n - 1$ for all $n \geq 1$ (when saving the maximum; otherwise as previously stated, the sum is less).

4.61

(a) For instance:

(b) Correct from the beginning, and the new curve cuts off part of each of the already present areas, and thus doubling the number of areas. (This was a very brief proof by induction!)

(c) Here we get 14 areas:

(d) First we analyse exactly what happens when a circle is drawn:

Start drawing the circle at the point where it crosses the edge of another circle. The circle goes into an area, and partitions it into two pieces. Every time it crosses an edge it starts to subdivide a new area. And so on until it joins the starting point. The circle therefore generates the same number of subareas as the number of borders it crosses, which is twice the number of circles already present.

Now for the proof!

Base: The statement is true for $n = 1$, since one circle subdivides the plane into two pieces (inside and outside), and $2 = 1^2 - 1 + 2$.

Induction: Assume that $n = p$ circles partition the area into $p^2 - p + 2$ pieces. What happens when we draw circle $p+1$?

The circle will generate $2p$ new subareas, and then the total number of subareas will be

$$\underbrace{(p^2 - p + 2)}_{\text{old}} + \underbrace{2p}_{\text{new}} =$$
$$= p^2 + 2p + 1 - p - 1 + 2$$
$$= (p+1)^2 - (p+1) + 2$$

So the statement still holds.

(e) The proof will closely resemble the one in example 4.21 on page 96.

4.65 All the numbers in the sequence are even squares. $S_n = (1+2+\cdots+n)^2$ is the statement which is to be proved.

4.66 All numbers in the sequence end with zero, so they are divisible by ten.

Chapter 5

5.2 No. The two outcomes aren't equally probable, since the drawing pin isn't symmetrical.

5.3

(a) $5/36$ **(b)** $10/36$ **(c)** $21/36$

5.4

(a) $(1/2)^3 = 1/8$

(b) $3/8$ (Outcomes allowed are heads-heads-heads, heads-heads-tails, heads-tails-heads.)

(c) $3/8$

5.5 The probability of having chosen the right door at the start is $1/5$, which is our chance of winning if not switching doors. If we choose the wrong door at the start (chance $4/5$) we are guaranteed to get the car if we switch. The probability is thus increased from $1/5$ to $4/5$.

5.6 Both questions: Yes. But the proof looks different.

5.7 $9/36 = 1/4$

5.8 $8/13$

5.9

$$P(A \text{ or } B) = \frac{|A \cup B|}{|\Omega|}$$
$$= \frac{|A| + |B| - |A \cap B|}{|\Omega|}$$
$$= \frac{|A|}{|\Omega|} + \frac{|B|}{|\Omega|} - \frac{|A \cap B|}{|\Omega|}$$
$$= P(A) + P(B) - P(A \text{ and } B).$$

5.10 $1/4 + 1/13 - 1/52 = 16/52 = 4/13$

5.11 The risk of losing is $3/36 + 6/36 + 1/2 - 4/36 = 23/36$. The chance of winning is thus $1 - 23/36 = 13/36$. The potential gain is 50 crowns and the potential loss 100 crowns. A small chance of gaining a little money doesn't compensate a great risk of losing a lot of money. It seems unwise to play this game. One says that the expected value is negative: the expected gain is $(13/36) \cdot 50 - (23/36) \cdot 100 = -45.83$, that is, on average one loses 45.83 crowns each time.

5.13

(a) 0.25 **(b)** 0.4 **(c)** 0.16

(d) 0.1 **(e)** 0.01

5.14

(a) The risk that the system will stop functioning during the year is 28 percent.

(b) Now it's slightly more than 4 percent. (To be precise: $1 - (1 - 0.10^3)(1 - 0.20^2)$.)

5.15 If the probability of a woman is p then the probability of a man (that is, not-woman) ought to be $1 - p$. And if we have 25 independent companies the probabily of a man in all of them is according to the multiplication principle $(1-p)^{25}$. If this probability is to be more than 50 % then p has to be less than $1 - \sqrt[25]{0.50} \approx 2.7$ %.

5.16

(a) The probability of having chosen correctly from the start is in the same way as before $1/5 = 3/15$. If we have the wrong door (chance $4/5$) the car will be hidden behind one of the three doors that the presenter hasn't opened. The probability that we will succeed in switching to precisely that door is $1/3$. The chance of getting the car if we switch is thus $4/5 \cdot 1/3 = 4/15$. So the probability is increased from $3/15$ to $4/15$.

(b) The probability that we'll have chosen correctly from the start is now $2/5$, and the probability that we'll have chosen wrongly (that is to say, chosen a door hiding a goat) $3/5$. If we have chosen wrongly there will

be cars behind two of the three doors that the presenter hasn't opened. If instead we have chosen correctly there are cars behind one of the three unopened doors. The probability of getting a car if switching is thus

$$\frac{3}{5}\cdot\frac{2}{3}+\frac{2}{5}\cdot\frac{1}{3}=\frac{6}{15}+\frac{2}{15}=\frac{8}{15}$$

5.17 $P(B \mid A)=0$; $P(A \mid B)=0$; $P(\text{not } A \mid B)=1$; $P(B \mid \text{not } A)=1/51$.

5.18 According to the definition the events A and B are independent if the probability that A occurs isn't affected by whether B occurs or not. That means exactly that $P(A \mid B)=P(A)$.

5.19

$$P(A \text{ and } B)=\frac{|A\cap B|}{|\Omega|}$$
$$=\frac{|A\cap B|}{|B|}\cdot\frac{|B|}{|\Omega|}=P(A \mid B)\cdot P(B).$$

5.20

(a) $6+11=17$ choices.

(b) $6\cdot 11=66$ choices.

5.21 The list starts: one-piece, nothing; one-piece, purple hat; and ends: yellow jacket, yellow pants, green hat.

5.22 If Svea or Hanna gets to be chairperson there are six possible ways of electing the treasurer. If Alfred or Rut gets to be chairperson the same person can't be treasurer so there are only five possiblities left. Therefore, in total there are $6+6+5+5=22$ ways of appointing the posts.

5.23 Regarding the first element we have to choose whether it is to belong to the subset or not; that is a choice between two possibilities. The same thing applies to the second element; there is another choice between two possibilities. In the same way, for each of the n elements we have a choice between two possibilities. According to the multiplication principle that gives in total

$$\underbrace{2\cdot 2\cdot 2\cdot\cdots\cdot 2}_{n\text{ 2:or}}=2^n$$

possible choices.

5.24 5. (Hint: draw a decision tree.)

5.25 Note: This is a true example.

(a) The addition principle, in spite of being a multiplication! It's the sum of 30 equal terms.

(b) No. The addition principle presupposes that the cases are *disjoint*, and it's very unlikely that these lists of ingredients are. (Quite a lot of the products include for instance *aqua*....)

5.28 For $n\geq 2$ it's true that

$$n!=n(n-1)(n-2)\cdots 1$$
$$=n\cdot[(n-1)(n-2)\cdots 1]=n\cdot(n-1)!$$

For $n=1$ it is true that $1!=1=1\cdot 1=1\cdot 0!$, since $0!$ is defined as 1.

5.30 A permutation of n symbols is the same thing as a bijection between the n symbols and the n places in the permutation.

5.31 A pack with jokers can be shuffled in 54! ways. A pack without jokers kan be shuffled in 52! ways. $54!/52!=54\cdot 53=2862$.

5.32

(a) $8!=40{,}320$ (b) $8!$

5.33 Rub out the last element in your answers to excercise 5.27:

123, 124, 132, 134, 142, 143, 213, 214, 231, 234, 241, 243, 312, 314, 321, 324, 341, 342, 412, 413, 421, 423, 431, 432

5.34 Start with the definition $n(n-1)(n-2)\cdots 1=n!$ and divide both sides by $(n-k)!$. For the rest, the version with the factorial sign uses less space on the paper when written, and is therefore more practical in "letter calculations", while the other one is a lot better when you have to use concrete numbers. (If you don't see what we mean, try to calculate $100\cdot 99\cdot 98$ using both formulas!)

5.35

(a) $20!/12!\approx 5\cdot 10^9$

(b) There are five cases depending on whether the two new queues have the lengths of $2+6$, $3+5$, $4+4$, $5+3$, or $6+2$. In each of these cases there are $20!/12!$ ways of forming the queues. In total there are then $5\cdot 20!/12!\approx 2.5\cdot 10^{10}$ ways.

5.36 21, 35, and 35.

5.37 $$\binom{n}{k}=\prod_{i=0}^{k-1}\frac{n-i}{k-i}$$

5.39

(a) $\binom{1000}{20}\approx 3\cdot 10^{41}$

(b) There are $\binom{999}{20}$ ways of choosing twenty rows among the 999 first rows. The probability for this to happen is thus $\binom{999}{20}/\binom{1000}{20}$. This quotient can be rewritten as:

$$\binom{999}{20}\Big/\binom{1000}{20}=$$
$$=\frac{999!}{20!\,979!}\Big/\frac{1000!}{20!\,980!}=\frac{980}{1000}$$
$$=98\ \%.$$

You can find this answer directly by regarding this as the problem of finding the probability that the last row will be among the 980 unchosen rows instead.

5.41 1, 6, 15, 20, 15, 6, 1.

5.42 We are to prove that $\binom{n}{0}=1$ and $\binom{n}{n}=1$ for all integers $n\geq 0$. This follows directly from the definition of the binomial numbers:

$$\binom{n}{0}=\frac{n!}{0!\,n!}=1 \qquad \binom{n}{n}=\frac{n!}{n!\,0!}=1$$

For the rest, the numbers along the left edge represent "the number of ways of taking zero objects among n possible ones" which can only be done in one way (leave them all), while the numbers along the right edge represent "the number of ways of taking n object among n possible ones" which can only be done in one way as well (take them all). Because of this, there should be ones in these positions.

5.43 We are to prove that the relationship $\binom{n}{k}=\binom{n-1}{k-1}+\binom{n-1}{k}$ holds for all integers $n\geq k\geq 0$. The right-hand side is, according to the definition, equal to

$$\frac{(n-1)!}{(k-1)!\,(n-k)!}+\frac{(n-1)!}{k!\,(n-k-1)!}=$$
$$=\frac{k\cdot(n-1)!}{k\cdot(k-1)!\,(n-k)!}$$
$$+\frac{(n-k)\cdot(n-1)!}{k!\,(n-k)\cdot(n-k-1)!}$$
$$=\frac{k\cdot(n-1)!}{k!\,(n-k)!}+\frac{(n-k)\cdot(n-1)!}{k!\,(n-k)!}$$
$$=\frac{(k+n-k)\cdot(n-1)!}{k!\,(n-k)!}$$
$$=\frac{n\cdot(n-1)!}{k!\,(n-k)!}=\frac{n!}{k!\,(n-k)!},$$

which by definition is equal to the right-hand side.

5.44 Subsets with k element from $\{1, 2, \ldots, n\}$ can be divided into two kinds: subsets where the element n isn't included and subsets where the element n is included. The former are subsets with k elements from $\{1, 2, \ldots, n-1\}$. The latter are, if element n is removed, subsets with $k-1$ elements from $\{1, 2, \ldots, n-1\}$.

5.46 You get a term of type $x^{n-k}y^k$ from the product $(x+y)^n$ by choosing the y-term from k of the factors (and thereby choosing the x-term from the $n-k$ remaining factors). This can be done in $\binom{n}{k}$ ways, which is thus the coefficient in front of $x^{n-k}y^k$.

5.47 $(x+y)^4 = x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + y^4$. If we put in $-y$ instead of y we get $(x-y)^4 = x^4 - 4x^3y + 6x^2y^2 - 4xy^3 + y^4$. (There will be a minus sign where there is an odd power of y.)

5.48 There are *many* answers to this question. It might make sense to regard the contents of a libary as a multiset, since there is often several copies of the same book. But if you work with people multisets don't seem useful, since only one copy of each person exists!

5.50 $\dfrac{10!}{4!\,3!\,3!} = 4200$

5.51 $\binom{6}{3,2,1}$

5.52 There are 2 m, 1 u, 2 l, 2 t, 4 i, 2 n, 2 o, 1 a, 2 c, 2 e, and 2 f, 22 letters in total. Can be ordered in

$$\binom{22}{2,1,2,2,4,2,2,1,2,2,2} = \frac{22!}{2!\,1!\,2!\,2!\,4!\,2!\,2!\,1!\,2!\,2!\,2!} \approx 1.7 \cdot 10^{17} \text{ ways}$$

5.53 $\binom{n}{k}$ tells us in how many ways you can select k things among n possibilities. That can be done by arranging the things in a row and then distributing k notes with the text "selected" and $n-k$ notes with the text "not selected". That corresponds to permuting k objects of one kind and $n-k$ of another kind, which $\binom{n}{k,n-k}$ counts.

5.54

$$\begin{aligned}
&\binom{3}{3,0,0}a^3b^0c^0 + \binom{3}{2,1,0}a^2b^1c^0 \\
&+ \binom{3}{2,0,1}a^2b^0c^1 + \binom{3}{1,2,0}a^1b^2c^0 \\
&+ \binom{3}{1,1,1}a^1b^1c^1 + \binom{3}{1,0,2}a^1b^0c^2 \\
&+ \binom{3}{0,3,0}a^0b^3c^0 + \binom{3}{0,2,1}a^0b^2c^1 \\
&+ \binom{3}{0,1,2}a^0b^1c^2 + \binom{3}{0,0,3}a^0b^0c^3 \\
&= a^3 + 3a^2b + 3a^2c + 3ab^2 + 6abc \\
&\quad + 3ac^2 + b^3 + 3b^2c + 3bc^2 + c^3
\end{aligned}$$

5.57

(a) The worst case is 20 socks. Sock number 21 then *has to* match one of the earlier ones.

(b) There are in total $40 \cdot 39 \cdot 38 \cdot \dots \cdot 21$ ways of picking out the first 20 socks from the machine. If no pair is to be formed the first sock can be picked in 40 ways, the second in 38 (any sock *except* the one matching the one we just picked), the third in 36, and so on. That gives a probability of

$$\frac{40 \cdot 38 \cdot 36 \cdot \dots \cdot 2}{40 \cdot 39 \cdot 38 \cdot \dots \cdot 21} = \prod_{i=1}^{20} \frac{2i}{20+i} \approx 7.6 \cdot 10^{-6}$$

(And I could have sworn that this happens every time!)

5.58 1, 15, 25, 10, 1

5.59 $\{\{1\},\{2,3\},\{4\}\}$;
$\{\{1,2\},\{3\},\{4\}\}$;
$\{\{1,3\},\{2\},\{4\}\}$;
$\{\{1,4\},\{2\},\{3\}\}$;
$\{\{1\},\{2,4\},\{3\}\}$;
$\{\{1\},\{2\},\{3,4\}\}$

5.60 You can partition the things into k piles, which can be done in $S(n,k)$ ways, and then distribute the piles to the persons, which can be done in $k!$ ways. In total: $k!S(n,k)$ ways.

5.61 The number of solutions to the equation

$$x_1 + x_2 + \dots + x_k = n$$

where $x_i \in \mathbb{N}$ för $i = 1, 2, \dots, k$ is

$$\binom{n+k-1}{k-1}$$

5.62

(a) $\binom{20+5-1}{5-1} = \binom{24}{4} = 10{,}626$ ways.

(b) Remove $5 \cdot 2 = 10$ biscuits. Distribute the remaining 10 in the standard way, which can be done in $\binom{10+5-1}{5-1} = \binom{14}{4} = 1001$ ways.

(c) If the mathematicians don't manage to eat all the biscuits, then they may eat zero or one or two or.... We can solve the problem for all these numbers and add, but the solution gets a lot simpler if we introduce the variable of the office secretary, who eats all the left-over biscuits. Then we have six eaters, and are guaranteed that all the biscuits are eaten. $\binom{20+6-1}{6-1} = \binom{25}{5} = 53{,}130$ ways.

5.63 Place n biscuits in a row. Then there are $n-1$ spaces. In each space we can either place or not place a boundary stick. Then there are 2^{n-1} possible choices, that each corresponds to one way of partitioning the number.

5.64

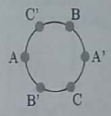
(a) 10^4 (each choice can be done in 10 ways). Example: 4-digit codes.

(b) $10 \cdot 9 \cdot 8 \cdot 7 = 5040$ (one less to choose among for each choice). Example: Eating 4 sweets from a plate containing 10 different ones.

(c) $\binom{10}{4} = 210$. Example: forming a 4-person committee from a group of 10.

(d) Here you have to use section 5.4.1 on page 126. Call the number of item i x_i. We need the number of solutions to the equation $x_1 + x_2 + \dots + x_{10} = 4$, $x_i \in \mathbb{N}$. There are $\binom{13}{9} = 715$ ones. Example: Ordering 4 drinks from a menu containing 10 kinds.

5.65 Call the women A, B, and C and their husbands A', B', and C' respectively. Seat A on one of the chairs. B' and C' have to sit next to A , and at their sides in their turn C and B respectively. Furthest from A A' has thus to sit.



This seating is acceptable and we have in total twelve possible ways of carrying it out: 6 possible placements of A, then 2 possible placements of B' and C' (that is: who shall sit to the left of A).

5.66 There are $\binom{35}{7}$ different Lotto rows. Out of those one has all the numbers correct. To get six numbers correct we have to choose six out of the seven winning numbers and one of the twenty-eight losing numbers, so there are $\binom{7}{6}\binom{28}{1}$ rows with six correct numbers. In the same way we get $\binom{7}{5}\binom{28}{2}$ rows with five correct numbers. The chance of getting at least five correct numbers is thus

$$\frac{1 + \binom{7}{6}\binom{28}{1} + \binom{7}{5}\binom{28}{2}}{\binom{35}{7}} \approx 0.12\ \%.$$

5.67 The answer depends on exactly what rules the casino in question uses, but in principle yes. But a very good strategy is needed and enough skill to carry it out. The rules for Black Jack are so complicated that it makes sense to use a simulation to see if the strategy you have chosen is good enough to beat the bank on average.

