Contents

Chapter 1

Introduction and Motivation

1.1 Discrete Mathematics and Computer Science

Discrete Mathematics is concerned with finite and countably infinite mathematical structures. There are at least three major reasons why discrete mathematics is of central importance in computer science

- **Discrete structures**
  Many objects studied in computer science are discrete mathematical objects, for example a graph modeling a computer network or an algebraic group used in cryptography. Many applications exploit sophisticated properties of the involved structures
- **Abstraction**
  A computer system can only be understood by considering a number of layers of abstraction, from application programs to the physical hardware
- **Programs as mathematicals objects**
  Understanding a computer program means to understand it as a discrete mathematical object

1.2 Discrete Mathematics: A Selection of Teasers

This section contains 8 examples, worth taking a look at.

1.3 Abstraction: Simplicity and Generality

In Discrete Mathematics, abstraction stands for simplicity and generality. In order to manage complexity, software systems are divided into components (called modules, layers, objects or abstract data types) that interact with each other and with the environment in a well-defined manner.

Abstraction means *simplification*. By an abstraction one ignores all aspects of a system that are not relevant for the problem at hand, concentrating on the properties that matter

Abstraction also means *generalization*. If one proves a property of a system described at an abstract level, then this property holds for any system with the same abstraction, independently of any details.

1.4 Programs as Discrete Mathematical Objects

A computer program is a discrete mathematical object. The correctness of a program is a (discrete) mathematical statement. The proof that a program is correct (i.e., that it satisfies its specification) is a mathematical proof, not a heuristic argument.

Chapter 2

Mathematical Reasoning, Proofs, and a First Approach to Logic

2.1 What is a Proof?

2.1.2 The concept of a proof

Definition 2.1 (Informal)
A *proof* of a statement S is a sequence of simple, easy verifiable consecutive steps. The proof starts from a set of axioms (things postulated to be true) and known (previously proved) facts. Each step corresponds to the application of a derivation rule to a few already proven statements, resulting in a newly proved statement, until the final step proves S.

2.1.3 Informal vs. Formal Proofs

Formal Proof: Uses mathematical symbols and language.
Informal Proof: Explained using common, everyday language.

There are at least three (related) reasons for using a more rigurous and formal type of proof:

- **Prevention of errors**
  A completely formal proof leaves no room for interpretation, and hence allows to exclude errors
- **Proof complexity and automatic verification**
  Certain proofs are too complex to be carried out and verified "by hand". A computer is required for the verification, which can only deal with rigurously formalized statements, not with semi-precise common language.
- **Precision and deeper understanding**
  A formal proof requires the formalization of the arguments and can lead to a deeper understanding (also for the author of the proof).

2.1.4 The Role of Logic

Logic defines the syntax of language for expressing statements and the semantics of such a language, defining which statements are true and which are false. A logical calculus allows to express and verify proofs in a purely syntactic fashion, for example by a computer.

2.2 Propositions and Logical Formulas

2.2.1 Propositions, True and False

Definition 2.2
A proposition is a (mathematical) statement that is either *true* or *false*. Alternative terms for "proposition" are assertion, claim or simply statement.

Statements like "Bob loves Carol", "it is raining", and "birds can fly" are NOT (mathematical) propositions, unless we assume that the validity of these statements is defined (outside of mathematics, e.g. by common sense) to be a fixed, constant value (true or false).

Definition 2.3
A true proposition is often called a *theorem*, a *lemma*, or a *corollary*

There is no fixed naming convention, but the term "theorem" is usually used for an important result, whereas a lemma is an intermediate, often technical result, possibly used in several subsequent proofs. A corollary is a simple consequence (e.g. a special case) of a theorem or lemma.

2.2.2 Logical constants, Operators and Formulas

Definition 2.4
The Logical values (constants) "true" and "false" are usually denoted as 1 and 0, respectively.

Definition 2.5

i) The *negation* (Logical NOT) of a proposition A, denoted as $\neg A$, is true if and only if A is false.

ii) The *conjunction* (Logical AND) of two propositions A and B, denoted $A \wedge B$, is true if and only if both A and B are true.

iii) The *disjunction* (Logical OR) of two propositions A and B, denoted $A \vee B$, is true if and only if A or B (or both) are true

A	$\neg A$
0	1
1	0

A	B	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

Logical operators can also be combined, in the usual way of combining functions. For example, if A, B and C are propositions, then

$$A \vee (B \wedge C)$$

is also a proposition. Its function table is

A	B	C	$A \vee (B \wedge C)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Definition 2.6
A correctly formed expression involving propositional symbols (like $A, B, C, \dots$) and logical operators is called a *formula* (of propositional logic)

We introduce a new, derived logical operator: *implication*, denoted as $A \to B$ and defined by[1]

$$A \to B :\Longleftrightarrow \neg A \vee B$$

A	B	$A \to B$
0	0	1
0	1	1
1	0	0
1	1	1

Example 2.2

Consider the following sentence: If you read the lecture note every week and do the exercises (A), then you will get a good grade in the exam (B). This is an example of an implication $A \to B$. Saying that $A \to B$ is true does not mean

[1]The symbol $:\Longleftrightarrow$ simply means that the left side (here $A \to B$) is defined to mean the right side (here $\neg A \vee B$)

that A is true and it is not excluded that B is true even if A is false, but it is excluded that B is false when A is true.

