Mathematical Logic

Boğaziçi - Math 411

Fall 2022

Ayhan Günaydın

Contents

Introduction

In simplest terms, the aim of this course is to define *truth* and to construct a *machinery of proof*. We compare these: *true vs. provable*. The study of these are respectively referred to as *sematics* and *syntax*.

Actually, we construct two machineries of proof, among many possible ones. They are *propositional logic* and *first order logic*. The former is a toy model to play with and get aquatinted with the rules of the game. Honestly, it is not very useful for mathematical purposes. However, first order logic is very useful and it actually solves problems.

In order to talk about truth, we would have to construct the object whose truth makes sense. In the case of propositional logic, these objects are *propositions*. One may think of these as basic daily life statements like "Mars is the third biggest planet on the solar system" and more complicated statements obtained from these by using connectives such as "and" and "if ... then". So far there is nothing about truth; these are just a bunch of symbols. We have to determine some of the basic statements as "true" and others as "false". For instance, the statement about Mars needs to be determined as false. Then we automatically know the truth or falsity of more complicated propositions. Next we introduce a totally formal proof system that defines which propositions can be proven from given propositions (sometimes called "premises"). The propositions that can be proven from given premises are the *theorems* of those premises. The interesting thing is these two concepts turn out to capture the same thing; namely a proposition is true under the assumption that certain propositions are true if and only if it is a theorem of those premises. This is called the *completeness theorem for propositional logic*. We develop the details of this in the first chapter.

Then we move on to do a similar thing in a much more complicated yet flexible setting, called first order logic. Here formulas take the place of propositions, and the main difference is that we have *quantifiers*. This means that we may start to talk about something being true for all elements (varying in a given set), or the existence of an element (from the given set) satisfying some properties. The concept of truth is much more complicated in this setting: we can talk about truth in structures. After introducing structures and truth in them, we again construct a proof system for formulas, and it turns out that we once again have that truth of a formula is the same thing as its being provable. In this setting, this fact is called *Gödel's Completeness Theorem*.

There are many equivalent ways to define the proof system for the first order logic; as long as we can prove the completeness theorem for a proof system, it has to be equivalent to the one we construct since provability is equivalent to the absolute concept of being true. You may check other systems in other books and lecture notes. Some great examples of such sources are Enderton's *A Mathematical Introduction to Logic* ([**2**]), van den Dries' *Mathematical Logic Lecture Notes* ([**1**]), and Loeser and Hils' *A First Journey Through Logic* ([**3**]).

After proving Gödel's Completeness, we give some applications of it. As a matter of fact, most of those applications are proven using a consequence of the proof of Gödel's Completeness, called the *Compactness Theorem*. These are done in the third chapter and we hope to emphasize the importance of the mathematical applications.

In the final part, we give a brief introduction to *Gödel's Incompleteness Theorem*. This theorem very roughly states that a mathematical theory *containing basic arithmetic* cannot be *axiomatized in a nice way*. The first italic part is easy to understand and to make precise. Most of the last chapter is devoted to defining the latter italic part: What does it mean to axiomatize a mathematical theory in a nice way? We need to introduce some basic computability theory in order to achieve this goal.

CHAPTER 1

Propositional Logic

1. Syntax

In order to have a 'logic', we need an alphabet and propositions written in that alphabet: An alphabet is just a set; or rather a union of some sets since we want to distinguish different types of elements of the alphabet. We shall call elements of a given alphabet *symbols* of that alphabet, and an *expression* of that alphabet is a finite sequence of symbols. The mathematically correct way to define "expression" is as follows: Let $\mathcal{A}$ be an alphabet. Then expressions are elements of $\bigcup_{n\in\mathbb{N}} \mathcal{A}^n$. However, we do not write elements of this union as $(a_1, a_2, \ldots, a_n)$, but instead we write $a_1 a_2 \cdots a_n$. The expression of length 0 is called the *empty expression*.

Propositions are certain expressions in the alphabet. We do not get into the details of these concepts in general and below we define the alphabet and the propositions of propositional logic. In the next chapter, we present the alphabet and propositions of first-order logic(s).