5.69 $1/2$

5.70 $2/4 = 1/2$

5.71

(a) 2, 3, 5, 7, 11, 13, 17 are the seven first prime numbers.

(b) There are $\binom{35}{7}$ different Lotto rows. The probability that a certain row has all the numbers correct at a certain draw is thus $1/\binom{35}{7}$. The probability of not having all numbers correct is then $1-1/\binom{35}{7}$. The probability of not having all the numbers correct twelve times running is thus

$$\left(1-\frac{1}{\binom{35}{7}}\right)^{12} \approx 0.9999982156.$$

5.72 The first nail can be put in at one out of 8 places. The second one in one of the 4 holes at the other side. The third in one out of 6 places, and so on. In total $8\cdot4\cdot6\cdot3\cdot4\cdot2\cdot2\cdot1 = 9216$ ways.

5.73

(a) The first group can be paired with one of the other $2n-1$ ones. When this is done, the remaining group with the lowest number can be paired with one of the $2n-3$ remaining ones, and so on. This gives

$(2n-1)(2n-3)\ldots3\cdot1$ ways

Alternatively one can select 2 groups out of the $2n$ available ones, then 2 among the $2n-2$ remaining ones, and so on. But then every pairing has been counted $n!$ times, one for each way of ordering the pairs. Therefore the answer is

$$\binom{2n}{2}\binom{2n-2}{2}\cdots\binom{2}{2}\Big/n! \text{ ways}$$

This expression can be simplified to the expression in the first solution. (Do it!)

(b) {ab, cd, ef}, {ab, ce, df}, {ab, cf, de}, {ac, bd, ef}, {ac, be, df}, {ac, bf, de}, {ad, bc, ef}, {ad, be, cf}, {ad, bf, ce}, {ae, bc, df}, {ae, be, cf}, {ae, bf, cd}, {af, bc, de}, {af, bd, ce}, {af, be, cd}

5.74 If you have chosen the first three letters the last two letters are also given – they are to be the same as the first two ones.

(a) The first letter can be chosen in 4 ways, the same with the second and third one. The fourth letter is to be the same as the second one, so it can only be chosen in one way. The fifth one is to be the same as the first one, one way. This gives in total $4\cdot4\cdot4 = 4^3 = 64$

(b) $4\cdot3\cdot2 = 24$

(c) $64 - 24 = 40$

(d) We won't get any more because of that – the first three letters are chosen, and the following ones are the same.

5.76

(a) The juggler can choose five balls of different colours in $3\cdot4\cdot6\cdot2\cdot6 = 864$ ways, according to the multiplication principle.

(b) For it to be possible for the juggler to find five balls of the same colour the colour clearly has to be red or purple. There are $\binom{6}{5}$ ways of choosing five out of six red balls. There are the same number of ways of choosing five out of six purple balls. The addition principle shows that there are $\binom{6}{5}+\binom{6}{5} = 12$ ways of choosing five red *or* five purple balls.

5.77

(a) We use Pascal's recursion, which we have already proved:

$$\begin{aligned}&\binom{n}{k}+2\binom{n}{k-1}+\binom{n}{k-2} = \\ &= \left[\binom{n}{k}+\binom{n}{k-1}\right] \\ &\quad+\left[\binom{n}{k-1}+\binom{n}{k-2}\right] \\ &= \binom{n+1}{k}+\binom{n+1}{k-1} \\ &= \binom{n+2}{k}\end{aligned}$$

(b) If we are to take k objects among $n+2$ possible ones we can either take k out of the first n (and neither of the last 2) *or* we can take one of the last two and $k-1$ out of the n first *or* we can take both the last ones and $k-2$ out of the n first.

5.78

(a) It seems easiest to start with both sides, hoping to meet somewhere in the middle.

$$\begin{aligned}\text{LHS} &= \binom{r}{m}\binom{m}{k} \\ &= \frac{r!}{m!\,(r-m)!}\cdot\frac{m!}{k!\,(m-k)!} \\ &= \frac{r!}{(r-m)!\,k!\,(m-k)!} \\ \text{RHS} &= \binom{r}{k}\binom{r-k}{m-k} \\ &= \frac{r!}{k!\,(r-k)!} \\ &\quad\cdot\frac{(r-k)!}{(m-k)!\,((r-k)-(m-k))!} \\ &= \frac{r!}{k!\,(m-k)!\,(r-m)!}\end{aligned}$$

(b) The left-hand side seems to describe a selection in two steps: First you select a group of m persons among r applicants, and then you choose k persons among these m. Then you will have made k out of the r original applicants happy, and at the same time made $m-k$ out of the $r-k$ not chosen really angry.

5.79

(a) $\binom{16+12}{7} = \binom{28}{7} = 1{,}184{,}040$

(b) Proportional must mean 4 men and 3 women.

$$\binom{16}{4}\binom{12}{3} = 400{,}400$$

(c) We solve part (d) first, and use that answer.

$$\binom{16+12}{7}-\binom{16}{7}-\binom{12}{7} = 1{,}171{,}808$$

(d) $\binom{16}{7}+\binom{12}{7} = 12{,}232$

5.80

(a) $2^8 = 256$ (b) $\binom{8}{2} = 28$

(c) Half of them.

5.81 The binomial theorem shows

$$2^n = (1+1)^n = \sum_{k=0}^{n}\binom{n}{k}.$$

The right-hand side is the sum of the binomial coefficients on row n in Pascal's triangle.
To get a subset of size k out of a set with n elements we have to choose k out of the elements, which can be done in $\binom{n}{k}$ ways. But the size of k can vary between 0 and n, which are disjoint cases. Thus there is a total of $\sum_{k=0}^{n}\binom{n}{k} = 2^n$ subsets.

5.83

(a) Choose one out of 4 suits, and then 5 out of 13 cards in this suit.

$$4\cdot\binom{13}{5} = 5148$$

(b) Choose one out of 13 ranks in which to get the four cards, take all the cards in that rank and one of the remaining 48 cards.

$$13 \cdot 48 = 624$$

(c) Choose one out of 13 ranks and 3 out of 4 cards of that rank, and then one of the remaining 12 ranks and 2 out of those 4 cards.

$$13 \cdot \binom{4}{3} \cdot 12 \cdot \binom{4}{2} = 3744$$

(d) Choose 2 out of the 13 ranks to pick pairs out of, and 2 out of the 4 cards in both those ranks, and one of the remaining 44 cards that don't belong to any of the already chosen ranks.

$$\binom{13}{2} \cdot \binom{4}{2}^2 \cdot 44 = 123{,}552$$

Note that it's *not* possible to reason "choose one of the 13 ranks for the first pair and one of the remaining 12 for the second" since then all the cases will be counted twice. For instance the card hand shown on page 120 will be counted *both* as a hand containing firstly a pair of eights and secondly a pair of threes *and* as a hand containing firstly a pair of threes and secondly a pair of eights.

(e) Check exercise 3.1 on page 37 for some hints!

5.84

(a) To get one point a jack has to be selected together with 12 zero-point cards. That can be done in $\binom{4}{1}\binom{36}{12} \approx 5 \cdot 10^9$ ways

(b) The probability is the number of favourable cases divided by the total number of cases, that is

$$\binom{4}{1}\binom{36}{12} \Big/ \binom{52}{13} \approx 0.79\ \%$$

5.85

(a) A triangle corresponds to a selection of three out of the eight corners. $\binom{8}{3} = 56$ triangles.

(b) It seems simpler to count the forbidden triangles and subtract them from the total number. A triangle sharing two sides with the octagon must have three corners lying side by side on the periphery of the octagon. Such a set can be identified by the corner in the middle, which can be chosen in 8 ways. A triangle sharing just one side with the octagon consists of one of the 8 sides and one out of the 4 corners that aren't adjacent to the side. The number of permitted triangles is thus $56 - 8 - 8 \cdot 4 = 16$.

(c) $\binom{n}{3}$ (d) $\binom{n}{3} - n - n \cdot (n-4)$

5.86

(a) Take one digit for the 1:th position, can be done in 10 ways. Take a different digit for the 2:nd position, can be done in 9 ways. Take a 3:rd digit for the 3:rd place, can be done in 8 ways. Take one of the digits already used for the last position, can be done in 3 ways.

(b) This gives the codes where the second copy of the doubled digit appears *last*. But can't it be put in other places as well? As in 0012, 3435, or 6778?

(c) Choose a digit to use twice; this can be done in 10 ways. Choose two positions for this; it can be done in $\binom{4}{2}$ ways. Choose another digit for the first gap; this can be done in 9 ways. Choose a third digit for the second gap; this can be done in 8 ways. This gives $10 \cdot \binom{4}{2} \cdot 9 \cdot 8 = 4320$ codes (twice as many!).

5.88 $\lfloor x \rfloor$ denotes "x rounded downwards". "Coprime to 5" means "not a multiple of 5". The number of integers between 1 and 100 that are multiples of 5 are $100/5$. The same with 7. "Coprime to 9", on the other hand, is synonymous with "not a multiple of 3".

$$\begin{aligned} &100 - (\lfloor 100/5 \rfloor + \lfloor 100/7 \rfloor + \lfloor 100/3 \rfloor) \\ &\quad + (\lfloor 100/35 \rfloor + \lfloor 100/15 \rfloor + \lfloor 100/21 \rfloor) \\ &\quad - \lfloor 100/105 \rfloor \\ &= 100 - (20{+}14{+}33) + (2{+}6{+}4) - 0 \\ &= 45 \end{aligned}$$

5.89

(a) $40 = 2^3 \cdot 5^1$. When we want to find a divisor we have to take between zero and three out of the availabe 2s, and zero or one 5. This means that there ought to be $(3+1)(1+1) = 4 \cdot 2 = 8$ divisors of 40. In general, a number having the prime factorisation $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$ has $(e_1+1)(e_2+1)\cdots(e_k+1)$ divisors.

(b) $6 = 2 \cdot 3$. Numbers having the prime factorisation p^5 and numbers having the prime factorisation $p_1^2 p_2$ have exactly six divisors (p^0, p^1, p^2, p^3, p^4, and p^5, and $p_1^0 p_2^0$, $p_1^1 p_2^0$, $p_1^2 p_2^0$, $p_1^0 p_2^1$, $p_1^1 p_2^1$, and $p_1^2 p_2^1$, respectively).

5.90 We pick a person, A, and divide the rest of the company into two groups: those who know A and those who don't. One of the groups has to consist of at least three persons. Let's say that it is the group who knows A. If two out of those three persons know one another as well, the we have (if including A) three persons who do know one another. If there aren't two persons in the group that know one another, well then we have a group of three who don't know one another! Repeat the line of argument using the other group.

5.91 All integers can be written as an odd number times a power of two. For instance, $14 = 7 \cdot 2^1$, $15 = 15 \cdot 2^0$, $16 = 1 \cdot 2^4$. There exist 50 different odd numbers between one and one hundred. Because of this, the 51 numbers we have taken can't all have different "odd-parts", but at least two of them have the same. If the numbers can be written as $o \cdot 2^m$ and $o \cdot 2^n$ respectively (where $m < n$) the second one can be written as 2^{n-m} times the first one, and is thus divisible by it.

5.95 $\binom{2n}{n}$ is the number of ways of choosing n elements among $2n$ possible ones. Start by dividing the set into two equal parts. If we want n elements we can take n from the first part and none from the other or $n-1$ from the first part and 1 from the other or.... So

$$\begin{aligned} \binom{2n}{n} &= \binom{n}{n} \cdot \binom{n}{0} + \binom{n}{n-1} \cdot \binom{n}{1} \\ &\quad + \binom{n}{n-2} \cdot \binom{n}{2} + \dots \\ &= \binom{n}{n} \cdot \binom{n}{n} + \binom{n}{n-1} \cdot \binom{n}{n-1} \\ &\quad + \binom{n}{n-2} \cdot \binom{n}{n-2} + \dots \\ &= \sum_{k=0}^{n} \binom{n}{k}^2 \end{aligned}$$

(since $\binom{n}{k} = \binom{n}{n-k}$).

Note that not all sum formulas are best suited for proof by induction!

5.96 We won't write the complete solution here, but will give some hints.

That which is to be proved is that *both* the formulas hold, and as an induction hypothesis one has to use "assume that both formulas hold for $n = p$" after which one proves that they then hold for $n = p+1$. In the proof, Pascal's recursion has to be used, see page 122.

Chapter 6

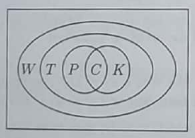
6.2 In non-discrete mathematics graphs of functions that illustrate the relationship between two variables (usually x and y) are common. An example is found on page 2.

6.3 CEO: 2; Director of Recreation: 3; Mayor: 2; Bicycle Councillor: 1. $2+3+2+1=8$, which is even.

6.4 When the sum of the degrees is calculated, each edge will add 2 to the sum, 1 for each end. Thus the sum of the degrees will be the number of edges times two, which is an even number.

6.5 Let the guests be nodes and draw an edge between two persons that have shaken hands. Then the guests having shaken hands an odd number of times correspond to the nodes having an odd degree. According to the previous exercise the sum of the degrees is an even number. Since the sum of the even degrees is always an even number, the sum of the odd degrees has to be even as well, which means that there has to be an even number of nodes having an odd degree. (You remember the rules for odd and even numbers?)

6.8



6.10 There are lots of examples; here are two!

(a) The network of roads in a city ought to be connected; it doesn't work if it's impossible to get from one part to another.

(b) The graph showing "who usally collaborates with whom" at a study programme at a university is often not connected. There are often groups who work with each other but not with anyone outside the group.

6.13 There is an edge between every pair of nodes. From n nodes $\binom{n}{2}$ pairs can be made.

6.14 From each of the m left nodes n edges start towards the right nodes. That gives mn edges in total.

6.15

(a) Yes. The central node is one part, the outer nodes the other.

(b) Yes. Points that are placed diagonally with respect to one another on a side of this cube belong to the same part.

(c) No. But it is *tripartite*. It is possible to divide the nodes into three groups, where there are no edges inside a group and all the edges are between nodes from different groups.

6.16 As a K_4 and a K_3. (All nodes inside the same part get connected, no edges between the parts.)

6.17 If all three of the nodes are included in the subgraph there are $2^3=8$ ways of choosing a subset of the edges. If only two of the nodes are to be included there are $\binom{3}{2}\cdot 2=6$ ways of choosing them and then decide whether the edge between them is to be included or not. If only one node is to be included there are $\binom{3}{1}=3$ ways if choosing it, and then no edge can be used. Lastly there is the empty subgraph that lacks both nodes and edges. In total, the triangle has $8+6+3+1=18$ subgraphs.

6.18 The complement graph has edges between persons that don't know each other.

6.19 Everyone knows everyone. This can occur for instance in a work team or in a family.

6.20 At classical pair dancing a woman always dances with a man in each pair. If we draw an edge between all those that dance with each other during the night the graph will be bipartite. (This will not hold on a dancing floor where for instance a ring dance is danced, or on courses in pair dancing where there is a surplus of one of the sexes and everyone wants to practise dancing.)

6.23

(a) Yes.

(b) Well, that depends a bit on what you are doing. If the only thing of interest is the number of exits from the node a loop is counted as 2 when calculating the degree. But if the interesting thing is the number of edges connected to the node it makes more sense to count it as 1! 2 can be said to be standard; if 1 is used it has to be pointed out.

(c) No. It has to be divided into "in-degrees", the number of edges arriving at the node, and "out-degrees", the number of edges departing from the node.

6.24 Yes. The same proof works.

6.25 The tip of a pen that draws all the edges in one move follows an Eulerian trail.

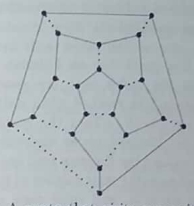
6.26 A graph having a Hamiltonian cycle can be drawn by first drawing the Hamiltonian cycle like a necklace with the nodes as pearls, and then drawing some extra threads for the remaining edges. Conversely, each such neclace with extra threads has a Hamiltonian cycle, namely the necklace itself.

6.27 None. According to the given dates of birth and death Hamilton died at the age of sixty.

6.28 Twenty.

6.29

(a)



(b) A route that visits every city but doesn't return to any of them before getting back to the starting point will in the graph show up as a closed path visiting all the vertices, that is, a Hamiltonian cycle. One is marked in the graph; there are several others.

6.30

(a) TCA → CAT → ATG → TGC → GCG → CGC → GCA is the unique Hamiltonian path in the graph, which gives us the DNA sequence TCATGCGCA.

(b) If a DNA sequence is broken up in three-letter subwords, consecutive words will automaticly have two letters in common, so that a directed edge is formed in the graph. Thus the whole sequence makes up a Hamiltonian path in the graph. A lot of other edges may be formed by chance, but if the Hamiltonian path is unique it has to correspond to the real sequence.

(c) If the words are AGA and GAG the sequence can have been both AGAG and GAGA.

6.32

6.33 Every walk has of course to arrive at a node the same number of times as it leaves it, if the node isn't the starting

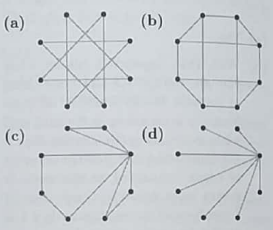
point or endpoint of the walk. But a circuit is closed and hasn't got any start or finish of that kind.

6.37 If a graph has an Eulerian circuit all the nodes that are touched by edges will be reached when walking along the circuit. Thus the graph is connected (isolated nodes discounted). When walking the Eulerian trail one enters each node the same number of times as one leaves it, and since each edge is used exactly once the degree of each node has to be even. For an Eulerian trail the same thing applies, except that the first and last node on the walk, if different, have an odd degree.

