1 Notations

$\mathbb{N}$ := set of natural numbers.
$\mathbb{R}$ := set of real numbers.
$x \in y := x$ is an element of set $y$.
$\{\text{true, false}\}$ := set of truth values.
$\land :=$ connective “and”.
$\lor :=$ connective “or”.
$\neg :=$ connective “not”.
$\implies :=$ connective “implication”.
$\forall :=$ quantifier “for all” or “every”.
$\exists :=$ quantifier “there exists” or “some”.
Propositional variable := variable whose possible values are the truth values.
Propositional formula := either a propositional variable or $A \land B$, $A \lor B$, $A \implies B$, $\neg A$, where $A, B$ are propositional formulas.
Tautology := propositional formulas which is true under every interpretation (for example, $A \lor \neg A$)
Universe := set of elements where we interpret the constants, the function symbols and the relation symbols.
$\forall x :=$ for all elements $x$, where $x$ ranges over the elements of the universe.
$\forall X :=$ for all subsets $X$, where $X$ ranges over the subsets of the universe.
Term := an expression denoting an element of the universe when we interpret the variables
Atomic formula := An $n$-ary relation symbol applied to $n$ terms
First-order formula := either an atomic formula or $\phi \land \psi$, $\phi \lor \psi$, $\phi \rightarrow \psi$, $\neg \phi$, $\forall x \phi$, $\exists x \phi$ where $\phi, \psi$ are first-order formulas. The quantification $\exists x$ and $\forall x$ is restricted to the elements of the universe.

$x, y, \ldots$ := sequences of indeterminate length.

$x^n :=$ sequence $x_1, \ldots, x_n$; and similarly, for $x^k, x^m, y^n, y^k, y^m$.

$z \in x := z$ is an element of the sequence $x$.

$x \cap y = \emptyset :=$ sequences $x$ and $y$ do not have common elements.

$X \subseteq Y := X$ is a finite subset of $Y$.

$1 := \text{true}$
$0 := \text{false}$

2. The language of mathematics

In mathematics we prove sentences (= theorems or propositions or lemmas) expressing properties of mathematical objects. The sentences are usually expressed in the natural language. The mathematical objects can be of different type: integers, reals, matrices, sequences, continuous functions, groups, etc. When we provide a proof of a sentence $\phi$, we express a judgment of truth: the sentence $\phi$ is true because of its proof. In such a case, the sentence $\neg \phi$ is false. No other truth value is allowed in classical mathematics. In the real life the situation is different.

Mathematical logic is a branch of mathematics, where sentences and proofs are formalized in a formal language. In this way sentences, proofs, and theories become mathematical objects as integers or groups, so that we can prove sentences expressing properties of formal sentences, proofs and theories.

2.1 Predicate symbols, terms and atomic formulas

In the language of mathematics atomic sentences are constructed by relating mathematical objects. Mathematical objects are denoted by terms (or expressions), where variables may occur. For example, in algebra the terms may denote the elements of a group (or a ring, vector space, etc.), while in analysis they may denote reals or continuous functions. It is also possible to consider terms for sets of objects (for example, subgroups, vector subspaces, etc.).

- Terms or expressions denote mathematical objects:
– 3 denotes (= is a name for) the natural number three
– III denotes (= is a name for) the natural number three in the roman representation
– 5 + 6 denotes (= is a name for) the natural number eleven
– 0.3147 denotes (= is a name for) the real number \( \frac{3}{10}+\frac{1}{100}+\frac{4}{1000}+\frac{7}{10000} \)
– \{3, 6, 7\} denotes (= is a name for) the set whose elements are 3, 6, 7
– “The author of Romeo and Juliet” is a term denoting (= a name for) Shakespeare
– “The father of Antonino Salibra” is a term denoting (= a name for) Luigi Salibra

• Terms or expressions with variables denote mathematical objects after interpretation of the variables in the universe:
  – \( x \) denotes a generic element of the universe
  – \( 5 + x \), where \( x \) ranges over the set of natural numbers, denotes a generic natural number greater than 5
  – \( 0.3xy7 \) denotes a generic real number of the form \( \frac{3}{10}+x/100+y/1000+7/10000 \)
  – \( \{x, y, z\} \) denotes a generic set of at most three elements
  – “The author of \( x \)” denotes a generic writer

• An atomic formula denotes a truth value:
  – 3 divides 21
  – 121 is a multiple of 11
  – 3 = 5 + 6
  – 11 + 6 is a prime number
  – 5 is odd
  – 3 \( \in \{3, 6, 7\} \)
  – \( \{3\} \subseteq \{3, 6, 7\} \)
  – John loves Mary
  – John is father of Mary
  – \( P(5) \)
  – \( Q(dog, cat, dogcat) \)
The words “divides”, “is a multiple of”, “=”, “⊆”, “loves” and “is a father of” are binary relations (or predicates). They relate pairs of elements of the universe of elements we are speaking or writing about. The words “is odd”, “is a prime number” and “P” are unary predicates (or “properties” of the elements of the universe we are writing about). The binary predicate “∈” relates elements and sets. The ternary predicate “Q” relates triple of elements of the universe.

- An atomic formula with occurrences of variables denotes a truth value after the interpretation of the variables in the universe:
  - \( x \) divides 122
  - \( x + y = y + x \)
  - \( 3 + 5 = 5 + 3 \) (it is a particular instance of the atomic formula \( x + y = y + x \), where \( x = 3 \) and \( y = 5 \))
  - \( x \) is a prime number
  - \( x \) is odd
  - \( 3 \in X \)
  - \( X \subseteq Y \)
  - \( \{3\} \subseteq \{3, 6, 7\} \) (it is a particular instance of the atomic formula \( X \subseteq Y \), where \( X = \{3\} \) and \( Y = \{3, 6, 7\} \))
  - John loves X
  - \( P(x) \)
  - \( Q(dog, x, y) \)
  - \( A \)

The symbol “A” represents a predicate of arity 0, in other words a propositional variable. “A” can be either true or false.

## 3 Propositional Logic: syntax and semantics

More complex formulas can be obtained by the atomic formulas applying the propositional connectives “and”, “or”, “not”, “implies”, “if and only if”. Formally,

\[
\wedge (\text{and}); \ \vee (\text{or}); \ \neg (\text{not}); \ \rightarrow (\text{if-then}); \ \leftrightarrow (\text{iff}),
\]

- 2 divides 122 and \( 3 + 5 = 5 + 3 \) (true)
• John likes Mary or Mary likes John

**Definition 1.** (Definition of the propositional formulas) Let $\mathcal{P} = \{A, B, C, \ldots\}$ be a set of propositional variables. The set $\mathcal{P}^- \mathcal{F}$ of propositional formulas is defined by induction as follows:

1. Every propositional variable is a propositional formula;
2. If $\phi, \psi$ are propositional formulas, then $\phi \land \psi$, $\phi \lor \psi$, $\phi \to \psi$, $\phi \leftrightarrow \psi$ and $\neg \phi$ are propositional formulas;
3. Nothing else is a propositional formula.

**Notation:** To save parenthesis, we define the following priorities among the logical symbols:

$1 : \neg; \quad 2 : \land, \lor; \quad 3 : \to, \leftrightarrow$.

For example,

1. $A \land B \to C \lor \neg D$ stands for $(A \land B) \to (C \lor (\neg D))$.
2. $A \to B \land C \lor \neg D$ is a wrong formula! We need parenthesis.
3. $A \to (B \land C) \lor \neg D$ stands for $A \to ((B \land C) \lor (\neg D))$.

We will write

$$A \to B \to C \to D \quad \text{for} \quad A \to (B \to (C \to D)).$$

The classical semantics of the propositional connectives is given by the truth tables.

1. $A \land B$ is true $\iff$ $A$ is true and $B$ is true;
2. $A \lor B$ is false $\iff$ $A$ is false and $B$ is false;
3. $\neg A$ is true $\iff$ $A$ is false;
4. $A \to B$ is false $\iff$ $A$ is true and $B$ is false.
5. $A \leftrightarrow B$ is true $\iff$ $A, B$ are both true or $A, B$ are both false.
To know the truth value of the propositional formula \((A \land B) \rightarrow C\), we need to know the truth values of the propositional variables \(A, B, C\). For example, if \(A, C\) are true and \(B\) is false, then \((A \land B) \rightarrow C\) is true. The formula \(A \land B\) is false because \(B\) is false, and any implication, whose premises is false, is true.

Then a model for a propositional language is just an interpretation of the propositional variables into the set of truth values. We compute the truth value of more complex propositional formulas by using the above semantics of the connectives.

We give the formal definition of interpretation. We recall that

**Definition 2.** Let \(\rho : V \rightarrow \{\text{true, false}\}\) be a function, called evaluation of the propositional variables. Then we extend \(\rho\) to a map \(|-|_\rho : \mathcal{PF} \rightarrow \{\text{true, false}\}\) of interpretation of the propositional formulas as follows:

1. \(|A|_\rho = \rho(A)\) for every propositional variable \(A\);
2. \(|\phi \land \psi|_\rho = \text{true iff } |\phi|_\rho = \text{true and } |\psi|_\rho = \text{true. Similarly for the other connectives.}\)

**Definition 3.**

1. A propositional formula is a tautology if it is true for every interpretation of its propositional variables.
2. Two propositional formulas \(\phi\) and \(\psi\) are logically equivalent iff \(\phi \leftrightarrow \psi\) is a tautology iff \(|\phi|_\rho = |\psi|_\rho\) for every evaluation \(\rho\).

The following formulas are tautologies:

- \(A \rightarrow A\);
- Tertium non datur: \(A \lor \neg A\);
- Peirce Law: \(((A \rightarrow B) \rightarrow A) \rightarrow A\);
- \(\neg(A \land \neg A)\);
- Axiom K: \(A \rightarrow B \rightarrow A\).
- Axiom S: \((A \rightarrow B \rightarrow C) \rightarrow (A \rightarrow B) \rightarrow A \rightarrow C\).

The following pairs of formulas are logically equivalent:

- De Morgan Law: \(\neg(A \land B)\) and \(\neg A \lor \neg B\);
- De Morgan Law: \(\neg(A \lor B)\) and \(\neg A \land \neg B\);
• Idempotent Law: \( A \land A \) and \( A \);

• Idempotent Law: \( A \lor A \) and \( A \);

• Commutative Law: \( A \land B \) and \( B \land A \);

• Commutative Law: \( A \lor B \) and \( B \lor A \);

• Distributive Law: \( A \land (B \lor C) \) and \( (A \land B) \lor (A \land C) \).

• Distributive Law: \( A \lor (B \land C) \) and \( (A \lor B) \land (A \lor C) \).

**Remark 1.** Another possible interpretation of the propositional formulas is given by using the Boolean algebra of the subsets of a set \( X \). The emptyset corresponds to the truth value “false”, and the truth value “true” corresponds to the full set \( X \). The propositional variables are interpreted by subsets of \( X \), the connective \( \land \) becomes the intersection, the \( \lor \) the union, the \( \neg \) the complement with respect to \( X \). And the implication \( \rightarrow \)?

\[
x \in (A \rightarrow B) \text{ iff } x \in A \rightarrow x \in B \text{ iff either } x \in B \text{ or } x \notin A.
\]

Then \( A \rightarrow B \) is true (i.e., \( A \rightarrow B = X \)) iff \( A \subseteq B \).

### 4 First-order Logic: syntax and semantics

Starting from atomic formulas, more complex formulas can be written by using the propositional connectives, the universal quantifier

\[
\forall \text{ (Every)}
\]

and the existential quantifier

\[
\exists \text{ (Some)}.
\]

Another idiomatic expression for the universal quantifier is “for all” and for the existential quantifier is “there exists”.

**Example 1.** Some examples in the natural language:

• Every man likes some book.

• Some odd natural number divides 122.

• John likes every woman.

• Every triangle admits an acute angle.
The same examples in the formal language of mathematics:

- $\forall x (\text{Man}(x) \rightarrow \exists y (\text{Book}(y) \land x \text{ likes } y))$
- $\exists x (\text{Odd}(x) \land x \text{ divides } 122)$
- $\forall x (\text{woman}(x) \rightarrow \text{John likes } x)$
- $\forall x (\text{triangle}(x) \rightarrow \exists y (\text{acute-angle}(y) \land y \text{ angle-of } x))$.

The strings $\text{Man}(x)$, $\text{Book}(y)$, $x \text{ likes } y$, $\text{Odd}(x)$ etc. represent atomic formulas.

The semantics of propositional connectives is given by the truth tables. The semantics of the quantifiers is more complex. Consider, for example, a unary predicate $P$ and a model $M$ of universe $M = \{a_1, a_2, \ldots, a_n, \ldots\}$, where this predicate should be interpreted (see below for the formal definition of a model).

1. The formula $\forall x P(x)$ is true in the model $M$ if, and only if, the formula (of length infinite if $M$ is infinite) is true:

   $P(a_1) \land P(a_2) \land \cdots \land P(a_n) \land \ldots$

   In other words, “Every element satisfies the property $P$” if and only if ($a_1$ satisfies $P$) and ($a_2$ satisfies $P$) and ($a_3$ satisfies $P$) and ... ($a_n$ satisfies $P$), ... etc.

2. The formulas $\exists x P(x)$ is true in the model $M$ if, and only if, the following formula (of length infinite if $M$ is infinite) is true:

   $P(a_1) \lor P(a_2) \lor \cdots \lor P(a_n) \lor \ldots$

In other words, “some element satisfies the property $P$” if and only if ($a_1$ satisfies $P$) or ($a_2$ satisfies $P$) or ($a_3$ satisfies $P$) or ... ($a_n$ satisfies $P$), ... etc.

Every sentence is either true or false in a model. Assuming that the universe of the model $M$ is recursively enumerable, we have two algorithms. The first one semidecides whether $\forall x P(x)$ is false, while the second one semidecides whether $\exists x P(x)$ is true. In the following two algorithms, if the value of the variable $x$ is $a_n$ then $S(x)$ denotes $a_{n+1}$.

$x := a_1$;
while $P(x)$ is true do $x := S(x)$;
Output($\forall x P(x)$ is false)

$x := a_1$;
while $P(x)$ is false do $x := S(x)$;
Output($\exists x P(x)$ is true)
To check that $\forall x P(x)$ is true in the model $M$ requires an infinite time, because we have to check that each formula in the infinite sequence $P(a_1), P(a_2), \ldots, P(a_n), \ldots$ is true.

To check that $\exists x P(x)$ is false in the model $M$ requires an infinite time, because we have to check that each formula in the infinite sequence $P(a_1), P(a_2), \ldots, P(a_n), \ldots$ is false.

This is the reason why we need proofs. A proof of the formula $\forall x P(x)$ gives us, in a finite time, the information that $\forall x P(x)$ is true. We pay the price to find the right proof in the infinite set of all possible proofs.

**Example 2.** Consider the sentence $\phi(x) \equiv A(x) \rightarrow \neg \exists y B(x, y)$. Let $M$ be a model of universe $\{0, 1, 2, 3\}$ such that $A(0)$, $A(2)$, $B(0, 3)$ and $B(2, 1)$ are true; in all other cases $A(x)$ and $B(x, y)$ are false. We now evaluate the truth value of the sentence $\forall x \phi(x)$:

1. $\forall x \phi(x)$ is true in $M$ iff the formulas $\phi(0), \phi(1), \phi(2)$ and $\phi(3)$ are all true in $M$.