Two-sided implication, denoted $A \leftrightarrow B$, is defined as follows[2]:

$$A \leftrightarrow B :\Longleftrightarrow (A \to B) \wedge (B \to A)$$

A	B	$A \leftrightarrow B$
0	0	1
0	1	0
1	0	0
1	1	1

Parenthesis can sometimes be dropped in a formula without changing its meaning, for example we can write $A \vee B \vee C$ instead of $A \vee (B \vee C)$ or $(A \vee B) \vee C$. Namely, $\wedge$ and $\vee$ bind stronger than $\to$ and $\leftrightarrow$. Also, $\neg$ binds stronger than $\wedge$ and $\vee$. For example, we can write $A \vee \neg B \to B \wedge C$ instead of $(A \vee (\neg B)) \to (B \wedge C)$

2.2.3 Formulas as Functions and Logical Equivalence

$(A \wedge (\neg B)) \vee (B \wedge (\neg C))$ is a formula, and corresponds to a function $\{0,1\}^3 \to \{0,1\}$.

Definition 2.7

The symbol $\top$ denote a function that is the constant 1 (true), and $\bot$ denotes the function that is constant 0 (false).

Definition 2.8

Two Formulas F and G (in propositional logic) are called *equivalent*, denoted as $F \Longleftrightarrow G$ (or also F$\equiv$ G), if they correspond to the same function (table).

For example, it is easy to see that $\wedge$ and $\vee$ are commutative and associative, i.e. for any formulas F, G, and H we have

$$F \wedge G \Longleftrightarrow G \wedge F \text{ and } F \vee G \Longleftrightarrow G \vee F$$

as well as

$$F \wedge (G \wedge H) \Longleftrightarrow (F \wedge G) \wedge H \Longleftrightarrow F \wedge G \wedge H$$

similarly

$$F \vee (G \vee H) \Longleftrightarrow (F \vee G) \vee H \Longleftrightarrow F \vee G \vee H$$

We also have

$$\neg(\neg(F)) \Longleftrightarrow F$$

[2]Note that $A \equiv B$ holds if and only if $A \leftrightarrow B$ is true, but $A \equiv B$ itself is not a logical formula.

2.2.4 Tautologies and Satisfiability

Definition 2.9

A formula F (in propositional logic) is called a *tautology* if it is true for all truth assignments of the involved propositional symbols. For example, the formulas $A \vee (\neg A)$ and $(A \wedge (A \to B) \to B$ are tautologies

Definition 2.10

A formula F (in propositional logic) is called *satisfiable* if it is true for at least one truth assignment of the involved propositional symbols, and it is called *unsatisfiable* otherwise

Lemma 2.1

F is a tautology if and only if $\neg F$ is unsatisfiable.

2.2.5 The Symbols $\Longrightarrow$ and $\Longleftrightarrow$

Definition 2.11

One writes $F \Longrightarrow G$ (or $F \Rightarrow G$)(The length of the arrow is completely irrelevant) to say that F implies G, i.e. that $F \to G$ is a tautology

As mentioned earlier, one writes $F \Longleftrightarrow G$ if F and G imply each other, i.e. if F and G are equivalent. $F \Longleftrightarrow G$ is also expressed as "F if and only if G".

To state $F \Longrightarrow G$ is the same as to state "$F \to G$ is a tautology". Note that $F \Longrightarrow G$ and $F \Longleftrightarrow G$ are not logical formulas since $\Longrightarrow$ and $\Longleftrightarrow$ are not allowed symbols in the syntax of logic.

Theorem 2.2

Implication is transitive: if $F \Longrightarrow G$ and $G \Longrightarrow H$, then $F \Longrightarrow H$

The theorem implies that, more generally, if one proves a statement F_1 as well as implications $F_1 \Longrightarrow F_2, F_2 \Longrightarrow F_3, \ldots F_{n-1} \Longrightarrow F_n$, then one has proved F_n.

2.3 Quantifiers and Predicate Logic

The extension of propositional logic is called *predicate logic* and is substantially more involved than propositional logic.

2.3.1 Predicates

Let us consider a set U as the universe in which we want to reason. For example, U could be the set $\mathbb{N}$ of natural numbers, the set $\mathbb{R}$ of real numbers, the set $\{0,1\}*$ of finite-length bit-strings, or a finite set like $\{0,1,2,3,4,5,6\}$.

Definition 2.12
A $k-$ary predicate P on U is a function $U^k \to \{0, 1\}$

A $k-$ary predicate P assigns to each list $(x_1, \ldots, x_k)$ of k elements of U the value $P(x_1, \ldots, x_k)$ which is either true (1) or false (0).