The naming of the objects below differ quite a bit from one author to another. I'll stick with these until we introduce first order logic, where we have similar but different meanings for them.

The *alphabet of propositional logic* contains the following symbols:

(1) *Logical Symbols:* $\neg \quad \to$
(2) *Atomic Proposition Symbols:* $A_1 \quad A_2 \quad \cdots$
(3) *Parentheses:* $(\quad)$

So there are two logical symbols, two parentheses, and countably many atomic proposition symbols.

These symbols do not have a meaning yet, however it is a good idea to think of the logical symbols with their usual meanings; namely *not* and *implies*. The atomic proposition symbols could be thought as daily life sentences. We will call them *atomic symbols* or more simply *atoms*.

Assumption. One subtlety is that we assume that none of the symbols is a finite sequences of others; for example A_{391} cannot be $\neg \to\to A_{34}\neg$.

Next, we'd like to introduce the propositions. First, we shall do that in a sloppy way, but then we do it in the correct way.

DEFINITION 1.1. The set $\mathcal{P}$ of *propositions* of propositional logic is the smallest set X of expressions such that

(1) Each A_i is in X,
(2) If ϕ is in X, then so is $(\neg\phi)$,
(3) If ϕ and ψ are in X, then so is $(\phi \to \psi)$.

The reason that this definition is sloppy is that it is not very clear what 'smallest' means. Also it is a little bit hard to show that an expression is a proposition with this definition. (It is easier to show that it is not.)

We let $\mathcal{P}_A$ denote the set of atoms; as a set it is just $\mathcal{A}$.

EXAMPLE 1.2. Let's see that the expression

$$(\neg(A_2 \to (\neg A_5)))$$

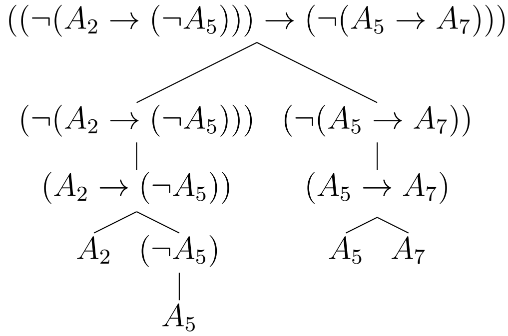
is a proposition.

Since A_5 is a proposition, so is $(\neg A_5)$. Similarly, $(A_2 \to (\neg A_5))$ is a proposition since A_2 is a proposition. Applying the second part to this proposition, we get that $(\neg(A_2 \to (\neg A_5)))$ is also a proposition.

EXAMPLE 1.3. The proposition above is quite simple. How about something slightly complicated as

$$((\neg(A_2 \to (\neg A_5))) \to (\neg(A_5 \to A_7)))?$$

We can go through the same recursive reasoning as above, but it is cumbersome to write it down. So we show this with a tree:



The way we construct this tree is that we separate –if we can– a given expression into smaller expressions using the (2) and (3) in the definition. There may be more than one way to do this! For instance, $((A_1 \to A_4) \to A_6)$ could be separated into $(A_1$ and $A_4) \to A_6$, but also into $(A_1 \to A_4)$ and A_6. We do this on every branch until we cannot do it any more. So we obtain a few different trees, because of the different ways of separating. If there is a tree in which all the leaves of the tree are atoms, then the expression we have started with is a proposition. This sort of gives an algorithm to determine whether an expression is a proposition; such algorithms are called *parsing algorithms*. However, giving a mathematical proof that this algorithm really determines the propositions requires intricate work; see Proposition 1.6 and Example 1.10 below.

Example 1.4. Even though it looks like one, the expression $A_3 \to A_5$ is not a proposition. One may just use the parsing algorithm in the previous example and say that we cannot continue even one step. In this particular case, we may give a rigorous mathematical proof. Consider $X = \mathcal{P} \setminus \{A_3 \to A_5\}$. Clearly, every atom is still in X as $A_3 \to A_5$ is not an atom according to our assumption. Suppose $\phi \in X$. Then $(\neg\phi)$ is not $A_3 \to A_5$, because the former start with (. Therefore $(\neg\phi)$ is in X. Similarly, one may see that X satisfies 3 in the definition of $\mathcal{P}$ as well. Since $X \subseteq \mathcal{P}$, we need to have $X = \mathcal{P}$. Thus $A_3 \to A_5$ is not in $\mathcal{P}$.