2. We show that $\phi(0) \equiv A(0) \rightarrow \neg \exists y B(0, y)$ is false, because $A(0)$ is true and $\neg \exists y B(0, y)$ is false. Since $B(0, 3)$ is true, then $\exists y B(0, y)$ is true, so that $\neg \exists y B(0, y)$ is false.

3. Since $\phi(0)$ is false in the model $M$, then $\forall x \phi(x)$ is also false in the model $M$.

We can compute the truth value of $\forall x \phi(x)$ by semantical methods (without a proof) because the universe of $M$ is finite.

### 4.1 Terms and formulas

A *language* $\mathcal{L}$ is given by a family of function symbols and a family of relation symbols. Each symbol has an arity, which is a natural number.

Function symbols of arity 0 are called *constants*. Relation symbols of arity 0 are called *propositional variables*.

**Example 3.** The language of arithmetics is given by two function symbols $+, \ast$ of arity 2 and two relation symbols $=, <$ of arity 2. The following are examples of formulas in the language of the arithmetics (see below for the formal definition of a term and of a formula):

(i) $\forall x (x + 0 = x)$;
(ii) \( \forall x \exists y(x < y) \);

(iii) \( \forall x \forall y \forall z(x \cdot (y + z) = (x \cdot y) + (x \cdot z)) \).

Terms (or expressions) and formulas are defined by induction according to the following definition.

**Definition 4.** (Definition of a term (or expression)) Let \( \mathcal{L} \) be a language and \( V = \{x, y, z, \ldots\} \) be an infinite set of individual variables.

1. Every individual variable is a term;
2. Every constant of \( \mathcal{L} \) is a term;
3. If \( f \in \mathcal{L} \) is a function symbol of arity \( n \) and \( t_1, \ldots, t_n \) are terms, then \( f(t_1, \ldots, t_n) \) is a term;
4. Nothing else is a term.

\( \text{FV}(t) \) denotes the set of individual variables occurring in a term \( t \). For example, \( \text{FV}(5 + 3) = \emptyset \) while \( \text{FV}(3 + x) = \{x\} \) and \( \text{FV}(3 + x + y) = \{x, y\} \).

**Definition 5.** (Definition of a formula) Let \( \mathcal{L} \) be a language and \( V = \{x, y, z, \ldots\} \) be an infinite set of individual variables.

1. Every propositional variable (i.e., relation symbol of arity 0) is an atomic formula;
2. If \( P \in \mathcal{L} \) is a relation symbol of arity \( n \) and \( t_1, \ldots, t_n \) are terms, then \( P(t_1, \ldots, t_n) \) is an atomic formula;
3. Every atomic formula is a formula;
4. If \( \phi, \psi \) are formulas, then \( \phi \land \psi, \phi \lor \psi, \phi \rightarrow \psi \) and \( \neg \phi \) are formulas;
5. If \( \phi \) is a formula and \( x \in V \) is an individual variable, then \( \forall x \phi \) and \( \exists \phi \) are formulas;
6. Nothing else is a formula.

**Notation:** To save parenthesis, we define the following priorities among the logical symbols:

\[
1 : \neg, \forall, \exists; \quad 2 : \land, \lor; \quad 3 : \rightarrow.
\]
For example, we write
\[ \forall x A(x) \land B \rightarrow C \lor \neg D \quad \text{for} \quad ((\forall x A(x)) \land B) \rightarrow (C \lor (\neg D)). \]

Consider an occurrence of the individual variable \( x \) in a formula \( \phi \). We say that the occurrence of \( x \) is bound if \( x \) is under the scope of a quantifier (either \( \forall x \) or \( \exists x \)).

**Example 4.** We have two occurrences of \( x \) in the formula \( \forall x R(x, y) \lor P(x) \): the first one is bound, and the second one is free. The underlined occurrence of \( x \) in \( \forall x R(x, y) \lor P(x) \) is under the scope of \( \forall x \), while the underlined occurrence of \( x \) in \( \forall x R(x, y) \lor P(x) \) is free. Recall that by our priority conventions, \( \forall x R(x, y) \lor P(x) \) stands for \( (\forall x R(x, y)) \lor P(x) \).

We do not care about the name of bound variables. For example, the syllogism “Every man is mortal” is formalised as: \( \forall x (\text{Man}(x) \rightarrow \text{Mortal}(x)) \), which is logically equivalent to \( \forall y (\text{Man}(y) \rightarrow \text{Mortal}(y)) \), which is logically equivalent to \( \forall z (\text{Man}(z) \rightarrow \text{Mortal}(z)) \), etc.

If \( \phi(x) \) is a formula and \( t \) is a term, then the formula \( \phi(t) \) (or \( \phi[t/x] \)) is the formula obtained by substituting \( t \) for every free occurrence of the variable \( x \) in \( \phi(x) \).

**WARNING:** NO FREE VARIABLE OF \( t \) MUST BECOME BOUND IN \( \phi(t) \) AFTER THE SUBSTITUTION.

**Example 5.** If \( \phi(y) \equiv \forall x R(x, y) \) and \( t \equiv x + y \), then \( \phi(x + y) \equiv \forall x R(x, x + y) \) is a WRONG substitution because the underlined variable \( x \) of \( t \equiv x + y \) become bound. For the right substitution we need to change name to the bound variable \( x \) in \( \forall x R(x, y) \). The result is \( \phi(x + y) \equiv \forall z R(z, x + y) \).

**Definition 6.** The set of all individual variables occurring free in a formula is defined by induction as follows:

- \( \text{FV}(P(t_1, \ldots, t_n)) = \text{FV}(t_1) \cup \cdots \cup \text{FV}(t_n) \);
- \( \text{FV}(\phi \ast \psi) = \text{FV}(\phi) \cup \text{FV}(\psi) \) for \( \ast = \land, \lor, \rightarrow \);
- \( \text{FV}(\neg \phi) = \text{FV}(\phi) \);
- \( \text{FV}(\forall x \phi) = \text{FV}(\exists x \phi) = \text{FV}(\phi) \setminus \{x\} \).

**Definition 7.** A formula \( \phi \) is called a sentence if \( \text{FV}(\phi) = \emptyset \).

For example, the formula \( \forall x \exists y R(x, y) \) is a sentence, but \( \forall x R(x, y) \) is not. If \( \phi \) is a formula, then we write \( \phi \equiv \phi(x_1, \ldots, x_n) \) if \( \text{FV}(\phi) \subseteq \{x_1, \ldots, x_n\} \).
4.2 Models

Terms and formulas are interpreted in a model.

**Definition 8.** (Definition of a model) Let $\mathcal{L}$ be a language. An $\mathcal{L}$-model $\mathcal{M}$ is given by a set $M$ of elements (called the universe of the model) and

1. For every function symbol $f \in \mathcal{L}$ of arity $n$, a function $f^\mathcal{M} : M^n \to M$;
2. For every relation symbol $R \in \mathcal{L}$ of arity $n$, a relation $R^\mathcal{M} \subseteq M^n$.

A unary predicate $P$ is interpreted in the model $\mathcal{M}$ by a subset $P^\mathcal{M} \subseteq M$ of the universe $M$, a binary predicate $R$ as a binary relation $R^\mathcal{M} \subseteq M \times M$, i.e. a set of pairs of elements of $M$, etc.

If $a \in P^\mathcal{M}$ then the sentence $P(a)$ is true in the model; If $a \notin P^\mathcal{M}$ then the sentence $P(a)$ is false in the model. If $(a, b) \in R^\mathcal{M}$ then the sentence $R(a, b)$ is true in the model; If $(a, b) \notin R^\mathcal{M}$ then the sentence $R(a, b)$ is false in the model.

**Example 6.** *(The model $\mathbb{N}_{Ar}$ of arithmetics)* The universe of the model is the set $\mathbb{N}$ of natural numbers. The operations are the usual arithmetical operations of sum $+ : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ and product $* : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$. We have (a) a binary relation $=^\mathbb{N}$ of equality such that $n =^\mathbb{N} k$ iff $n$ and $k$ are the same natural number; (b) a binary relation $<^\mathbb{N}$ such that $n <^\mathbb{N} k$ iff there exists $i \in \mathbb{N} \setminus \{0\}$ such that $n +^\mathbb{N} i = k$.

We can express the truth values of the following sentences:

(i) “3 divides 21” is true in the model because $21 =^\mathbb{N} 3 \times^\mathbb{N} 7$.
(ii) “121 is a multiple of 11” is true because $121 =^\mathbb{N} 11 \times^\mathbb{N} 11$
(iii) “$3 = 5 + 6$” is false because $5 +^\mathbb{N} 6 =^\mathbb{N} 11$ and $3 =^\mathbb{N} 11$ is false
(iv) “$3 < 5 + 6$” is true because $5 + 6 =^\mathbb{N} 11$ and $11 =^\mathbb{N} 3 +^\mathbb{N} 8$

Notice that the predicates “divides”, “is a multiple of” and ‘less or equal”’are not primitive in the model. They can be defined in terms $+, \times$ and equality:

$x$ divides $y$ iff there exists $z$ such that $y = z \times x$

Formally,

$x$ divides $y$ iff $\exists z (y = z \times x)$.

For the other predicates we have:

$x$ is a multiple of $y$ iff there exists $z$ such that $x = y \times z$ iff $\exists z (x = y \times z)$
\[ x < y \text{ iff there exists } z \in \mathbb{N} \setminus \{0\} \text{ such that } y = x + z \text{ iff } \exists z(y = x + z \land \neg(z = 0)). \]

The atomic formula “\( x \) divides 122” is neither true neither false in the model. Before giving a judgment of truth we need to give a value to the variable \( x \). The same for the formulas “\( x \) is a prime number” and “\( x \) is odd”.

**Example 7.** *(The model \( \mathcal{B}_{Ar} \) of truth values for the language of arithmetics)* The universe of the model is the set \{true, false\} of truth values. The operation symbols are interpreted as follows: \( +^B \) = the propositional connective “or”; \( *^B \) = the propositional connective “and”; \( 0^B \) = false; \( 1^B \) = true; \( 0 <^B 1 \). Then the sentence \( 1 + 1 = 1 \) is true in \( \mathcal{B}_{Ar} \) but not in \( \mathbb{N}_{Ar} \).

**Example 8.** The language of set theory is given by a function symbol \( \emptyset \) of arity 0 and a binary relation symbol \( \in \) of arity 2. The following are examples of formulas in the language of set theory:

(i) \( \forall x \neg(x \in \emptyset) \);
(ii) \( \forall x \exists y \forall z(z \in x \rightarrow z \in y) \).

We now give the formal definition of interpretation of terms and formulas in a model.

If \( V \) is the set of individual variables and \( \mathcal{M} \) is a model of universe \( M \), then a function \( \rho : V \rightarrow M \) is called environment of interpretation of the terms and formulas.

**Definition 9.** Let \( \mathcal{L} \) be a language, \( \mathcal{M} \) be a \( \mathcal{L} \)-model and \( \rho : V \rightarrow M \) be an environment.

 Interpretation of the terms:

(i) If \( x \in V \), then \( x^\mathcal{M} = \rho(x) \);
(ii) If \( c \in \mathcal{L} \) is a constant, then \( c^\mathcal{M} \) is the interpretation of the constant \( c \);
(iii) \( f(t_1, \ldots, t_n)^\mathcal{M} = f^\mathcal{M}((t_1)^\mathcal{M}, \ldots, (t_n)^\mathcal{M}) \).

 Interpretation of the formulas:

(i) \( R(t_1, \ldots, t_n)^\mathcal{M} \) is true iff \( ((t_1)^\mathcal{M}, \ldots, (t_n)^\mathcal{M}) \in R^\mathcal{M} \);
(ii) \( \forall x \phi(x)^\mathcal{M} \) is true iff \( \phi(a)^\mathcal{M} \) is true for all elements \( a \in M \);
(iii) \( \exists x \phi(x)^\mathcal{M} \) is true iff \( \phi(a)^\mathcal{M} \) is true for some element \( a \in M \);
(iv) \( (\phi \land \psi)^\mathcal{M} \) is true iff \( \phi^\mathcal{M}, \psi^\mathcal{M} \) are both true; similarly for the other connectives.
The interpretation of a formula in a model depends only on the interpretation of its free variables.

**Proposition 1.** If \( \phi \equiv \phi(x_1, \ldots, x_n) \) and \( \rho, \sigma \) are two environments such that \( \rho(x_i) = \sigma(x_i) \) for every \( 1 \leq i \leq n \), then \( \phi^M_\rho = \phi^M_\sigma \).

If a formula \( \phi \) is a sentence, then its truth value in \( M \) does not depend on the environment \( \rho \). Then we will write \( \phi^M \) for \( \phi^M_\rho \).

**Definition 10.** (i) A model \( M \) satisfies a sentence \( \phi \), and we write \( M \models \phi \), if \( \phi^M = \text{true} \).

(ii) The theory of a model \( M \) is the set

\[
\text{Th}(M) = \{ \phi : \phi \text{ is a sentence and } M \models \phi \}.
\]

Notice that \( \phi \in \text{Th}(M) \iff \neg \phi \notin \text{Th}(M) \); \( \phi \land \psi \in \text{Th}(M) \iff \phi, \psi \in \text{Th}(M) \) and so on. Moreover, \( \forall x \phi(x) \in \text{Th}(M) \) (resp. \( \exists x \phi(x) \in \text{Th}(M) \)) iff \( \phi(x)^M_\rho = \text{true} \) for all (some) environment \( \rho \).

**Example 9.** Consider a language with a unary predicate \( P \), a binary predicate \( R \) and two constants \( a \) and \( b \). We provide a model \( M \) for this language.

- Universe \( M = \mathbb{N} \), the set of natural numbers;
- \( a^M = 4 \) and \( b^M = 3 \);
- \( P^M(x) \) iff “\( x \) is a prime number”, i.e., \( P^M = \{2, 3, 5, 7, 11, 13, 17, 19, 23, \ldots \} \);
- \( R^M(x, y) \) iff “\( x \) divides \( y \)”, i.e., \( R^M = \{(k, n) : (\exists y) n = y \ast k\} \).

The following is an example of environment \( \rho \):

\[
x \mapsto_\rho 5 \ (i.e. \ \rho(x) = 5); \quad y \mapsto_\rho 0; \quad z \mapsto_\rho 3; \quad u \mapsto_\rho 2^8; \ldots
\]

The interpretation of the formula \( \phi(x, u) \equiv R(x, b) \land P(u) \) w.r.t. the environment \( \rho \) means

\[
5 \text{ divides } 3 \text{ and } 2^8 \text{ is prime}.
\]

We have that \( \phi(x, u)^M_\rho = \text{false} \).

The sentence \( R(a, b) \lor P(b) \) means “4 divides 3 or 3 is prime”, is true in the model.

The sentence \( \forall x \exists y R(x, y) \) is true in the model. Why?