2.3.2 Definition of $\exists$ and $\forall$

Definition 2.13
For a universe U and predicate $P(x)$ we define the following logical statements

- $\forall x P(x)$ is the statement that $P(x)$ is true for all $x \in U$
- $\exists x P(x)$ is the statement that $P(x)$ is true for some $x \in U$, i.e. there exists an $x \in U$ for which $P(x)$ is true.

Sometimes one uses an abbreviated notation in which one specifies a condition for the variable x directly. For example, we can write $\forall x \geq 5(x^2 \geq 25)$ instead of $\forall x(x \geq 5) \to (x^2 \geq 25)$.

2.3.3 Nested Quantifiers

Quantifiers can also be nested. For example, if $P(x)$ and $Q(x, y)$ are predicates, then

$$\forall x(P(x) \vee \exists y Q(y, x))$$

is a logical formula.

Check out multiple examples on page 19 about nested quantifiers

2.3.4 Tautologies and Satisfiability

A formula is *satisfiable* if, informally there is an interpretation that makes the formula true. Moreover, a formula is a *tautology* if it is true for all interpretations, i.e. for all choices of the universe and for all interpretations of the predicates. For example:

$$\forall x((P(x) \wedge Q(x)) \to (P(x) \vee Q(x)))$$

is a tautology. As mentioned in section 2.2.5, we write $F \implies G$ to mean that $F \to G$ is a tautology.

If the universe is fixed and there are no unspecified parts like predicate symbols, then a formula can trivially only be a tautology or unsatisfiable.

2.3.5 Some Rules

We list a few useful rules for predicate logic

- $\forall x P(x) \wedge \forall x Q(x) \iff \forall x (P(x) \wedge Q(x))$
- $\exists x (P(x) \wedge Q(x)) \implies \exists x P(x) \wedge \exists x Q(x)$
- $(\exists x P(x) \wedge \exists x Q(x)) \rightarrow (\exists x (P(x) \wedge Q(x)))$ is not a tautology
- $\neg \forall x P(x) \iff \exists x \neg P(x)$
- $\neg \exists x P(x) \iff \forall x \neg P(x)$

2.4 Some Proof Patterns and Techniques

What is to be proven (e.g. a statement called a theorem) is given by a logical formula F, or by a sentence in natural language that could be made precise as (and hence stands for) a logical formula F. Typically F is a proposition, i.e. it is either true or false, and for formulas in predicate logic the universe is understood.

2.4.1 Modus Ponens

The following theorem states that in order to prove G one can try to identify a formula F such that one can prove both F as well as $F \rightarrow G$. The proof is similar to the proof of theorem 2.2, using the fact that the propositional formula

$$((A \wedge (A \rightarrow B))) \rightarrow B$$

is a tautology.

Theorem 2.3

If F and $F \rightarrow G$ are tautologies, then G is also a tautology.

2.4.2 Direct Proof of an Implication

Definition 2.14

A *direct proof* of an implication $F \rightarrow G$ works by assuming F and then deriving G (from F), where the derivation can possibly involve several proof steps.

2.4.3 Indirect Proof of an Implication

Definition 2.15

An *indirect proof* of an implication $F \rightarrow G$ works by assuming $\neg G$ and deriving $\neg F$, i.e. by proving $\neg G \rightarrow \neg F$

This is correct because $F \rightarrow G$ id logically equivalent to $\neg G \rightarrow \neg F$, i.e.

$$F \rightarrow G \iff \neg G \rightarrow \neg F$$

2.4.4 Proofs by Contradiction

The following theorem states that in order to prove F one can try to identify a formula G such that one can prove both $\neg G$ as well as $\neg F \to G$. The proof is similar to the proof of Theorem 2.2, using the fact that the propositional formula

$$((\neg A \to B) \wedge \neg B) \to A$$

is a tautology.

Theorem 2.4

If $\neg F \to G$ and $\neg G$ are tautologies, then F is also a tautology (or: If $\neg F \to \bot$ is a tautology, then F is also a tautology.

Such proof usually works by assuming $\neg F$, and deriving from this assumption a formula H and also its negation $\neg H$, which is a contradiction. In other words, one proves $\neg F \to H$ as well as $\neg F \to \neg H$, and hence also $\neg F \to (H \wedge \neg H)$, i.e. the formula G of the theorem is $G = H \wedge \neg H$

2.4.5 Existence Proofs

Definition 2.16
An existence proof is the proof of a statement of the form $\exists x P(x)$

There are constructive existence proofs, which demonstrate an a for which $P(a)$ is true, as well as non-constructive existence proofs which simply show the existence of an a for which $P(a)$ is true, without exhibiting such an a.