6.38 We are going to show that it is possible to make up an Eulerian circuit in an arbitrary graph that is connected (isolated nodes discounted) and where all the nodes have an even degree. Walk along randomly chosen edges in the graph until stopped at a node where all the edges have been used. Since all the nodes have an even degree this can only happen at our starting point. Remove all the edges belonging to this circuit (that we can call circuit A). Then we get a graph where it still holds that all the nodes have an even degree, since we have removed an even number of edges from each node. If some edge touching the old walk was unused we can walk along it and walk on along randomly chosen edges until stopped. In the same way as before, this can only happen at the starting point for this new walk, so now we have a new circuit, say circuit B. We can now weave the circuits A and B together into a single circuit: Start at the original starting point and follow circuit A until the starting point for circuit B is reached. Now follow cicuit B in its full length. Now we are back at the starting point of circuit B and can continue with the remainder of circuit A. If there are still some unused edges alongside the walk we make a new circuit and weave it together with the old one. Keep on like this until all the edges are used up.

For a graph with two nodes, say A and B with an odd degree an Eulerian trail is manufactured in the same way. The only difference is that the first walk is to start in one of these nodes, say A, and that it then will stop in B. After that circuits are woven in as above.

6.39 Of course there are lots (but not an infinite number) of answers to this exercise. These are suggestions:



6.40 Many answers exist! A trivial one can be the park keeper, who has to find an efficient route to empty the waste bins and an efficient route to drive the snow plough. Presumably you only want to pass each bin once, and plough each road once.

6.41 Since the starting point doesn't matter we choose a starting node at random, say v_1. From v_1 the next node in the cycle, say v_2, can be chosen among $n-1$ possible nodes (all except v_1). From v_2 in the same way the next node in the cycle, say v_3, can be chosen among $n-2$ possible nodes (all except for v_1 and v_2). In general, from node v_k the next node can be chosen in $n-k$ ways (all except $v_1, \ldots, v_k$). In total we thus get $(n-1)(n-2)\cdots = (n-1)!$ possible Hamiltonian cycles.

6.43 The nodes were given in different orders in the different matrices.

6.45

(a) Directed graphs where there is some edge that isn't bidirected.

(b) Graphs with loops.

6.46 In $4! = 24$ ways.

6.47 It is plain that the nodes with degree 1 and the nodes with degree 3 have to correspond to each other. But then we have two nodes that are tied to the one with degree 3 and connected to each other, and which of them is which can be chosen freely. $\phi(1) = \mathrm{C}$, $\phi(2) = \mathrm{D}$, $\phi(3) = \mathrm{A}$, and $\phi(4) = \mathrm{B}$ works just as well.

6.48 One isomorphism is given by $\phi(a) = 5$, $\phi(b) = 2$, $\phi(c) = 4$, $\phi(d) = 3$, and $\phi(e) = 1$.

6.49 Both graphs have two nodes of degree three and three nodes of degree two, but the graph on the left has a cycle of length three which the graph on the right lacks.

6.50 The graph on the left is isomorphic to the one on the right. The graph in the middle contains 4-cycles, which the other two don't.

6.51

	1	2	3	4	5	6
a	1	0	0	0	1	1
b	1	1	0	0	0	0
c	0	1	1	0	0	1
d	0	0	1	1	0	0
e	0	0	0	1	1	0

	7	8	9	10	11	12
f	1	1	1	0	0	0
g	1	0	0	1	0	0
h	0	1	0	0	1	0
i	0	0	1	0	0	1
j	0	0	0	1	1	1

6.52 The size of the adjacency matrix is $|V| \cdot |V|$ while the size of the incidency matrix is $|V| \cdot |E|$. Thus the incidency matrix will be smaller than the incidency matrix if the number of edges is smaller than the number of nodes.

6.53 Two graphs are isomorphic if their incidency matrices are alike, at least after permuting the rows.

6.56 Since a tree is already connected, another edge between two nodes will combined with the existing path between those nodes form a cycle.

6.57 If an edge is removed from a tree the two nodes will be separated from each other, since the tree doesn't have any cycles. If there had been an alternative path between them it would have formed a cycle combined with the edge removed.

6.58 Since the tree has at least two nodes it must have at least one edge. Start in the middle of this edge and walk away, to the left and to the right. Since there are no cycles, these walks have to end some time, each one in a leaf.

6.59 In every non-empty tree the number of nodes is one more than the number of edges.

6.61 We are to show that the relationship $|V| = |E| + 1$ holds for all trees.

Base step: For the tree having just one node ($|V| = 1$) and zero edges ($|E| = 0$) the relationship clearly holds.

Inductive step: Let n be an integer greater than one. Assume that the relationship holds for all the trees with $n-1$ nodes. We are to show that it then holds for each tree with n nodes. If we remove a leaf and the attached edge from a tree having n nodes, what remains is a tree having $n-1$ nodes, which, according the inductive hypothesis, has $n-2$ edges. Since we removed exactly one edge the original tree must have had $n-2+1 = n-1$ edges. Thus the relationship holds for trees having n nodes as well if it holds for trees having $n-1$ nodes. Since the relationship holds for all the trees with one node it then, according to the inductive principle, holds for all the trees having at least one node.

6.63 It's possible to remove any edge from the 4-cycle and any edge from the

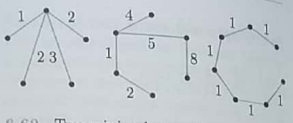
3-cycle, but it's not possible to remove the edge they have in common both times. Thus the graph has $4 \cdot 3 - 1 = 11$ spanning trees.

6.65 An n-cycle has n spanning trees; exactly one of the n edges is to be removed.

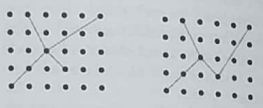
6.66 Kruskal's algorithm consists in each step of taking the lightest among the not yet chosen edges that doesn't form a cycle combined with the edges already chosen, until a spanning tree is formed. How do we know that the tree we have found using this algorithm really is the minimal spanning tree? Otherwise you would get the minimal spanning tree T_{min} by not choosing the lightest edge e_{light} available in some step and from this point forward only taking edges heavier than e_{light}. If we add e_{light} to the the tree T_{min} a cycle is formed. Some other edge in this cycle must have been added after the time when e_{light} was considered, and that edge must thus be heavier than e_{light}. If we remove that edge we get an even lighter spanning tree than T_{min}, but that is impossible since we have defined T_{min} as the minimal spanning tree! Thus Kruskal's algorithm has to give the minimal spanning tree.

6.67 Assume that the graph has n nodes and e edges. The spanning tree will have n nodes and $n-1$ edges, according to theorem 6.3. Thus $e-(n-1)$ edges have to be selected for removal. The alternative according to Kruskal is that $n-1$ edges are selected for inclusion. The latter is to be prefered if $n-1 < e-(n-1)$, that is, if the number of edges is greater than $2n-2$.

6.68 For the first two graphs the solution is unique; for the last one, on the other hand, there are seven equally good solutions. (We have contented ourselves with drawing one of them.)



6.69 Two minimal spanning trees exist:



6.72 A rooted tree is

- an empty tree, or
- a root below which a number of rooted trees are hanging.

6.73

(a) Breadth-first: a – b – c – d – e – f – g – h – i
Depth-first: a – b – d – e – h – i – c – f – g

(b) Breadth-first: 1 – 2 – 8 – 9 – 3 – 4 – 10 – 11 – 5 – 6 – 7 – 12 – 13.
Depth-first: 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13

6.74

(a) This is an implementation of a depth-first search.

(b) The problem is that there are no direct connections between nodes at the same level. That can be solved by making a list of the nodes step by step in the order they are to be visited. At the start, only the root is included in the list. When the first node is visited, its children are added last in the list, and the first node is removed. When this is repeated the effect is that one level at a time is visited.

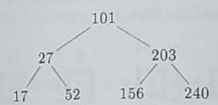
6.75 Breadth-first search. We want to search all the levels down to the first level where we find a game that ends with black winning.

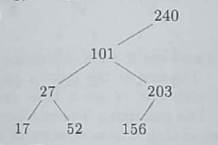
6.77 A binary tree is

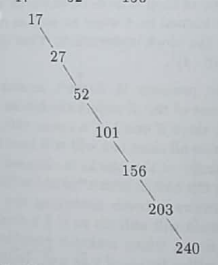
- an empty tree, or
- a root under which a binary tree to the left and a binary tree to the right are hanging.

6.78 10, 15, 16, 6, 2, 8, 13 for instance. The important thing is that we must have been given the *root* of a subtree earlier than the parts below. Except for that, the numbers may have arrived in any order.

6.79

(a)


(b)


(c)


6.80 Of course there can't be more levels below the root than there are nodes.

6.81 2^{n-1} ways. For each level we have to choose whether the node is to be placed to the left or to the right of the parent.

6.82 It is a bad thing if the sequence is sorted. Then the tree grows only to the right and gets maximally unbalanced.

6.83 All the trees will print in the following order: 17, 27, 52, 101, 156, 203, 240.

6.84
```
inorder(ROT) :
If ROOT isn't an empty tree,
    run inorder(ROOT.left),
    print ROOT.value,
    run inorder(ROOT.right).
```

6.85 Induction over the number of nodes in the binary tree: empty trees are of course printed in increasing order. For a non-empty tree our induction hypothesis that in-order will print all the smaller trees in increasing order. The subtrees on the left and right are smaller trees and will thus each be printed in increasing order. In an non-empty tree the subtree on the left will only contain numbers that are smaller than the root, and the subtree on the right only larger numbers. Thereby the whole tree will be printed in increasing order.

6.86

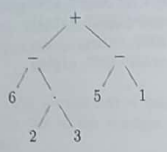
(a) Pre-order:
$+ \cdot 2\,3\,/\,/\,4\,5 - 6\,7$
Post-order:
$2\,3 \cdot 4\,5\,/\,6\,7 - /\,+$
Normal way:
$2 \cdot 3 + \frac{4/5}{6-7}$

(b) Pre-order:
$/\,/ - 9\,8 + 7\,6\,/ \cdot 5 - 4\,3 + 2\,1$
Post-order:
$9\,8 - 7\,6 + /\,5\,4\,3 - \cdot\,2\,1 + /\,/$
Normal way:
$\frac{9-8}{7+6} / \frac{5 \cdot (4-3)}{2+1}$

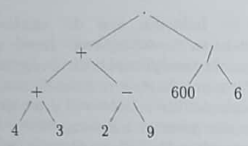
(c) Pre-order:
$+ \cdot + 4\,7 - 1\,1\,/ - \cdot 6\,7\,3\,7$
Post-order:
$4\,7 + 1\,1 - \cdot\,6\,7 \cdot 3 - 7\,/\,+$
Normal way:
$(4+7) \cdot (1-1) + \frac{6 \cdot 7 - 3}{7}$

6.87

(a)



(b)



(c) The expressions have the values $(6-2\cdot 3)+(5-1)=4$ and $((4+3)+(2-9))\cdot(600/6)=0$ respectively.

6.88

(a) The minus sign. Minus can be both a binary operation (as in $5-3$) and a unary operation (as in -17).

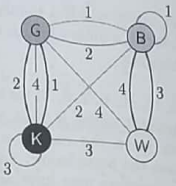
(b) In a binary expression tree unary operations are only given one child, which is usually placed on the right.

6.89 The graph is found in figure 3 on the facing page. The work will take 14 days in total. The tasks A, E, and L are time critical.

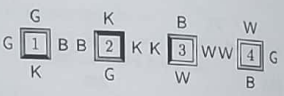
6.90 There are fewer ways of solving the problem in other ways for the first line, since it isn't possible to move words to or from the previous line (since no such line exists).

6.91 Most work according to the simple principle "squeeze as much as possible onto each line".

6.93 Here is the graph, one collection of edges in black, another one in blue and the edges we've chosen not to use in gray:



This selection of edges corresponds to the following placement of the blocks (if we let the black ones represent front and back and the blue ones right and left):



6.95 There are 4! ways of ordering the blocks. The one at the bottom can be placed with one of its 6 faces downwards. The remaining 3 blocks have to have one of their 6 faces downwards, and can be turned in 4 ways as well in relation to the block underneath. That gives $4!\cdot 6\cdot(6\cdot 4)^3$.

For the problem it doesn't matter in which one of the 4! orders the blocks are placed, since if one has a tower with all colours on all sides one will still have one if the order of the blocks is changed, as long as the blocks aren't turned around. Furthermore a tower satisfying the requirements will still do so if it's turned upside-down, which makes it possible to reduce with a factor of 2 as well. In total that gives

$$\frac{1{,}990{,}656}{4!\cdot 2}=41{,}472$$

cases to consider.

6.97

(a) 2.

(b) 4. Don't forget that all the answers in (a) are classified as trails as well!

(c) Any number. We may run along each edge backwards and forwards however many times we want to.

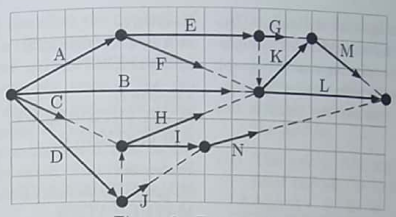
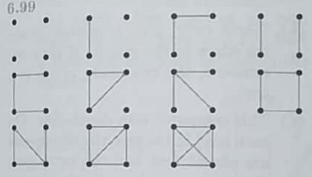
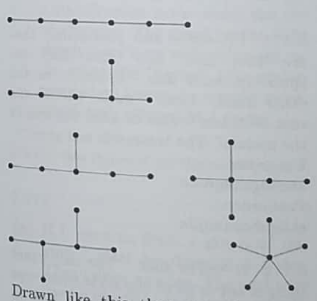


Figure 3: *Exercise 6.89*

6.99



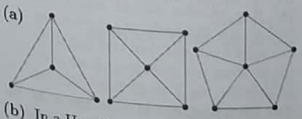
6.100



Drawn like this they strongly resemble different isomers of the hydrocarbon hexane. (Not the last one, though, since coal has no more than four valence electrons.)

6.101

(a)



(b) In a Hamiltonian cycle all the nodes are to be included, the hub among them. To reach the hub, we have to choose a spoke, then we have to exit via the one next to the right or to the left of the in-spoke. Then we follow the rim, to get the remaining nodes. The in-spoke can be chosen in n ways, then the out-spoke in 2, in total $2n$ Hamiltonian cycles.

6.102

(a)

(b) If we are to walk along an Eulerian circuit we have to walk around each leaf once. Our choice is firstly in which order we are to take the leaves, secondly if we are to walk around them clockwise or anticlockwise. If we decide upon a start leaf we can then pass the others in $(n-1)!$ different orders, and each of the n leaves in one of two directions as well. This gives $(n-1)!\,2^n$ circuits.

(c) For a spanning tree we have to remove one of the three edges on each of the n leaves. Can be done in 3^n ways.

6.103 The graphs are isomorphic; one bijection is: $\phi(a)=2$, $\phi(b)=3$, $\phi(c)=6$, $\phi(d)=1$, $\phi(e)=4$, $\phi(f)=5$. (There are other answers.)

6.104 They are not isomorphic, and the easieast way see that is to study

the complement graphs instead. (Isomorphic graphs have isomorphic complements.) The complement of the graph on the left is an 8-cycle, while the complement of the one on the right is two squares.

6.105

(a)

	a	b	c	d	e	f	g	h
a	0	1	1	0	1	0	1	1
b	1	0	1	1	0	1	0	1
c	1	1	0	1	1	0	1	0
d	0	1	1	0	1	1	0	1
e	1	0	1	1	0	1	1	0
f	0	1	0	1	1	0	1	1
g	1	0	1	0	1	1	0	1
h	1	1	0	1	0	1	1	0

(b)

	a	b	c	d	e	f	g	h
a	0	0	0	1	0	1	0	0
b	0	0	0	0	1	0	1	0
c	0	0	0	0	0	1	0	1
d	1	0	0	0	0	0	1	0
e	0	1	0	0	0	0	0	1
f	1	0	1	0	0	0	0	0
g	0	1	0	1	0	0	0	0
h	0	0	1	0	1	0	0	0

(c) Except on the diagonal: exchange 1 for 0 and vice versa.

(d) A lot more difficult, since one has to start by naming all the edges that aren't present in the graph.

6.106 Breadth-first:
TAYRHOUGPERFNH

Depth-first:
TARGHPYOEHRUFN

In-order:
GRAPHTHEORYFUN

Pre-order:
TARGHPYOEHRUFN

Post-order:
GRPHAHEROFNUYT

6.107
Breadth-first:
SCCASMIIVTITETT
VDREAEAHEHM

Depth-first:
SCAIVVSTIDHECMT
REEITATEHAM

In-order:
IVAVCTSHDEISRTE
MECATIEHTMA

Pre-order:
SCAIVVSTIDHECMT
REEITATEHAM

Post-order:
VIVATHEDISCRETE
MATHEMATICS

6.110

(a) Catalogues are inner nodes, documents are leaves.

(b) "Short cuts" give a number of extra edges in the graph. If they are removed, the remaining graph is a tree.

(c) This computer uses depth-first. On each level in the tree the documents are placed first and the catalogues after that, so that the documents are investigated before diving into the sub-catalogues.

6.111 Pre-order and post-order, that are "root first" and "root last", respecively, work fine. In-order, on the other hand, doesn't work; should the root be taken before or after the tree in the middle? The traversals will give:
Pre-order:
abefklgchdijmno
Post-order:
eklfgbhcimnojda.

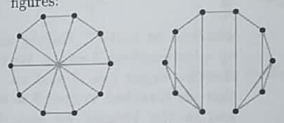
6.112 It works fine! The important thing is not to go to any node where you have already been. If we decide that at each crossroad where there is a choice we will choose the road on the right (seen from the point of view of the person walking) we will go
Breadth-first:
ad, ab, dg, de, bc, gh, ef, hi
Depth-first:
ad, dg, gh, hi, if, fc, cb, be

6.114

(a) The problem is the same as determining whether the connected

graph that describes the network of roads has an Eulerian circuit, that it has if and only if the degrees of all the nodes are even. The degree of a node is the same thing as the number of roads starting in the corresponding town, that is, the odd number three. Thus it's certain that the mule can't walk along all the roads.