**Example 10.** (The model of naive sets) The universe of the model \( M \) is the class \( \text{SET} \) of all sets and elements. We have the binary predicate “=” of equality and the binary predicate “\( \in \)” such that \( x \in y \) iff \( x \) is an element of set \( y \).
• $3 \in \{3, 6, 7\}$ is trivially true

• “$\{3\} \subseteq \{3, 6, 7\}$” is true because the unique element of set $\{3\}$ is also element of $\{3, 6, 7\}$. However, the predicate $\subseteq$ is not primitive. It must be defined:

$$x \subseteq y \text{ iff } \forall z (z \in x \rightarrow z \in y).$$

• $\{3\} \in x$ can be formalized as follows: $\exists z (z \in x \land \forall y (y \in z \rightarrow y = 3)).$

• “There exists the empty set” can be written as follows: $\exists x \forall z (\neg (z \in x))$, that is, there is a set without elements. The symbol $\neg$ is the logical connective of negation.

• “There is an infinite set” can be expressed as follows: $\exists x (\exists z (z \in x) \land \forall y (y \in x \rightarrow \{y\} \in x))$. Why is $x$ infinite? if $z \in x$ then $\{z\}, \{\{z\}\}, \{\{\{z\}\}\}, \ldots$ are elements of $x$.

If $\phi$ is a formula, the closure of $\phi$ is the formula $\forall x_1 \ldots \forall x_n \phi$ such that $\text{FV(} \phi \text{)} = \{x_1, \ldots, x_n\}$.

**Definition 11.** 1. A sentence $\phi$ is a logical truth, and we write $\models \phi$, if, for every model $\mathcal{M}$, we have that $\phi^\mathcal{M} = \text{true}$. A formula $\psi$ is a logical truth if its closure is such.

2. Two sentences $\phi$ and $\psi$ are logically equivalent iff, for every model $\mathcal{M}$, $\phi^\mathcal{M} = \psi^\mathcal{M}$ (that can be written $\models \phi \leftrightarrow \psi$).

The following sentences are logical truths:

• Every (propositional) tautology is a logical truth;

• $\forall x (P(x) \lor \neg P(x))$

The following pairs of sentences are logically equivalent. The symbol $\ast$ represents $\ast \equiv \land, \lor$ and the symbol $Q$ represents $Q \equiv \forall, \exists$.

(P1) $Qx\phi(x) \leftrightarrow Qy\phi(y)$.

WARNING: The formula $\forall x \exists y R(x, y)$ is not logically equivalent to $\forall y \exists y R(y, y)$. First we must rename the bound variable $y$: $\forall x \exists y R(x, y)$ is logically equivalent to $\forall y \exists z R(y, z)$.

(P2) $\neg \forall x \phi \leftrightarrow \exists x \neg \phi$.

(P3) $\neg \exists x \phi \leftrightarrow \forall x \neg \phi$. 

15
(P4) $Qx(\phi * \psi) \leftrightarrow (Qx\phi * Qx\psi)$.

(P5) $(\psi * Qx\phi) \leftrightarrow Qx(\psi * \phi)$ if $x \notin \text{FV}(\psi)$.

(P6) $(\phi \rightarrow Qx\psi) \leftrightarrow Qx(\phi \rightarrow \psi)$ if $x \notin \text{FV}(\phi)$.

(P7) If $x \notin \text{FV}(\psi)$, then $(\forall x \phi \rightarrow \psi) \leftrightarrow \exists x(\psi \rightarrow \psi)$ and $(\exists x \phi \rightarrow \psi) \leftrightarrow \forall x(\psi \rightarrow \psi)$.

**Definition 12.** A formula is in prenex form if all its quantifiers are at the beginning of the formula.

The formula $\forall x \exists y \forall z(P(x) \rightarrow Q(y, z))$ is in prenex form, while the formula $\forall x \forall z(P(x) \rightarrow \exists y Q(y, z))$ is not.

**Proposition 2.** Every formula is logically equivalent to a formula in prenex form.

*Proof.* Use the logical equivalences before the definition of a prenex form. We must carefully change the name of some bound variables to avoid mismatch of variables.

**Example 11.**
5 Axiomatic Systems

Mathematicians prove sentences from a set of axioms which they assume “true” (for example, the axioms of euclidean geometry or to the axioms of set theory). In this section we provide a certain number of axiomatic systems.

Example 12. Consider a language with two binary predicates ≤, = and a constant 0. A partially ordered set with minimum element is any model of the following axioms:

1. \( \forall x (x \leq x) \)
2. \( \forall xyz (x \leq y \land y \leq z \rightarrow x \leq z) \)
3. \( \forall xy (x \leq y \land y \leq x \rightarrow x = y) \)
4. \( \forall x (0 \leq x) \).

The theory of posets with minimum element is the set of all sentences provable from the above axioms.

Example 13. Consider a language with a binary operator +, a constant 0 and the binary predicate of equality. A group is any model of the following axioms:

1. \( \forall xyz (x + (y + z) = (x + y) + z) \)
2. \( \forall x (x + 0 = x = 0 + x) \)
3. \( \forall x \exists y (x + y = y + x = 0) \)

The set of integer numbers with the usual operation of addition constitutes a group. The set of bijective maps from a set \( A \) onto \( A \) constitutes a group with respect to the composition of functions (in other words, the symbol + is interpreted as the composition). In this case the constant 0 is interpreted as the identity map \( Id : A \rightarrow A \). We recall that any bijective map is invertible: \( f \circ f^{-1} = f^{-1} \circ f = Id \).

The theory of groups is the set of all sentences provable from the axioms of a group.
5.1 ZFC: Zermelo-Franklin set theory

The intuitive set theory is contradictory by Russel’s paradox. The property “$x \notin x$” does not define a set. At the beginning of last century Zermelo and Franklin introduced an axiomatic system for set theory. The language of ZFC is constituted by two binary predicate $\in$, $\ldots$. The formule $x \in y$ means “the set $x$ is an element of the set $y$”. All mathematics (numbers, functions, etc.) can be described within ZFC.

We do not know whether ZFC is consistent!

The axioms of ZFC are the following:

1. **Extensionality**: Two sets are equal iff they have the same elements.

   $$\forall xy(\forall z(z \in x \leftrightarrow z \in y) \rightarrow x = y)$$

2. **Empty set**: There exists the empty set.

   $$\exists x \forall y(\neg y \in x)$$

   The set $x$, whose existence is assumed in this axiom, is denoted by the constant $\emptyset$. In other words, we have:

   $$\forall y(\neg y \in \emptyset)$$

3. **Pair**: given two sets $x$ and $y$, we can build the set $\{x, y\}$.

   $$\forall xy\exists z\forall t(t \in z \leftrightarrow t = x \lor t = y)$$

   The set $z$, whose existence is assumed in this axiom, is denoted by $\{x, y\}$. In other words, we have:

   $$\forall xy(t \in \{x, y\} \leftrightarrow t = x \lor t = y)$$

   The symbol $\{\ldots, \ldots\}$ is a symbol function of arity 2.

4. **Union**:

   $$\forall x \exists y \forall z(z \in y \leftrightarrow \exists t(t \in x \land z \in t))$$

   The set $y$, whose existence is assumed in this axiom, is denoted by $\cup x$.

   For example, if $x = \{\emptyset, \{2\}, \{5\}, 7\}$ then $\cup x = \emptyset, 2, \{5\}, 7$. 

18
5. Power set

\[ \forall x \exists y \forall z (z \in y \iff z \subseteq x) \]

where \( z \subseteq x \) is an abbreviation of the formula \( \forall u (u \in z \rightarrow u \in x) \). The set \( y \), whose existence is assumed in this axiom, is denoted by \( \mathcal{P}(x) \).

For example, if \( x = \{3, 5\} \) then \( \mathcal{P}(x) = \{\emptyset, \{3\}, \{5\}, \{3, 5\}\} \).

**Remark 2.** For short, we will use the following further abbreviations:

- \( \{x\} \) stands for \( \{x, x\} \)
- \( x \cup y \) stands for \( \cup\{x, y\} \)

Notice that we can express a formula, where abbreviations occur, directly in the language of ZF. For example, the formula \( \emptyset \in \{x, y\} \) becomes

\[ \exists t z[\forall u (\neg u \in t) \land \forall u (u \in z \leftrightarrow u = x \lor u = y) \land t \in z]. \]

6. Axiom of infinite

There exists an infinite set.

\[ \exists x (\emptyset \in x \land \forall y (y \in x \rightarrow y \cup \{y\} \in x)) \]

In other words, the following infinite chain of sets are elements of \( x \):

\( \emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \{\emptyset, \{\emptyset\}\}, \ldots \)

We codify the natural numbers with these elements. Then

\( 0 = \emptyset, 1 = \{\emptyset\}, 2 = \{\emptyset, \{\emptyset\}\}, 3 = \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}, \ldots \)

7. Schema of separation

If \( \phi(x) \) is a first order formula with one free variable (i.e., a formula in the language of set theory with quantification over sets) and \( a \) is a set, then there exists the set \( b \) of all elements of \( a \) satisfying \( \phi(x) \).

\[ \forall a \exists b \forall y (y \in b \leftrightarrow y \in a \land \phi(y)). \]

Warning: the “collection” \( \{x : \phi(x)\} \) does not define a set. It is in general too big! For example, the universe, defined by the collection \( \{x : x = x\} \), is not a set; similarly \( \{x : x \notin x\} \) is not a set because of Russel’s paradox.

8. Schema of replacement

If the first-order formula \( \phi(x, y) \) with two free variables is a function (i.e., \( \phi(x, y) \land \phi(x, z) \rightarrow y = z \)), then, for every set \( a \) of inputs, there exists the set \( b \) of the corresponding outputs:

\[ \forall a [\forall xyz (\phi(x, y) \land \phi(x, z) \rightarrow y = z) \rightarrow \exists b \forall y (y \in b \leftrightarrow \exists x (x \in a \land \phi(x, y)))]. \]
9. **Foundation axiom**

\[ \forall x (x \neq \emptyset \rightarrow \exists y (y \in x \land y \cap x = \emptyset)) \]

Foundation axiom forbids self-reference. For example, there is no set \( x \) such that \( x = \{x\} \) and there is no infinite decreasing chain of memberships:

\[ \ldots y_n \in y_{n-1} \in \ldots \in y_2 \in y_1 \in y_0 \]

Foundation axiom is equivalent to the following induction principle, where \( \phi(x) \) is any formula in the language of set theory:

\[ \forall y [\forall x (x \in y \rightarrow \phi(x)) \rightarrow \phi(y)] \rightarrow \forall y \phi(y) \]

In other words, the property \( \phi \) holds for all sets if the empty set satisfies \( \phi \) and any set \( y \) satisfies \( \phi \) whenever every element of \( y \) satisfies \( \phi \). This principle implicitly say us that sets are built in a cumulative way.

10. **Choice Axiom**

A set \( X \) is **well-ordered** by a strict total order if every non-empty subset of \( X \) has a least element under the ordering. Zermelo has shown the following theorem:

**Theorem 1.** (Zermelo’s Theorem) For every set \( x \), there exists a well-ordering on \( x \).

Zermelo’s theorem cannot be proven in ZF set theory and it is equivalent to a choice principle independent of ZF set theory. This principle is non-constructive and it is now accepted by the community of mathematicians. The axiomatic set theory with choice axiom is denoted by ZFC:

\[ \forall x (\emptyset \notin x \rightarrow \exists f (f \text{ is a function from } x \text{ to } \bigcup x \land \forall y (y \in x \rightarrow f(y) \in y)), \]

where \( \bigcup x = \{z : (\exists y) y \in x \land z \in y\} \).

5.1.1 **Ordinal numbers**

Ordinal numbers extend the natural numbers to transfinite. We should think to the natural numbers as the sets defined by induction as follows:

\[ 0 = \emptyset; \quad n + 1 = n \cup \{n\}. \]

The following definition generalises the natural numbers.
**Definition 13.** A set \( o \) is an ordinal number if \( o \) is strictly well-ordered with respect to set membership and every element of \( o \) is also a subset of \( o \).

For example, the number \( 4 = \{0, 1, 2, 3\} \) is an ordinal because \( 0 \in 1 \in 2 \in 3 \) is a well-ordering and \( 0 = \emptyset \subseteq 4, 1 = \{0\} \subseteq 4, 2 = \{0, 1\} \subseteq 4 \) and \( 3 = \{0, 1, 2\} \subseteq 4 \).

The set \( \omega = \bigcup_{n \in \mathbb{N}} n \) is an ordinal, but also \( \omega + 1 = \omega \cup \{\omega\} \) is an ordinal.

The successor of an ordinal \( \alpha \) is defined as the ordinal \( \alpha \cup \{\alpha\} \). An ordinal is called a *limit ordinal* if it is not the successor of another ordinal. For example, \( \omega \) is a limit ordinal.

Ordinals are well-ordered by the relation \( \in \). A limit ordinal \( \alpha \) is the union of all ordinals \( < \alpha \), i.e., \( \alpha = \bigcup_{\beta < \alpha} \beta \).

The collection \( \text{Ord} \) of all ordinals is NOT a set, because otherwise \( \text{Ord} \in \text{Ord} \) against foundation axiom.

A *cardinal number* is a limit ordinal number \( \alpha \) such that \( \alpha \) is not in bijective correspondence with any \( \beta < \alpha \).

The first-order formula \( \text{ord}(x) \) can be defined as follows:

\[
\text{ord}(x) \equiv (x, \in) \text{ is a strict well-ordering } \land \forall y (y \in x \rightarrow y \subseteq x).
\]

**Remark 3.** There are three operations we define over the class of ordinals: addition, multiplication and exponentiation. The definitions are given by transfinite induction.

1. \( \alpha + 0 = \alpha \);
2. \( \alpha + (\beta + 1) = (\alpha + \beta) + 1 \);
3. If \( \beta \) is a limit ordinal, \( \alpha + \beta = \bigcup_{\gamma < \beta} (\alpha + \gamma) \).

*Using this definition, \( \omega + 3 \) can be seen to be a successor ordinal (it is the successor of \( \omega + 2 \)), whereas \( 3 + \omega \) is a limit ordinal, namely, the limit of \( 3 + 0 = 3, 3 + 1 = 4, 3 + 2 = 5 \), etc., which is just \( \omega \).*

*Zero is the neutral element \( \alpha + 0 = 0 + \alpha = \alpha \). Addition is associative, and is strictly increasing and continuous in the right argument:*

\[
\alpha < \beta \Rightarrow \gamma + \alpha < \gamma + \beta,
\]

*but the analogous relation does not hold for the left argument; instead we only have:*

\[
\alpha < \beta \Rightarrow \alpha + \gamma \leq \beta + \gamma.
\]
Ordinal addition is left-cancellative: if \( \alpha + \beta = \alpha + \gamma \), then \( \beta = \gamma \). On the other hand, right cancellation does not work: \( 3 + \omega = 0 + \omega = \omega \) but \( 3 \neq 0 \).

The multiplication \( \alpha \cdot \beta \) is defined as the well-ordered set obtained by \( \beta \), replacing every element of \( \beta \) by a copy of \( \alpha \). For example, \( \omega \cdot 2 \) is the set

\[
0 < 1 < 2 < 3 < \cdots < 0' < 2' < 3' < \ldots
\]

which is the ordinal \( \omega + \omega \). However, the ordinal \( 3 \cdot \omega \) is the ordinal

\[
0 < 1 < 2 < 0' < 1' < 2' < 0'' < 1'' < 3'' < \ldots
\]

which is just \( \omega \). The multiplication is associative but not commutative.