2.4.6 Inexistence Proofs

Definition 2.17
An inexistence proof is a proof of a statement of the form $\neg\exists x P(x)$

To prove the inexistence, one can use

$$\neg\exists x P(x) \Longleftrightarrow \forall x \neg P(x)$$

2.4.7 Proofs by Counterexample

Definition 2.18
A proof by counterexample is a proof of a statement of the form $\neg\forall x P(x)$ for some fixed predicate P, using

$$\neg\forall x P(x) \Longleftrightarrow \exists x \neg P(x)$$

An a for which $\neg P(a)$ is true is called a counterexample

2.4.8 Proofs by Induction

A proof by induction consists in two steps:

1. **Basis Step**: Prove $P(0)$
2. **Induction Step**: Prove $\forall n(P(n) \to P(n+1))$

Theorem 2.5

For every predicate P on $\mathbb{N}$ we have

$$P(0) \wedge \forall n(P(n) \to P(n+1)) \Longrightarrow \forall n P(n)$$

Fact 2.1

The natural numbers are well-ordered, i.e. every nonempty set of natural numbers has a least element[3]

[3]The well-ordering principle could be stated more formally as

$$\exists n P(n) \Longrightarrow \exists m\,(P(m) \wedge \forall k < m \neg P(k))$$

but we will not need this formula.

Chapter 3

Sets, Relations and Functions

There seems to be no simpler mathematical concept than a *set* (Menge), a collection of objects.

"Unter eine Menge verstehen wir jede Zusammenfassung M von bestimmten wohlunterschiedenen Objekten unserer Anschauung oder unseres Denkens (welche die Elemente von M genannt werden) zu einem Ganzen" - Georg Cantor, 1895

3.1 Sets and Operations on Sets

3.1.1 Set Memebership, Set Equality, and Cardinality

We assume that for every object x and set A it is defined whether x is an *element* of A, denoted $x \in A$, or whether it is not an element of A, denoted $x \notin A$ (instead of $\neg(x \in A)$)

Example 3.1

If $\mathbb{Z}$ denotes the set of integers, then $5 \in \mathbb{Z}$, but $\frac{1}{2} \notin \mathbb{Z}$. Two sets A and B are equal ($A = B$) if and only if they contain the same elements, independently of how A and B are described.

Definition 3.1

$$A = B :\Longleftrightarrow \forall x(x \in A \leftrightarrow x \in B)$$

Definition 3.2

The number of elements of a finite set A is called its **cardinality** and is denoted $|A|$

3.1.2 Set Description

A natural way to specify a finite set is by listing its elements. We use the standard notation $A = \{a, b, c\}$ to say that the set contains the three elements a, b, c and no other element. If two elements in the list are equal, then one of them can be dropped.

Example 3.2

$$\{1, 2, 3\} = \{1, 2, 3, 1, 3\} = \{3, 1, 2\} = \left\{2, \frac{18}{6}, \frac{6}{3}, 1\right\}$$

A set can also be described by defining a property of its elements. If A is a set and P is a predicate defined on A, then

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

which denotes the set of elements of A having property P

Example 3.3

For $A = \{n \in \mathbb{Z} \mid 1 \leq n \leq 10\}$ we have $\{n \in A \mid \text{ prime}(n)\} = \{2, 3, 5, 7\}$

3.1.3 Sets as Elements

Sets can themselves be elements of a set. For example, the set $\{3, \{4, 5\}\}$ is a set with two elements, the number 3 and the set $\{4, 5\}$.

For the operation of forming an *ordered pair* of two objects a and b, denoted (a, b), we have

$$(a, b) = (c, d) :\Longleftrightarrow a = c \wedge b = d$$

3.1.4 Subsets

An important relation between sets is the *subset* (or *inclusion*) relation.

Definition 3.3

The set A is a subset of the set B, denoted $A \subseteq B$, if every element of A is also an element of B, i.e.

$$A \subseteq B :\Longleftrightarrow \forall x(x \in A \rightarrow x \in B)$$

3.1.5 The Empty Set

Definition 3.4

The *empty set*, denoted $\varnothing$ or $\{\}$, is the set with no elements, i.e. $\forall x(x \notin \varnothing)$

Lemma 3.1

The empty set is a subset of every set, i.e. $\forall A(\varnothing \subseteq A)$.

Lemma 3.2

The empty set is unique.

3.1.6 Constructing Sets form the Empty Set

The empty set can be used to construct new sets, without using any other predefined objects as elements. The set $\{\varnothing\}$ is a set with a single element, the empty set $\varnothing$. It is important not to confuse $\varnothing$ with $\{\varnothing\}$. Note that $|\{\varnothing\}| = 1$ while $|\varnothing|$. Sets with different (finite) cardinality are different. One can define a whole sequence of such sets:

$$\varnothing, \{\varnothing\}, \{\{\varnothing\}\}\{\{\{\varnothing\}\}\}, \ldots$$

Note that except for the empty set, all these sets have cardinality 1.