Definition 1.5. Let ϕ be an expression. A *construction sequence* for ϕ is a sequence $\phi_1, \ldots, \phi_n$ of expressions where $\phi_n = \phi$ and for each $i \in \{1, \ldots, n\}$, the expression ϕ_i is either an atom or is one of of the forms $(\neg\phi_j)$ and $(\phi_j \to \phi_k)$ with $j, k < i$.

Proposition 1.6. *An expression is a proposition if and only if it has a construction sequence.*

Proof. Let $\mathcal{C}$ be the set of expressions that have a construction sequence. Clearly, $\mathcal{C} \subseteq \mathcal{P}$. It is also clear that $\mathcal{C}$ satisfies (1), (2), (3) in the definition of $\mathcal{P}$. Hence $\mathcal{C} = \mathcal{P}$. □

So we could have defined propositions as expressions that have a construction sequence. Now this is the less sloppy definition.

Next result somehow reduces defining functions on $\mathcal{P}$ to defining functions on $\mathcal{P}_A$.

Proposition 1.7. *Let X be a set and let $H_\neg : X \to X$ and $H_\to : X \times X \to X$ be functions. Then every function $G_A : \mathcal{P}_A \to X$ has a unique extension G to $\mathcal{P}$ such that*

(1) $G((\neg\phi)) = H_\neg(G(\phi))$,
(2) $G((\phi \to \psi)) = H_\to(G(\phi), G(\psi))$.

Proof. HW... □

Example 1.8. Let $X = \mathbb{N}$ and we would like $G(\phi)$ to be 'the number of parentheses in ϕ'. So we define $G_A(A_i) = 0$, $H_\neg(x) = x + 2$, and $H_\to(x, y) = x+y+2$. It is easy to see that the function G extending G_A as in the proposition is the one we're looking for. It is easy to see that the image of G is the set of even natural numbers. This shows that the number of parenthesis in a proposition has to be even. Therefore gives an easy way to determine some expressions as non-propositions.

Example 1.9. Again let $X = \mathbb{N}$. Define $G_A(A_i) = 0$, $H_\neg(x) = x + 1$, and $H_\to(x, y) = \max(x, y) + 1$. We call the function G as in the proposition *the rank function*. It will be useful when we prove or define things by induction.

EXAMPLE 1.10. This example differs from the previous ones as the set X is the set of all finite (rooted) trees. Our aim is to define the tree we constructed in Example 1.3 for every proposition.

So we define $G_A(A_i)$ to be the tree with one node labeled A_i.

Define $H_\neg(T)$ to be the following tree:

$$\begin{array}{c} (\neg T) \\ | \\ T \end{array}$$

So we add a node 'on top of' T and label it as $(\neg T)$.

Similarly we define $H_\to(T_1, T_2)$ to be:

$$\begin{array}{c} (T_1 \to T_2) \\ \diagup \quad \diagdown \\ T_1 \quad T_2 \end{array}$$

Note that we really need the trees to be rooted for this to makes sense.

Setting G to be the function given by Proposition 1.7, we see that, when ordered correctly, the nodes of $G(\phi)$ gives a construction sequence for the proposition ϕ.

EXAMPLE 1.11. This example defines $G(\phi)$ to be the set of 'sub-propositions' of ϕ. So the set X is the power set of $\mathcal{P}$. We do not define G_A, $H_\neg$, and $H_\to$. We define G directly:

- $G(A_i) = \{A_i\}$,
- $G((\neg\phi)) = G(\phi) \cup \{(\neg\phi)\}$,
- $G((\phi \to \psi)) = G(\phi) \cup G(\psi) \cup \{(\phi \to \psi)\}$.

Readability Conventions. As illustrated by Example 1.4, there are some expressions that look like propositions, but are not actually propositions. The reason is the extra parentheses. It would be good to remove irrelevant parentheses for the ease of reading. Here is how we do it.