The multiplication can be also defined by transfinite induction:

1. \( \alpha \cdot 0 = 0 \);
2. \( \alpha \cdot (\beta + 1) = (\alpha \cdot \beta) + \alpha \);
3. If \( \beta \) is a limit ordinal, \( \alpha \cdot \beta = \bigcup_{\gamma < \beta} (\alpha \cdot \gamma) \).

5.1.2 The models of set theory

The ZF set theory defines a cumulative hierarchy of sets starting from the empty set. Since ZF is defined by first-order axioms we have infinite different models of ZF. In this section we define two models. The first one is the cumulative hierarchy of sets introduced by von Neumann and Zermelo. It is defined by transfinite induction:

- \( V_0 = \emptyset \);
- \( V_{\alpha+1} = \mathcal{P}(V_\alpha) \) (powerset of \( V_\alpha \));
- \( V_\alpha = \bigcup_{\beta < \alpha} V_\beta \) if \( \alpha \) is a limit ordinal;
- \( V = \bigcup_{\alpha \in \text{Ord}} V_\alpha \).

\((V, \in)\) is a model of ZFC, but \( V \) is not a set. The axiom of foundation can be proven equivalent to this statement:

Every set belongs to \( V_\alpha \) for some ordinal \( \alpha \).

Define Gödel’s constructible universe \( L \) as follows:

\[
\text{Def}(X) := \{ u \subseteq X : \exists z_1, \ldots, z_n \in X \forall y (y \in u \rightarrow (X, \in) |= \phi(y, z_1, \ldots, z_n)) \}.
\]

The class \( L \) of constructible sets is defined by transfinite recursion as follows:
\begin{itemize}
\item $L_0 := \emptyset$;
\item $L_{\alpha+1} := \text{Def}(L_\alpha)$;
\item If $\lambda$ is a limit ordinal, then $L_\lambda := \bigcup_{\alpha<\lambda} L_\alpha$;
\item $L := \bigcup_{\alpha \in \text{Ord}} L_\alpha$.
\end{itemize}

Here \text{Ord} denotes the class of all ordinals.

The elements of $L$ are called “constructible” sets. $(L, \in)$ is a model of ZFC, where it holds the continuum hypothesis CH (i.e., there are no cardinal numbers between the cardinal $\omega$ and the cardinal of the power set of $\omega$).

In sixties Cohen, by using a new technique of forcing, was able to construct a model of ZFC where the negation of CH holds. Then CH is independent of ZFC.

\section{5.2 Peano Arithmetics}

Second-order Peano Arithmetic (PA2) is constituted by all sentences in the language of arithmetics provable from the following four axioms (PA1)-(PA2)-(PA3)-(FI):

(PA$_1$) “The function successor is injective and 0 is not the successor of any other number”:
\[ \forall x \neg(0 = x + 1); \quad \forall xy(x + 1 = y + 1 \rightarrow x = y) \]

(PA$_2$) “The function ‘+’ is defined by induction:
\[ \forall x(x + 0 = x); \quad \forall xy(x + (y + 1) = (x + y) + 1) \]

(PA$_3$) “The function ‘∗’ is defined by induction:
\[ \forall x(x \ast 0 = 0); \quad \forall xy(x \ast (y + 1) = (x \ast y) + x) \]

The full induction principle:

(IF) Full Induction: $\forall P(P(0) \land \forall x(P(x) \rightarrow P(x + 1)) \rightarrow \forall x P(x))$.

\textbf{Remark 4.} In full induction principle (FI) we have a quantification over all subsets (predicates) of the universe. What is the difference between a quantification over subsets and over elements of the universe? Consider the following sentences:

\begin{itemize}
\item Every natural number is the sum of two prime numbers:
\[ \forall n \exists x \exists y(n = x + y \land \text{Prime}(x) \land \text{Prime}(y)). \]
\end{itemize}
Every set of natural numbers has minimum element:
\[ \forall X (X \text{ non-empty} \rightarrow X \text{ has-minimum}) \]

- Full Induction (FI)
- Every continuous function is derivable.

The variable \( n \) ranges over “atomic” elements (in this case, natural numbers) which do not have an “inner” structure. Each natural number has a meaning thanks to the relations connecting it to the other natural numbers. These relations are created by the operations of addition and multiplication and the predicate of equality.

Subsets, continuous functions and properties have an inner structure. For example, we can apply the induction principle to an arbitrary predicate expressed in an arbitrary language. There is no linguistic restriction to the ways to describe predicates, continuous functions and subsets.

Peano considers a restriction of the full induction principle to all first-order definable subsets of the universe. This means that (PA\(_4\)) below is an infinite set of axioms, one for each first-order formula \( \phi(x) \) in the language of arithmetic.

\[(\text{PA}_4) \quad \text{First-order Induction: } \phi(0) \land \forall x (\phi(x) \rightarrow \phi(x + 1)) \rightarrow \forall x \phi(x)\]

For example, the first-order induction principle for the formula \( \text{Even}(x) =_{\text{def}} \exists y (x = 2 \times y) \) is expressed by the following formula:

\[\text{Even}(0) \land \forall x (\text{Even}(x) \rightarrow \text{Even}(x + 1)) \rightarrow \forall x \text{Even}(x)\]

that is,

\[\exists y (0 = 2 \times y) \land \forall x (\exists y (x = 2 \times y) \rightarrow \exists y (x + 1 = 2 \times y)) \rightarrow \forall x \exists y (x = 2 \times y).\]

**Definition 14.** Peano arithmetic \( \text{PA} \) is the theory axiomatized by the axioms \( \text{PA}_1-\text{PA}_4 \), while \( \text{PA}_2 \) is the theory axiomatized by the axioms \( \text{PA}_1-\text{PA}_3 \) and the full induction principle (FI).

It is obvious that the axioms \( \text{PA}_1-\text{PA}_4 \) are true in the standard model of natural numbers, because the standard model satisfies the full induction principle.

### 5.3 A comparison of \( \text{PA} \) and \( \text{PA}_2 \)

**Definition 15.** Two models \( \mathcal{M} \) and \( \mathcal{P} \) are isomorphic if there exists a bijective map \( h : \mathcal{M} \rightarrow \mathcal{P} \) satisfying the following conditions, for all \( a_1, \ldots, a_n \in \mathcal{M} \), all function symbols \( f \) and all relation symbols \( R \):
(i) \( h(f^M(a_1, \ldots, a_n)) = f^P(h(a_1), \ldots, h(a_n)) \);

(ii) \((a_1, \ldots, a_n) \in R^M \iff (h(a_1), \ldots, h(a_n)) \in R^P \).

**Proposition 3.** Two isomorphic models \( M \) and \( P \) satisfy the same sentences (that is, \( \text{Th}(M) = \text{Th}(P) \)).

We show that all the models of PA2 are isomorphic to the standard model \( \mathbb{N}_{Ar} \) of the arithmetic.

**Proposition 4.** PA2 is categorical, i.e., all the models are isomorphic to the standard model of the arithmetic \( \mathbb{N}_{Ar} \).

**Proof.** Assume that \( M \) is a model of PA2. We define by induction a map \( f : \mathbb{N} \to M \) having the set \( \mathbb{N} \) of natural numbers as domain:

\[
f(0^\mathbb{N}) = 0^M, \quad f(x +^\mathbb{N} 1^\mathbb{N}) = f(x) +^M 1^M, \quad \text{for every natural number } x \in \mathbb{N}.
\]

\( f \) is well-defined by the induction principle. Then the conclusion follows from the following considerations.

1. Every model of second-order arithmetics satisfies \( \forall x (x \neq 0 \to \exists y (x = y + 1)) \) and the associative property for the addition and multiplication. These properties can be proven by induction. As an exercise we prove the associativity of addition:

\[
\forall xyz ((x + (y + z) = (x + y) + z).
\]

The proof is by induction over \( z \). If \( z = 0 \), then \( x + (y + 0) = x + y = (x + y) + 0 \) by PA2. Assume that \( x + (y + z) = (x + y) + z \). Then we have:

\[
(x + y) + (\color{red}{z + 1}) = ((x + y) + z) + 1 \quad \text{by PA}_2
\]

\[
= (x + (y + z)) + 1 \quad \text{by induction hyp.}
\]

\[
= x + ((y + z) + 1) \quad \text{by PA}_2
\]

\[
= x + (y + (z + 1)) \quad \text{by PA}_2
\]

2. \( f \) is injective, that is, \( \forall xy (f(x) = f(y) \to x = y) \). The proof is by induction over the parameter \( x \in \mathbb{N} \).

- \( x = 0 \): If \( y \neq 0 \) then by the above item (1) \( y = n + 1 \) for some \( n \), so that \( f(y) = f(n + 1) = f(n) +^M 1 \). If \( f(0) = f(y) \) then \( 0^M = f(n) +^M 1 \), but this contradicts PA1.
• Assume by induction hypothesis that \( f(x) = f(y) \) implies \( x = y \). Let \( f(x + 1) = f(x) +^M 1 = f(y) \), so that \( y \neq 0 \). Then \( y = n + 1 \) and \( f(x) +^M 1 = f(n) +^M 1 \). By \( \text{PA}_1 \) we obtain \( f(x) = f(n) \). By the induction hypothesis \( x = n \), then \( x + 1 = n + 1 = y \).

3. \( f \) is onto, that is, \( \forall y \exists x (y = f(x)) \). In this case we do not apply the usual induction principle satisfied by of the natural numbers. We apply the full induction principle (FI), satisfied by the model \( M \), to the predicate \( P(y) \equiv \exists x \in \mathbb{N} (f(x) = y) \).

• Base of induction: \( P(0^M) \) is true because \( f(0) = 0^M \) by definition of \( f \).
• Assume by induction hypothesis that \( P(y) \) is true. Then we prove \( P(y +^M 1) \). Since \( P(y) \) is true, then there exists a natural number \( x \) such that \( f(x) = y \). Then we have:

\[
y +^M 1 = f(x) +^M 1 \quad \text{by induction hyp.}
\]
\[
y +^M 1 = f(x + 1) \quad \text{by def of } f.
\]

The full induction principle for \( M \) allows us to conclude that \( \forall y P(y) \), i.e., the map \( f \) is onto.

4. \( f \) is a homomorphism with respect to the sum: \( f(x + y) = f(x) + f(y) \). We have:

\[
f(x + 0) = f(x) = f(x) +^M 0^M = f(x) +^M f(0).
\]

Assume by induction hypothesis that \( f(x + y) = f(x) +^M f(y) \). Then we have:

\[
f(x + (y + 1)) = f((x + y) + 1) \quad \text{by PA2}
\]
\[
= f(x + y) +^M 1 \quad \text{by def of } f
\]
\[
= (f(x) +^M f(y)) +^M 1 \quad \text{by induction hyp.}
\]
\[
= f(x) +^M (f(y) +^M 1) \quad \text{by associativity of } +^M
\]
\[
= f(x) +^M f(y + 1) \quad \text{by def of } f
\]

5. \( f \) is a homomorphism with respect to the product: \( f(x \ast y) = f(x) \ast f(y) \). As above.

\( \square \)

The first-order theory \( \text{PA} \) is not categorical, that is, it admits non-isomorphic models. Why does the proof of Prop. 4 fail for \( \text{PA} \)? Assume that \( M \) is a model of
PA. As in Prop. 4, we define by induction a map $f : \mathbb{N} \to M$ having the set $\mathbb{N}$ of natural numbers as domain:

$$f(0) = 0^M; \quad f(x + 1) = f(x) +^M 1,$$

for every natural number $x \in \mathbb{N}$.

As in the proof of Prop. 4, we can prove that $f$ is injective, $f$ is a homomorphism with respect to the sum and product, because the full induction principle holds for the standard model of arithmetic. But we cannot prove that $f$ is onto because in this case we should use the first-order induction principle for the property $P = \{y \in M : (\exists x \in \mathbb{N}) f(x) = y\}$ of the model $\mathcal{M}$. But we do not know whether the above property is definable with a first-order formula in the language of arithmetics, where quantification is over individuals. It is possible to prove that there exist models of PA, the so called non-standard models of arithmetic, for which $P$ is not first-order definable.

**Proposition 5.** PA is not categorical, i.e., there are models of PA that are not isomorphic to the standard model of natural numbers.

**Proof.** The existence of non-standard models of arithmetic can be demonstrated by an application of the Compactness Theorem 1.

We define a set of axioms in a language including the language of Peano arithmetic together with a new constant symbol “c”. The axioms PA* consist of the axioms of Peano arithmetic PA together with another infinite set of axioms: $0 < c, 1 < c, 2 < c, 3 < c, 4 < c, \ldots$. In other words, for each standard natural number $n$, the axiom $c > n$ is included. Any finite subset $X$ of PA* contains only a finite number of the new axioms $n < c$. Then there exists a largest natural number $n_0$ such that $n_0 < c$ belongs to $X$. Then the axioms in $X$ are satisfied by a model which is the standard model of arithmetic plus $c$ interpreted as $n_0 + 1$. Thus by the compactness theorem there is a model $\mathcal{M}$ satisfying all the axioms PA*. Since any model of PA* is a model of PA (a model of a set of axioms is obviously also a model of any subset of that set of axioms), we have that our extended model is also a model of the Peano axioms. The element of this model $\mathcal{M}$ corresponding to “c” cannot be a standard number as it is greater than any standard natural number.

6 Compactness and Lowenheim-Skolem theorems

If $\Sigma$ is a set of sentences and $\phi$ is a sentence, then we say that $\phi$ is a semantical consequence of $\Sigma$, and we write $\Sigma \models \phi$, if, for every model $\mathcal{M}$,

$$\forall \psi \in \Sigma(\mathcal{M} \models \psi) \Rightarrow \mathcal{M} \models \phi.$$
We also write $\mathcal{M} \models \Sigma$ for $\forall \psi \in \Sigma(\mathcal{M} \models \psi)$.

A set $\Sigma$ of sentences is satisfiable if there exists a model $\mathcal{M}$ such that $\mathcal{M} \models \phi$ for every formula $\phi \in \Sigma$.

A set $\Sigma$ of sentences is finitely satisfiable if, for every finite subset $\Gamma$ of $\Sigma$ there exists a model $\mathcal{M}$ such that $\mathcal{M} \models \phi$ for every formula $\phi \in \Gamma$. Note that the model $\mathcal{M}$ depends on the choice of the finite set $\Gamma$.

**Theorem 2.** (Compactness Theorem) A set $\Sigma$ of first-order sentences (where the quantification is restricted to individuals) admits a model if, and only if every finite subset of $\Sigma$ admits a model.

**Proof.** The proof is divided in lemmas.

**Lemma 1.** Let $\Sigma$ be a finitely satisfiable set of $L$-sentences. Then, for every sentence $\phi$, either $\Sigma \cup \{\phi\}$ or $\Sigma \cup \{\neg \phi\}$ is finitely satisfiable.