3.1.7 The Power Set

Definition 3.5

The *power set* of a set A, denoted $\mathcal{P}(A)$ or sometimes 2^A, is the set of all subsets of A:

$$\mathcal{P}(A) := \{S \mid S \subseteq A\}$$

For a finite set with cardinality k, the power set has cardinality 2^k (hence the name 'power set' and the alternative notation 2^A)

3.1.8 Union, Intersection, Difference and Complement

Definition 3.6

The *union* of two sets A and B is defined as

$$A \cup B := \{x \mid x \in A \vee x \in B\} \text{ or } \forall x(x \in A \cup B \leftrightarrow x \in A \vee x \in B)$$

and their *intersection* is defined as

$$A \cap B := \{x \mid x \in A \wedge x \in B\}$$

Let $\mathcal{A}$ be a set of sets, with finite or infinite cardinality. The only restriction on $\mathcal{A}$ is that its elements must be sets. Then we define the union of all sets in $\mathcal{A}$ as the set of all x that are an element of at least one of the sets in $\mathcal{A}$:

$$\bigcup \mathcal{A} := \{x \mid \exists A \in A : x \in A\}$$

Similarly, we define the intersection of all sets in $\mathcal{A}$ as the set of all x that are an element of every set in $\mathcal{A}$:

$$\bigcap \mathcal{A} := \{x \mid \forall A \in A : x \in A\}$$

Definition 3.7

For a given universe of discourse U, the *complement* of a set A, denoted $\overline{A}$ is

$$\overline{A} := \{x \in U \mid x \notin A\}$$

or simply $\overline{A} = \{x \mid x \notin A\}$[1].

If union and intersection are understood as logical OR and AND, respectively, then the complement can be understood as the logical negation.

Problem with footnote inside framed environment!!

Definition 3.8

The *difference* of sets B and A, denoted $B - A$ (or sometimes $B \backslash A$) is the complement of A, relative to B:

$$B - A := \{x \in B \mid x \notin A\}$$

Theorem 3.3

For any sets A, B and C, and a universe U, the following laws hold:

Idempotence:	$A \cap A = A$
	$A \cup A = A$
Commutativity:	$A \cap B = B \cap A$
	$A \cup B = B \cup A$
Associativity:	$A \cap (B \cap C) = (A \cap B) \cap C$
	$A \cup (B \cup C) = (A \cup B) \cup C$
Absorption:	$A \cap (A \cup B) = A$
	$A \cup (A \cap B) = A$
Distributivity:	$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
	$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
Complementarity:	$A \cap \overline{A} = \varnothing$
	$A \cup \overline{A} = U$
Consistency:	$A \subseteq B \iff A \cap B = A \iff A \cup B = B$

3.1.9 The Cartesian Product

Definition 3.9

The *Cartesian product* $A \times B$ of two sets A and B is the set of all ordered pairs with the first component from A and the second component from B:

$$A \times B = \{(a, b) \mid a \in A \wedge b \in B\}$$

For finite sets, the cardinality of the Cartesian product of some sets is the product of their cardinalities.

3.1.10 Russell's Paradox

The problem with Cantor's intuitive set theory is that, because it was not clearly axiomatized, it makes the following apparently innocent (yet false) assumption. Whenever one specifies a precise condition, allowing to distinguish between objects that satisfy the condition and objects that don't, then $\{x \mid P(x)\}$, the set of objects satisfying the definition, is well-defined. Russell proposed the set

$$R = \{A \mid A \notin A\}$$

of sets that are not themselves. Note that there seems nothing wrong with a set being an element of itself. For example, the set of sets containing at least 10 elements is certainly an element of itself, as it contains 10 elements.

Check again, maybe find a better and shorter explanation

3.2 Relations

3.2.1 The Relation Concept

Definition 3.10

A (binary) relation ρ from a set A to a set B is a subset of $A \times B$. If $A = B$, then ρ is called a relation on A

Instead of $(a, b) \in \rho$ one usually writes

$$a \rho b$$

and sometimes we write $a \not\rho b$ if $(a, b) \notin \rho$

3.2.2 Representations of Relations

For finite sets A and B, a (binary) relation ρ from A to B can be represented as a Boolean $|A| \times |B|$ matrix M^ρ with the rows and columns labeled by the elements of A and B respectively. For $a \in A$ and $b \in B$, the matrix entry $M^\rho_{a,b}$ is 1 if $a \rho b$, 0 otherwise.

3.2.3 Set Operations on Relations

Relations are sets, and therefore we can apply any operation defined on sets: union, intersection and complement.

3.2.4 The Inverse of a Relation

Definition 3.12

The *inverse* of a relation ρ from A to B is the relation $\widehat{\rho}$ from B to A such that

$$a \rho b \iff b \widehat{\rho} a$$

An alternative notation for the inverse of ρ is ρ^{-1}

In the matrix representation, taking the inverse of a relation corresponds to the transposition of the matrix. In the graph representation, taking the inverse corresponds to inverting the direction of all edges.

3.2.5 Composition of Relations

Definition 3.13

Let ρ be a relation from A to B and let σ be a relation from B to C. Then the *composition* of ρ and σ, denoted $\rho\sigma$ (or also $\rho \circ \sigma$), is the relation from A to C where

$$a\rho\sigma c :\Longleftrightarrow \exists b \in B : (a\rho b \wedge b\sigma c)$$

The $n-$fold composition of a relation ρ on a set A is denoted ρ^n

Lemma 3.4

The composition of relations is associative, i.e. we have $\rho(\sigma\phi) = (\rho\sigma)\phi$

Lemma 3.5

Let ρ be a relation from A to B and let σ be a relation from B to C. Then the inverse $\widehat{\rho\sigma}$ of $\rho\sigma$ is the relation $\widehat{\sigma}\widehat{\rho}$

3.2.6 Special Properties of Relations

Definition 3.14

A relation ρ on a set A is called *reflexive* if $a\rho a$ for every $a \in A$, i.e. if $\mathrm{id} \subseteq \rho$

In other words, a relation is reflexive if it contains the equality relation (=). In the matrix representation of relations, reflexive means that the diagonal is all 1. In a graph representation, reflexive means that every vertex has a loop (an edge from the vertex to itself).