- The outmost parentheses are omitted. So we write the expression $A_1 \to A_4$ instead of $(A_1 \to A_4)$.
- Parentheses around negation are omitted. So instead of $(\neg\phi)$, we write $\neg\phi$ even when it is not the outmost parentheses. For instance, $A_{42} \to \neg A_3$ means $(A_{42} \to (\neg A_3))$.

Of course, we do not apply the recursive definition of propositions to these simplified versions. In other words, we make the simplifications at the very end.

Another convention is about atomic proposition symbols. We use the letters P, Q, R for atomic proposition symbols. This is to say that when we write P, it means $P = A_i$ for some i. This way, we may add subscripts and primes to P, Q, R.

We also introduce the usual shorthand uses for certain propositions. Let ϕ, ψ be propositions.

- $(\phi \vee \psi)$ means $((\neg\phi) \to \psi)$; so it is $\neg\phi \to \psi$ with the new parentheses conventions.
- $(\phi \wedge \psi)$ means $(\neg(\phi \to (\neg\psi)))$; or $\neg(\phi \to \neg\psi)$ with the parentheses convention.
- $(\phi \leftrightarrow \psi)$ means $(\phi \to \psi) \wedge (\psi \to \phi)$. (Here we used the conventions; it is quite cumbersome to write down the actual formula.)

We apply the parentheses convention to these new definitions as well; so we write $A_1 \vee \neg A_4$ rather than $(A_1 \vee (\neg A_4))$.

Our last convention is that $\vee$ and $\wedge$ are stronger than $\to$ and $\leftrightarrow$. For instance, $\phi \vee \psi \to \theta$ is $(\phi \vee \psi) \to \theta$.

2. Semantics

Now we start to attach meanings to propositions. More rigourously, given truth values of atoms, we determine the truth values of propositions.

A *value assignment* assigns values *false* and *true* to atomic proposition symbols, but we shall use 0 and 1 in the place of false and true respectively. So letting $\mathbb{A} = \{0, 1\}$ a *value assignment* is a function $v_A : \mathcal{P}_A \to \mathbb{A}$. There are $2^{\aleph_0}$ many of such functions.

We'd like to extend any value assignment to $\mathcal{P}$. In order to do this, we define $H_\neg$ and $H_\to$ on $\mathbb{A}$ as follows:

$$H_\neg(0) = 1 \quad H_\neg(1) = 0,$$

$$H_\to(0,0) = H_\to(0,1) = H_\to(1,1) = 1 \quad \text{and} \quad H_\to(1,0) = 0.$$

Now we can extend v_A to $v : \mathcal{P} \to \mathbb{A}$ using Proposition 1.7. We call this function a value assignment on propositions; most of the times, we will simply call it a *valuation*.

So for propositions ϕ, we have $v(\neg\phi) = 1$ if and only if $v(\phi) = 0$.

We sometimes write $[\![\phi]\!]_v$ in the place of $v(\phi)$. We sometimes even drop the letter v and write $[\![\phi]\!]$.

A given proposition ϕ is a finite string of symbols, hence in particular it contains finitely many atoms; say $P_1, \dots, P_n$. Then for a valuation v, the value $v(\phi)$ depends only on the values $v(P_1), \dots, v(P_n)$. Also the values of $v(P_i)$ are independent in the sense that any two of the 2^n possible value assignments to $P_1, \dots, P_n$ give a different valuation.

We illustrate the possible values of $v(\phi)$ depending on $v(P_1), \ldots, v(P_n)$ using *truth tables*. For instance if ϕ is $\neg P$, then we have the following truth table:

P	$\neg P$
0	1
1	0

If ϕ is $P \to Q$, then the truth table is:

P	Q	$P \to Q$
0	0	1
0	1	1
1	0	0
1	1	1

Let's construct the truth table for $P \vee Q$. Recall that $P \vee Q$ is $\neg P \to Q$.

P	Q	$\neg P$	$\neg P \to Q$
0	0	1	0
0	1	1	1
1	0	0	1
1	1	0	1

Let's consider something more complicated: Let ϕ be $P \wedge Q$. Recall that this means ϕ is $\neg(P \to \neg Q)$.