**Proof.** Assume that $\Sigma \cup \{\phi\}$ is not finitely satisfiable. Then there exists $\Gamma \subseteq_{\text{fin}} \Sigma$ such that $\Gamma \cup \{\phi\}$ is unsatisfiable. By hypothesis $\Gamma$ has models. Then we have, for every model $\mathcal{M}$:

$$\mathcal{M} \models \Gamma \Rightarrow \mathcal{M} \models \neg \phi.$$  \hspace{1cm} (1)

We now show that $\Sigma \cup \{\neg \phi\}$ is finitely satisfiable. Let $\Psi \subseteq_{\text{fin}} \Sigma$ be an arbitrary finite subset of $\Sigma$. Then by hypothesis $\Gamma \cup \Psi$ is a finite satisfiable subset of $\Sigma$. By (1) a model of $\Gamma \cup \Psi$ is also a model of $\neg \phi$, and we get the conclusion. \hfill \Box

**Lemma 2.** Let $\Sigma$ be a finitely satisfiable set of $L$-sentences, $\phi(x)$ be a formula and $c$ a constant neither occurring in $\Sigma$ nor in $\phi(x)$. Then, $\Sigma \cup \{\exists x \phi(x)\}$ is finitely satisfiable iff $\Sigma \cup \{\phi(c)\}$ is finitely satisfiable.

**Proof.** Given a model satisfying $\exists x \phi(x)$, we interpret $c$ as an element making $\exists x \phi(x)$ true. \hfill \Box

**Lemma 3.** Let $\Sigma$ be a finitely satisfiable set of $L$-sentences. Then there exists a language $L' \supset L$ and a set $\Sigma' \supset \Sigma$ of $L'$-sentences with the following properties:

(i) $\Sigma'$ is finitely satisfiable;

(ii) For every sentence $\phi$ of $L'$, either $\phi \in \Sigma'$ or $\neg \phi \in \Sigma'$, but not both;

(iii) If $\exists x \phi(x) \in \Sigma'$ then there exists a constant $c$ of $L'$ such that $\phi(c) \in \Sigma'$;

(iv) $|L'| + \omega = |L| + \omega$. 

28
Proof. Without loss of generality, we assume the language \( L \) to be countable. Let \( c_0, c_1, \ldots, c_n, \ldots \) be a countable set of new constants. Let \( L' = L \cup \{c_0, c_1, \ldots, c_n, \ldots\} \). Let \( \phi_0, \phi_1, \ldots, \phi_n, \ldots \) be an enumeration of all \( L' \)-sentences. Define a sequence of finitely satisfiable sets of \( L' \)-sentences by putting \( \Sigma_0 = \Sigma \). Assuming that \( \Sigma_n \) is finitely satisfiable, we put:

- If \( \Sigma_n \cup \{\phi_n\} \) is not finitely satisfiable, then by Lemma 1 \( \Sigma_n \cup \{\neg \phi_n\} \) is finitely satisfiable and then we define \( \Sigma_{n+1} = \Sigma_n \cup \{\neg \phi_n\} \).
- If \( \Sigma_n \cup \{\phi_n\} \) is finitely satisfiable, then
  - if \( \phi_n \) is not an existential formula, define \( \Sigma_{n+1} = \Sigma_n \cup \{\phi_n\} \);
  - if \( \phi_n \equiv \exists x \psi(x) \) is an existential formula, then \( \Sigma_{n+1} = \Sigma_n \cup \{\phi_n, \psi(c_j)\} \), where \( c_j \in L' \) is a new constant.

At the end define \( \Sigma' = \bigcup_{n \in \mathbb{N}} \Sigma_n \).

Lemma 4. The set \( \Sigma' \) is satisfiable, so that \( \Sigma \) is also satisfiable.

Proof. Recall that the new language \( L' \) is defined as \( L' = L \cup \{c_0, c_1, \ldots, c_n, \ldots\} \). Define a model \( M \) of \( \Sigma' \) as follows. The universe of \( M \) is the set \( M = \{c_0, c_1, \ldots, c_n, \ldots\} \) of the new constants. The relation symbols are defined as follows:

\[(c_i_1, \ldots, c_i_n) \in R_M \iff R(c_i_1, \ldots, c_i_n) \in \Sigma'.\]

The definition is well done because, for every formula \( \phi \), either \( \phi \in \Sigma' \) or \( \neg \phi \in \Sigma' \). Then we prove by induction over the complexity of the formulas \( \phi \) that \( Th(M) = \Sigma' \). If \( \phi = \phi_1 \land \phi_2 \in \Sigma' \), then \( M \models \phi \) iff \( M \models \phi_1 \) and \( M \models \phi_2 \) iff, by induction, \( \phi_1, \phi_2 \in \Sigma' \). Similarly, for the other formulas.

This concludes the proof of the compactness theorem.

Corollary 1. Let \( \Sigma \) be a set of first-order sentences. If \( \Sigma \models \phi \) then there exists a finite subset \( \Sigma_0 \) of \( \Sigma \) such that \( \Sigma_0 \models \phi \).

Proof. Assume that for every finite subset \( \Gamma \) of \( \Sigma \) we have that \( \Gamma \cup \{\neg \phi\} \) is satisfiable (We are negating the conclusion of the corollary). The by the compactness theorem the set \( \Sigma \cup \{\neg \phi\} \) is satisfiable, contradicting the hypothesis.

Theorem 3. (Upward Lowenheim-Skolem Theorem) Let \( \Sigma \) be a set of sentences. If \( \Sigma \) has either (a finite model of cardinality \( n \) for every \( n > 0 \)) or (an infinite model), then for every cardinal \( \kappa \geq |L| + \omega \), \( \Sigma \) has a model of cardinality \( \kappa \).
Proof. Consider a set $C$ of new constants of cardinality $\kappa$ and the language $\mathcal{L}' = \mathcal{L} \cup C$. Let $\Sigma'$ be the theory axiomatised by $\Sigma$ and the axioms $c \neq d$ for all distinct $c, d \in C$. The theory $\Sigma'$ is finitely satisfiable. Then by compactness $\Sigma'$ has a model which has cardinality $|\mathcal{L}'| + \omega = \kappa$. $\square$
7 Proofs in natural deduction

Proof theory, in contraposition to truth value interpretation of logic, gives a new interpretation to the formulas. To assert the formula \( A \) means “there is a proof of \( A \)”, while to assert the formula \( \neg A \) means “\( A \) is contradictory (= from \( A \) it follows a contradiction)”.

*Classical logic* believes that the Boolean logic of truth values “not true = false” and “not false = true” is the same of the logic of proofs: “not provable = contradictory” and “not contradictory = provable”. Every sentence \( A \) is either “true = provable” or “false = contradictory”. This explains the law of excluded middle: for any sentence \( A \), there exists a proof of either \( A \) or \( \neg A \).

*Intuitionistic logic* refutes the law of excluded middle: may be, there are sentence \( B \) such that \( B \) is neither provable nor contradictory.

We use the new symbol \( \vdash \) with the following interpretation:

\[
A_1, \ldots, A_k \vdash B
\]

iff from the assumption that proofs of \( A_1, \ldots, A_k \) exist we prove \( B \) (by applying the logical rules of deducibility). “\( A_1, \ldots, A_k \vdash B \)” is called a *sequent*.

In the context of proof theory, the word “theorem” is used as a sinonimo of “provable sentence”.

A mathematician proves sentences. A computer scientist programs problems (to find algorithmic solutions to the problems). A sentence with its proof and a problem with its program are the result of a creative act. It is difficult to be a good mathematician as a good programmer.

A sentence may not have a proof, a problem may not have a program (for example, the halting problem). There are different reasons for a sentence without a proof: (i) it is false in the standard (i.e., intended) model (for example, \( 0 = 1 \) in arithmetic); (ii) it is not provable from the axioms; (iii) it is independent from the axioms (for example, the continuum hypothesis in Zermelo-Franklin set theory).

Let \( \phi \) be a sentence, \( T \) be a set of sentences and \( \mathcal{M} \) be a model. We denote by \( \vdash \) the notion of provability (to be defined below). We prove only true sentences, that is, if \( T \vdash \phi \) (\( \phi \) is provable from \( T \)) then \( T \models \phi \) (\( \phi \) is a semantical consequence of \( T \)).

<table>
<thead>
<tr>
<th>Proofs from semantics</th>
<th>Proofs from syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \models \phi ) (logical truth)</td>
<td>( \vdash \phi ) (provable from the empty set of axioms)</td>
</tr>
<tr>
<td>( T \models \phi ) (( \phi ) is a semantical consequence of ( T ))</td>
<td>( T \vdash \phi ) (( \phi ) is provable from ( T ))</td>
</tr>
<tr>
<td>( \mathcal{M} \models \phi ) (i.e., ( \phi \in \text{Th}(\mathcal{M}) ))</td>
<td>( T \vdash \phi ) where ( T ) is a complete theory</td>
</tr>
</tbody>
</table>
7.1 Deduction rules for propositional connectives

We introduce the simple rules of natural deduction of propositional logic. We start with an example of informal proof.

Proofs in intuitionistic logic will be labeled by (IL), the other ones by (CL).

**Example 14. (IL)** This example is from [?]. Consider the following sentence:

\[ (A \rightarrow B) \rightarrow (A \rightarrow (B \rightarrow C)) \rightarrow A \rightarrow C. \]

It is a tautology by the truth values interpretation of the propositional variables. However, in mathematics the above sentence would be proven as follows: Assume \( A \rightarrow B, A \rightarrow (B \rightarrow C) \) and \( A \). We would like to show \( C \). From \( A \) and \( A \rightarrow B \) we deduce \( B \). From \( A \) and \( A \rightarrow (B \rightarrow C) \) we obtain \( B \rightarrow C \). Finally, from \( B \) and \( B \rightarrow C \) we get \( C \).

\((\land_e)\) (read: and elimination) From a proof of \( A \land B \) we can construct a proof of \( A \) and a proof of \( B \). Formally,

\[ A \land B \vdash A; \quad A \land B \vdash B. \]

Another way to write the rule in tree notation:

\[
\begin{array}{c}
A \land B \\
\hline
A \\
\hline
[\land_e]
\end{array}
\quad
\begin{array}{c}
A \land B \\
\hline
B \\
\hline
[\land_e]
\end{array}
\]

\((\land_i)\) (read: and introduction) From a proof of \( A \) and a proof of \( B \) we construct a proof of \( A \land B \). Formally,

\[ A, B \vdash A \land B. \]

Another way to write the rule in tree notation:

\[
\begin{array}{c}
A \\
\hline
B \\
\hline
A \land B \\
[\land_i]
\end{array}
\]

\((\rightarrow_i)\) (read: arrow introduction) \( A \rightarrow B \) is provable if from the assumption \( A \) we can construct a proof of \( B \). Formally,

\[ \text{If } A \vdash B \text{ then } \vdash A \rightarrow B \]

The assumption \( A \) is discharged.
The rule in tree notation:

\[
\begin{array}{c}
[A] \\
\vdots \\
B \\
\hline \\
A \rightarrow B
\end{array}
\]

\[\rightarrow_i\]

The assumption \(A\) is discharged.

\((\rightarrow_e)\) (read: arrow elimination or Modus Ponens (MP))

\[A, A \rightarrow B \vdash B\]

The rule in tree notation:

\[
\begin{array}{c}
A \rightarrow B \\
A \\
\hline \\
B
\end{array}
\]

\[\rightarrow_e\]

\((\lor_i)\) (read: or introduction)

\[A \vdash A \lor B; \quad B \vdash A \lor B\]

The rule in tree notation:

\[
\begin{array}{c}
A \\
\hline \\
A \lor B
\end{array}
\]

\[\lor_i\]

\[\begin{array}{c}
B \\
\hline \\
A \lor B
\end{array}\]

\[\lor_i\]

**Remark 5.** (See [?]) The meaning of the rules \(\lor_i\) is not evident, because nobody seems to have an interest to prove \(A \lor B\), when we already have, for example, a proof of \(A\). However, in some cases we need to prove \(A \lor B\). For example, assume that we have shown the following sentence: \(\forall x(A(x) \lor B(x) \rightarrow C(x))\) concerning natural numbers, and we would like to prove \(\forall x C(x)\). Then it is sufficient to prove \(A(n) \lor B(n)\) for every natural number \(n\), that is, to prove \(A(n)\) or \(B(n)\) for every \(n\); may be, we are able to prove \(A(n)\) for every even \(n\), and \(B(k)\) for every odd \(k\).

**Example 15.** (IL) (See [?, Example 2.1]) We prove \(A \land B, B \land A \rightarrow C \vdash C \lor D\).

We write the proof as a tree (written bottom-up) whose root is labeled by \(C \lor D\) and whose leaves are labeled by the assumptions.
We now analyze the other logical rules.

(\lor e) \textbf{(read: or elimination)} Consider the rule of and-introduction: \( A, B \vdash A \land B \). If we reverse the rule by “applying” the * map introduced at the beginning of this section we get: \( \neg A \lor \neg B \vdash \text{either } \neg A \text{ or } \neg B \). If we put \( A \) for \( \neg A \) and \( B \) for \( \neg B \), we have:

\[
A \lor B \vdash \text{either } A \text{ or } B
\]

with the obvious meaning: from the assumption that a proof of \( A \lor B \) exists it follows that either a proof of \( A \) or a proof of \( B \) exists. However, this is not a deduction rule because we do not conclude with a proof of something! However, if we are able to prove a formula \( C \) by assuming either \( A \) or \( B \) then we get the right deduction rule for or-elimination:

\[
(\lor e) \text{ If } A \vdash C \text{ and } B \vdash C \text{ then } A \lor B \vdash C
\]

The assumptions \( A \) and \( B \) are discharged.

The rule in tree notation:

\[
\begin{array}{c}
[A] \\
\vdots \\
A \lor B \\
C \\
\vdots \\
B \\
\vdots \\
C
\end{array}
\]

\[
C \vdash [\lor e]
\]

The assumptions \( A \) and \( B \) are discharged.

(\bot e) \textbf{(read: false elimination)} We denote the “false” or the “absurd” with \( \bot \).

\[
\bot \vdash A
\]

If have a proof of the absurd then we can prove anything.

The rule in tree notation:
The right proof-theoretical interpretation of the formula \( \neg A \) is the following:

a proof of \( \neg A \) iff from the assumption \( A \) we prove the absurd \( \bot \)

In other words,

\[
\vdash \neg A \text{ iff } \vdash A \rightarrow \bot
\]

Then the rule of \( (\neg e) \):

\[
A, \neg A \vdash \bot
\]

The rule in tree notation:

\[
\infer{\bot}{\neg A \quad A \quad \neg e}
\]

(\( \neg i \)) (read: not introduction)

If \( A \vdash \bot \) then \( \vdash \neg A \)

The assumption \( A \) is discharged.