Definition 3.16

A relation ρ on a set A is called *symmetric* if $\rho = \widehat{\rho}$, i.e. if

$$a\rho b \Longleftrightarrow b\rho a$$

In the matrix representation of relations, symmetric means that the matrix is symmetric (with respect to the diagonal).

Definition 3.17
A relation ρ on a set A is called *antisymmetric* if $\rho \cap \widehat{\rho} \subseteq \mathrm{id}$, i.e.

$$a\rho b \wedge b\rho a \Longrightarrow a = b$$

In a graph representation, antisymmetric means that there is no cycle of length 2, i.e. no distinct vertices a and b with edges both from a and b and from b to a. Note that antisymmetric is **not** the negation of symmetric.

Definition 3.18
A relation ρ on a set A is called *transitive* if

$$a\rho b \wedge b\rho c \Longrightarrow a\rho c$$

Lemma 3.6

A relation ρ is transitive if and only if $\rho^2 \subseteq \rho$.

3.2.7 Transitive Closure

Definition 3.19
The transitive closure of a relation ρ on a set A, denoted ρ^*, is

$$\rho^* = \bigcup_{n=1}^{\infty} \rho^n$$

In the graph representation of a relation ρ on A, we have $a\rho^k b$ if and only if there is a path of length k from a to b in the graph. The transitive closure is the reachability relation, i.e. $a\rho^* b$ if and only if there is a path (of arbitrary finite length) from a to b.

3.3 Equivalence Relations

3.3.1 Definition of Equivalence Relations

Definition 3.20
An *equivalence relation* is a relation that is reflexive, symmetric and transitive

Definition 3.21

For an equivalence relation θ on a set A and for $a \in A$, the set of elements of A that are equivalent to a is called the *equivalence class of* a and is denoted as $[a]_\theta$:

$$[a]_\theta := \{b \in A \mid b\theta a\}$$

Lemma 3.7

The intersection of two equivalence relations is an equivalence relation.

Example 3.44

The intersection of $\equiv_5$ and $\equiv_3$ is $\equiv_{15}$.

3.3.2 Equivalence Classes Form a Partition

Definition 3.22

A *partition* of a set A is a set $\{S_i \subseteq A\}_{i \in I}$ of mutually disjoint subsets of A that cover A, i.e.

$$S_i \cap S_j = \varnothing \quad \text{for } i \neq j \qquad \text{and} \qquad \bigcup_{i \in I} S_i = A$$

Partitions and equivalence relations capture the same (simple) abstraction.

Definition 3.23

The set of equivalence classes of an equivalence relation θ, denoted by

$$A/\theta := \{[a]_\theta \mid a \in A\}$$

is called the *quotient set of* A by θ, or simply A modulo θ, or A mod θ.

Theorem 3.8

The set A/θ of equivalence classes of an equivalence relation θ on A is a partition of A.

Maybe add proof on page 45

3.3.3 Example: Definition of the Rational Numbers

send back to the book!

3.4 Partial Order Relations

3.4.1 Definition

Definition 3.24
A *partial order* on a set A is a relation that is reflexive, antisymmetric and transitive. A set A together with a partial order $\preceq$ on A is called a *partially ordered set* (or simply *poset*) and is denoted as $(A; \preceq)^2$

In a graph representation of relations, a partial order has no cycles.

For a partial order relation $\preceq$ we can define the relation $a \prec b$ similar to how the relation $<$ is obtained from $\leq$:

$$a \prec b :\Longleftrightarrow a \preceq b \wedge a \neq b$$

Definition 3.25
For a poset $(A; \preceq)$, two elements a and b are called *comparable* if $a \preceq b$ or $b \preceq a$; otherwise they are called *incomparable*

Definition 3.26
If any two elements of a poset $(A; \preceq)$ are comparable, then A is called *totally ordered* (or *linearly ordered* by $\preceq$

Definition 3.27
A poset $(A; \preceq)$ is *well-ordered* if it is totally ordered and if every non-empty subset of A has a least element

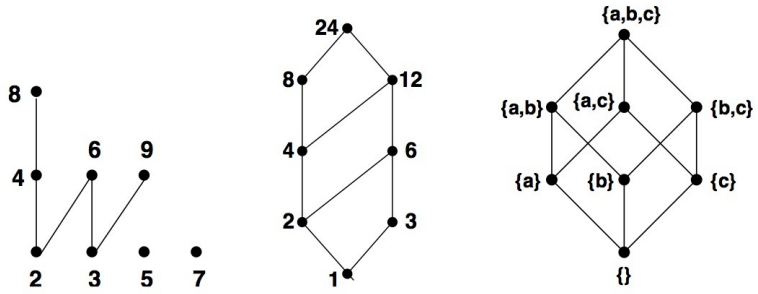
Note that every totally ordered finite poset is well-ordered. The property of being well-ordered is of interest only for infinite posets. Any subset of the natural numbers is also well-ordered.

