Automata, Formal Languages, and Computation

Lecture Notes

Prof. Dr. Titus Dose

OTH Regensburg – Faculty of Computer Science and Mathematics

October 8, 2024

Table of Contents I

Table of Contents II

Table of Contents III

1
Mathematical Basics

1.1
Foundations from Propositional Logic

This lecture is about proving mathematical statements.
A mathematical statement is either true (1) or false (0). So it always has a unique truth value.[1]

We will be using the following logical connections that turn statements into more complicated statements:

- "and", conjunction, $\wedge$
- "or", disjunction, $\vee$
- "not", negation, $\neg$
- implication, $\Rightarrow$, "if ..., then ..."
- equivalence, $\Leftrightarrow$, "if and only if ..., then ..."

In general, the natural language description should be preferred to the formula notation using symbols: e.g., "and" should be preferred to "$\wedge$".

[1] So we do not consider the liar's paradox ("This statement is false") as a statement. The liar's paradox will be discussed later on.

The following truth value table defines the connections:

A	B	$\neg A$	$A \wedge B$	$A \vee B$	$A \Rightarrow B$	$A \Leftrightarrow B$
1	1	0	1	1	1	1
1	0	0	0	1	0	0
0	1	1	0	1	1	0
0	0	1	0	0	1	1

Some typical mistakes and misunderstandings:

- the disjunction is inclusive (contrary to "either-or"), i.e., the following statement is true: "2 is even or 2 is a prime".
- the negation of the statement "Each prime is odd" is **not** "Each prime is even", but "There is an even prime".
- Everything is implied by a false statement. E.g., the following statement is true: "If 9 is a prime, then there are infinitely many twin primes" (even if we do not know whether there are infinitely many twin primes).[2]

[2] This is in accordance with everyday language: the statement "If it rains tomorrow, then the street will get wet" is considered to be true even if it does not rain the next day.

- Note: $A \Leftrightarrow B$ is equivalent with $(A \Rightarrow B) \wedge (B \Rightarrow A)$

A	B	$A \Rightarrow B$	$B \Rightarrow A$	$(A \Rightarrow B) \wedge (B \Rightarrow A)$	$A \Leftrightarrow B$
1	1	1	1	1	1
1	0	0	1	0	0
0	1	1	0	0	0
0	0	1	1	1	1

A typical mistake is to forget one of the two implications, especially when having a longer chain of equivalent statements. As an example, consider the following statements about a real number x

$$\begin{aligned} x = 2 &\Leftrightarrow x + 1 = 3 \\ &\Leftrightarrow (x+1)^2 = 9 \text{ WRONG!} \\ &\Leftrightarrow 2^{(x+1)^2} = 512 \end{aligned}$$

However, the statements "$x = 2$" and "$2^{(x+1)^2} = 512$" are not equivalent as $-4 \neq 2$, but $2^{(-4+1)^2} = 512$.

Tip: In the first semester, mainly do without $\Leftrightarrow$ and instead prove $A \Rightarrow B$ and $B \Rightarrow A$ separately.

1.2
Methods of Proof

We will be using 3.5 methods of proof.

1. Direct proof
2. Indirect proof
    2.1 Proof by contradiction
    2.2 Proof by contraposition
3. Induction

Direct Proof

Whenever we prove a mathematical statement, we start with a statement A, assume it to be true, and show that then another statement B is also true (i.e., we always prove implications).

The condition A is not always stated explicitly. We can e.g. prove the statement "2 is a prime". But where's the premise?

When proving the statement, we will have to use some premises, e.g., certain properties of the multiplication of natural numbers, properties of natural numbers,...
We cannot squeeze water from a stone. We always need axioms.

Direct proof: we assume that A is true and show by a chain of logical inferences[3] that B is true.

[3] It can be proven that a couple of simple inference rules suffice. Thus, it is easy to write a program that can check arbitrary mathematical proofs for their correctness (if they are written down detailed enough).

Direct Proof – Example

Let p be a natural number. Let A be the statement "p is an even prime" and B the statement "p is less than 5". We now prove $A \Rightarrow B$.

Proof.