The rule in tree notation:

\[
\infer{\bot}{A \quad \neg i}
\]

**Example 16.** (IL) We show that the formulas \( \neg A \) and \( A \rightarrow \bot \) are logically equivalent. This result can be obtained by using (MP), \((\rightarrow i)\), \(\neg i\) and \(\neg e\). We do not use (RRA).

\[
\infer{\bot}{[A]^* \quad \neg A \quad \neg e}
\]

\[
\infer{\bot}{[A]^* \quad A \rightarrow \bot \quad \rightarrow e}
\]

By the definition \( \neg A \equiv A \rightarrow \bot \), the rules \((\neg i)\) and \((\neg e)\) are particular cases of \((\rightarrow i)\) and \((\rightarrow e)\).
Example 17. (IL) To prove \((\neg A \land \neg B) \rightarrow \neg (A \lor B)\) we must prove \(\neg (A \lor B)\) from the assumption \(\neg A \land \neg B\).

\[
\begin{array}{c}
[\neg A \land \neg B]^* \quad [\land_e] \\
\neg A \quad [A]^* \\
\perp \\
\hline
\end{array}
\quad
\begin{array}{c}
[\neg A \land \neg B]^* \quad [\land_e] \\
\neg B \quad [B]^* \\
\perp \\
\hline
\end{array}
\]

\[
[\neg (A \lor B)]^* \quad [\neg_i;^*] \\
\perp \\
\hline
(\neg A \land \neg B) \rightarrow \neg (A \lor B) \quad [\neg_i;^*]
\]

Example 18. (IL) A proof of \(A \rightarrow \neg \neg A\):

\[
\begin{array}{c}
[A]^* \quad [\neg A]^* \\
\perp [\neg_e] \\
\hline
\neg A \quad [\neg_i;^*] \\
\perp \\
\hline
A \rightarrow \neg \neg A \quad [\neg_i;^*]
\end{array}
\]

(RRA)] (proof by contradiction) A constructive logic has two main properties:

- the **disjunction property** - \(\Gamma \vdash A \lor B\) iff either \(\Gamma \vdash A\) or \(\Gamma \vdash B\)
- the **witness property** - \(\Gamma \vdash \exists x. A(x)\) iff \(\Gamma \vdash A(t)\) for some witness \(t\)

Intuitionistic logic (IL) is constructive, but classical logic (CL) not (ex: \(A \lor \neg A\) and \(\exists x. (Ax \rightarrow \forall y Ay)\)). The rule RRA makes non constructive classical logic.

If \(\neg A \vdash \perp\) then \(\vdash A\).

The assumption \(\neg A\) is discharged.

The rule in tree notation:

\[
\begin{array}{c}
[\neg A] \\
\vdots \\
\perp \quad [RRA] \\
\hline
A
\end{array}
\]

This rule implies, for example, the tertium non datur \(A \lor \neg A\) of classical logic.
Example 19. \((CL)\) \((Tertium non datur)\) We get a proof of \(A \lor \neg A\) by (RRA) if we prove \(\bot\) from the assumption \(\neg(A \lor \neg A)\).

(1) We prove \(\neg(A \lor \neg A) \vdash \bot\). Assume \(\neg(A \lor \neg A)\). This is the principal assumption that we can use in all steps (1) of the proof.

(1.1) We prove \(A \vdash \bot\). Assume \(A\) (this assumption can be used only in step (1.1) of the proof); then \(A \vdash A \lor \neg A\). Since we can use the principal assumption \(\neg(A \lor \neg A)\), then we have \(A \lor \neg A, \neg(A \lor \neg A) \vdash \bot\). In conclusion, we have a proof of \(\bot\) from \(A\).

(1.2) \(\vdash \neg A\) by (1.1) and the rule \(\neg_i\).

(1.3) \(\neg A \vdash A \lor \neg A\).

(1.4) \(A \lor \neg A, \neg(A \lor \neg A) \vdash \bot\) (we use the proven formula \(A \lor \neg A\) and again the principal assumption)

(1.5) In conclusion, we have a proof \(\bot\) from \(\neg(A \lor \neg A) \vdash \bot\).

(2) \(\vdash A \lor \neg A\) by (1) and the rule \(RRA\).

The proof can be written in tree notation:

```
A \lor \neg A \quad \neg_i
\begin{array}{c}
\hline
A \lor \neg A
\end{array}
\begin{array}{c}
\hline
\neg(A \lor \neg A)\quad \neg_i
\end{array}
\begin{array}{c}
\hline
\bot
\end{array}
\begin{array}{c}
\hline
\neg A\quad \neg_i
\end{array}
\begin{array}{c}
\hline
A \lor \neg A
\end{array}
\begin{array}{c}
\hline
\neg(A \lor \neg A)\quad \neg_i
\end{array}
\begin{array}{c}
\hline
\bot
\end{array}
\begin{array}{c}
\hline
\neg_i
\end{array}
\begin{array}{c}
\hline
A \lor \neg A\quad RRA
\end{array}
```

The assumption \(A\) is discharged (and this is written by including the assumption within square bracket) by \(\neg_i\), while the principal assumption \(\neg(A \lor \neg A)\) is discharged by RRA.
Example 20. (CL) The Peirce law

\[
\begin{align*}
\frac{[A]^*}{\perp \quad \to_i,*} \\
\frac{B}{\quad \to_i,*} \\
\frac{\perp \quad \to_i,*}{A \to B} \\
\frac{\quad [(A \to B) \to A]^*}{A \to B} \\
\frac{\quad [A \to \bot]^*}{\quad \to_e} \\
\frac{\quad \perp \quad [RRA,*]}{\quad A} \\
\frac{\quad (A \to B) \to A \to_i,*}{} \\
\frac{\quad \bot \quad [RRA,*]}{\quad A} \\
\frac{\quad [A \to \bot]^*}{\quad \to_e} \\
\frac{\quad \bot \quad [\neg_i,*]}{\quad \neg \neg A} \\
\frac{\quad \bot \quad [\neg_i,*]}{\quad \neg \neg A} \\
\frac{\quad (A \to B) \to A \to_i,*}{\quad \neg \neg A} \\
\end{align*}
\]

The intuitionistic version of Peirce law:

\[
\begin{align*}
\frac{[A]^*}{\perp \quad \to_i,*} \\
\frac{B}{\quad \to_i,*} \\
\frac{\perp \quad \to_i,*}{A \to B} \\
\frac{\quad [(A \to B) \to A]^*}{A \to B} \\
\frac{\quad [A \to \bot]^*}{\quad \to_e} \\
\frac{\quad \perp \quad [RRA,*]}{\quad A} \\
\frac{\quad (A \to B) \to A \to_i,*}{} \\
\frac{\quad \bot \quad [\neg_i,*]}{\quad \neg \neg A} \\
\frac{\quad \bot \quad [\neg_i,*]}{\quad \neg \neg A} \\
\frac{\quad (A \to B) \to A \to_i,*}{\quad \neg \neg A} \\
\end{align*}
\]
Example 21. We prove the formula \((A \rightarrow B) \vee (B \rightarrow A)\):

\[
\begin{array}{c}
[A]^* [A \rightarrow \bot]^* \\
\hline
\bot [\bot] \\
B [\rightarrow_i,]^* \\
A \rightarrow B [\vee_1]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
(A \rightarrow \bot) \rightarrow \bot
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
(A \rightarrow \bot) \rightarrow \bot
\end{array}
\]

\[
\begin{array}{c}
B \rightarrow A [\rightarrow_i,]^* \\
[\bot]^2 [A]^1 \\
\hline
B \rightarrow A [\vee_1] \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
A \rightarrow \bot [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
A \rightarrow \bot [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
A \rightarrow \bot [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

\[
\begin{array}{c}
\bot [\rightarrow_i,]^* \\
[(A \rightarrow B) \vee (B \rightarrow A) \rightarrow \bot]^2 [\rightarrow_a]
\end{array}
\]

7.1.1 RRA and other equivalent rules

Consider the following rule of double negation elimination:

\[\neg \neg A \quad [\bot_c]\]

Proposition 6. Assuming the other rules of natural deduction, (RRA) and \((\bot_c)\) are equivalent.

Proof. 

\[
\begin{array}{c}
\neg \neg A [\neg A]^* \\
\hline
\bot [\bot_e]
\end{array}
\]

\[
\begin{array}{c}
\bot [\bot_e] \\
\hline
A [RRA;]^*
\end{array}
\]

\[
\begin{array}{c}
\neg \neg A [\neg A]^* \\
\hline
\bot [\bot_e]
\end{array}
\]

\[
\begin{array}{c}
\bot [\bot_e] \\
\hline
A [RRA;]^*
\end{array}
\]

The Peirce law \(((A \rightarrow B) \rightarrow A) \rightarrow A\) is very important for classical logic (see below). If we transform this law in a rule, then we get the so-called Peirce rule:
Proposition 7. Assuming the other rules of natural deduction, (RRA) is equivalent to the \((\bot)+(\text{Peirce})\) rules.

Proof. Assume the rule (RRA).

\((\bot)\):

\[
\begin{array}{c}
\vdots \\
\bot \\
\hline \\
\bot [\neg A]^* \\
\hline \\
\bot [\text{RRA;*}] \\
\end{array}
\]

Recall that we can always add useless hypotheses to a proof.

We now prove the Peirce rule:

\[
\begin{array}{c}
\vdots \\
\bot \\
\hline \\
\bot [\neg A]^* \\
\hline \\
\bot [\text{RRA;*}] \\
\end{array}
\]

Conversely, assume \((\bot)\) and (Peirce). We now show the RRA rule:

\[
\begin{array}{c}
\vdots \\
\bot \\
\hline \\
\bot [\neg A]^* \\
\hline \\
\bot [\text{RRA;*}] \\
\end{array}
\]

\[
\begin{array}{c}
\vdots \\
\bot \\
\hline \\
\bot [\neg A]^* \\
\hline \\
\bot [\text{RRA;*}] \\
\end{array}
\]

40
7.2 Deduction rules for quantifiers

(∃ᵢ) (read: “there exists” introduction) An existential statement ∃ₓP(x) can be proven in two different ways. The first one is a constructive method: we provide an element a satisfying P(a):

\[ P(a) \vdash \exists x P(x) \]

In tree notation:

\[ P(a) \quad \exists x P(x) \quad [\exists_i] \]

The second method is not constructive: we assume that there exists a proof of ¬∃ₓP(x); then, we use a reasoning to get a contradiction (for example, 0 = 1). Then there is no proof of ¬∃ₓP(x). In classical logic it holds the law of excluded middle: for any sentence A, either A is provable, or ¬A is provable. Then in classical logic we can conclude that there exists a proof of ∃ₓP(x), without providing an element which satisfies property P. In intuitionist logic the fact that there is no proof of ¬∃ₓP(x) does not imply that there exists a proof of ∃ₓP(x).

Example 22.  1. The sentence ∃ₓPrime(x) is true because Prime(2) is true.

2. We would like to show by contradiction that there exist two irrational numbers a and b such that aᵇ is rational. Assume, by the way of contradiction, that the sentence is false. It is well known that √2, that is the length of the diagonal of the unitary square, was shown irrational by Pitagora. Then, by using our hypothesis √2√2 and (√2√2)√2 are both irrational. But (√2√2)√2 = √2√2 = √2 = 2. Contradiction. Then the original sentence is true. By the proof we have the further information that at least one of the following pairs a = √2, b = √2 or a = √2√2, b = √2 is a pair of irrationals satisfying aᵇ rational. We do not know whether the first pair is irrational or whether the second pair is irrational.

(∀ᵢ) (read: “for all” introduction) In order to show a universal statement ∀ₓP(x) either we give a schema of proof (starting as follows: “Let x be...”) by using the general properties of the variable x, or we assume that there is no proof of ∀ₓP(x) and we get a contradiction. This second type of reasoning only holds in classical logic.
\[
\frac{P(x)}{\forall x P(x)} \quad [\forall_i; x \text{ not free in the hypotheses of the proof of } P(x)]
\]

It is fundamental to require that \( x \) not free in the hypotheses of the proof of \( P(x) \).
Here is an example of wrong application of the \( \forall_i \)-rule.

\[
\frac{[x \text{ is a prime number}]^*}{\forall x (x \text{ is a prime number})} \quad [\forall_i; \text{(wrong application)}]
\]

\[
\frac{(x \text{ is a prime number}) \rightarrow \forall x (x \text{ is a prime number})}{\forall x [(x \text{ is a prime number}) \rightarrow \forall x (x \text{ is a prime number})]} \quad [\forall_i; \text{(right application)}]
\]

But the above sentence is false. If the sentence holds for all \( x \), then we can choose \( x = 3 \). Then the assumption “3 is a prime number” is true. This implies that the conclusion “\( \forall x (x \text{ is a prime number}) \)” is also true: every natural number is prime!

**Example 23.** The statement \( \forall x (x^2 + 7x + 12 \text{ is even}) \) is true. Let \( x \) be a natural number. Then we have two cases: \( x \) even and \( x \) odd. If \( x \) is even, then \( x^2 + 7x + 12 \) is sum of even numbers, so that it is even. If \( x \) is odd, then \( x^2 \) is odd and \( 7x \) is also odd, so that \( x^2 + 7x \) is even. Finally, \( x^2 + 7x + 12 = (x^2 + 7x) + 12 \) is even because sum of two even numbers.

\[
(\forall_e) \ (\text{read: “for all” elimination})
\]

\[
\frac{\forall x P(x)}{P(a)} \quad [\forall_e]
\]

For example, the following is a proof:

\[
\frac{\forall x (x^2 + 7x + 12 \text{ is even})}{3^2 + (7 \times 3) + 12 \text{ is even}} \quad [\forall_e]
\frac{42 \text{ is even}}{\quad \text{[after evaluation of the expression]}}
\]

**Example 24.** In order to show \( \neg \forall x P(x) \) either we give a counter-example (i.e., an element \( a \) such that \( P(a) \) is false) or we assume that there exists a proof of \( \forall x P(x) \) and we get a contradiction. In this second case, we conclude that there is no proof of \( \forall x P(x) \), which is a proof of \( \neg \forall x P(x) \).
The statement $\forall n(n^2 + 3n + 4$ is a perfect square) is false because we have a counter-example: $n = 1$. Formally,

$$\forall n(n^2 + 3n + 4$ is a perfect square)$] \rightarrow [\forall e]

\frac{1^2 + (3 \times 1) + 4$ is a perfect square}{8 is a perfect square} \quad [\text{evaluation}] \quad \frac{8$ is not a perfect square}{\exists e}$]

\frac{\bot}{\neg \forall n(n^2 + 3n + 4$ is a perfect square)} [\neg_i,*]

$(\exists_e)$ (read: “there exists” elimination)

$$\exists x P(x) \quad C \quad [x \text{ not free in hypotheses of the proof of } \exists x P(x) \text{ and (except } P(x) \text{) of } C]$$

This deduction rule is often used to show $\neg \exists x P(x)$: give a schema of proof (starting as follows: “Let $x$ be such that $P(x)$ is true....” by using the general properties of the variable $x$.

$$\frac{[P(x)]^*}{\exists x P(x)^{**} \quad \bot [\exists_e,*]} \quad \bot [\exists_e,**]$$

Example 25.

$$\frac{[P(x) \land \neg P(x)]^*}{\exists x(P(x) \land \neg P(x))^{**} \quad \bot [\exists_e,*]} \quad \bot [\exists_e,**]$$

because we have:

$$\frac{P(x) \land \neg P(x) \quad P(x) \land \neg P(x)}{P(x) \quad \neg P(x)} [\land_e] \quad [\land_e]$$

43
Lemma 5. We have the following logical equivalences:

- $\neg \forall x \phi(x) \iff \exists x \neg \phi(x)$.
- $\neg \exists x \phi(x) \iff \forall x \neg \phi(x)$.