3.4.2 Hasse Diagrams

Definition 3.28
In a poset $(A; \preceq)$ an element b is said to *cover* an element a if $a \prec b$ and there exists no c with $a \prec c$ and $c \prec b$ (i.e. between a and b)

Definition 3.29
The *Hasse diagrams* of a (finite) poset $(A; \preceq)$ is the directed graph whose vertices are labeled with the elements of A and where there is an edge from a to b if and only if b covers a

The Hasse diagram is a graph with directed edges. It usually drawn such that whenever $a \prec b$, then b is placed higher than a. This means that all arrows are directed upwards ad therefore can be omitted.



3.4.3 Chains and Antichains

Definition 3.30
A totally ordered subset (with respect to $\preceq$) $C \subseteq A$ of a poset $(A; \preceq)$ is called a chain $(\text{in}(A; \preceq))$

Definition 3.31
A subset $B \subseteq A$ of a poset $(A; \preceq)$ is called an *antichain* if any two distinct elements in B are incomparable

3.4.4 Combinations of Posets and the Lexicographic Order

Theorem 3.9

For given posets $(A; \preceq)$ and $(B; \sqsubseteq)$, the relation $\leq_{\text{lex}}$ defined on $A \times B$ by

$$(a_1, b_1) \leq (a_2, b_2) :\Longleftrightarrow a_1 \preceq a_2 \wedge b_1 \sqsubseteq b_2$$

is a partial order relation.

Theorem 3.10

For given posets $(A; \preceq)$ and $(B; \sqsubseteq)$, the relation $\leq_{\text{lex}}$ defined on $A \times B$ by

$$(a_1, b_1) \leq_{\text{lex}} (a_2, b_2) :\Longleftrightarrow a_1 \prec a_2 \vee (a_1 = a_2 \wedge b_1 \sqsubseteq b_2)$$

is a partial order relation.[3]

3.4.5 Special elements in Posets

[3] As mentioned earlier, for a partial order $\preceq$ we can define the relation $\prec$ as $a \prec b :\Longleftrightarrow a \preceq b \wedge a \neq b$

Definition 3.32

Let $(A; \preceq)$ be a poset, and let $S \subseteq A$ be some subset of A. Then

1. $a \in S$ is a *minimal (maximal) element* of S if there exists no $b \in S$ with $b \prec a$ ($b \succ a$)
2. $a \in S$ is the *least (greatest) element* of S if $a \preceq b$ ($a \succeq b$) for all $b \in S$
3. $a \in A$ is a *lower (upper) bound* of S if $a \preceq b$ ($a \succ b$) for all $b \in S$
4. $a \in A$ is the *greatest lower bound (least upper bound)* of S if a is the greatest (least) element of the set of all lower (upper) bounds of S

Minimal, maximal, least and greatest elements are easily identified in a Hasse diagram.

The greatest lower bound and the least upper bound of a set S are sometimes denotes as $\text{glb}(S)$ and $\text{lub}(S)$, respectively.

3.4.6 Meet, Join and Lattices

Definition 3.33

Let $(A; \preceq)$ be a poset. If a and b (i.e. the set $\{a, b\} \subseteq A$) have a greatest lower bound, then it is called the *meet* of a and b, often denoted $a \wedge b$. If a and b have a least upper bound, then it is called the *join* of a and b, often denoted $a \vee b$.

Definition 3.34

A poset $(A; \preceq)$ in which every pair of elements has a meet and a join is called a *lattice*

3.5 Functions

3.5.1 The Function Concept

Definition 3.35

A function $f : A \to B$ from a *domain* A to a *codomain* B is a relation $A \times B$, i.e. a subset $f \subseteq A \times B$, with the special properties (using the relation notation afb):

1. $\forall a \in A \;\; \exists b \in B \;\; afb$ (f is totally defined)
2. $afb \wedge afb' \;\; \Rightarrow b = b'$ (f is well-defined)

The set of all functions $A \to B$ is denoted as B^A. This notation is motivated by the fact that if A and B are finite, then there are $|B|^{|A|}$ such functions (see Section 4.1).