(b) The problem is the same as determining whether the graph that describes the network has a Hamiltonian cycle. A graph with ten nodes and three edges starting at each node can both have and not have a Hamiltonian cycle, as seen in the figures:



The graph on the left has clearly got a Hamiltonian cycle along the circumference. The graph on the right clearly hasn't got a Hamiltonian cycle, since the "bridge" (the uppermost edge) has to be crossed twice, which isn't allowed in a cycle.

6.115 See figure 4 on the next page.

6.116

(a) If I want to draw a graph of this kind I start by drawing 4 nodes. Then I draw an edge. The next edge can be placed so that it doesn't connect to the previous one. Then the graph consists of two lines; the third line can in this case only be placed in such a way that it connects one end of one line with one edge of the other line, which is isomorphic to the graph in the middle. Or the second edge is placed in connection with the first one; the third edge can then be drawn from the node in the middle, and then I get something isomorphic to the graph on the left. Or the edge is drawn from an end node; then I can choose to let it lead to the other end node, which will be isomorphic to the graph on the right, or to the fourth node, which will be isomorphic to the graph in the middle.

(b) The two to the right. In the leftmost one all the nodes have an odd degree.

(c) The answer is four, since what we can choose is in which of the four nodes the three edges are to converge.

(d) In a graph with four nodes it's possible to draw at most $\binom{4}{2} = 6$ edges. If we only want three out of those, we can pick them out in $\binom{6}{3} = 20$ ways. Four out of those graphs are isomorphic to the leftmost graph, which was the only one not having an Eulerian trail. The probability in question is thus $4/20$.

6.117 There is room for 2^k nodes at level k in a binary tree. The lowest possible height of a binary tree with n nodes is thus the smallest number h such that $n \leq 1+2+2^2+2^3+\cdots+2^h = 2^{h+1}-1$, that is, the smallest number h that is greater than or equal to $(\log_2(n+1)) - 1$. In other words we have to round $(\log_2(n+1)) - 1$ upwards.

6.118 *Base:* If $n = 1$ there is of course only one complete ternary tree, and it has $3 = 2n + 1$ leaves.

Inductive step: For a given $k \geq 1$, assume that it is true that all ternary trees with $n = k$ inner nodes have $2k+1$ leaves. We then want to prove that all the ternary trees with $n = k+1$ inner nodes have $2(k+1)+1 = 2k+3$ leaves.

For an arbitrary tree of that kind we can choose a leaf and remove it together with its two sisters. Then an inner node is laid bare and turned into a leaf. We now have a complete ternary tree with k inner nodes and thus, according to the hypothesis, $2k+1$ leaves. Thus the original tree had $2k+1+3-1 = 2k+3$ leaves.

```
Add-to-tree(X, ROOT):
  If ROOT is an empty tree,
    return
      Build-tree(Empty-tree, X, Empty-tree).
  IF X>ROOT.value,
    return
      Build-tree(ROOT.left,
                 ROOT.value,
                 Add-to-tree(X, ROOT.right)),
  ELSE return
    Build-tree(Add-to-tree(X, ROOT.left),
               ROOT.value,
               ROOT.right)
```

Figure 4: *Exercise 6.115*

According to the inductive principle then all the complete ternary trees with $n \geq 1$ inner nodes have $2n+1$ leaves.

6.119

(a) If the numbers are to be different we can choose the number on the first edge in 10 ways, the second in 9, the third in 8, and the last in 7 ways, In total $10 \cdot 9 \cdot 8 \cdot 7 = 5040$ variants, whereas if the numbers may be alike each one of them can be chosen in 10 ways, which gives $10^4 = 10{,}000$ variants.

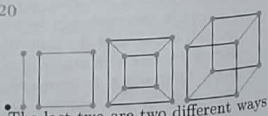
(b) We can place the 10 squares in a row in $10! = 3{,}628{,}800$ ways. Furthermore, each of the 10 squares can be rotated in 4 different ways. In total, that gives $10! \cdot 4^{10} = 3{,}805{,}072{,}588{,}800$ ways.

(c) We draw the graph, see figure 5 on the facing page. (The thick lines will be explained later.)

In the graph, we pick edges as follows: We *have* to take the loop attached to node 1, otherwise that node won't be represented. Then we can't use the other edge marked 2. Node 8 is only attached to two edges, so we have to pick them. That excludes the other edges marked 7 and 9. Furthermore, we have to take the two edges attached to node 7, thereby using up edge-numbers 8 and 3. We take the two edges (no. 6 and no. 10) that are attached to node 5 as well, and in the bargain we have chosen two edges attached to node 9. The only edge number connected to node 6 that hasn't already been picked is no. 1, so we take that. Out of the edges attached to node 0 only no. 4 can be used. And lastly we connect node 5 to itself. Now each node is attached to two edge ends, and all the edge-numbers have been used once!

The selection we made here can be used for the placement of the ten squares, so that each number will be represented twice: once on the upper edge and once on the lower edge of the row.

6.120

(a) 

The last two are two different ways of drawing the 3-dimensional figure. (One is a cube in central perspective, the other one a cube seen at an angle from above.)

(b) *Base:* The statement is true for

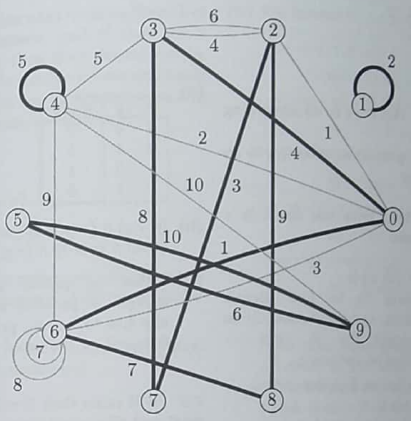


Figure 5: *Exercise 6.119*

$n = 0$, since the zero-dimensional hypercube has $1 = 2^0$ nodes.

Induction: Assume that a p-dimensional hypercube has 2^p nodes. When the $p+1$-dimensional hypercube is put together, two copies of the p-dimensional cube are drawn, and no nodes are added in other places. Therfore the $p+1$-dimensional cube will include the same number of nodes as two p-dimensional ones, which (according to the hypothesis) will mean $2 \cdot 2^p = 2^{p+1}$ nodes.

(c) *Base:* The statement is true for $n = 0$, since a zero-dimensional hypercube has zero edges, and $0 = 0 \cdot 2^{0-1}$.

Induction: Now assume that a p-dimensional hypercube has $p \cdot 2^{p-1}$ edges. The $p+1$-dimensional hypercube will include all the edges already present in the two p-dimensional hypercubes that it is based on, and on top of that edges between these hypercubes. One connecting edge starts from each node in one of the hypercubes to the corresponding node in the other, so the number of edges of this kind must be equal to the number of *nodes* in the p-dimensional hypercube, which according to the previous proof is 2^p.

So the $p+1$-dimensional hypercube has (according to the hypothesis and the previous argument) $2 \cdot p \cdot 2^{p-1} + 2^p = p \cdot 2^p + 2^p = (p+1)2^p$ edges.

(d) We won't print the complete argument, but it is necessary to explain both why all the nodes get different names and why the names of adjacent nodes differ in exactly one place. (These things combined show that a Hamiltonian cycle generates a Gray code.)

Chapter 7

7.1

(a) *If*
there are two ways to do something
or
there are more than two ways to do something
and
one of those ways can result in a catastrophe
then
someone will do it
There seem to be four atomic propositions, and three connectives (if we count the "if...then..."-composition as one unit).

(b) A graph has an Eulerian circuit
if and only if
the graph is connected
and
all the nodes have an even degree
Three atomic propositions and two connectives (of which one has the unwieldy formulation "if and only if").

7.2 For instance "Sweden isn't part of America" and "$2 + 2 \neq 5$", respecively.

7.3 For instance "We start at 8 o'clock in the morning and I don't like rising that early" and "6 has the divisors 2 and 3".

7.4 Gray-code; see page 78, for instance.

7.5 For instance "it's wet outdoors, or it doesn't rain" and "an integer greater than 1 is composite or prime".

7.6 Out of the statements given here the first one is an and/or statement, because it's possible that both parts are true at once, for instance when it just stopped raining. The other one, on the other hand, is an either/or statement, since numbers can't be both prime and composite at the same time. How the case may be with the statement you choose yourself we can't tell.

7.7

(a)

p	q	$p \oplus q$
0	0	0
0	1	1
1	0	1
1	1	0

(b) $(p \vee q) \wedge \big(\neg(p \wedge q)\big)$ or alternatively $\big((\neg p) \wedge q\big) \vee \big(p \wedge (\neg q)\big)$

(c) Easier to prove, among other things. A proof of *either/or* usually has to be one proof of *and/or* followed by the proof of *not both*.

7.9 "If it rains then it will be wet outside" and "if a number is greater than a million then it is greater than zero" are two examples.

7.10 The reversal doesn't hold in either of the examples given here. It can be wet outside even if it's not raining (thaw, for instance), and for instance 17 is greater than zero without being greater than a million.

7.11 We start by assuming that the speaker is telling the truth. The last sub-clause in the implication, that is "I want to eat my hat" is a clearly false statement. If an implication with a false end clause is to be counted as true, the beginning has to be false as well. So the speaker wants to hint that "You win the game" is a false statement. (This shows that not just mathematicians and logicians use the truth-table for implication given here, but normal people do as well.)

7.12 "Clothes should be washed if and only if they need washing" (at least if you want to take both the environment and hygiene into account) and "that a number is even is the same thing as that it is equal to zero modulo two".

7.13 "If clothes need washing then they should be washed, and if they don't need washing then they should not be washed" and "a number is even and equal to zero modulo two or it's not even and not equal to zero modulo two".

7.14

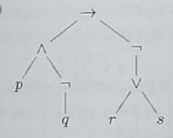
(a) $\big((p \vee q) \wedge r\big) \rightarrow s$

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

7.18 No. The tables are alike irrespective of the placement of the parentheses.

7.19 $p \vee \neg q \rightarrow \neg r \wedge q$. All the parentheses were unnecessary!

7.20



7.24
$$
\begin{aligned}
& p \rightarrow q \vee r \\
& \quad \Leftrightarrow \quad \text{Rewr. impl.} \\
& \neg p \vee q \vee r \\
& \quad \Leftrightarrow \quad \text{Commutative law} \\
& q \vee \neg p \vee r \\
& \quad \Leftrightarrow \quad \text{Double negation} \\
& \neg(\neg q) \vee \neg p \vee r \\
& \quad \Leftrightarrow \quad \text{Rewr. impl.} \\
& \neg q \rightarrow \neg p \vee r
\end{aligned}
$$

"If he is happy then he has passed the exam or won the pools" has the correct structure, and can be formulated as "if he hasn't passed the exam then he isn't happy or he has won the pools".

7.25 No. The expression is false for $p = 1$, $q = 0$, $r = 1$, $s = 0$, $t = 0$, and true for all other combinations of values.

7.26

(a) We skip this one; it takes lots of space.

(b) For instance:
$$
\begin{aligned}
& (p \rightarrow q) \wedge p \rightarrow q \\
& \quad \Leftrightarrow \quad \text{Rewr. impl} \\
& \neg\big((\neg p \vee q) \wedge p\big) \vee q \\
& \quad \Leftrightarrow \quad \text{De Morgan} \\
& \big(\neg(\neg p \vee q) \vee \neg p\big) \vee q \\
& \quad \Leftrightarrow \quad \text{Assoc. law} \\
& \neg(\neg p \vee q) \vee (\neg p \vee q) \\
& \quad \Leftrightarrow \quad \text{Com. law} \\
& (\neg p \vee q) \vee \neg(\neg p \vee q) \\
& \quad \Leftrightarrow \quad \text{Inverse} \\
& \quad 1
\end{aligned}
$$
Note that when rewriting the implication in the first step it's the *whole* left part that is to be negated, which means that we have to enclose it with parentheses.

7.27

(a) "If it is Sunday, then the bank will be closed, and the bank isn't closed. So it can't be Sunday", for instance.

(b) We skip this one.

(c) For instance:
$$
\begin{aligned}
& (p \rightarrow q) \wedge \neg q \rightarrow \neg p \\
& \quad \Leftrightarrow \quad \text{Rewr. impl} \\
& \neg\big((\neg p \vee q) \wedge \neg q\big) \vee \neg p \\
& \quad \Leftrightarrow \quad \text{De Morg; D. neg.} \\
& \big(\neg(\neg p \vee q) \vee q\big) \vee \neg p \\
& \quad \Leftrightarrow \quad \text{Assoc. law} \\
& \neg(\neg p \vee q) \vee \big(q \vee \neg p\big) \\
& \quad \Leftrightarrow \quad \text{Com. law} \\
& (\neg p \vee q) \vee \neg(\neg p \vee q) \\
& \quad \Leftrightarrow \quad \text{Inverse} \\
& \quad 1
\end{aligned}
$$

7.28 $A^c = \{\, x \mid \neg(x \in A) \,\}$
$A \cap B = \{\, x \mid x \in A \wedge x \in B \,\}$
$A \cup B = \{\, x \mid x \in A \vee x \in B \,\}$
That $\cap$ is turned in the same way as $\wedge$ and $\cup$ as $\vee$ is not chance but deliberate.

7.30 If you decide that anything that isn't zero is one, most of the rules seem

OK. But the distrubutive law on the left may take some time getting used to.

7.31

(a) $\neg\big(x \vee (\neg y \wedge z)\big)$

(b) $\overline{x\overline{y} + \overline{z}\,\overline{w}}$

An observation is that the digital-technical notation takes up a lot less space.

7.33 Demonstration of the calculation of the first row:

$$(0+\overline{0})\overline{0} + \overline{0\cdot 0}\cdot 0 =$$
$$= (0+1)1 + 1\cdot 0 = 1$$

The complete table is:

x	y	z	w	$f(x,y,z,w)$
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

7.34 2^{2^n}, since there are 2^n possible combinations of values, and each combination is to be coupled to one out of two possible values. The number is large, but not infinite.

7.35 Write down the tables of values for the expressions, and check that they correspond to the one given in example 7.6. Alternatively, rewrite the expressions into each other using the calculation rules, but that is fairly difficult!

7.37

$$f(x,y,z,w) = (x+\overline{y})\overline{z} + \overline{w}\overline{z}y$$
distr. law, De Morgan
$$= x\overline{z} + \overline{y}\,\overline{z} + (\overline{w}+\overline{z})y$$
distr. law
$$= x\overline{z} + \overline{y}\,\overline{z} + y\overline{w} + yz$$
inverse and identity laws
$$= x(y+\overline{y})\overline{z}(w+\overline{w}) + (x+\overline{x})\overline{y}\,\overline{z}(w+\overline{w}) + (x+\overline{x})y(z+\overline{z})\overline{w} + (x+\overline{x})y\overline{z}(w+\overline{w})$$
distr. law
$$= xy\overline{z}w + xy\overline{z}\,\overline{w} + x\overline{y}\,\overline{z}w + x\overline{y}\,\overline{z}\,\overline{w} + x\overline{y}\,\overline{z}w + x\overline{y}\,\overline{z}\,\overline{w} + \overline{x}y\overline{z}\,\overline{w} + \overline{x}y\overline{z}\,\overline{w} + \overline{x}y\overline{z}w + \overline{x}y\overline{z}\,\overline{w} + \overline{x}y\overline{z}w + \overline{x}y\overline{z}\,\overline{w}$$
See below
$$= \overline{x}\,\overline{y}\,\overline{z}\,\overline{w} + \overline{x}\,\overline{y}\,\overline{z}w + \overline{x}y\overline{z}\,\overline{w} + \overline{x}y\overline{z}w + x\overline{y}\,\overline{z}\,\overline{w} + x\overline{y}\,\overline{z}w + xy\overline{z}\,\overline{w} + xy\overline{z}w$$

Last step: Sorting and removal of duplicates, which uses the associative law, the commutative law and the idempotence law.

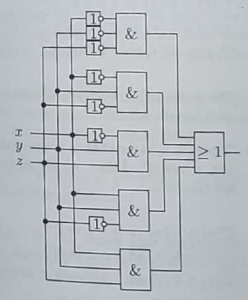
7.39 An extended table of values is found in table 1 on the next page. From that we can get the function in normal form:

$$f(x,y,z,w) = \overline{x}\,\overline{y}\,\overline{z}\,\overline{w} + \overline{x}\,\overline{y}\,\overline{z}w + \overline{x}y\overline{z}\,\overline{w} + \overline{x}y\overline{z}w + \overline{x}y\overline{z}\,\overline{w} + x\overline{y}\,\overline{z}\,\overline{w} + x\overline{y}\,\overline{z}w + xy\overline{z}\,\overline{w} + xy\overline{z}w + xyz\overline{w}$$
$$= (x+y+\overline{z}+w)(x+y+\overline{z}+\overline{w})\cdot (x+\overline{y}+\overline{z}+\overline{w})(\overline{x}+y+\overline{z}+w)\cdot (\overline{x}+y+\overline{z}+\overline{w})(\overline{x}+\overline{y}+\overline{z}+\overline{w})$$

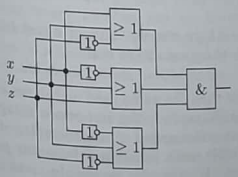
7.41 Disjunctive normal form:

x	y	z	w	$f(x,y,z,w)$	minterms	maxfactors
0	0	0	0	1	$\overline{x}\,\overline{y}\,\overline{z}\,\overline{w}$	
0	0	0	1	1	$\overline{x}\,\overline{y}\,\overline{z}w$	
0	0	1	0	0		$x+y+\overline{z}+w$
0	0	1	1	0		$x+y+\overline{z}+\overline{w}$
0	1	0	0	1	$\overline{x}y\overline{z}\,\overline{w}$	
0	1	0	1	1	$\overline{x}y\overline{z}w$	
0	1	1	0	1	$\overline{x}yz\overline{w}$	
0	1	1	1	0		$x+\overline{y}+\overline{z}+\overline{w}$
1	0	0	0	1	$x\overline{y}\,\overline{z}\,\overline{w}$	
1	0	0	1	1	$x\overline{y}\,\overline{z}w$	
1	0	1	0	0		$\overline{x}+y+\overline{z}+w$
1	0	1	1	0		$\overline{x}+y+\overline{z}+\overline{w}$
1	1	0	0	1	$xy\overline{z}\,\overline{w}$	
1	1	0	1	1	$xy\overline{z}w$	
1	1	1	0	1	$xyz\overline{w}$	
1	1	1	1	0		$\overline{x}+\overline{y}+\overline{z}+\overline{w}$

Table 1: *Table of values for exercise 7.39.*



Conjunctive normal form:



7.42 The output is 1 only if all the input signals are 0.

7.43

(a)
$$p \to q$$
$$\Leftrightarrow$$
$$\neg p \vee q$$
$$\Leftrightarrow$$
$$\neg(p \wedge \neg q)$$
$$\Leftrightarrow$$
$$\neg\big(p \wedge \neg(q \wedge q)\big)$$
$$\Leftrightarrow$$
$$p \mathbin{\overline{\wedge}} (q \mathbin{\overline{\wedge}} q)$$

(b) "It's not true that you wash while it's not true that I dry and I dry", is an effort. (This shows perhaps why the decision has been made not to use the fact that everything can be expressed using NAND in propositional logic.)