We assume that A is true. So p is an even prime.
A implies statement C: "$p = 2$" because every other even number is divisible by 2 and thus no prime.
C implies B as 2 is less than 5. □

Contraposition

The statement $A \Rightarrow B$ is equivalent to $\neg B \Rightarrow \neg A$.[4]

Instead of proving $A \Rightarrow B$ directly, we give a direct proof for the statement $\neg B \Rightarrow \neg A$.

Example: Let p be a natural number. Let A be the statement "p^2 is not divisible by 4" and B the statement "p is odd". We prove $A \Rightarrow B$.

Proof.

We give a proof by contradiction.
So we start with the statement $\neg B$: "p is even".
Then $p = 2 \cdot q$ for some natural number q.
Then $p^2 = (2 \cdot q)^2 = 4 \cdot q^2$, so p^2 is divisible by 4, which is the statement $\neg A$. □

[4] Write down the truth value table if you are not convinced. Cf. page 20.

Proof by Contradiction

The proof by contradiction is similar to the proof by contraposition.

We prove $A \Rightarrow B$ by starting with the statement $A \wedge \neg B$ and inferring a contradiction. Then our assumption $A \wedge \neg B$ must have been wrong (otherwise, it wouldn't imply a contradiction).

Thus, it is impossible that A is true and B is not. In other words, if A is true, then B is also true, i.e., $A \Rightarrow B$.

Proof by Contradiction – Example

Example: Let us prove that every natural number greater than 1 has a divisor that is prime (prime divisor).

Proof.

Let's assume for a contradiction that there exists some number that does not have a prime divisor. Let p be the least such number.

Then p is no prime (p is divisor of p). Thus, $p = r \cdot s$ for natural numbers $r, s > 1$.

As p is the least number > 1 without any prime divisors, $r > 1$, and $r < p$, the number r has a prime divisor t, i.e., $r = t \cdot u$ for a natural number u.

But then $p = r \cdot s = t \cdot u \cdot s$ and since t is prime, p has a prime divisor. This is a contradiction. So our assumption was wrong and every natural number > 1 has a prime divisor. □

Proof by Contradiction – 2nd Example

Now we can prove with the same technique that there are infinitely many primes.

Proof.

Let's assume for a contradiction that there exist only finitely many primes, i.e., there are some $n \in \mathbb{N}$ and primes $p_1, \dots, p_n$ such that these are the only primes.

Consider $p = p_1 \cdot p_2 \cdot \dots \cdot p_n + 1$.
As p is greater than 1, it has a prime divisor (by the statement proven on the last page). So there is some i so that p_i is a divisor of p, i.e., $p = p_i \cdot s$ for a natural number s.
Thus $s = \frac{p}{p_i} = \frac{p_1 \cdot p_2 \cdot \dots \cdot p_n + 1}{p_i} = \frac{p_1 \cdot p_2 \cdot \dots \cdot p_n}{p_i} + \frac{1}{p_i}$, i.e., $\frac{1}{p_i} = s - \frac{p_1 \cdot p_2 \cdot \dots \cdot p_n}{p_i}$.
So $\frac{1}{p_i}$ is the difference of two natural numbers and hence, an integer itself. However, as p_i is prime, $0 < \frac{1}{p_i} \leq \frac{1}{2}$, a contradiction. □

Complete Induction

Complete Induction is a method of proof we will be using extensively.

Consider the statement: For all natural numbers n the following statement A_n is true:

$$\sum_{i=1}^{n} i = \frac{n^2+n}{2} \quad .$$[5]

Proving all statements A_n for $n \in \mathbb{N}$ directly is not straightforward. The idea of complete induction is to prove the statement A_0 (base case, BC) and "for all $k \in \mathbb{N}$, if A_k is true, then so is A_{k+1}" (induction step, IS). This proves that all A_n are true:

$$A_0 \Rightarrow A_1 \Rightarrow A_2 \Rightarrow A_3 \Rightarrow A_4 \Rightarrow \dots$$

Note: BC and IS can be proven using any of the proof methods.

[5] $\sum_{i=1}^{n} i = 1 + 2 + \cdots + (n-1) + n$

Complete Induction – Example

For all natural numbers n the following statement A_n is true:

$$\sum_{i=1}^{n} i = \frac{n^2 + n}{2}$$

Proof.