P	Q	$\neg Q$	$P \to \neg Q$	$\neg(P \to \neg Q)$
0	0	1	1	0
0	1	0	1	0
1	0	1	1	0
1	1	0	0	1

Although the values of atoms are independent, the values of propositions might depend on each other. For instance, the value of a proposition certainly depends on the atoms appearing in it.

Definition 2.1. Let Γ be set of propositions and ϕ a proposition. We say that ϕ is a *logical consequence of* Γ if $v(\phi) = 1$ for every valuation v with $v(\psi) = 1$ for every $\psi \in \Gamma$. We denote this by $\Gamma \models \phi$.

So in human words this means *'if all the elements of* Γ *are true, then so is* ϕ*'*.

Example 2.2. If $\phi \in \Gamma$, then $\Gamma \models \phi$. A slightly more complex example is when $\Gamma = \{P, Q\}$ and ϕ is $P \wedge Q$.

If Γ is a finite set, say $\Gamma = \{\psi_1, \ldots, \psi_n\}$, then we write $\psi_1, \ldots, \psi_n \models \phi$ in the place of $\Gamma \models \phi$.

DEFINITION 2.3. If $\emptyset \models \phi$, then we say that ϕ is a *tautology* and we write $\models \phi$.

In other words, a proposition is a tautology if it is true under every valuation. For instance, using truth tables one may easily conclude that $\phi \to \neg\neg\phi$ is a tautology for every ϕ, or $\phi \to (\psi \to \phi)$ is also a tautology for every ϕ, ψ.

PROPOSITION 2.4. *Let ϕ and ψ be propositions. Then the following hold:*

(1) *$\phi \to \psi$ is a tautology if and only if $\phi \models \psi$.*
(2) *$\phi \to \psi, \phi \models \psi$.*

PROOF. Suppose that $\phi \to \psi$ is a tautology and let v be a valuation with $v(\phi) = 1$. Then $v(\phi \to \psi) = 1$, which can only happen when $v(\psi) = 1$. So $\phi \models \psi$. Conversely, let $\phi \models \psi$ and take a valuation v. If $v(\phi) = 0$, then $v(\phi \to \psi) = 1$ regardless of $v(\psi)$. If $v(\phi) = 1$, then $v(\psi) = 1$ by the assumption, and hence $v(\phi \to \psi) = 1$, proving (1).

For (2), let v be a valuation with $v(\phi \to \psi) = v(\phi) = 1$. But then $v(\psi) = 1$.

□

According to this proposition, given formulas ϕ and ψ, the proposition $\phi \leftrightarrow \psi$ is a tautology if and only if $\phi \models \psi$ and $\psi \models \phi$. When that situation occurs we say that ϕ and ψ are *logically equivalent* and we denote this by $\phi \approx \psi$.

PROPOSITION 2.5. *Logical equivalence is indeed an equivalence relation on $\mathcal{P}$.*

PROOF. Only transitivity is not totally trivial. Let $\phi \approx \psi$ and $\psi \approx \theta$. We would like to show that $\phi \approx \theta$. It suffices to show that $\phi \models \theta$, since the roles of ϕ and θ are symmetric. So let v be a valuation with $v(\phi) = 1$. Then $v(\psi) = 1$ and hence $v(\theta) = 1$. □

Next theorem lists some logical equivalences that we shall use frequently, whose proofs are left as exercises.

THEOREM 2.6. *Let ϕ, ψ, σ be propositions. Then we have the following equivalences:*

(1) $(\phi \vee \psi) \vee \sigma \approx \phi \vee (\psi \vee \sigma)$.
(2) $(\phi \wedge \psi) \wedge \sigma \approx \phi \wedge (\psi \wedge \sigma)$.
(3) $\phi \vee \psi \approx \psi \vee \phi$.
(4) $\phi \wedge \psi \approx \psi \wedge \phi$.
(5) $\phi \vee (\psi \wedge \sigma) \approx (\phi \vee \psi) \wedge (\phi \vee \sigma)$.
(6) $\phi \wedge (\psi \vee \sigma) \approx (\phi \wedge \psi) \vee (\phi \wedge \sigma)$.
(7) $\neg(\phi \vee \psi) \approx \neg\phi \wedge \neg\psi$.
(8) $\neg(\phi \wedge \psi) \approx \neg\phi \vee \neg\psi$.