7.44 For instance the monadic predicate "is a woman", the dyadic predicate "is the mother of", and the triadic predicate "are the parents of". Usage: "Silvia is a woman", "Silvia is the mother of Victoria", "The King and Silvia are the parents of Victoria" are all true statements.

7.45 For instance "is even" "is a divisor of" "... plus ... equal ...". Usage:

"2 is even" "3 is a divisor of 6", "2 plus 2 equals 4" are all true statements.

7.46

$$\underbrace{\text{For every individual in the universe}}_{\forall}$$
$$\underbrace{\text{call the said person } x}_{x}$$
it is true that if
$$\underbrace{\text{the said person is a student}}_{S(x)}$$
$$\underbrace{\text{then}}_{\rightarrow}$$
$$\underbrace{\text{the said person is lazy}}_{L(x)}$$
and
$$\underbrace{\text{It's not true that}}_{\neg}$$
$$\underbrace{\text{there is some individual in the universe}}_{\exists}$$
$$\underbrace{\text{call the said person } x}_{x}$$
such that
$$\underbrace{\text{the said person is a teacher}}_{T(x)}$$
$$\underbrace{\text{and}}_{\wedge}$$
$$\underbrace{\text{the said person is reasonable}}_{R(x)}$$

7.47 "There is one object, call it x, having property P, and one object, call it y, having property P, and these two objects are not identical."

$$\exists x \exists y \left(P(x) \wedge P(y) \wedge \neg(x = y)\right)$$

There is no problem with writing $x \neq y$ instead of $\neg(x = y)$, if that's preferred.

7.48

(a) False, a isn't a consonant.

(b) True.

(c) True.

(d) False.

(e) "All the letters are consonants". False, since $C(a)$ is false.

(f) "Some letter is a consonant". True, since $C(b)$ is true.

(g) "All the letters are vowels". False, since $V(b)$ is false.

(h) "Some letter is a vowel". True, since $V(a)$ is true.

(i) "Every letter is a consonant or a vowel". True, which we can see this way:

$$\begin{aligned} C(a) \vee V(a) &\Leftrightarrow 0 \vee 1 \Leftrightarrow 1 \\ C(b) \vee V(b) &\Leftrightarrow 1 \vee 0 \Leftrightarrow 1 \\ C(c) \vee V(c) &\Leftrightarrow 1 \vee 0 \Leftrightarrow 1 \\ C(c) \vee V(d) &\Leftrightarrow 1 \vee 0 \Leftrightarrow 1 \\ C(e) \vee V(e) &\Leftrightarrow 0 \vee 1 \Leftrightarrow 1 \\ C(f) \vee V(f) &\Leftrightarrow 1 \vee 0 \Leftrightarrow 1 \end{aligned}$$

The proposition was true for all possible values of u.

(j) "There is a letter that is a consonant and a vowel". False, which we can see this way:

$$\begin{aligned} C(a) \wedge V(a) &\Leftrightarrow 0 \wedge 1 \Leftrightarrow 0 \\ C(b) \wedge V(b) &\Leftrightarrow 1 \wedge 0 \Leftrightarrow 0 \\ C(c) \wedge V(c) &\Leftrightarrow 1 \wedge 0 \Leftrightarrow 0 \\ C(d) \wedge V(d) &\Leftrightarrow 1 \wedge 0 \Leftrightarrow 0 \\ C(e) \wedge V(e) &\Leftrightarrow 0 \wedge 1 \Leftrightarrow 0 \\ C(f) \wedge V(f) &\Leftrightarrow 1 \wedge 0 \Leftrightarrow 0 \end{aligned}$$

There was no value of v that made the proposition true.

Note that the name used on the variable doesn't matter; the important thing is what will happen when we replace the variable with an object from the universe. In this case we should avoid the names a–f, since they are objects in the universe. But apart from that, we may use any symbol we want.

7.49

$$\forall x \Big\{ P(x) \wedge \underbrace{\exists y\, Q(y,x)}_{y} \rightarrow \underbrace{\forall z \big(R(x,z) \vee \underbrace{\exists w\, S(z,w)}_{w} \big)}_{z} \Big\}$$

x

7.50 The y in $Q(y)$ belongs to the all-quantifier, the other two belong to the existence-quantifier.

7.52 The implication in the bottom step can be turned, to $\forall x\big(R(x) \rightarrow \neg T(x)\big)$, $\forall x\big(\mathit{Reasonable}(x) \rightarrow \neg \mathit{Teacher}(x)\big)$, "Reasonable beings aren't teachers".

7.53 The original x in $P(x,y)$ would suddenly be regarded as linked to the existence-quantifier, since that one is nearer, instead of to the all-quantifier.

7.54 The x in $Q(x)$, which belongs to the existence-quantifier, would instead be linked to the all-quantifier.

7.56 $\forall x \forall y \big(G(y) \vee \neg P(x)\big)$. (There are altenative solutions.)

7.57 In the latter case everybody has to have the *same* name, which isn't required in the first case.

7.58 If we are discussing the positive real numbers and let $P(x,y)$ stand for "x has the inverse y" then $\forall x \exists y\, P(x,y)$ means "every number has an inverse" which is true, while $\exists y \forall x\, P(x,y)$ means "there is a number which is the inverse of all numbers" which hardly holds.

7.59 If we study a group of people, and $P(x,y)$ would mean x is a citizen and y is king then the expressions mean "All are citizens and there is a king" and "There is a king and all are citizens", respectively, and those statements are equivalent.

7.60 No. If the same y is good enough for all x's then each x is guaranteed to be able to find a y.

7.61 $\exists x \exists y \big\{ \mathit{Exam}(y) \wedge \mathit{Designed}(x,y) \wedge \forall z \big(\mathit{Student}(z) \rightarrow \mathit{Hates}(x,z) \big) \big\}$

"There is someone, call them x, and something, call it y, such that y is an exam and x has designed y, and for each individual (call them z) it holds that if z is a student then x hates z." Or put more briefly: "There is an exam designer who hates all students". (We prefer not to discuss the truth of this sentence.)

7.62

(a) "Every object in the whole universe (apples included) is both a student and hard-working".

(b) "There is something in the universe such that if it's a student then it's hard-working as well". That was true at the time of the dinosaurs, because if we put a *Tyrannosaurus rex* into the sentence the first clause is false, and the expression thereby true. But the sentence we were to translate was *not* true at that time!

7.63

(a) "For all pairs of things it's true that if the first one is an exam and the second one has designed the first one the second one thinks that the first one is good" or in *good* English: "Whoever has designed an exam thinks that it's good".

(b) "If no-one who has been tested on an exam had passed it, the designers of the exam won't be happy".

(c) "A student who has been tested on an exam and has passed it is happy and thinks that the exam was good".

7.64 There are several alternative solutions to each subexercise.

(a) "For each pair of things it's true that if the first one is a student and the second one an exam and the first one hasn't passed the second one then the first one doesn't think that the second one was good" or using symbols:

$$\forall x \forall y \big(S(x) \wedge E(y) \wedge \neg P(y,x) \rightarrow \neg G(y,x)\big)$$

(b) $\exists x \exists y \big\{E(x) \wedge D(y,x) \wedge \big(\exists z\, P(z,x) \rightarrow H(y)\big)\big\}$

(c) $\forall x \big\{E(x) \rightarrow \big[\forall y(T(y,x) \rightarrow P(y,x)) \rightarrow \neg\exists z \neg G(z,x)\big]\big\}$

Note the placement of the parentheses!

7.65 "For all objects it holds that if the object has the property P then there exists something having the property P".

A possible interpretation: as a universe we take humanity, and decide that $P(x)$ means "x is over 100 years of age". The proposition then means "For each human being it is the case that if the said person is over 100 years old then there exists someone who is over 100 years".

This is for instance true if by "humanity" we mean the students at the university, because there the first clause of the implication is false and the implication thereby true in all cases. Since we have found an interpretation that makes the proposition true it can't be a contradiction. (Actually, it's a tautology.)

7.70

(a) Tautology. **(b)** Neither.

(c) Contradiction. **(d)** Neither.

(e) Subexercise (a) was a tautology:

$$\begin{aligned} &(\neg q \wedge p) \vee (p \rightarrow q) \\ &\quad\Leftrightarrow \quad \text{D. neg; rewr. impl.} \\ &(\neg q \wedge \neg(\neg p)) \vee (\neg p \vee q) \\ &\quad\Leftrightarrow \quad \text{De Morgan} \\ &\neg(q \vee \neg p) \vee (\neg p \vee q) \\ &\quad\Leftrightarrow \quad \text{Com. law} \\ &(\neg p \vee q) \vee \neg(\neg p \vee q) \\ &\quad\Leftrightarrow \quad \text{Inverse law} \\ &\quad 1 \end{aligned}$$

Subexercise (b) was a contradiction:

$$\begin{aligned} &(p \wedge q) \wedge \neg\big((p \leftrightarrow q) \vee (p \vee q)\big) \\ &\quad\Leftrightarrow \quad \text{De Morgan} \\ &(p \wedge q) \wedge \neg(p \leftrightarrow q) \wedge \neg(p \vee q) \\ &\quad\Leftrightarrow \quad \text{Com. law} \\ &(p \wedge q) \wedge \neg(p \vee q) \wedge \neg(p \leftrightarrow q) \\ &\quad\Leftrightarrow \quad \text{De Morgan} \\ &(p \wedge q) \wedge (\neg p \wedge \neg q) \wedge \neg(p \leftrightarrow q) \\ &\quad\Leftrightarrow \quad \text{Rewr. equiv.} \\ &\quad (p \leftrightarrow q) \wedge \neg(p \leftrightarrow q) \\ &\quad\Leftrightarrow \quad \text{Inverse law} \\ &\quad 0 \end{aligned}$$

Of course, there are several alternative solutions.

7.71

(a) Neither. **(b)** Contradiction.

(c) Tautology. **(d)** Neither.

(e) Too much work to print!

7.72

(a)

x	y	z	f_1
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

(b) For instance

$$\begin{aligned} f_1(x,y,z) &= xz + xy + x\overline{y}z + y\overline{z} \\ &= (x + \overline{y} + \overline{z})(x + y)(\overline{x} + y + z) \end{aligned}$$

(c) $f_2(x,y,x) = x\overline{y}z + xy\overline{z} + xyz$
$= (x + y + z)(x + y + \overline{z})(x + \overline{y} + z)$
$\cdot (x + \overline{y} + \overline{z})(\overline{x} + y + z)$

(d) Takes too much space.

7.73

(a)

x	y	z	f_2
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

(b) For instance

$$f_2(x,y,z) = xy + xz = x(y + z)$$

(c) $f_2(x,y,z) = x\overline{y}z + xy\overline{z} + xyz$
$= (x + y + z)(x + y + \overline{z})(x + \overline{y} + z)$
$\cdot (x + \overline{y} + \overline{z})(\overline{x} + y + z)$

7.74

(a) True, 0 is divisible by all numbers and then, among others, by 2.

(b) False, 1 isn't divisible by 3.

(c) True, since for instance 4 is divisible by 2.

(d) False, since for instance 5 isn't divisible by 3.

(e) True, 6 is divisible by both 2 and 3.

(f) False, for instance 7 isn't divisible by either of 2 and 3.

7.75 For instance (for there are several correct solutions):

(a) $\forall x \big\{Z(x) \wedge \neg N(x) \rightarrow \exists y\big(Z(y) \wedge I(y,x)\big)\big\}$

(b) $\exists x\big(Z(x) \wedge I(x,x)\big)$

(c) $\exists x \big\{Z(x) \wedge \exists y\big(Z(y) \wedge I(y,x)\big)\big\} \wedge \exists x \big\{Z(x) \wedge \neg\exists y\big(Z(y) \wedge I(y,x)\big)\big\}$

(d) $\forall x \big\{Z(x) \wedge \neg\exists y\big(Z(y) \wedge I(y,x)\big) \rightarrow N(x)\big\}$

7.76

(a) $\forall x\big(M(x) \rightarrow S(x,x)\big)$

(b) $\exists x \big\{M(x) \wedge \forall y\big(M(y) \rightarrow S(x,y)\big)\big\}$

(c) $\forall x \forall y \big\{M(x) \wedge M(y) \wedge S(x,y) \rightarrow \forall z\big(E(z,x) \rightarrow E(z,y)\big)\big\}$

(d) $\forall x \forall y \big\{M(x) \wedge M(y) \wedge D(x,y) \rightarrow \neg\exists z\big(E(z,x) \wedge E(z,y)\big)\big\}$

7.77 "A graph that has an Eulerian circuit is connected". Not quite true; the graph may consist of several components, where all components except one consist of a single node.

7.78 "A graph that isn't connected can't have an Eulerian circuit". The same objection as in the previous exercise!

7.79 $\exists x \forall y\big(\neg P(x) \vee \neg Q(y) \vee \neg R(y)\big)$. (There are equivalent solutions.)

7.80 $\forall x \forall y\big(P(x) \wedge \neg Q(y)\big)$. As a matter of fact, it can be shortend to $\forall x\big(P(x) \wedge \neg Q(x)\big)$, but then you have to use some predicate-logical rules that we haven't covered in this book.

7.81

(a) Draw a truth-table, or use rules of equivalence.

(b) For instance "if it pours down or I'm in a hurry I take the bus to the university", which reformulated becomes "if it pours down I take the bus to the university, and if I'm in a hurry I take the bus to the university as well".

(c) The left-hand variant is a very common structure of statements. The right-hand variant is very easy to prove (prove the parts independently). That means that statements with this structure should be reformulated before being proved.

7.82

(a) Draw a truth-table, or use rules of equivalence.

(b) "If I pass the exam, I will, if I have money left, go to the pub" can be reformulated as "if I pass the exam and have money left I will go to the pub".

(c) The right-hand variant is (somewhat) simpler to express and prove, so statements with the left-hand structure gain by being reformulated before being proved.

7.83 When working with computers it can be seen that the assertion that statements are either true or false doesn't quite accord with reality. They can have the truth-value "makes the computer crash" as well. (Division by zero is a typical example.) $p \wedge q \to r$ and $q \to (p \to r)$ are logically equivalent, but

```
if ((1/x > 1) && (x != 0))
    y = x+1;
```

crashes when $x = 0$, while the logicaly equivalent code

```
if (x != 0)
    if (1/x > 0)
        y = x+1;
```

works. When programming, time spent trying to find out the most efficient way of expressing a logical condition is often well spent.

7.85 A path is a trail without repeated nodes. Analyse "trail that isn't a path":

$$\begin{aligned}
&\text{trail} \wedge \neg\text{path} \\
&\qquad \Leftrightarrow \quad \text{Def. path} \\
&\text{trail} \wedge \neg(\text{trail} \wedge \neg\text{repeated nodes}) \\
&\qquad \Leftrightarrow \quad \text{De Morgan} \\
&\text{trail} \wedge (\neg\text{trail} \vee \text{repeated nodes}) \\
&\qquad \Leftrightarrow \quad \text{Distr. law} \\
&(\text{trail} \wedge \neg\text{trail}) \vee (\text{trail} \wedge \text{repeated nodes}) \\
&\qquad \Leftrightarrow \quad \text{Inverse} \\
&0 \vee (\text{trail} \wedge \text{repeated nodes}) \\
&\qquad \Leftrightarrow \quad \text{Identity} \\
&\text{trail} \wedge \text{repeated nodes}
\end{aligned}$$

A trail that isn't a path is a trail that has repeated nodes.

7.86

(a) Truth-table, rewriting, or reasoning according to example 7.4 on page 189 works.

(b) "The maid or the butler committed the murder and it wasn't the maid, so it must have been the butler".

(c) This is known as the **process of elimination**, where one finds out what is true by removing all the cases that can't be until only one is left.

7.87

(a) "If there is something with both the property P and property Q, then there is firstly something having the property P and secondly something having property Q.

(b) Well, if there is something having both the properties there plainly exsists something – namely this object – having the first property, and something – this object as well – having the second property.

(c) No. The reversal could be interpreted as "if there is something that is a pawn and something that is a queen then there is something that is both pawn and queen", which doesn't work out that well. (The pawn and the queen don't have to be the *same* playing piece!)