Proof.

\[
\begin{array}{c}
\exists x \neg \phi(x) \\
\downarrow \\
\neg \forall x \phi(x)
\end{array}
\]

\[
\begin{array}{c}
\forall x \neg \phi(x) \\
\downarrow \\
\neg \exists x \phi(x)
\end{array}
\]

Example 26. *(Proofs by contradiction)* We prove by contradiction that there exist infinite prime numbers. Assume, by contradiction, that there exists only a finite number of prime numbers: $p_0 < p_1 < \cdots < p_n$. Let $n = (p_0 \cdots \cdot p_n) + 1$. When we divide $n$ by $p_i$ we have a remainder of 1, so $n$ is not a multiple of any of these prime numbers $p_i$. Then it must exist a prime greater than $p_n$. Contradiction.

Example 27. *(Proofs by contrapositive)* We prove the following statement: If $x \geq 4$ is a prime number, then $x + 1$ is not a perfect square. It is sufficient to prove that, if $x + 1$ is a perfect square, then $x$ is not a prime. In fact, if $x + 1 = k^2$ then $x = k^2 - 1 = (k + 1)(k - 1)$ is factorable if $k \neq 2$ (i.e., $x \geq 4$).

8 The completeness theorem

If $\Sigma$ is a set of sentences and $\phi$ is a sentence, then we say that $\phi$ is a provable from $\Sigma$, and we write

\[\Sigma \vdash \phi,\]

if there exists a proof of $\phi$ in natural deduction with assumptions in $\Sigma$. 

44
\( \Sigma \) is a theory if it is closed under derivability, i.e.,
\[
\Sigma \vdash \phi \implies \phi \in \Sigma.
\]

We write \( \Sigma \vdash \Gamma \) for \( \forall \psi \in \Gamma (\Sigma \vdash \psi) \).

Recall that a set \( \Gamma \) of sentence is consistent if \( \Gamma \not\vdash \bot \).

**Theorem 4.** (Gödel’s Completeness Theorem) Let \( \Sigma \) be a set of sentences.

(i) \( \Sigma \) is consistent iff \( \Sigma \) has a model.

(ii) \( \Sigma \vdash \phi \) iff \( \Sigma \models \phi \).

*Proof.* (i) (\( \iff \)) Since the deduction rules preserve the truth, if \( \Sigma \) has a model, then \( \Sigma \) is consistent, otherwise \( \bot \) would have a model!

(i) (\( \implies \)) Assume \( \Sigma \) to be consistent. The proof is divided in lemmas.

**Lemma 6.** Let \( \Gamma \) be a consistent set of \( \mathcal{L} \)-sentences. Then, for every sentence \( \phi \), either \( \Gamma \cup \{ \phi \} \) or \( \Gamma \cup \{ \neg \phi \} \) is consistent.

*Proof.* Assume that \( \Gamma \cup \{ \phi \} \) is inconsistent. Then there exists a proof of \( \bot \) assuming \( \Gamma \cup \{ \phi \} \). An application of \( (\neg_i) \) provides a proof of \( \neg \phi \) from \( \Gamma \), that is \( \Gamma \vdash \neg \phi \). It follows that \( \Gamma \cup \{ \neg \phi \} \) is consistent, otherwise \( \Gamma \) would be inconsistent. \( \square \)

**Lemma 7.** Let \( \Sigma \) be a consistent set of \( \mathcal{L} \)-sentences, \( \phi(x) \) be a formula and \( c \) a constant neither occurring in \( \Sigma \) nor in \( \phi(x) \). Then, \( \Sigma \cup \{ \exists x \phi(x) \} \) is consistent iff \( \Sigma \cup \{ \phi(c) \} \) is consistent.

*Proof.* If \( \Sigma \cup \{ \phi(c) \} \) is consistent then \( \Sigma \cup \{ \exists x \phi(x) \} \) is also consistent. By the rule \( (\exists_i) \) and by the assumption \( \phi(c) \) we can deduce \( \exists x \phi(x) \).

For the converse, assume by contraposition that \( \Sigma \cup \{ \phi(c) \} \) is inconsistent. Then we have a proof of \( \bot \) from \( \phi(c) \) (and \( \Sigma \)). By applying the rule \( (\neg_i) \) we get a proof of \( \neg \phi(c) \) from \( \Sigma \). Since \( c \) is a new constant occurring only in \( \phi(c) \), we can consider it as a variable. We then apply the rule \( (\forall_i) \) to get a proof of \( \forall c \neg \phi(c) \) from \( \Sigma \). Since \( \forall c \neg \phi(c) \) is logically equivalent to \( \neg \exists x \phi(x) \), this contradicts the hypothesis that \( \Sigma \cup \{ \exists x \phi(x) \} \) is consistent. \( \square \)

**Lemma 8.** Let \( \Sigma \) be a consistent set of \( \mathcal{L} \)-sentences. Then there exists a language \( \mathcal{L}' \supset \mathcal{L} \) and a set \( \Sigma' \supset \Sigma \) of \( \mathcal{L}' \)-sentences with the following properties:

(i) \( \Sigma' \) is consistent;

(ii) For every sentence \( \phi \) of \( \mathcal{L}' \), either \( \phi \in \Sigma' \) or \( \neg \phi \in \Sigma' \), but not both;
(iii) If $\exists x \phi(x) \in \Sigma'$ then there exists a constant $c$ of $\mathcal{L}'$ such that $\phi(c) \in \Sigma'$.

Proof. Without loss of generality, we assume the language $\mathcal{L}$ to be countable. Let $c_0, c_1, \ldots, c_n, \ldots$ be a countable set of new constants. Let $\mathcal{L}' = \mathcal{L} \cup \{c_0, c_1, \ldots, c_n, \ldots\}$. Let $\phi_0, \phi_1, \ldots, \phi_n, \ldots$ be an enumeration of all $\mathcal{L}'$-sentences. Define a sequence of consistent sets of $\mathcal{L}'$-sentences by putting $\Sigma_0 = \Sigma$. Assuming that $\Sigma_n$ is consistent, we put:

- If $\Sigma_n \cup \{\phi_n\}$ is not consistent, then by Lemma 1 $\Sigma_n \cup \{\neg \phi_n\}$ is consistent and then we define $\Sigma_{n+1} = \Sigma_n \cup \{\neg \phi_n\}$.
- If $\Sigma_n \cup \{\phi_n\}$ is consistent, then
  - if $\phi_n$ is not an existential formula, define $\Sigma_{n+1} = \Sigma_n \cup \{\phi_n\}$;
  - if $\phi_n \equiv \exists x \psi(x)$ is an existential formula, then $\Sigma_{n+1} = \Sigma_n \cup \{\phi_n, \psi(c_j)\}$, where $c_j \in \mathcal{L}'$ is a new constant.

At the end define $\Sigma' \equiv \bigcup_{n \in \mathbb{N}} \Sigma_n$. If $\Sigma' \vdash \bot$, then there exists a finite subset $\Sigma'_0 \subseteq \Sigma'$ such that $\Sigma'_0 \vdash \bot$. By the inductive definition of $\Sigma'$, there is $n$ such that $\Sigma'_0 \subseteq \Sigma_n$. This contradicts the consistency of $\Sigma_n$. Then $\Sigma'$ is consistent.

Lemma 9. The set $\Sigma'$ is satisfiable.

Proof. Recall that the new language $\mathcal{L}'$ is defined as $\mathcal{L}' = \mathcal{L} \cup \{c_0, c_1, \ldots, c_n, \ldots\}$. Define a model $\mathcal{M}$ of $\Sigma'$ as follows. The universe of $\mathcal{M}$ is the set $\mathcal{M} = \{c_0, c_1, \ldots, c_n, \ldots\}$ of the new constants. The relation symbols are defined as follows:

$$(c_{i_1}, \ldots, c_{i_n}) \in R^\mathcal{M} \text{ iff } R(c_{i_1}, \ldots, c_{i_n}) \in \Sigma'.$$

The definition is well done because, for every formula $\phi$, either $\phi \in \Sigma'$ or $\neg \phi \in \Sigma'$. Then we prove by induction over the complexity of the formulas $\phi$ that $Th(\mathcal{M}) = \Sigma'$. If $\phi = \phi_1 \land \phi_2 \in \Sigma'$, then $\mathcal{M} \models \phi$ iff $\mathcal{M} \models \phi_1$ and $\mathcal{M} \models \phi_2$ iff, by induction, $\phi_1, \phi_2 \in \Sigma'$. Similarly, for the other formulas.

This concludes the proof of Item (i) of the completeness theorem.

(ii) $\Sigma \not\models \phi$ if $\Sigma \cup \{\neg \phi\}$ is consistent iff (by (i)) $\Sigma \cup \{\neg \phi\}$ is satisfiable iff $\Sigma \not\models \phi$. □

9 Gödel's Incompleteness Theorem

In this section Form denotes a set of formulas (non necessarily first-order formulas). If $\phi \in \text{Form}$, then we write $\phi = \phi(x_1, \ldots, x_n)$ if $x_1, \ldots, x_n$ are all variables occurring...
free in $\phi$. A model $\mathcal{M}$ is a set $M$ together with an $n$-ary relation $\phi^\mathcal{M} \subseteq M^n$ for every formula $\phi(x_1, \ldots, x_n)$.

**Definition 16.** Let $\mathcal{M}$ be a model. A relation $R \subseteq M^n$ is definable in $\mathcal{M}$ if there exists a formula $\phi(x_1, \ldots, x_n)$ such that $\mathcal{M} \models \phi(a_1, \ldots, a_n)$ iff $(a_1, \ldots, a_n) \in R$.

A model $\mathcal{M}$ is complemented if an $n$-ary relation $R$ is definable in $\mathcal{M}$ iff $M^n \setminus R$ is definable in $\mathcal{M}$.

**Lemma 10.** (Diagonalization lemma) Let $M$ be a set and $R \subseteq M \times M$ be a binary relation. Then there exist no element $m_0 \in M$ satisfying the following condition:

$$\forall x \in M[(m_0, x) \in R \text{ iff } (x, x) \notin R].$$

Intuitive meaning: each element $b \in M$ codifies (is a name of) the unary relation $\{x \in M : (b, x) \in R\}$. The unary relation $\{x \in M : (m, m) \notin R\}$ has no name.

**Theorem 5.** (Tarski’s theorem on undefinability of truth) Let $\mathcal{M}$ be a model for which there exists a bijective map $\lceil \rceil : \text{Form}_1 \rightarrow M$ (Gödel numbering)

$$\phi(x) \in \text{Form}_1 \mapsto [\phi(x)] \in M$$

from the set of unary formulas onto the universe $M$ of $\mathcal{M}$. Then,

(i) The complement of $\text{Truth} = \{(\lceil \varphi(x) \rceil, m) : \mathcal{M} \models \varphi(m)\}$ is not definable in $\mathcal{M}$.

(ii) If $\mathcal{M}$ is complemented neither $\text{Truth}$ nor the complement of $\text{Truth}$ is definable in $\mathcal{M}$.

**Proof.** Assume, by contradiction, the complement of $\text{Truth}$ to be definable in $\mathcal{M}$ by a formula $false(x, y)$. Then the unary formula $\Delta(x) = false(x, x)$ satisfies for every $m \in M$:

$$([\Delta(x)], m) \in \text{Truth} \text{ iff } (m, m) \notin \text{Truth}.$$ 

contradicting the diagonalization lemma. \[\square\]

**Corollary 2.** If $\mathcal{M}$ is an $\mathcal{L}$-structure of domain $\mathbb{N}$, where all semidecidable sets are definable (for example, the standard model of $PA$), then $\text{Truth}$ is not semidecidable.

**Corollary 3.** The Halting Problem is not decidable.
Proof. Consider the set of formulas $P_n$, where $P$ is a program and $n \geq 1$ is a natural number. We define a model for this logical language as follows:

Universe: the set $P$ of all programs.

Interpretation: $P^P_n = \{(Q_1, \ldots, Q_n) : P \downarrow (Q_1, \ldots, Q_n)\}$.

Then $\text{Truth} = \{(P, Q) : P \downarrow Q\}$ is not decidable and the complement of $\text{Truth}$ is not semidecidable. □

Theorem 6. (Gödel’s first incompleteness theorem) Let $\vdash$ be a notion of provability of formulas. Let $\mathcal{M}$ be a complemented model such that there exists a bijective map $\lceil \cdot \rceil : \text{FORM}_1 \rightarrow M$ (Gödel numbering).

If $\text{Prov} = \{\lceil \psi(x) \rceil, a) : \vdash \psi(a)\}$ is definable in $\mathcal{M}$, then there exists a formula $\varphi(x)$ such that

1. $\mathcal{M} \models \varphi(\lceil \varphi \rceil)$ iff $\not\vdash \varphi(\lceil \varphi \rceil)$ (intuitive meaning: $\varphi(\lceil \varphi \rceil)$ says ‘I am not provable’) so that $\text{Prov} \neq \text{Truth}$.

2. If the system $\vdash$ is consistent (that is, we can prove only true sentences: $\text{Prov} \subseteq \text{Truth}$), then $\mathcal{M} \models \varphi(\lceil \varphi \rceil)$ and $\varphi(\lceil \varphi \rceil)$ is not provable. The formula $\lnot \varphi(\lceil \varphi \rceil)$, which says ‘$\varphi(\lceil \varphi \rceil)$ is provable’, is also not provable.

Proof. Let $\text{Prov}(x, y)$ be a formula such that

$\mathcal{M} \models \text{Prov}(\lceil \psi(x) \rceil, a) \text{ iff } \vdash \psi(a)$.

Define $\varphi \equiv \lnot \text{Prov}(x, x)$. Then we have:

$\mathcal{M} \models \varphi(\lceil \varphi \rceil)$ iff $\not\vdash \varphi(\lceil \varphi \rceil)$.

□

The formula $(0 = 1)$ is false. Then the consistency of provability $\vdash$ can be expressed by the formula

$\text{Cons} \equiv \lnot \text{Prov}(\lceil 0 = x \rceil, 1)$.

Theorem 7. (Gödel’s second incompleteness theorem) Let $\mathcal{M}$ be a complemented model of a logic $\vdash$ such that there exists a Gödel numbering

$\lceil \cdot \rceil : \text{FORM}_1 \rightarrow M$

and $\text{Prov} = \{\lceil \psi(x) \rceil, a) : \vdash \psi(a)\}$ is definable in $\mathcal{M}$.

If the system $\vdash$ can internalise the proof of the first incompleteness theorem

$\vdash \text{Cons} \rightarrow \varphi(\lceil \varphi \rceil)$,

then $\not\vdash \text{Cons}$ (where $\varphi(\lceil \varphi \rceil)$ means ‘I am not provable’).

Proof. If $\vdash \text{Cons}$ and $\vdash \text{Cons} \rightarrow \varphi(\lceil \varphi \rceil)$ then by Modus Ponens $\vdash \varphi(\lceil \varphi \rceil)$. This contradicts First Incompleteness Theorem. □
10 Exercises: formalization

In algebra we may write formulas to express that a binary operation is commutative or that a vector sub-space has dimension 3; in analysis, formulas to express that a sequence is convergent or that a function is continuous; in set theory, formulas to express the inclusion of two sets. We now provide examples of sentences and formulas.