(9) $\phi \vee \phi \approx \phi$.
(10) $\phi \wedge \phi \approx \phi$.
(11) $\neg\neg\phi \approx \phi$.

PROOF. HW... $\square$

Boolean Functions and Propositions. Let ϕ be a proposition such that the atoms appearing in ϕ are among $\{P_1, \ldots, P_n\}$. (So some of these symbols might not appear in ϕ.) Then we can define a map $f_\phi : \mathbb{A}^n \to \mathbb{A}$ as follows: Let $\vec{a} = (a_1, \ldots, a_n) \in \mathbb{A}^n$ and take a valuation v such that $v(P_i) = a_i$. Then we define $f_\phi(\vec{a}) = v(\phi)$. Note that this is well-defined as $v(\phi)$ depends only on $v(P_1), \ldots, v(P_n)$.

This f_ϕ is an example of a Boolean function: Any function $\mathbb{A}^n \to \mathbb{A}$ for some $n \geq 1$ is called a *Boolean function*. So it makes sense to call f_ϕ as the Boolean function attached to ϕ, but we need to fix an ordering of $P_1, \ldots, P_n$ and assume that this list is a minimal list. This is to say that these are exactly the atoms appearing in ϕ; so no irrelevant P_i's appear.

We state some easy facts in the next result.

PROPOSITION 2.7. *Let ϕ, ψ be propositions such that the atoms appearing in ϕ are among $\{P_1, \ldots, P_n\}$. Then we have the following.*

(1) *$\phi \models \psi$ if and only if $f_\phi(\vec{a}) \leq f_\psi(\vec{a})$ for every $\vec{a} \in \mathbb{A}^n$.*
(2) *$\phi \approx \psi$ if and only if $f_\phi = f_\psi$.*
(3) *ϕ is a tautology if and only if f_ϕ is constantly 1.*

PROOF. First note that every valuation v gives $\vec{a} \in \mathbb{A}^n$ such that $f_\phi(\vec{a}) = v(\phi)$ and $f_\psi(\vec{a}) = v(\psi)$. Also $\phi \not\models \psi$ if and only if there is a valuation v such that $v(\phi) = 1$ and $v(\psi) = 0$. So (1) follows.

The second part is a direct consequence of the first part.

Tha last part is clear from the correspondence of valuations and elements of $\mathbb{A}^n$ as mentioned in the first sentence of the proof. $\square$

Next we prove that every Boolean function indeed arises from a proposition.

THEOREM 2.8. *Let $f : \mathbb{A}^n \to \mathbb{A}$ be a Boolean function with $n > 0$. Then there is a proposition ϕ such that $f = f_\phi$.*

PROOF. If $f = 0$, then we may define ϕ to be $A_1 \wedge \neg A_1$. So we may assume that there are $\vec{a}_1, \ldots, \vec{a}_k \in \mathbb{A}^n$ such that $f(\vec{a}_i) = 1$ for every i and $f(\vec{b}) = 0$ for $\vec{b} \in \mathbb{A}^n \setminus \{\vec{a}_1, \ldots, \vec{a}_k\}$.

Write $\vec{a}_i = (a_{i1}, \ldots, a_{in})$ and put $\psi_{ij} = A_j$ if $a_{ij} = 1$ and $\psi_{ij} = \neg A_j$ if $a_{ij} = 0$. Now define $\phi = \phi_1 \vee \cdots \vee \phi_k$, where $\phi_i = \psi_{i1} \wedge \cdots \wedge \psi_{in}$ for $i = 1, \ldots, k$.

It is easy to check that $f = f_\phi$. $\square$

Remark. Note that the proposition ϕ constructed in this proof is of the form

$$\bigvee_{i=1}^{k} \bigwedge_{j=1}^{n} \psi_{ij},$$

where each ψ_{ij} is either an atom or is the negation of an atom. Such propositions are said to be in *disjunctive normal form*; we often shorten this as *DNF*. Combining the theorem with the second part of the previous proposition we get that any proposition is logically equivalent to a formula that is in DNF.