BC: We prove A_0: It holds $\sum_{i=1}^{0} i = 0 = \frac{0^2+0}{2}$.

IS: We prove “for all $k \in \mathbb{N}$, if A_k is true, then so is A_{k+1}”. Let $k \in \mathbb{N}$. We assume that A_k is true (**induction hypothesis**). So $\sum_{i=1}^{k} i = \frac{k^2+k}{2}$.
In order to complete the proof, we need to prove A_{k+1}

Recall: We have assumed the statement A_k:

$$\sum_{i=1}^{k} i = \frac{k^2+k}{2}.$$

We want to prove the statement A_{k+1} (in order to complete IS):

$$\sum_{i=1}^{k+1} i = \frac{(k+1)^2+k+1}{2}.$$

Then

$$\sum_{i=1}^{k+1} i = k+1+\underbrace{\sum_{i=1}^{k} i}_{\overset{A_k}{=}\frac{k^2+k}{2}} = k+1+\frac{k^2+k}{2} = \frac{2(k+1)+k^2+k}{2}$$

$$= \frac{k^2+2k+1+k+1}{2} = \frac{(k+1)^2+k+1}{2}$$

holds[6]. Thus A_{k+1} holds, IS is proven, and the proof is complete.

□

[6] Note: the last equation is due to Francesco Binomi (1727 – 1643).

Inductive Definitions

Having introduced the concept of proofs via complete induction, we can now use the same concept for definitions (inductive definition).

We define the summation operation $\sum$. Let $s_1, s_2, \ldots$ be natural numbers. We define:

BC: $\sum_{i=1}^{0} s_i := 0$.
IS: $\sum_{i=1}^{k+1} s_i := s_{k+1} + \sum_{i=1}^{k} s_i$.

Why does this define e.g. the expression $\sum_{i=1}^{5} i$?

$$\sum_{i=1}^{5} i \overset{\text{IS}}{=} 5 + \sum_{i=1}^{4} i \overset{\text{IS}}{=} 5 + 4 + \sum_{i=1}^{3} i \overset{\text{IS}}{=} 5 + 4 + 3 + \sum_{i=1}^{2} i$$

$$\overset{\text{IS}}{=} 5 + 4 + 3 + 2 + \sum_{i=1}^{1} i \overset{\text{IS}}{=} 5 + 4 + 3 + 2 + 1 + \sum_{i=1}^{0} i$$

$$\overset{\text{BC}}{=} 5 + 4 + 3 + 2 + 1 + 0.$$

Structural Induction

Inductive proofs are not only useful when considering statements that hold for all natural numbers.

Generally, they can be used for recursively/inductively defined structures. E.g., for **polynomials** over the integers with one variable:

BC Let x be a variable. Then

- x is a polynomial
- each integer z is a polynomial

IS If p_1 and p_2 are polynomials over the integers with one variable x, then $(p_1 + p_2)$ and $(p_1 \cdot p_2))$ are polynomials.

Example:[7] $((5 \cdot ((((x \cdot x) \cdot x) - (3 \cdot (x \cdot x))) + 12)) \cdot ((((3 \cdot x) - 5))))$
5, x, 3, 12 are polynomials by BC. Applying IS iteratively finally leads to the polynomial above.

[7] Note: later on we will omit unnecessary brackets and write $5 \cdot (x^3 - 3x^2 + 12) \cdot (3x - 5)$.

Structural Induction – Example

Let's prove: All polynomials have the same number of opening and closing brackets.

Proof.

BC: Neither x nor an integer contains any bracket.

IS: Let p_1 be polynomial with a opening and a closing brackets. Let p_2 be polynomial with b opening and b closing brackets.

The polynomial $(p_1 + p_2)$ has $a + b + 1$ opening and closing brackets. The same holds for $(p_1 \cdot p_2)$. This completes the proof.