Remark 6. The sentences “Every professor is...”, “Every multiple of 3 is...”, in general, “Every P is...” are translated formally as follows:

$$\forall x (P(x) \rightarrow ....)$$

The sentences “Some professor is ...”, “There is a multiple of 3 such that...”, in general, "Some P is...” are translated as follows:

$$\exists x (P(x) \land ...)$$.

10.1 Formalization of the natural language

The language of human beings contains:

- Constants: Mario, Maria, Giovanni, ...
- Unary functions: “The-father-of”, “The-mother-of”, ...
- Predicates: =, love, eat, like, ...

The following are examples of terms: The-father-of(Maria), The-mother-of(Maria), while the sentence “The father of Maria eats” is formalized as Eat(The-father-of(Maria)).

Example 28. Consider a binary predicate \(A\), a unary predicate \(P\) and a constant \(j\), with the following intuitive meaning:

\(A(x, y) \equiv x \text{ likes } y; \quad P(x) \equiv x \text{ is a professor}; \quad j \equiv \text{John}\)

1. John likes every professor: \(\forall x (P(x) \rightarrow A(j, x)).\)
2. John likes some professor: \(\exists x (P(x) \land A(j, x)).\)
3. **John likes only the professors.** The above sentence is logically equivalent to:

John likes all professors and does not like the other people. Formally,

\[ \forall x(Px \rightarrow A(j, x)) \land \forall x(\neg Px \rightarrow \neg A(j, x)), \]

which is logically equivalent to

\[ (*) \quad \forall x[(Px \rightarrow A(j, x)) \land (\neg Px \rightarrow \neg A(j, x))]. \]

because it holds the following logical equivalence (Q ed R are arbitrary unary predicates):

\[ \forall x(Qx \land Rx) \leftrightarrow (\forall xQx \land \forall xRx). \]

Another way to write the sentence is the following. We have the following logical equivalence:

\[ (A \rightarrow B) \leftrightarrow (\neg B \rightarrow \neg A). \]

Then, (*) becomes

\[ \forall x[(Px \rightarrow A(j, x)) \land (A(j, x) \rightarrow P(x))], \]

that is,

\[ \forall x(Px \leftrightarrow A(j, x)). \]

4. **Only one professor likes Mary:** \( \exists x(Px \land A(x, m) \land \forall z(Pz \land A(z, m) \rightarrow z = x)). \)

We have the following logical equivalences:

- \( \neg \forall xRx \leftrightarrow \exists x\neg Rx; \)
- \( \neg \exists xRx \leftrightarrow \forall x\neg Rx; \)
- \( (A \rightarrow B) \leftrightarrow (\neg A \lor B); \)
- \( (\neg A \lor \neg B) \leftrightarrow \neg(\neg(A \land B)). \)

Then,

\[ \forall z(Pz \land A(z, m) \rightarrow z = x) \leftrightarrow \forall z(\neg(Pz \land A(z, m)) \lor z = x) \]
\[ \leftrightarrow \forall z(\neg(Pz \land A(z, m)) \lor \neg(z = x)) \]
\[ \leftrightarrow \forall z(\neg Pz \land A(z, m) \land \neg(z = x)) \]
\[ \leftrightarrow \neg \exists z(Pz \land A(z, m) \land \neg(z = x)) \]

Then the formula formalizing “Only one professor likes Mary” can be written as follows:

\[ \exists x(Px \land A(x, m) \land \neg \exists z(Pz \land A(z, m) \land \neg(z = x))). \]
5. Two professors like Mary: \( \exists xy (Px \land Py \land x \neq y \land A(x, m) \land A(y, m)) \), where \( x \neq y \) stands for \( \neg(x = y) \).

6. Only two professors like Mary: \( \exists xy (Px \land Py \land x \neq y \land A(x, m) \land A(y, m) \land \neg \exists z (Pz \land z \neq x \land z \neq y \land A(z, m))) \).

**Example 29.** Consider a binary predicate \( B \) and two unary predicates \( C \) and \( S \) with the following intuitive meaning:

\[
B(x, y) \equiv x \text{ likes } y; \quad Px \equiv x \text{ is a university course}; \quad Sx \equiv x \text{ is a student}
\]

1. No student likes every course: \( \neg \exists x (Sx \land \forall y (Cy \rightarrow B(x, y))) \)
2. No course is loved by all students: \( \neg \exists x (Cx \land \forall y (Sy \rightarrow B(y, x))) \)

### 10.2 Formalization of mathematical sentences

#### 10.2.1 The language of arithmetic

- Constants: 0, 1, 2, 3, \ldots, \( n \), \ldots
- Binary function symbols: \(+, \ast\)
- Binary relation symbol: \( = \)

The standard (or intended) model of arithmetics has the set \( \mathbb{N} \) of natural numbers as universe and the arithmetical interpretation of the above symbols.

**Example 30.** Every multiple of 6 is a multiple of 3. Formally:

1. \( x \) multiple-of \( y \) iff \( \exists k (x = y \ast k) \)
2. \( y \) divides \( x \) iff \( \exists k (x = y \ast k) \)
3. \( \forall x (x \text{ multiple-of } 6 \rightarrow x \text{ multiple-of } 3) \).

**Example 31.** The predecessor of a multiple of 4 is a prime number. Formally:

1. \( \text{Prime}(z) \equiv \neg(z = 0) \land \neg(z = 1) \land \forall x (x \text{ divides } z \rightarrow x = 1 \lor x = z) \)
2. \( \forall x (x \text{ multiple-of } 4 \rightarrow \exists z (\text{Prime}(z) \land x - 1 = z)) \).

**Example 32.** The sum \( + \) is commutative. Formally: \( \forall x \forall y (x + y = y + x) \).

**Example 33.** Existential sentences:
• There is a number $n$ such that $n^2 = 543$. Formally: $\exists n(n^2 = 543)$.

• There are natural numbers $a, b, c$ such that $a^2 + b^2 = c^2$. Formally: $\exists a \exists b \exists c (a^2 + b^2 = c^2)$.

We cannot express in the language of arithmetics the induction principle because we have a universal quantification over the subsets of $\mathbb{N}$ (or all unary predicates on $\mathbb{N}$):

$$\forall Y(Y(0) \land \forall x(Y(x) \rightarrow Y(x + 1)) \rightarrow \forall xY(x)).$$

First-order logic only admits quantification over individual variables. See Section 5.2 below.

### 10.2.2 The language of set theory

We admit quantification over sets if sets become the elements of the intended model. For example, in set theory we have the following language.

- Constants: $\emptyset$ (empty set), $\mathbb{N}$ (the set of natural numbers), $\mathbb{R}$ (the set of real numbers), etc.

- Function symbols of arity 2: $\cup$ (union), $\cap$ (intersection), $\setminus$ (complementation)

- Relation symbols of arity 2: $=$ (equality), $\in$ (belonging to), $\subset$ (contained within), $\subseteq$ (contained within or equal to)

The following strings $x$, $\emptyset$, $x \cup (x \cap y)$, $x \cap \emptyset$, $x \setminus y$, ... are terms, while the following is a formula expressing De Morgan’s law:

$$\forall x \forall y [(z \setminus (x \cup y)) = (z \setminus x) \cap (z \setminus y)].$$

**Example 34.** We express the sentence “The set $X$ has infinite elements” with a unique sentence:

$$\exists y (y \in X) \land \forall z (z \in X \rightarrow \exists x (x \in X \land \forall y (y \in x \iff y = z))).$$

In other words, $X$ is nonempty and, whenever $z \in X$, then $\{z\} \in X$. Thus, if $z \in X$, then $\{z\}, \{\{z\}\}, \{\{\{z\}\}\}, \ldots \in X$.

**Example 35.** We introduce a binary predicate $B$ and we formalize the sentence “$B$ is an injective but not surjective function”.

- $B$ is a function: $\forall xyz (B(x, y) \land B(x, z) \rightarrow y = z)$
• $B$ is a total function: $\forall xyz (B(x, y) \land B(x, z) \rightarrow y = z) \land \forall x \exists y B(x, y)$

• $B$ is a total injective function:
  \( B \) is a total function \( \land \forall xyz (B(x, z) \land B(y, z) \rightarrow x = y) \)

• $B$ is an injective but not surjective function:
  \( B \) is a total injective function \( \land \exists y \forall x \neg B(x, y) \).

Example 36. Let $\mathbb{N}$ be the set of natural numbers. We formalize the sentence
Every nonempty subset of natural numbers has minimum element. The above sentence is equivalent to induction principle. Formally:

1. $X$ has-minimum iff $\exists x (x \in X \land \forall z (z \in X \rightarrow x \leq z))$

2. $\forall X (X \subseteq \mathbb{N} \land X$ non-empty $\rightarrow X$ has-minimum).

10.2.3 The language $\mathcal{L}$ of real analysis

• Constants: 0, 1, $e, \pi, \ldots$

• Function symbols of arity 2: + (addition), * (product), − (subtraction), etc.

• Function symbols of arity 1: $| |$ (absolute value), $\sin, \cos, \ln$ (natural logarithm), and other function symbols: $f, g, h$ etc.

• Relation symbols of arity 1: $\mathbb{N}$ (to be a natural number), $\mathbb{Z}$ (to be an integer), etc.

• Relation symbols of arity 2: = (equality), ≤ (less than or equal to), < (less than), > (greater than), ≥ (greater than or equal to)

The standard (or intended) model of the above language has the set $\mathbb{R}$ of real numbers as universe and the usual interpretation of the above symbols in real analysis.

Example 37. The following strings 0, 1, 0 + 1, $x + 0$, $\sin(x + 0), \ldots$ are terms, while the sentence “the function $f$ is continuous” is formalized by

$$\forall x_0 \forall \varepsilon (\varepsilon > 0 \rightarrow \exists \alpha (\alpha > 0 \land \forall x (|x - x_0| < \alpha \rightarrow |f(x) - f(x_0)| < \varepsilon)).$$

Example 38. We recall that if $\langle A, \leq \rangle$ is partially ordered set and $X \subseteq A$, then $X$ has an upper bound if there exists $z \in A$ such that $\forall x (x \in X \rightarrow x \leq z)$. The least upper bound of $X$, when it exists, is the least upper bound w.r.t. $\leq$. 

53
Let $\mathbb{R}$ be the set of real numbers and $\leq$ be the usual partial ordering on $\mathbb{R}$. The axiom of continuity of the real line can be expressed as follows:

*Every nonempty upper bounded set of real numbers has least upper bound.*

It cannot be formalised in the language $\mathcal{L}$, whose function symbols and relation symbols are defined at the beginning of this section. We have a quantification over all subsets of $\mathbb{R}$. Then we need the language of set theory.

1. $X$ non-empty iff $\exists x (x \in X)$;
2. $y$ upper-bound-of $X$ iff $\forall x (x \in X \rightarrow x \leq y)$;
3. $X$ upper-bounded iff $\exists y (y$ upper-bound-of $X)$ iff $\exists y \forall x (x \in X \rightarrow x \leq y)$;
4. $X$ has-lub iff $\exists z (z$ upper-bound-of $X \land \forall y (y$ upper-bound-of $X \rightarrow z \leq y))$
5. $\forall X (X \subseteq \mathbb{R} \land X$ non-empty $\land X$ upper-bounded $\rightarrow X$ has-lub).

**Example 39.** We recall that a Cauchy sequence is a sequence $(f(n) : n \in \mathbb{N})$ of real numbers such that for every $\epsilon > 0$ there exists $k$ such that $|f(i) - f(j)| < \epsilon$ for all $i, j > k$. We formalise the sentence: *Every Cauchy succession is convergent.*

Formally:

1. $\text{Cauchy}(f) \iff \forall \epsilon \exists k (\mathbb{N}(n) \land \mathbb{N}(m) \land n > k \land m > k \rightarrow |f(n) - f(m)| < \epsilon)$
2. $\forall f (\text{Cauchy}(f) \rightarrow \exists x \forall \epsilon \exists k (\mathbb{N}(k) \land \mathbb{N}(n) \land (n > k \rightarrow |x - a_n| < \epsilon)))$.

The last sentence is outside first-order logic, because the quantification is not over the element of the universe.

**10.2.4 The language of plane geometry**

- Constants: $p_1, \ldots, p_n$ (points), $r_1, \ldots, r_k$ (lines)
- Unary predicates: $P$ (to be a point), $L$ (to be a line),
- Binary predicates: $=, \in$ (a point belongs to a line), $I$ (a line intersects another line)
- Ternary predicate: $B$ (a point is between two other points)

We formalize the following sentences:
1. “Two points belong exactly to a line”
   \[ \forall xy(P(x) \land P(y) \land x \neq y \rightarrow \exists !z(L(z) \land x \in z \land y \in z)) \]

2. “Every line contains at least two points”
   \[ \forall x(L(x) \rightarrow \exists yz(P(y) \land P(z) \land y \in x \land z \in x \land y \neq z)) \]

3. • “\(x, y, z\) points and \(x, y, z \notin \text{line } u\)”
   \[ P(x) \land P(y) \land P(z) \land L(u) \land x \notin u \land y \notin u \land z \notin u \]
   • “There exists at least three points which do not belong to a line”
   \[ \exists xyz \forall u(x, y, z \text{ points and } x, y, z \notin \text{ line } u) \]

4. “If a point \(y\) is between a point \(x\) and \(z\), then \(x, y, z\) are three distinct points of a line and \(y\) is also between \(z\) and \(x\)”
   \[ P(x) \land P(y) \land P(z) \land B(y, x, z) \rightarrow B(y, z, x) \land x \neq y \land x \neq z \land y \neq z \land \exists u(L(u) \land x, y, z \in u) \]

5. “For every line and every point not in the line, there exists a line containing the point which is parallel to the given line”:
   \[ \forall x \forall y(L(x) \land P(y)) \land \neg(y \in x) \rightarrow \exists z(L(z) \land \neg I(x, z))) \]

6. “Given a line and a point not in the line, there exists a unique line containing the point which is “perpendicolare” to the given line”
   \[ \forall x \forall y(L(x) \land P(y)) \land \neg(y \in x) \rightarrow \exists !z(L(z) \land y \in z \land z \text{ perpendicolare } x)) \]

7. “Let \(x, y, z\) be three points not in the same line and \(a\) be a line such that \(x, y, z\) are not in \(a\). If the line \(a\) crosses the segment \(xy\) then \(a\) crosses either the segment \(xz\) or the segment \(yz\).”
   • \(T(x, y, z) \equiv \text{def } x, y, z \text{ are three points not in the same line} \)
   \[ P(x) \land P(y) \land P(z) \land \neg(B(x, y, z) \lor B(y, x, z) \lor B(z, x, y)) \]
   • \(x, y, z \notin a \equiv \text{def } x \notin a \land y \notin a \land z \notin a \)
   • \(T(x, y, z) \land x, y, z \notin a \land \exists z(z \in a \land B(z, x, y)) \rightarrow \exists u(u \in a \land (B(u, x, z) \lor B(u, y, z)) \].