COROLLARY 2.9. *Let $P_1, \dots, P_n$ be atoms. Then there are exactly 2^{2^n} many (logical) equivalence classes of propositions whose atoms are among $P_1, \dots, P_n$.*

PROOF. Clear. □

Homework 1. Write down all the equivalence classes of propositions containing two distinct atoms P, Q.

Homework 2. A proposition is said to be in *conjunctive normal form (CNF)* if it is $\bigwedge_{i=1}^{k} \bigvee_{j=1}^{n} \psi_{ij}$, where each ψ_{ij} is either an atom or is the negation of an atom. Show that any proposition is equivalent to a formula in CNF.

Homework 3. Understand what the following sentence might mean: Every proposition can be written as an expression in the alphabet obtained by replacing $\rightarrow$ by $\vee$ in the alphabet of Propositional Logic.

Homework 4. Show that the sentence in the previous homework is correct. (I guess you need to understand it in a way that it is correct.)

3. Natural Deduction

Now we are going to introduce an axiomatic system for deriving conclusions from premises. We do it in such a general way that it applies to first order logic as well as propositional logic.

DEFINITION 3.1.

(I) A *formal theory*, $\mathfrak{T}$ consists of the following data:
  1. *Alphabet:* A set of symbols.
  2. *Well-formed Formulas:* A set W of expressions in the alphabet; we shortly write *wff* in the place of well-formed formula.
  3. *Axioms:* A set A of wff's; so $A \subseteq W$.

(4) *Inference Rules:* Finitely many relations $R_1, \dots, R_n$ among wff's; this means that each R_j is a subset of W^{m_j+1} for some $m_j > 0$. If $\phi_1, \dots, \phi_{m_j}, \psi$ are wff's such that

$$(\phi_1, \dots, \phi_{m_j}, \psi) \in R_j,$$

then we say that ψ *follows from* $\phi_1, \dots, \phi_{m_j}$ *by* R_j.

(II) Let $\mathfrak{T}$ be a formal theory. A finite sequence $\phi_1 \dots, \phi_n$ of wff's of $\mathfrak{T}$ is called a *proof (in* $\mathfrak{T}$*)*, if for each $i \in \{1, \dots, n\}$ the wff ϕ_i is either an axiom or follows from previous wff's by some inference rule. If $\phi_1 \dots, \phi_n$ is a proof, then ϕ_n is called a *theorem (of* $\mathfrak{T}$*)*.

(III) Let $\mathfrak{T}$ be a formal theory, Γ a set of wff's, and ψ a wff. We say that ψ is a *deductive consequence of* Γ if there is a sequence $\phi_1 \dots, \phi_n$ of wff's such that $\psi = \phi_n$ and each ϕ_i either is an axiom, or is an element of Γ, or follows from previous ϕ_j's by an inference rule. We denote this by $\Gamma \vdash_{\mathfrak{T}} \psi$ and we sometimes say 'ψ *follows from* Γ' in the place of 'ψ is a deductive consequence of Γ'. We also drop $\mathfrak{T}$ from notations if there is not doubt about which formal theory we are working in.

One can work in this generality, but we will do that very briefly and then focus on two kinds of formal theories: Propositional Logic (just one theory) and First Order Theories (many theories).

Note that $\Gamma \vdash_{\mathfrak{T}} \psi$ means that if we add Γ to the set of axioms of $\mathfrak{T}$ to get a new formal theory $\mathfrak{T}'$, then ψ is a theorem of $\mathfrak{T}'$. In particular, $\emptyset \vdash_{\mathfrak{T}} \psi$ just means that ψ is a theorem of $\mathfrak{T}$ and in this case we simply write $\vdash_{\mathfrak{T}} \psi$.

We record the following facts.