In the following we use the term "induction" for both "complete induction" and "structural induction". □

1.3
Basic Notations and Terms

We use ( ) and [ ] to bracket mathematical expressions. Arguments of functions are placed in ( ). Sets are bracketed with { }.

$$\begin{aligned}
\mathbb{N} &\overset{\mathit{df}}{=} \{0, 1, \dots\} = \text{Set of natural numbers}\\
\mathbb{N}^{+} &\overset{\mathit{df}}{=} \{1, 2, \dots\} = \text{Set of positive natural numbers}\\
\mathbb{P} &\overset{\mathit{df}}{=} \{2, 3, 5, 7, 11, 13, 17, \dots\} = \text{Set of primes}\\
\mathbb{Z} &\overset{\mathit{df}}{=} \{\dots, -2, -1, 0, 1, 2, \dots\} = \text{Set of integers}\\
\emptyset &\overset{\mathit{df}}{=} \text{empty set}
\end{aligned}$$

Important: **0 is a natural number.**

For $x \in \mathbb{N}$, $y \in \mathbb{N}^{+}$ let $(x \text{ mod } y)$ denote the rest of the division x/y, i.e., $(x \text{ mod } y) = x - zy$, where $z \in \mathbb{N}$ is the greatest number with $zy \leq x$.

Sets

Notation for the definition of sets:

$$\begin{aligned}\{n : n \in \mathbb{N} \text{ and } n \geq 5\} &= \{n \in \mathbb{N} : n \geq 5\} \\ &= \{n \mid n \in \mathbb{N} \text{ and } n \geq 5\} \\ &= \{n \in \mathbb{N} \mid n \geq 5\} \\ &= \{5, 6, \ldots\}\end{aligned}$$

Element relationship and inclusion:

- $a \in M \overset{df}{\Longleftrightarrow}$ a is an element of the set M
- $a \notin M \overset{df}{\Longleftrightarrow}$ a is no element of the set M
- $M \subseteq N \overset{df}{\Longleftrightarrow}$ for all a, if $a \in M$, then $a \in N$ (M is subset of N)
- $M \nsubseteq N \overset{df}{\Longleftrightarrow}$ it does not hold $M \subseteq N$ (M is no subset of N)
- $M \subsetneq N \overset{df}{\Longleftrightarrow}$ $M \subseteq N$ and $M \neq N$ (M is proper subset of N)

Set Operations

$$
\begin{array}{rcll}
A \cap B & \overset{df}{=} & \{a \mid a \in A \text{ and } a \in B\} & \text{(Intersection of } A \text{ and } B\text{)} \\
A \cup B & \overset{df}{=} & \{a \mid a \in A \text{ or } a \in B\} & \text{(Union of } A \text{ and } B\text{)} \\
A \setminus B & \overset{df}{=} & \{a \mid a \in A \text{ and } a \notin B\} & \text{(Difference of } A \text{ and } B\text{)} \\
A - B & \overset{df}{=} & A \setminus B & \text{(Difference of } A \text{ and } B\text{)} \\
\overline{A} & \overset{df}{=} & M \setminus A & \text{(Complement of } A \text{ relative} \\
 & & & \text{ to a fixed base set } M\text{)} \\
\mathcal{P}(A) & \overset{df}{=} & \{B \mid B \subseteq A\} & \text{(Power set of } A\text{)}
\end{array}
$$

$\#A = |A| \overset{df}{=}$ number of elements of a finite set A

Tuple (Vector) and Cartesian Product

For $n \in \mathbb{N}$ we define the following.

- $(a_1, a_2, \ldots, a_n) \overset{df}{=}$ sequence of elements $a_1, a_2, \ldots, a_n$ in this order (n-tuple, n-dimensional vector)
- $A_1 \times A_2 \times \ldots \times A_n \overset{df}{=} \{(a_1, a_2, \ldots, a_n) \mid a_i \in A_i \text{ for all } i\}$ (Cartesian product of sets A_1,$A_2, \ldots, A_n$)
- $A^n \overset{df}{=} \underbrace{A \times A \times \ldots \times A}_{n \text{ times}}$ (n-dim. Cartesian product of set A)

The first definition yields the empty tuple $()$ for $n = 0$. Thus $A^0 = \{()\}$ and $|A^0| = 1$.

Quantifiers

Quantifiers are merely abbreviations:

- $\exists =$ "there exist(s)"
- $\forall =$ "for all"

Instead of "Every even natural number greater or equal 4 is the sum of two (non-necessarily distinct) primes" we may write as shortcut $\forall_{n \in \{2 \cdot i \mid i \in \mathbb{N}, i \geq 2\}} \exists_{p,q \in \mathbb{P}}\ n = p + q$.