7.88

(a) *Sketch* of the first proof: *Base* $n = 2$, true according to the ordinary De Morgan. *Induction* Assume true for $n = q$, show true for $n = q + 1$:

$$\begin{aligned}
&\neg(p_1 \wedge p_2 \wedge \cdots \wedge p_q \wedge p_{q+1}) \\
&\qquad \Leftrightarrow \quad \text{Assoc. law} \\
&\neg\big((p_1 \wedge p_2 \wedge \cdots \wedge p_q) \wedge p_{q+1}\big) \\
&\qquad \Leftrightarrow \quad \text{De Morgan} \\
&\neg(p_1 \wedge p_2 \wedge \cdots \wedge p_q) \vee \neg p_{q+1} \\
&\qquad \Leftrightarrow \quad \text{Acc. hyp.} \\
&(\neg p_1 \vee \neg p_2 \vee \cdots \vee \neg p_q) \vee \neg p_{q+1} \\
&\qquad \Leftrightarrow \quad \text{Assoc. law} \\
&\neg p_1 \vee \neg p_2 \vee \cdots \vee \neg p_q \vee \neg p_{q+1}
\end{aligned}$$

(b) A for-all expression can be regarded as a very large conjunction: this is true for this and this and this object, where one rattles off all the objects that exist. An existence expression can in the same way be regarded as a disjunction: this is true for this object or this object or this one, all the way through the universe. When the for-all expression, the conjunction, is negated you get a disjunction of negated expressions, the existence expression with a negation. The same line of argument applies for the second law.

7.89 "If it is true that the statement holds in the simplest case and that if the statement holds in one case then it holds in the next as well, so then the statement will hold in all cases"

$$\begin{aligned}
&\forall x\big(S(x) \to P(x)\big) \\
&\wedge \forall x\big[P(x) \to \forall y\big(N(y,x) \to P(y)\big)\big] \\
&\qquad\qquad \to \ \forall x\, P(x)
\end{aligned}$$

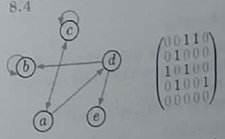
(Other variants are possible as well. In this version we have assumed that there may be several "simplest cases", and that there may be several "next cases". If you know that there is only one simplest case $\exists x\big(S(x) \wedge P(x)\big)$ – there is something that is the simplest case and where the statement holds – would be just as good a start.

Chapter 8

8.1 For instance: *lives-in-the-same-house-as*, *younger-than*, *brother-in-law-of*; *approximately-equal-to*, *half-of*, *inverse-of*.

8.3 Everyone is related to everyone, so we get a complete graph, with loops in addition. Since there are nine persons there will be $\binom{9}{2} = 36$ bidirected edges, and nine loops added to that. The matrix only contains ones.

8.4



$$\begin{pmatrix} 0&0&1&1&0 \\ 0&1&0&0&0 \\ 1&0&1&0&0 \\ 0&1&0&0&1 \\ 0&0&0&0&0 \end{pmatrix}$$

The picture may look different depending on how one chooses to place the nodes on the paper. In the matrix, the nodes are taken in alphabetical order.

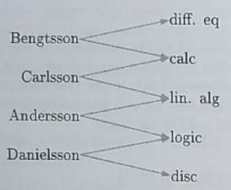
8.5 $\{(a,b),\ (b,a),\ (b,b),\ (b,e),\ (c,c),\ (c,d),\ (e,c),\ (e,d)\}$

8.6

(a) If we take the persons in alphabetical order and the courses in the order given (maybe it's according to the course code):

$$\begin{pmatrix} 1&0&0&1&0\\ 0&0&1&0&1\\ 0&0&1&1&0\\ 1&1&0&0&0 \end{pmatrix}$$

(b)



This shows fairly well the importance of thinking when drawing graphs.

(c) The matrix looks much better as well.

$$\begin{pmatrix} 1&1&0&0&0\\ 0&1&1&0&0\\ 0&0&1&1&0\\ 0&0&0&1&1 \end{pmatrix}$$

8.9 $\mathcal{R}_1 \circ \mathcal{R}_2 = \{(a,\alpha), (c,\alpha), (d,\gamma)\}$.

8.11 *In-the-same-form-in-school* and *greater-than-or-equal-to* are reflexive, *have-different-family-names*, and *is-a-proper-subset-of* aren't.

8.12 Everyday:

"I need water for a cup of tea. How should I measure, so that I heat the right amount?"
"Measure in the cup, of course!"

Mathematical:

"Tell me a divisor of 4711."
"4711."

8.14 *Relation-of* and *inverse-of* are symmetric. *Grandchild-of* and *divisor-of* aren't.

8.15 Everyday:

"Help, my son has just told me that he needs cross-country skis with boots the day after tomorrow! Now wait: he has the same size in shoes as me, and that must mean that I have the same size in shoes as him. I'll step into the sports shop over here and pick out some stuff that fits myself." (Yes, this has happened!)

Mathematical:

"If $5 \cdot 7 = 35$ then $35 = 5 \cdot 7$, and that factorisation will help when simplifying this fraction."

Often information is stored in some way in the brain, and it takes an active effort to realise that the whole thing can be turned around.

8.16 $\forall x \forall y (x \mathcal{R} y \vee y \mathcal{R} x \vee x = y)$ and $\neg\exists x \exists y (x \mathcal{R} y \wedge y \mathcal{R} x \wedge x \neq y)$ are two variants.

8.18 *Father-of* and *less-than-or-equal-to* are anti-symmetric, *cousin-of* and *approximatly-equal-to* aren't.

8.19 That depends. If we use positive numbers it is, but not if we use all the integers. (Then for instance $-5 \mid 5$ and $5 \mid -5$, but $-5 \neq 5$.)

8.20

(a) In the final part of example 3.3 on page 53 the fact that if $k \geq 1151$ at the same time as $1151 \geq k$ then $k = 1151$ has to be the case is used. In all the formal proofs of equality of sets in section 2.3 on page 18 the anti-symmetry of the subset relation is used.

(b) "Do you know if Ebba is the big sister of Aline?"
"Hmm... I *know* that Aline is the big sister of Ebba, and then Ebba can hardly be Aline's – big sistership is usually not mutual."

8.22 *In-the-same-form-in-school* and *less-than* are examples of transitive relations, *grandmother-of* and *inverse-of* aren't.

8.23 Everyday:

Madicken and Lisabet are going to slide on the ice on the creek. Alva has checked that the ice is strong enough. "And if it's strong enough for Alva it's strong enough for us", Lisabet says.

Mathematical:

"Do you know what the limit of a factorial divided by a logarithm is, as the input approaches infinity?"
"Hmm... Logarithms grow slower than powers, and powers grow slower than exponential functions, and exponential functions grow slower than factorials, and in that case the numerator grows faster than the denominator, so the quotient approaches infinity as well."

8.24 $\mathcal{R}_1$ is transitive, while $\mathcal{R}_2$ isn't. (There we have $a \mathcal{R}_2 b$ and $b \mathcal{R}_2 a$, but $a \not\mathcal{R}_2 a$.)

8.25

(a) "Everyone is in love with themself"

(b) "All love is reciprocated"

(c) "The only reciprocated love is the love of oneself"

(d) "You always love the one that your beloved loves"

8.26

	ref	sym	a-s	tra
$\neq$		X		
$\parallel$	X	X		X
$\subseteq$	X		X	X
$\perp$		X		
$<$			X	X
$=$	X	X	X	X
$\mid$	X		?	X
$\equiv$	X	X		X
$\approx$	X	X		
$\geq$	X		X	X
$\heartsuit$				

The doubts in the case of *divisor-of* is because it's not clear from the question whether we are using positive numbers or not; see exercise 8.19 on page 226. Otehrwise, if you don't share the opinions of the authors concerning some relation, discuss the question with a fellow student.

8.27 No.

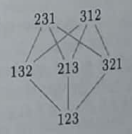
8.28 As the forms in the school.

8.29 There are three classes:

$$\{x \mid x = 3y, y \in \mathbb{Z}\}$$
$$\{x \mid x = 3y + 1, y \in \mathbb{Z}\}$$
$$\{x \mid x = 3y + 2, y \in \mathbb{Z}\}$$

The first class consists of everything that is equivalent to zero, the second one of everything equvalent to one, and the third one of everything equivalent to two.

8.30



There is a direct connection from one permutation to another if it's possible to get the second one by letting two digits in the first one swap places. The "correctly ordered" permutation is at the bottom.

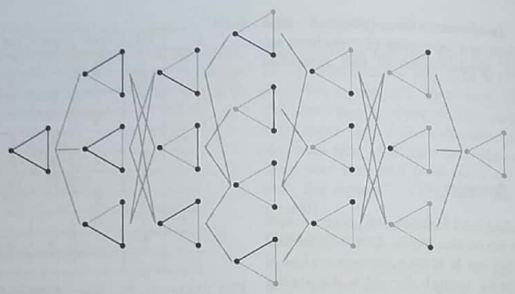


Figure 6: *Exercise 8.31 (for reasons of space drawn horisontally).*

8.31 See figure 6.

8.32

(a) As a lot of isolated nodes, each with a loop. The graph can either be regarded as a collection of complete graphs (K_1:s, all of them), and thus the graph of an equivalence relation, or as a degenerated Hasse diagram, with all the elements at the same level.

(b) Since there are no connections between different elements it's possible both to maintain that all the connections are bidirected and that none are. That makes the relation both symmetric and anti-symmetric.

(c) No.

8.33 "Sine" can be calculated for all real numbers, so the domain $\mathbb{R}$ seems good. The result is a real number between -1 and 1, so as the domain we can take either $\{y \mid y \in \mathbb{R} \ \wedge \ -1 \leq y \leq 1\}$ or simply $\mathbb{R}$. "Tangent" can be calculated for all real numbers except for odd multiples of $\pi/2$ (where you get division by zero). The domain $\mathbb{R} \setminus \{x \mid x = (2n+1)\pi/2, n \in \mathbb{Z}\}$ seems reasonable. We can get any real number as a result, so the codomain $\mathbb{R}$ seems good.

8.34 All humans have (or have had) a mother, so the domain *humanity alive at this moment*, and the codomain *all women that have been alive during the last 200 years* may work. Alternatively we extend the domain and codomain to *all human beings that have ever existed*, but then we get problems with Adam and Eve, alternatively with deciding exactly when *Homo sapiens* appeared.

"Costs" sounds like a function from articles to prices in a shop. The domain *all the articles in Coop*, and the codomain $\mathbb{R}$ seems to work.

8.35 For sine: $\{y \mid y \in \mathbb{R} \wedge -1 \leq y \leq 1\}$, for tangent: $\mathbb{R}$; for *has-the-mother* all women that have borne children who are still alive, alternatively all women that ever existed that gave birth to children; for *costs* the prices that are actually found in the shop.

8.36

(a) Not a function, 6 isn't tied to any value.

(b) Function.

(c) Not a function, 2 is tied to two different values.

8.37 It has got one and only one 1 in every row.

8.38 We haven't used negative numbers when working with these concepts, and if one sticks to this the domain of the gcd is $(\mathbb{N} \times \mathbb{N}) \setminus \{(0,0)\}$ (because calculating $\gcd(0,0)$ doesn't work). For the lcm $\mathbb{N} \times \mathbb{N}$ will do.

8.39

(a) $(2x)^2 = 4x^2$ (b) $2x^2$

(c) Yes.

8.40

$$\neg\exists x_1 \exists x_2 \big(f(x_1) = f(x_2) \ \wedge \ x_1 \neq x_2\big)$$
$$\Leftrightarrow \text{Negation of quant.}$$
$$\forall x \forall y \neg \big(f(x_1) = f(x_2) \ \wedge \ x_1 \neq x_2\big)$$
$$\Leftrightarrow \text{De Morgan}$$
$$\forall x \forall y \big(f(x_1) \neq f(x_2) \ \vee \ x_1 = x_2\big)$$
$$\Leftrightarrow \text{Rewr. implication}$$
$$\forall x_1 \forall x_2 \big(f(x_1) = f(x_2) \ \rightarrow \ x_1 = x_2\big)$$

"For all pairs of x-values it's true that they can't have the same function values at the same time as the x-values are different" and "For all pairs of x-values it's true that the function values are different or the x-values alike", respectively.

8.41

(a) No, definitely not! $f(1) = 1^3 = 1$, $f(2) = 2^3 = 8 = 7 + 1 \equiv 1$, $f(4) = 4^3 = 64 = 63 + 1 \equiv 1$. Moral: Beware of taking for granted that things work exactly the way they usually do in modular arithmetic!

(b) No. If you are to talk about "*the* cube root" of a number it has to be unique.

8.42 Yes, different persons have different personal identity numbers (that's the whole point). No, different persons may have the same name. (Try looking up "Andersson" in the phone book.)

8.43 Usually the derivative is checked. If the derivative is positive, then the function is increasing, and then two different x-values can't belong to the same y.

8.44 There is *at most* one 1 in each column.

8.45 No in both cases. For instance there is probably no-one with the personal identity number 640231-0024, nor is there anyone named Xxxzzzww.

8.46

(a) Surjective and injective.

(b) Injective but not surjective. (There is for instance no integer that raised to the power of four is 2.)

(c) Neither surjective nor injective. (Both 1 and -1 are mapped to 1, and concerning the surjectivity, see above.)

(d) Surjective, but not injective. (Both 1 and -1 are mapped to 1.)

8.47 Usually, one looks for the greatest and smallest value of the function. If it's continuous, it takes all the values in between as well.

8.48 There is *at least* one 1 in each column.

8.49

(a) There are infinitely many that fit! The whole set $\{x \mid x = \pm\pi/3 + n2\pi, n \in \mathbb{Z}\}$ has the property that $f(x) = 0.5$.

(b) No such x exists, at least not among the real numbers!

(c) Neither. That's what caused the problems.

8.50 When using matrices, if you have studied linear algebra, and in modular arithmetic, if you have been reading this book.

8.51

(a) subtraction (b) division

(c) logarithmation

8.52 The function itself. If you turn something back-to-front twice you are back where you started.

8.53 The answer is x. Since f^{-1} performs the opposite operation of f the effects cancel out irrespective of whether you start with f or with f^{-1}.

8.54 Even values. For them $(y-3)/2$ isn't an integer.

8.55

(a) There are $|B|^{|A|}$ functions from A to B. Each of the $|A|$ elements in A have $|B|$ "targets" to choose from.

(b) None at all if $|B| < |A|$, since then the elements in B won't be enough to allow all the elements in A to get one of their own. In other cases there are

$$|B|\cdot(|B|-1)\cdot(|B|-2)\ldots(|B|-|A|+1) = \\ = \prod_{k=0}^{|A|-1}(|B|-k) = \frac{|B|!}{(|B|-|A|)!}$$

The first element in A has $|B|$ things to choose from, the second one $|B|-1$ (since it's not allowed to take the same one as the first one did) and so on.

(c) None at all if $|B| > |A|$, because then the elements in A won't be enough to cover all the elements in B. Otherwise we can partition the elements in A in $|B|$ piles, after which we assign each element in B a pile, corresponding to the elements in A that are mapped to this element. The number of ways of partitioning $|A|$ different things into $|B|$ piles are $S(|A|,|B|)$ (see page 126). Assigning $|B|$ different piles to $|B|$ different recipients can be done in $|B|!$ different ways. In total $S(|A|,|B|)\cdot|B|!$.

(d) A bijective function is both surjective and bijective. To make a surjective function possible, we must have $|A| \geq |B|$. To make an injective function possible, we must have $|B| \geq |A|$. For these two things to be true at the same time we must have $|A| = |B|$. ("$\geq$" is an antisymmetric relation.)

(e) According to the previous subexercise none at all if $|A| \neq |B|$. In other cases, $|A|! = |B|!$. (We can get that either by insterting $|A| = |B|$ into the exercise about surjectivity or into the one about injectivity.)

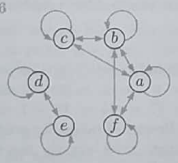
(f) If $|A| = |B|$ a surjective function is guaranteed to be injective (and vice versa), otherwise not.

(g) Since A is guaranteed to have the same number of elements as itself the answer has to be yes.

(h) No. Counterexample: excercise 8.46(d) on page 235. This was an example of a statement of the kind discussed on page 210, one that is true in all finite universes, but not in all infinite ones.

8.56

(a)



(b)
$$\begin{pmatrix} 1&1&1&0&0&1\\ 1&1&1&0&0&1\\ 1&1&1&0&0&0\\ 0&0&0&1&1&0\\ 0&0&0&1&1&0\\ 1&1&0&0&0&1 \end{pmatrix}$$

(c) Yes, all the nodes have loops.

(d) Yes, all the connections are bidirected.

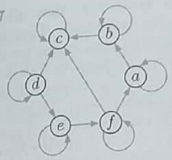
(e) No. There are connections that aren't unidirected.

(f) No. We have $f \mathcal{R} b$ and $b \mathcal{R} c$ without having $f \mathcal{R} c$.

(g) Neither an equivalence relation nor a partial order.

8.57

(a)



(b)
$$\begin{pmatrix} 1&1&0&0&0&0\\ 0&1&1&0&0&0\\ 0&0&1&0&0&0\\ 0&0&1&1&1&0\\ 0&0&0&0&1&1\\ 1&0&0&0&0&1 \end{pmatrix}$$

(c) Yes, there are loops on all the nodes.

(d) No, there are connections that aren't bidirected.

(e) Yes, all the connections are unidirected.

(f) No, we have $f \mathcal{R} a$ and $a \mathcal{R} b$ without having $f \mathcal{R} b$.

(g) Neither an equivalence relation nor a partial order.

8.58

(a) $\{(a,a),(b,a),(c,a),(d,a),(d,b),(d,c),(d,d),(d,e),(d,f),(e,a),(f,a)\}$

(b)
$$\begin{pmatrix} 1&0&0&0&0&0\\ 1&0&0&0&0&0\\ 1&0&0&0&0&0\\ 1&1&1&1&1&1\\ 1&0&0&0&0&0\\ 1&0&0&0&0&0 \end{pmatrix}$$

(c) No, there are elements (for instance B) that aren't related to themselves.

(d) No, there are unidirected connections.

(e) Yes, all the connections are unidirected.

(f) Yes.

(g) Neither an equivalence relation nor a partial order.

8.59

(a) $\{(a,a),(c,c),(e,e)\}$

(b)
$$\begin{pmatrix} 1&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&1&0\\ 0&0&0&0&0&0 \end{pmatrix}$$

(c) No, there are elements (for instance b) that aren't related to themselves.