Definition 3.36
A *partial function* is a relation $A \times B$ such that condition 2. above holds

Definition 3.37
Two (partial) functions with common domain A and codomain b are *equal* if they are equal as relations (i.e. as sets)

$f = g$ is equivalent to saying that the function values of f and g agree for all arguments (including, in case of partial functions, whether or not it is defined)

Definition 3.38
For a function $f : A \to B$ and a subset S of A, the *image* of S under f, denoted $f(S)$, is the set

$$f(S) := \{f(a) \mid a \in S\}$$

Definition 3.39
The subset $f(A)$ of B is called the *image* (or *range*) of f and is also denoted $\mathrm{Im}(f)$

Definition 3.40
For a subset T of B, the *inverse image* (or *preimage*) of T, denoted $f^{-1}(T)$, is the set of values in A that map into T:

$$f^{-1}(T) := \{a \in A \mid f(a) \in T\}$$

3.5.2 Properties of Functions and Function Composition

Definition 3.41
A function $f : A \to B$ is called

1. *Injective* if $a \neq b \Rightarrow f(a) \neq f(b)$, i.e. no two distinct values are mapped to the same function value (there are no "collisions")
2. *Surjective* (or *onto*) if for every $b \in B$, $b = f(a)$ for some $a \in A$, i.e. if $f(A) = B$ (every value in the codomain is taken on for some argument)
3. *Bijective* if it is both surjective and injective.

Definition 3.42

The *composition* of a function $f : A \to B$ and a function $g : B \to C$, denoted by $g \circ f$, or simply gf, is defined by $g \circ f(a) = g\,(f(a))$

Lemma 3.11

Function composition is associative, i.e. $(h \circ g) \circ f = h \circ (g \circ f)$

Chapter 4

Combinatorics and Counting

4.1 Basic Counting Principles

4.1.1 The Addition, Multiplication and Bijection Principles

The cardinality of the union n disjoint sets $A_1, \ldots, A_n$ is equal to the sum of the cardinalities. This is known as the *addition principle*:

$$\forall i, j, 1 \leq i < j \leq n : A_i \cap A_j = \emptyset \Longrightarrow |A_1 \cup \cdots \cup A_n| = \sum_{i=1}^{n} |A_i|$$

The *multiplication principle* for finite sets states the obvious fact that $|A \times B| = |A| \cdot |B|$ and, more generally

$$|A_1 \times \cdots \times A_n| = \prod_{i=1}^{n} |A_i|$$

Bijection Principle: If there is a bijection (or one-to-one correspondence) between the finite sets A and B, then $|A| = |B|$

4.1.2 The Inclusion-Exclusion Priniciple

We now consider the generalization of the addition principle when the sets $A_1, \ldots, A_n$ are not disjoint. We have

$$|A \cup B| = |A| + |B| - |A \cap B|$$

which is obvious when drawing a Venn diagram. More formally, $A \cup B$ is the union of three disjoint sets $A - B$, $A \cap B$, and $B - A$, and we have $|A - B| = |A| - |A \cap B|$ and $|B - A| = |B| - |A \cap B|$. More generally

$$|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C|$$

Theorem 4.1

For any finite sets $A_1, \ldots, A_n$

$$\begin{aligned} |A_1 \cup \cdots \cup A_n| = & \sum_{i=1}^{n} |A_i| \\ & - \sum_{1 \le i_1 < i_2 \le n} |A_{i_1} \cap A_{i_2}| \\ & + \sum_{1 \le i_1 < i_2 < i_3 \le n} |A_{i_1} \cap A_{i_2} \cap A_{i_3}| \\ & - \ldots \\ & + (-1)^{n-1} |A_1 \cap \cdots \cap A_n| \end{aligned}$$

Theorem 4.2

$$|A_1 \cup \cdots \cup A_n| \ge \sum_{i=1}^{n} |A_i| - \sum_{1 \le i_1 < i_2 \le n} |A_{i_1} \cap A_{i_2}|$$

4.1.3 Drawing Elements From a Set

One of the most fundamental counting problems is to count the number of ways in which one can select k elements from a set of cardinality n. There are four different flavors of this problem, depending on whether or not the k selected elements must be distinct (with or without repetition), and whether or not the order in which they are drawn matters. We recall that $k! = 1 \cdot 2 \cdot 3 \ldots (k-1) \cdot k$

- **Ordered selection with repetition**
  We have already seen earlier that there are n^k strings of length k over an alphabet of size n. There is an obvious one-to-one correspondence between these strings and all the ways of selecting an ordered list of k objects (with repetitions allowed) from n objects.

- **Ordered selection without repetition**
  If the k elements must be distinct, there are n choices for the first elements, $n-1$ choices for the second element, etc. In total there are:

  $$n^{\underline{k}} := n(n-1)(n-2)\ldots(n-k+1) = \prod_{i=0}^{k-1} (n-1) = \frac{n!}{(n-k)!}$$

  possibilities[1]. For $n = k$, this corresponds to the number $k!$ of bijections from $\{1, 2, \ldots, k\}$ to $\{1, 2, \ldots, k\}$

[1] We define $0! := 1$

- **Unordered selection without repetition**
  If the order in the above selection is irrelevant, then the $n^{\underline{k}}$ possibilities can be divided into equivalence classes of size $k!$, each class being characterized by the (unordered) set of selectd elements. Therefore the number of possibilities is:

$$\left(\begin{array}{c} n \\ k \end{array} \right) := \frac{n^{\underline{k}}}{k!} = \frac{n\,(n-1)\,(n-2)\ldots(n-k+1)}{k!} = \frac{n!}{k!\,(n-k)!}$$

-