For better readability, we will go without subscripts when using quantifiers, i.e., $\forall n \in \{2 \cdot i \mid i \in \mathbb{N}, i \geq 2\} \exists p, q \in \mathbb{P}\ n = p + q$.

Negating such statements can be done mechanically, replace $\forall$ with $\exists$ and vice versa, and negate the remaining statement: $\exists n \in \{2 \cdot i \mid i \in \mathbb{N}, i \geq 2\} \forall p, q \in \mathbb{P}\ n \neq p + q$.

Functions

A function f from A to B (notation $f : A \to B$) is determined by the source set (aka domain) A, the target set (aka codomain) B, and the graph $G_f \subseteq A \times B$, where for every $a \in A$ there is at most one $b \in B$ with $(a, b) \in G_f$.

If $(a, b) \in G_f$, then the function f at the position a has the function value b (notation: $f(a) = b$).

If there is no $b \in B$ with $f(a) = b$, then $f(a)$ is not defined (notation: $f(a) = \text{n.d.}$).

Example: The functions g and h are different, although $G_g = G_h$.

- $g : \mathbb{N} \to \mathbb{N}$ with $g(x) = 0$ for all $x \in \mathbb{N}$
- $h : \mathbb{Z} \to \mathbb{Z}$ with $h(x) = 0$ if $x \geq 0$ and $h(x) = \text{n.d.}$ otherwise

Definition 1.1

Let $f\colon A \to B$ and $g\colon B \to C$ be functions.

- $\boldsymbol{g \circ f}$ denotes the function $A \to C$ with $(g \circ f)(x) \overset{df}{=} g(f(x))$ (composition of functions).
- **Domain of definition** of f: $D_f \overset{df}{=} \{a \in A \mid \exists b \in B \; f(a) = b\}$
- **Range** of f: $R_f \overset{df}{=} \{b \in B \mid \text{there is } a \in A \text{ with } f(a) = b\}$
- f is **total** $\overset{df}{\iff} D_f = A$
- f is **surjective** $\overset{df}{\iff} R_f = B$
- f is **injective** $\overset{df}{\iff} f(a_1) \neq f(a_2)$ for all distinct $a_1, a_2 \in D_f$
- f is **bijective** $\overset{df}{\iff}$ f is total, surjective, and injective
- If f injective, there exists the **inverse function** $f^{-1}\colon B \to A$ with $\boldsymbol{f^{-1}(b)} \overset{df}{=}$ the $a \in A$ with $f(a) = b$.
  Note: for an injective function f it holds $f^{-1}(f(a)) = a$ for $a \in D_f$ and $f(f^{-1}(b)) = b$ for $b \in R_f$.

2
Computability

Computability – Outline

- Everyone has a certain idea of what computers can do. We want to take a closer look at this question and find out which tasks can in principle be solved with computers. This requires meaningful and precise definitions of the terms *algorithm* and *computability*. The creation of these definitions is considered one of the most important contributions to mathematics in the 20th century and was the result of a development that can be traced back to discussions of philosophical questions in antiquity.
- In this chapter, we will first get to know various computation models: While programs (WHILE), Random access machines (RAM), Turing machines (TM).

Computability – Outline II

- We show the equivalence of these computational models and thus substantiate the thesis that they capture exactly what computers can do.
- We have now defined the terms *algorithm* and *computability* in a meaningful and precise way.
- Advantage: We can now also recognize the limits of computability. We show that many easily formulated tasks cannot be solved by computers.
- For example, the question of whether two given computer programs compute the same function cannot be solved by computers. This shows that the automatic verification of arbitrary computer programs cannot work.

2.1
History of the Concept of Algorithms

Etymology of the Term “Algorithm”

The word *algorithm* is derived from the name of the mathematician *Muhammad Al-Chwarizmi*, who lived in Baghdad around 780–850. He wrote a book, that was important at the time, on methods for dealing with algebraic equations.

In the Middle Ages, the word *algorismus* was used to describe the art of calculating with Arabic numerals.