(d) Yes. We can't find any unidirected connections.

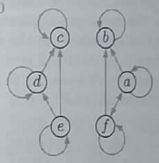
(e) Yes. There are no connections not being unidirected either.

(f) Yes.

(g) No to both questions.

8.60

(a)



(b) $\{(a,a),(a,b),(a,f),(b,b),(c,c),(d,c),(d,d),(e,c),(e,d),(e,e),(f,a),(f,b),(f,f)\}$

(c) Yes.

(d) No, there are unidirected connections.

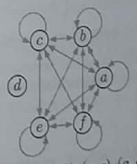
(e) No, there are bidirected connections.

(f) Yes.

(g) No to both questions.

8.61

(a)



(b) $\{(a,a),(a,b),(a,c),(a,e),(a,f),(b,a),(b,b),(b,c),(b,e),(b,f),(c,a),(c,b),(c,c),(c,e),(c,f),(e,a),(e,b),(e,c),(e,e),(e,f),(f,a),(f,b),(f,c),(f,e),(f,f)\}$

(c) No. $d \mathcal{R} d$.

(d) Yes.

(e) No, there are bidirected connections.

(f) Yes.

(g) No to both questions.

8.62 If we take two numbers, no matter which ones, there is one and only one number that is the sum of those. That is the very definition of function: a rule that to each element (in this case pair of numbers) assign one and only one element in the codomain (that consists of numbers).

Addition is defined in all standard sets, so $\mathbb{C} \times \mathbb{C}$ is the most general domain. All numbers can be written as the sum of two numbers (e.g., itself and zero), so the range is then $\mathbb{C}$. (If you work in $\mathbb{Z}_+$, it's a bit more complicated!)

The proponents of *prefix notation*, see page 163, are of the opinion that one of the advantages of writing + 2 2 instead of 2 + 2 is that it emphasises that it is a case of calculating a function. The standard for functions is putting the name in front and the operands after.

8.63 Domain: $\{0,1\} \times \{0,1\} \times \{0,1\}$ (all triples of 0:s and 1:s). Codomain: $\{0,1\}$

8.64

(a) Yes.

(b) No. For instance no x exists such that $f(x) = 1$.

(c) The range consists of the even numbers.

(d) Yes.

(e) Yes. (A table of values answers both this question and the previous one.)

(f) $f^{-1}(y) = 5y$

(g) Yes, what made the function invertible is that the number 2 has an inverse (namely 5) when calculating modulo 9.

8.66 If it's symmetric then all the connections are bidirected; if it's antisymmetric then no connections are bidirected. This is only possible if there are no connectiona, loops excluded. If there are no connections between different elements the only candidates for $x \mathcal{R} y, y \mathcal{R} z$ are such that $x = y = z$, and if $x \mathcal{R} x$ and $x \mathcal{R} x$ are true, of course $x \mathcal{R} x$ follows.

8.67 No. If a relation is both symmetric and transitive every element that is related to something else has to be related to itself as well. But there may be elements that aren't related to anything!

8.68

(a) Reflexive, then each element is related to itself. In a function, an element may only be related to one object, so these connections must be the only ones there are. The function has to be the **identity function**, which is defined as $f(x) = x$. (The identity function is the name the world of functions uses on the equality relation.)

(b) All connections are mutual. That means that if $f(a) = b$ then $f(b) = a$, which in its turn gives $f(f(a)) = f(b) = a$, that is to say: the function is its own inverse!

(c) No connections are bidirected, while at the same time one arrow exits from each object. We will then have an object pointing at another one, that points at a third, that points at... The graph of the function will consist of a number of directed cycles with the length at least 3, and possibly some isolated nodes with loops.

(d) In a transitive relation every multi-step path has a corresponding one-step path. Since no more than one arrow may start at an object this means that there mustn't be any multi-step paths. The arrows starting at the objects then have to double back at them, so this means as well that we study the identity function.

8.69

(a) From a set with n elements it's possible to make n^2 ordered pairs. When a relation is defined, for each pair a choice has to be made whether it is to be included or not. In total 2^{n^2}, then.

(b) If the relation is to be reflexive, all the pairs along the main diagonal in the matrix have to be included. That's n pairs. The remaining $n^2 - n$ pairs we can choose whether they are to be included or not. 2^{n^2-n} reflexive relations.

(c) If the relation is to be symmetric, then if we include (a, b) we have to include (b, a) as well. There are $n^2 - n$ ordered pairs where the elements are different, which gives us $(n^2 - n)/2$ unordered pairs. Out of the n pairs where the halves are identical we may pick as many as we like. $2^{(n^2-n)/2} 2^n = 2^{(n^2+n)/2}$.

(d) If the relation is to be antisymmetric we may only include one of the pairs (a, b) and (b, a). For each pair-pair we thus have three choices: include the first pair, include the second pair, and include neither of the pairs. The pairs on the diagonal we may have or lose. $3^{(n^2-n)/2} \cdot 2^n$.

(e) Well, who can say! (The level of difficulty of this question is far above this course!)

(f) Each equivalence relation corresponds to a partition of the set. Thus we can count the number of partitions. How that is done we will cover in the next book!

(g) See question (e).

8.70 No.

8.71 NAND can take an arbitrary number of signals as input (and gives one signal as output). If we let $\mathcal{B}$ denote $\{0,1\}$ the domain is

$$\mathcal{B} \cup \mathcal{B}^2 \cup \mathcal{B}^3 \cup \ldots = \bigcup_{n=1}^{\infty} \mathcal{B}^n$$

(The large union sign has the same meaning as the sum symbol, but with union instead of plus.)

8.72

(a) There are several reasonable answers to this exercise. Here are two of them: left pocket – right pocket – left side seam – right side seam – left inside seam – right inside seam – left hem – right hem – crotch – waistband; and left pocket — left side seam – left inside seam – left hem – right pocket – right side seam – right inside seam – right hem – crotch – waistband.

(b) Here as well there are several strategies. One is to look through the Hasse diagram breadth-first. (That generates the first of the suggestions.) Another one is to start at one end, and then follow a maximally long path. When one reach a point where something else has to be done before one restarts, and follows another maximally long path. (That generates the second of the suggestions.) And there are more methods.

(c) A lot of answers! Almost all housework; computer programming; making syllabuses at the university, are just some examples.

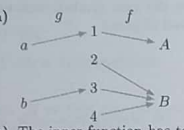
(d) At building projects it's possible to do several things at the same time, which a single person sewing a pair of trousers usually can't.

8.73 We won't give the answer, but some hints:

- Check that the domain and codomain really are the specified ones!
- Yes, the exercise is possible to solve.
- Note that nowhere in the definition of *function* is stated that there has to exist any nice formula for describing it.

8.74

(a)



(b) The inner function has to be injective. The outer function has to map the elements in the range of the inner one onto different elements, but it doesn't matter what it does to the remaining elements.

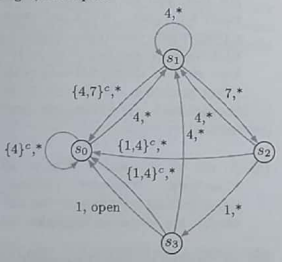
(c) The same answer as in (a) works.

(d) The outer function has to be surjective. That's not enough, but it has to spread the whole range of the inner function over the whole codomain. What it decides to do with remaining elements, if any, doesn't matter.

Chapter 9

9.1 0000010001000

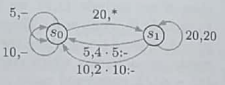
9.2 The input alphabet is clearly the digits 0–9. As the output alphabet we can take *flash* (a lamp that flashes when a button is pushed), which we mark using *, and *open*.



9.3

(a) A fairly sensible behaviour seems to be this: if a twenty is put in a lamp is lit (marked with *). If you then push 10 you get two 10:s, if you push 5 you get four 5:s. At the same time the lamp is switched off. If you push any button without having put in a note nothing happens (marked with –); if you put in a note when there is a note already in the slot it is spat out again.

(b)



9.4 If you put in ten crowns you get a cup of coffee.

9.6 It's always possible to switch the machine types. From Moore to Mealy it's really simple, you just have to move the output signals backwards to the arrows. In the other direction you may have to split one state into several, one for each possible output signal on the way to it. Since the machine may expand when going from Mealy to Moore, but not when going in the other direction, the Moore machine is usually larger.

9.8

s_0	haven't read anything of interest
s_1	have read "s"
s_2	have read "se"
s_3	have read "sea"
s_4	have read "sear"
s_5	have read "searc"
s_5	have read "search"

9.9

(a) When we had read "se" and then found "s" and not "r" we went back to s_0. The correct thing is to go back to s_1 since we just read an "s",

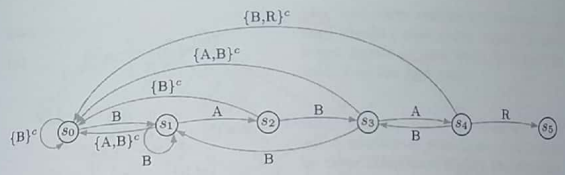


Figure 7: *Exercise 9.11*

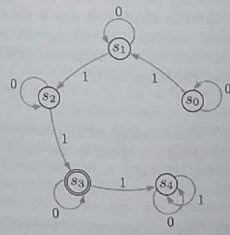
which may be the start of the word "search".

(b) From all the states there have to be arrows marked $\{s\}$ that end at s_1, and all the arrows to s_0 should be supplemented with "not s".

9.11 See figure 7.

9.12 See figure 8 on the following page.

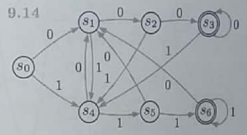
9.13



The meanings of the states:

s_0	We haven't read any 1.
s_1	We have read exactly one 1.
s_2	We have read exactly two 1:s.
s_3	We have read exactly three 1:s.
s_4	We have read more than three 1:s.

9.14



9.16

(a) The language of words from the alphabet $\{a\}$ where the number of a:s is divisible by 3.

(b) The language of binary strings that somewhere contain two consecutive instances of the same symbol.

9.17 To recognise a palindrome the machine by its design has to "remember" the whole word, so that it's possible to check whether the first letter matches the last one, etc. Since the words can be arbitrarily long an infinite memory is needed, and thus an infinitely large machine.

9.18 There are difficulties with definition – when are two closely related languages to be regarded as dialects of the same language? There are dynamic difficulties – languages develop and languages die out.

9.19 In this book we have used a little more than 150 different symbols.

9.20 For instance the Chinese written language has more than 50 000 characters, according to the Swedish National Encyclopedia.

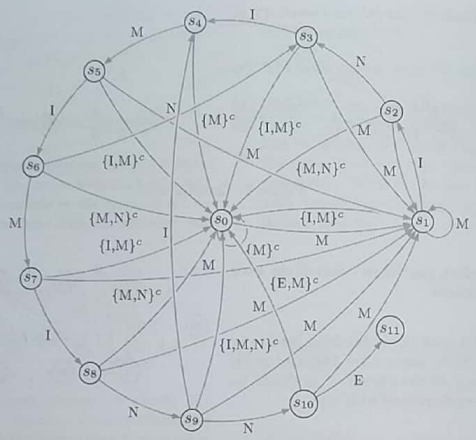


Figure 8: *Exercise 9.12.*

9.21 $\lambda w = w = w\lambda$ according to the definition of λ.

9.22 Assume that $v \neq \lambda$. Then v will have some intial letter. Choose a word w that starts with some other letter. Then $vw \neq wv$.

9.23

(a) $\mathcal{L}_1\mathcal{L}_2 = \{000, 010, 100, 110, 001, 011, 101, 111\} = \mathcal{L}_2\mathcal{L}_1$.

(b) No. If $\mathcal{L}_1 = \{0\}$ and $\mathcal{L}_2 = \{1\}$ then $\mathcal{L}_1\mathcal{L}_2 = \{01\}$ while $\mathcal{L}_2\mathcal{L}_1 = \{10\}$.

9.24 There are nm possible ways of concatenating a word from $\mathcal{L}_1$ (n words) and a word from $\mathcal{L}_2$ (m words). Of these compound words some may be alike, so that there are fewer than nm words in total. For instance, the concatenation of $\mathcal{L}_1 = \{1, 10\}$ and $\mathcal{L}_2 = \{1, 01\}$ gives only three words: $\mathcal{L}_1\mathcal{L}_2 = \{11, 101, 1001\}$.

9.25 The number 0 is included in $S_0 + SS_0(S_0)^*$ but not in $S(S_0)^*$.

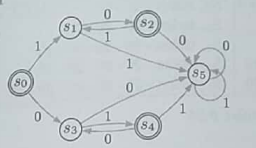
9.28 $a+$ ends with $+$, which can't happen in the regular expressions. The corresponding problem with the other expressions is that $(bc))$ has a right parentesis without a matching left parenthesis and that $^*c$ lacks an expression before the Kleene star.

9.29 The language of all binary strings of odd length is given by the regular expression $(\mathcal{B}^2)^*\mathcal{B}$.

9.30 There is no way of indicating in a regular expression that the first letter has to be the same as the last one – except by explicitly writing the letters in question, but then all the palindromes have to be written explicitly and that isn't possible in a regular expression, since there is an infinite number.

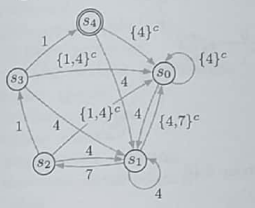
9.31

(a)



(b)

(c)



9.32

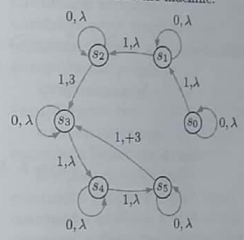
(a) The acceptor recognises all the strings that begins with zeros and end with ones. 0^*1^*

(b) The machine accepts anything that starts with one or with zero followed by some more digit. $1(0+1)^*+0(0+1)(0+1)^* = \big(1+0(0+1)\big)(0+1)^*$

9.34 We are to design the deterministic machine A. Every sequence w of input signals in the non-deterministic machine I leads to a set S_w of states in I. For each such set S_w we define a state in A. A is finite since there is only a finite number of sets that can be formed from the finite number of states in I. The transitions in A are now of course defined so that input a in state S_w leads to state S_{wa}. At last we define that a state S_w is accepting if and only if the set S_w includes some accepting state in I. Now we have a deterministic finite-state machine A that accepts exactly the same language as I.

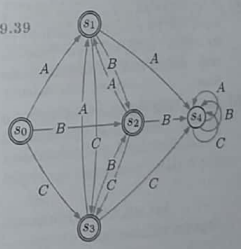
9.35 Five different states corresponding to the input levels zero, twenty, forty, fifty, and sixty crowns are needed. A table showing *next state, output* for different inputs in each state is found in table 2 on the next page.

9.36 Keeping track of when we have got a number of ones that is divisible by three takes three states. But besides that, we have to differentiate between the first three (that should not be preceded by $+$) and later threes. Thus six states are needed in the machine:



9.38

9.39



9.40 It's the language in the alphabet $\{1, 2, 3\}$ where every second symbol is a 1.

$(\lambda + 2 + 3)\big(1(2+3)\big)^*(\lambda + 1)$

	20	50	100
S_0	S_1, λ	S_3, λ	S_0, ticket+20+10
S_1	S_2, λ	S_0, ticket	S_0, ticket+20+20+10
S_2	S_4, λ	S_0, ticket+20	S_0, ticket+20+20+20+10
S_3	S_0, ticket	S_0, ticket+20+10	S_0, ticket+20+20+20+20
S_4	S_0, ticket+10	S_0, ticket+20+20	S_0, ticket+20+20+20+20+10

Table 2: *Exercise 9.35*

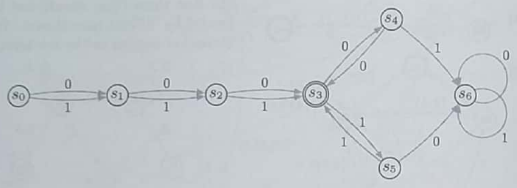


Figure 9: *Exercise 9.45*

9.41

(a) ABC = {september, septober, octember, october, november, novober, december, decober}. C^{10} = {berberberberberberberberber}.

(b) $|B^{10}| = 2^{10}$.

9.42

(a) If the alphabet has only one letter, a word will be completely defined by its length. w_1w_2 and w_2w_1 have the same length and are thus the same word.

(b) Let w_1 and w_2 start with different letters. Then w_1w_2 and w_2w_1 as well will start with different letters and thus $w_1w_2 \neq w_2w_1$ has to hold.

9.45 We take the input order (1,1), (2,2), (3,3), (1,2), (2,1), (1,3), (3,1), (2,3), (3,2). A relation is symmetric if and only if a one on one side of the main diagonal always corresponds to a one on the other side. (What's written on the main diagonal doesn't matter.) Thus we will here read the first three digits (but ignore their values), after which we check that the digits following come in pairs, in which case we are to end up in an accepting state, otherwise not. The machine is drawn in figure 9.

English-Swedish Glossary

accepting state accepterande tillstånd
addition principle additionsprincipen
adjacency matrix grannmatris
adjacent granne
algorithm algoritm
alphabet alfabet
antisymmetric relation anti-symmetrisk relation
arc båge
atomic proposition atomär sats
bijection bijektion
bijective function bijektiv funktion
binary digit binär siffra
binary expression tree binärt uttryckträd
binary operation binär operation
binary relation on a set binär relation på en mängd
binary search tree binärt sökträd
binary tree binärt träd
binomial coefficient binomialkoefficient
binomial numbers binomialtal
binomial theorem binomialsatsen
bipartite graph bipartit graf
Boolean algebra boolesk algebra
Boolean function boolesk funktion
Boolean value booleskt värde
Boolean variable boolesk variabel
breadth-first search bredden-förstsökning
cardinality kardinalitet
child barn
circuit krets
closed walk sluten väg
codomain kodomän
combinatorics kombinatorik
common divisor gemensam delare
complement graph komplementgraf
complement set komplementmängd
complete bipartite graph komplett bipartit graf
complete graph komplett graf
complex numbers komplexa tal
composite function sammansatt funktion
composite number sammansatt tal
composite relation sammansatt relation
concatenation sammansättning
conditioned probability betingad sannolikhet
conjunction konjunktion
conjunctive form konjunktiv form
conjunctive normal form konjunktiv normalform
connected graph sammanhängande graf
connective konnektiv
contradiction kontradiktion
coprime numbers relativt prima tal
countably infinite set uppräkneligt oändlig mängd
cycle cykel
decision tree beslutsträd
degree grad
depth-first search djupet-förstsökning

direct proof direkt bevis
directed graph riktad graf
disjoint events disjunkta händelser
disjoint sets disjunkta mängder
disjunction disjunktion
disjunctive form disjunktiv form
disjunctive normal form disjunktiv normalform
divide dela
divisor delare
domain domän
edge kant
element element
empty set tomma mängden
empty string tomma strängen
equivalence class ekvivalensklass
equivalence relation ekvivalensrelation
equivalence ekvivalens
Eulerian circuit eulerkrets
Eulerian graph eulersk graf
Eulerian trail eulerväg
event händelse
exclusive or exklusivt eller
factor faktor
finite state machine ändlig automat
function funktion
game tree spelträd
gate grind
graph graf
greatest common divisor största gemensamma delare
Hamiltonian cycle hamiltoncykel
Hamiltonian graph hamiltonsk graf
Hamiltonian path hamiltonstig
height of a rooted tree höjden av ett rotat träd
implication implikation
in-order inordning
incidency matrix incidensmatris
independent events oberoende händelser
indirect proof indirekt bevis
induction basis bassteg
induction induktion
inductive step induktionssteg
initial state starttillstånd
injective function injektiv funktion
integers heltal
intersection snitt
intersect skära
inverse function inversfunktion
isomorphic graphs isomorfa grafer
isomorphism isomorfi
Kleene star kleenestjärna
language språk
least common multiple minsta gemensamma multipel
leaf löv
linear combination linjärkombination
loop ögla
lower limit undre gräns
matrix matris
Mealy machine mealyautomat
minimal spanning tree minimalt spännande träd
modular aritmetic modulär aritmetik
multigraph multigraf
multiple edges multipla kanter
multiple multipel
multiplication principle multiplikationsprincipen
multiset multimängd
natural numbers naturliga tal
negate negera
negation negation
node nod
nondeterministic finite state machine ickedeterministisk ändlig automat
number base talbas
number sequence talföljd
pair par
parent förälder
partial order partialordning
path stig
pigeonhole principle postfacksprincipen
post-order postordning

power set potensmängd
pre-order preordning
predicate logic predikatlogik
predicate predikat
prime number primtal
principal remainder principal rest
principle of inclusion/exclusion principen om inklusion/exklusion
probability theory sannolikhetslära
product set produktmängd
proof by contradiction bevis genom motsägelse
proper subset äkta delmängd
proposition calculus satslogik
proposition sats
quantifier kvantifikator
quotient kvot
range värdemängd
rational numbers rationella tal
real numbers reella tal
recursion rekursion
reflexive relation reflexiv relation
regular expression reguljärt uttryck
regular language reguljärt språk
relation between sets relation mellan mängder
rooted tree rotat träd
sample space utfallsrum
satisfiable proposition satisfierbar sats
sentence sentens
set builder mängdbyggaren
set theory mängdlära
set mängd
simple graph enkel graf
spanning tree spännande träd
statement påstående
state tillstånd
straight insertion insättningssortering
string sträng
subgraph delgraf
subset delmängd
summation index summationsindex
surjective function surjektiv funktion
symmetric relation symmetrisk relation
tautology tautologi
total order totalordning
trail väg
transition övergång
transitive relation transitiv relation
tree träd
truth value sanningsvärde
truth-table sanningsvärdestabell
uncountably infinite överuppräkneligt oändlig mängd
undirected graph oriktad graf
uniformly distributed probability likformigt fördelad sannolikhet
union union
universe universum
upper limit övre gräns
valid statement logiskt giltig sats
Venn diagram venndiagram
vertex hörn
walk vandring
weight vikt
word ord

Index

$\cap$, 16
$\cup$, 16
$\emptyset$, 16
$\in$, 16
$\wedge$, 181
$\leftrightarrow$, 184
$\vee$, 182
$\neg$, 181
$\notin$, 16
$\not\subseteq$, 16
$\oplus$, 182
$\rightarrow$, 183
$\setminus$, 16
$\subseteq$, 16

A-format, 71
absolute value, 13
absorption laws
  in Boolean algebra, 193
  in propositional logic, 187
  in set theory, 20
accepting state, 250
acceptor, 249, 250
according to the hypothesis, 88
addition principle
  in combinatorics, 114
  in probability, 111
addition table, 57
adjacency matrix, 152
adjacent, 140
Agenda 21, 11
algorithm, 36
Alice in Wonderland, 9
alphabet, 251
AND-gate, 197
anti-symmetric relation, 225
arc, 140
arithmetical number sequence, 82
artificial language, 243, 251
associative laws
  in Boolean algebra, 193
  in propositional logic, 187
  in set theory, 20
  of addition, 18
  of multiplication, 18
atomic proposition, 180
automatically, 244
automaton, 244

balanced tree, 161
base part, 72
base step, 86
bijection, 28, 236
bijective function, 236
binary digit, 62
binary expression tree, 163
binary number system, 62
binary operation, 164
binary operator, 163
binary relation on a set, 218
binary search tree, 160
binary string, 135
binary tree, 160
binomial coefficient, 123
binomial number, 120
binomial theorem, 122
bioinformatics, 148
bipartite graph, 144
  for a relation, 221
bit, 62
Boole, G., 191
Boolean algebra, 191
Boolean function, 192
Boolean value, 192
Boolean variable, 192
breadth-first search, 158
bridges of Königsberg, 149

C, 251
$\mathbb{C}$, 27
calculation rules
  in Boolean algebra, 193
  in predicate logic, 205
  in propositional logic, 187, 188
  in set theory, 20
calculus, 97
Cantor's diagonal argument, 30
Cantor, G., 30
car and goats, 110, 113
card game, 37, 119
cardinality, 16
Carroll, L., 9
change machine, 247
children, 157
circuit
  in a graph, 142
  logical, 197
circular definitions, 74
circular proof, 101
clock arithmetic, 54
closed path, 142
closed trail, 142
c n f, 194
codomain, 230
coefficient, 50
combinatorial game theory, 157
combinatorics, 75, 105, 114
common divisor, 42, 43
commutative laws

in Boolean algebra, 193
in propositional logic, 187
in set theory, 20
of addition, 18
of multiplication, 18
complement
of an event, 109
complement graph, 143
complement set, 16
complete bipartite graph, 144
complete graph, 143
complete ternary tree, 175
complex numbers, 27
composite function, 232
composite number, 39
composite relation, 221
computers, 5
concatenation, 251
of languages, 252
conditioned probability, 113
congruence
modulo n, 54
conjugate rule, 6
conjunction, 181
conjunctive form, 194
conjunctive normal form, 194
connected component, 142
connected graph, 142
connective, 180
constructive proof, 212
contradiction, 189
conversion
from binary to decimal, 62
from decimal to binary, 63
convex, 94
coordinates, 69, 262
coprime numbers, 43
corollary, 49
countably infinite, 29
counting in $\mathbb{Z}_n$, 54
cycle, 142

database, 33, 261
De Morgan's laws, 19
generalised, 216
in Boolean algebra, 193
in propositional logic, 187
in set theory, 20
decimal number system, 62
decision tree, 116, 157
degree, 141
depth-first search, 159
difference
between sets, 16
diofantine equation, 50
complete solution, 51
direct proof, 211
directed graph, 145
discrete mathematics, 1
discretisation, 2
disjoint events, 111
disjoint sets, 14
disjunction, 182
disjunctive form, 194
disjunctive normal form, 194
distributive laws
in Boolean algebra, 193
in ordinary arithmetics, 18
in propositional logic, 187
in set theory, 20
divide, 38
divisibility rules, 69
divisible, 38
division algorithm, 36
divisor, 38
divisor graph, 41
DNA sequence, 148
dnf, 194
domain, 230
dominance laws
in Boolean algebra, 193
in propositional logic, 187
in set theory, 20
domino effect, 87
door code reader, 247
double complement, 20
double negation
in Boolean algebra, 193
in proposition, 187
dyadic predicate, 200

E, 140
edge, 140
element, 12, 16
EMACS, 255
empty string, 251
encryption, 40
equal to, 201
equivalence, 184
between expressions, 186
modulo n, 54
equivalence class, 228
equivalence relation, 228
Esperanto, 251
Euclid, 40
Euclidian algorithm, 43
Euler, L., 147
Eulerian circuit, 147
Eulerian graph, 150
Eulerian trail, 146
event, 108
exact arithmetic, 38
exclusive or, 182
existence statement, 202
proof of, 212
expected gain, 277
expected value, 277
exponential function, 96
expression tree, 163

factor, 38
factorial function, 98
female CEO, 113
Fibonacci, 76
Fibonacci numbers, 76, 103
computation, 77
finite-state machine, 243, 244
flow chart, 117
for-all statement, 201
proof of, 212
function, 29, 217, 230
bijective, 236
injective, 234
"nicely combed", 234
onto, 235
surjective, 235
fundamental theorem of arithmetic, 40, 49
proof, 48

game tree, 157
gate, 197
gcd, 43
geometrical sequence, 83
good mathematical presentation, 6
grammar, 243
graph, 139, 140
Gray code, 78
greatest common divisor, 42

Hamilton, W. R., 147
Hamiltonian cycle, 146
Hamiltonian graph, 150
Hamiltonian path, 146
hand shaking, 125
hand-shaking lemma, 141
Hasse diagram, 41, 229
height of a rooted tree, 161
hexadecimal form, 65
hypercube, 176

idempotence laws
in Boolean algebra, 193
in propositional logic, 187
in set theory, 20
identity function, 314
identity laws
in Boolean algebra, 193
in propositional logic, 187
in set theory, 20
of addition, 18
of mutliplication, 18
if and only if, 184
implication, 183
proof of, 211
in-order, 161
incidency matrix, 153
inclusion/exclusion, 23, 131, 137
independent events, 112
indirect proof, 211
induction, 71
induction basis, 86
induction hypothesis, 86
inductive step, 86
inequalities, 94
calculation rules, 95
infinite recursion, 73
infinite sets, 26
infix notation, 163
initial state, 244
injection, 234
injective function, 234
input alphabet, 244
input signal, 244
Instant Insanity, 168
integers, 27
interest, 84
interpretation, 210
intersect, 14
intersection, 14
inverse, 60, 236
inverse function, 236
inverse laws
in Boolean algebra, 193
in propositional logic, 187
in set theory, 20
invert, 181
inverter, 197
invertible, 60
isomorphic graphs, 151
isomorphism, 152

Kaliningrad, 150
Kant, I., 150
Kernighan, B. W., 251
Kleene star, 252
$K_{m,n}$, 144
K_n, 143
Knuth, D. E., 167
Knuth-Morris-Pratt-machine, 249
Kruskal's algorithm, 156
Königsberg, 150

language, 251
$+$ and $\cdot$, 253
artificial, 251
natural, 251
regular, 255
last "non-vanishing" remainder, 43
last digit, 87
law of double complement, 20
lcm, 46
leaf, 154
least common denominator, 46
least common multiple, 46
lemma, 48
line breaking, 166
using regular expressions, 255
linear combination, 42
linear equation, 50
linear time, 151
Lisp, 163
logic, 177
loop, 145
Lotto, 129, 133
lower limit, 80

machine code, 251
Maple, 38
mathematical model, 2
mathematical presentation, 6
mathematicians
and biscuits, 127
and sheep, 177
and singing, 184
matrix, 25
of a composite relation, 222
of relation, 219
matrix algebra, 61
maxfactors, 196
Mealy machine, 244
minimal spanning tree, 156
minterms, 196
model, 2
modular arithmetic, 54
equationsolving, 57
equivalence classes in, 228
modulo n, 54
modus ponens, 190
modus tollens, 191

monadic predicate, 199
Monty Hall problem, 110
Moore machine, 247
multigraph, 145
multinomial coefficient, 124
multiple, 38
multiple edges, 145
multiplication principle
  in combinatorics, 115
  in probability, 112
multiset, 123
Murphy's law, 180

$\mathbb{N}$, 26
ν, 245
n choose k, 120
$n!$, 98, 118
NAND-gate, 198
natural language, 251
natural numbers, 27
negate, 181
negation, 181
network
  for computer communication, 154
  social, 145
next number, 26
next state, 245
n-factorial, 118
node, 140
nondeterministic finite-state machine, 256
NOR, 198
NOT-gate, 197
noughts and crosses, 157
number base, 62
number circle, 56
number sequence, 74
numeracy, 4

Ω, 108
ω, 245
octal form, 64
office secretary, 281
optimisation problem, 117
OR-gate, 197
ordered selection, 119
output alphabet, 245
output signal, 244

Π, 84
pair, 24
palindrome, 133, 250
parent, 157
partial order, 228
partitioning, 126
Pascal's recursion, 122
Pascal's triangle, 121, 135
Pascal, B., 122
path, 142
permutation, 117
pigeonhole principle, 124
place-value system, 62
polish notation, 163
Polya, G., 7
polynomial, 96
positional system, 62
positive integers, 26
post-order, 162
postfix notation, 163
PostScript, 163
power set, 23
  cardinality, 89
pre-order, 162
predicate, 199
predicate logic, 199
  calculation rules, 205
  satisfiability in, 210
  syntax rules, 201
prefix notation, 163
prenex form, 205
prime, 39
prime factorise, 39
prime number, 39
principal remainder, 36
principle of inclusion/exclusion, 23, 131, 137
probability estimation, 107
probability theory, 105
problem-solving skills, 7
process of elimination, 306
product, 84
product set, 24
product symbol, 84
proof by contradiction, 40, 211
proof technique, 210
proper subset, 24
proposition, 179
propositional logic, 179
propositional variables, 185
pyramid, 101

$\mathbb{Q}$, 27
Q.E.D., 19
quantifiers, 199
quod erat demonstrandum, 19
quotient, 36

$\mathbb{R}$, 27
$\mathbb{R}^2$, 69
range, 231
rational numbers, 27
real numbers, 27
recursion, 71
recursive, 72
recursive algorithm, 78
recursive definition, 71
  of regular languages, 254
  of remainder in division, 73
  of rooted tree, 158
recursive equation, 88, 93
recursively defined number sequences, 74
reflexive relation, 224
registration number, 134
regular expression, 243, 252, 254
regular language, 255
relation, 217
  between sets, 221
relation graph, 219
remainder, 36
remainder in division
  recursive definition, 73
reversal
  of implication, 183
reverse polish notation, 163
rewriting
  equivalence, 188
  implications, 188
Ritchie, D. M., 251

root, 157
rooted tree, 157
rules for quadratic expansion, 6

Σ, 79
sample space, 108
satisfiable proposition, 189
schedule, 165
sedecimal form, 65
set, 12
set builder, 13
set theory, 11
sieve of Eratosthenes, 264
simple graph, 145
size
  of a set, 16
solution
  to recursive equation, 88
spanning tree, 155
square root, 231
Stalin, J., 150
state, 244
Stirling numbers of the second kind, 127
Stirling, J., 127
straight insertion, 272
string, 251
subgraph, 144
subjective probability, 107
subset, 14, 16
subtree, 158
sum, 79
summation symbol, 79
surjection, 235
surjective function, 235
symmetric relation, 224
symmetrical difference, 263

tautology, 187, 189
ternary tree, 173
tetrahedron, 101
TEX, 166, 167
time critical tasks, 166
tossing
  coins, 106
  dice, 106
total order, 229, 241
trail, 142
tranferable knowledge, 4
transition, 245
transitive relation, 226
traverse, 161
tree, 154
trigonometry, 136, 138
truth-table, 180, 185
truth-value, 180
Turing machine, 243
two-valued Boolean algebra, 192

$\mathcal{U}$, 16
unary number system, 267
unary operation, 164
uncountably infinite, 31
undirected graph, 145
uniformly distributed probability, 108
union, 14
universe, 16
unordered selection, 119
upper limit, 80

V, 140
valid statement, 189
Venn diagram, 12, 20, 131
vertex, 140

walk, 141
weight, 156
wheelgraph, 171
word, 251

XOR, 182

$\mathbb{Z}$, 27
Zamenhof, L. L., 251
zero divisor, 61
zero factor law, 61
$\mathbb{Z}_n$, 54
$\mathbb{Z}_n^2$, 69

Kimmo Eriksson is a professor specialised in discrete mathematics at Mälardalen University. He has been assigned teaching awards both at KTH and Mälardalen University. Kimmo has also written several other books in mathematics. At Studentlitteratur, he and Hillevi Gavel have published a second part of this book: *Diskret matematik fördjupning*.

Hillevi Gavel is a computer scientist and works as a lecturer in mathematics at Mälardalen University. She is a highly regarded teacher who has been a recipient of the university's distinguished teaching award. Her special area is discrete mathematics. Hillevi really enjoys making mathematical text both easy to comprehend and nice to look at.

Discrete Mathematics and Discrete Models

This book is intended as a course book in a first course in discrete mathematics. A special aim of the book is to provide an understanding of the role of discrete mathematics in modelling. Phenomena that are modelled are picked from for instance computer science, electronics, and politics.

An introductory chapter discusses what is meant by discrete mathematics and discrete models, and reviews numeracy, problem solving, and mathematical presentation. Each chapter is started by a summary of the main points that will be covered. The presentation is always based on real problems that in a natural way lead to the introduction of the mathematical concepts.

The text is written in an informal way, with many illustrations and exercises, in most cases with complete solutions.

This version in English is a direct translation of the second edition in Swedish, which means that reading instructions intended for the Swedish book can be used for this one without any modification.

Art.nr 38939

ISBN 978-91-44-10642-7

www.studentlitteratur.se