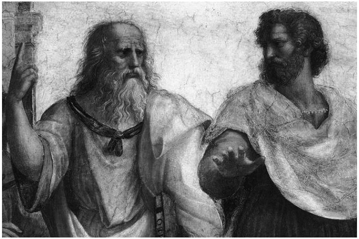
Later, the term *algorithm* was generally used to describe a procedure for solving a problem.

Al-Chwarizmi

Historical Development

The first half of the 20th century saw a rapid development in mathematical logic, which led to the precise definition of the terms *algorithm* and *computability* in the 1930s. The reason for this was not the emergence of computing technology, but the question of the axiomatizability of mathematics:

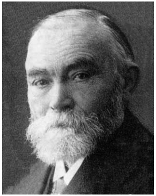
- Even in ancient times, there were discussions about whether there are fundamental limitations to knowledge and predictability. The liar's paradox has been known since the 4th century BC: "This statement is false". The basic ideas behind today's concept of axiomatization can already be found in Plato and Aristotle (400/350 BC).



- Gottfried Leibniz (1646–1716) has a vision of a machine that can manipulate symbols and thus determine the truth value of mathematical statements.

- 1889: Giuseppe Peano (1858–1932) presents the first formal axioms for natural numbers.

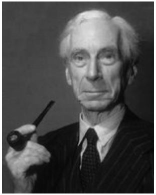
- 1893: Gottlob Frege (1848–1925) writes his “Grundgesetze der Arithmetik” (Fundamentals of arithmetic), which are intended to give all of mathematics a purely logical foundation (logicism). All theorems of arithmetic should be traceable to logical truths.



- 1900: David Hilbert (1862–1943) in his speech at the Mathematics Congress in Paris: Paradoxes (like the liar’s paradox) in mathematics are based on the ambiguity of natural language. They can be avoided by axiomatization.

- 1902: Bertrand Russel (1872–1970) points out the following paradox to Frege in a letter (*Russel's antinomy*). For $R \overset{df}{=} \{x \mid x \notin x\}$ it holds

$$R \in R \Leftrightarrow R \notin R.$$



- Such contradictions lead to the so-called *foundational crisis of mathematics*.
- 1920: Hilbert takes up his idea from 1900 again and develops the so-called Hilbert's program: Call for the complete formalization of mathematics. Theorems should be derived mechanically by applying rules from axioms.

- 1928: Hilbert formulates the decision problem: Is there an algorithm that, given a mathematical statement, determines whether it follows from a given set of axioms?

- 1931: Kurt Gödel (1906–1978) shows the hopelessness of Hilbert's program. *Gödel's incompleteness theorem*: Non-trivial axiomatic systems (e.g. Peano arithmetic) always leave the truth value of certain sentences open.
  "Either mathematics is too big for the human mind or the human mind is more than a machine."

- 1936: Alonzo Church (1903-1995) and Alan Turing (1912-1954) independently show that Hilbert's decision problem cannot be solved by any algorithm. This requires the definition of the terms *algorithm* and *computability*.

  Church uses the lambda calculus, Turing uses the Turing Machine (TM) named after him. It can be seen later that both models have the same computational power.

  Gödel's incompleteness theorem and, above all, the coding of formulas and programs developed by him with natural numbers (Gödelization) had a major influence on the work of Church and Turing.

- Church-Turing thesis (1936): The functions computable in the intuitive sense are exactly the functions computable by TMs. This means that the terms *algorithm* and *computability* can be defined by TMs.
- Further computation models are defined in the following years. It turns out that *all these models are equivalent*. This is how the Church-Turing thesis develops into a generally recognized thesis.
- 1960s: Only now definitions of the term “algorithm” emerge that are characterized by computing technology (e.g. RAMs as a mathematical model for real computers).
- 1970: Extended Church-Turing thesis: Any meaningful computational model can be efficiently simulated (polynomial runtime difference) on a probabilistic TM.

- 1990s: Investigation of new computational models such as quantum computers and DNA computers. It is assumed that certain problems can be solved much faster than in classical computation models.
  However, all problems that can be solved with these computation models can also be solved with classical computation models.

The mathematical characterization of the terms *algorithm* and *computability* is one of the most important scientific achievements of the 20th century.