PROPOSITION 3.2. *Let $\mathfrak{T}$ be a formal theory, Γ, Δ sets of wff's, and ϕ a wff. Then we have the following.*

(1) *If $\Gamma \subseteq \Delta$ and $\Gamma \vdash_{\mathfrak{T}} \phi$, then $\Delta \vdash_{\mathfrak{T}} \phi$.*
(2) *$\Gamma \vdash_{\mathfrak{T}} \phi$ if and only if there is a finite set $\Gamma_0 \subseteq \Gamma$ such that $\Gamma_0 \vdash_{\mathfrak{T}} \phi$.*
(3) *If $\Delta \vdash_{\mathfrak{T}} \phi$ and for every $\psi \in \Delta$ we have $\Gamma \vdash_{\mathfrak{T}} \psi$, then $\Gamma \vdash_{\mathfrak{T}} \phi$.*

PROOF. Straightforward. □

3.1. Propositional Logic as a Formal Theory. Now we present the long promised formal system for propositional logic. It is a formal theory $\mathfrak{T}_P$ whose alphabet is the same as the alphabet of propositional logic (namely $\{(,), \neg, \to\} \cup \{A_1, A_2, \dots\}$) and its wff's are propositions. We still need to introduce the axioms and the inference rules. Below are the axioms of propositional logic:

(1) For every wff's ϕ, ψ we have the axiom:

$$\phi \to (\psi \to \phi)$$

(2) For every wff's ϕ, ψ, θ we have the axiom:

$$(\phi \to (\psi \to \theta)) \to ((\phi \to \psi) \to (\phi \to \theta))$$

(3) For every wff's ϕ, ψ we have the axiom:

$$(\neg\phi \to \neg\psi) \to ((\neg\phi \to \psi) \to \phi)$$

Note that we are using the conventions for propositions. By an earlier exercise, we know that we could have given all the axioms in terms of $\vee$ in the place of $\to$, after making the necessary modifications. However, it will be more convenient to use $\to$ in the forthcoming proofs.

Note that we have as many axioms as the number of propositions, although we seem to have only three of them at first look. We call each of these three items an *axiom scheme*.

The formal theory of propositional logic has a single inference rule, called *Modus Ponens (MP)*:

ψ follows from $\phi \to \psi$ and ϕ by Modus Ponens for every ϕ, ψ.

We show this as

$$\frac{\phi \to \psi, \phi}{\psi}.$$

The correct way of presenting this inference rule is to say that it is the following ternary relation on $\mathcal{P}$:

$$\big\{(\phi \to \psi, \phi, \psi) : \phi, \psi \in \mathcal{P}\big\}.$$

Our eventual goal is that being a logical consequence is the same as being a deductive consequence in this formal system for the propositional logic. In particular, we would like to show that tautologies of propositional logic are exactly the theorems of propositional logic. Actually, that's all we are going to do and leave the general case as an exercise. We handle the general case when dealing with first order theories.

It is easier to show that theorems are tautologies and that implication is called the *Soundness Theorem (for Propositional Logic)*. The harder implication is to construct proofs for tautologies and that is called the *Completeness Theorem (for Propositional Logic)*.

A particular case of the Soundness Theorem is that the axioms are tautologies.

Homework 5. Show that the axioms of the formal theory of propositional logic are tautologies. (*Hint: Just use truth tables.*)

After these remarks, let's start to deduce things. Below we do not refer to the formal theory as it is fixed as Proposition Logic; for instance we simply say 'ϕ is a theorem', rather than saying 'ϕ is a theorem of Propositional Logic'. Similarly we use $\vdash$ rather than $\vdash_{\mathfrak{T}_P}$.

LEMMA 3.3. *The proposition $\phi \to \phi$ is a theorem for every proposition ϕ.*

PROOF. We write down each proposition in the proof and next to it, we indicate why is it allowed:

(1) $(\phi \to ((\phi \to \phi) \to \phi)) \to ((\phi \to (\phi \to \phi)) \to (\phi \to \phi))$ (Axiom 2 with $\phi \to \phi$ and ϕ in the places of ψ and θ respectively)
(2) $\phi \to ((\phi \to \phi) \to \phi)$ (Axiom 1 with $\phi \to \phi$ in the place of ψ)
(3) $(\phi \to (\phi \to \phi)) \to (\phi \to \phi)$ (MP applied to 1 and 2 above)
(4) $\phi \to (\phi \to \phi)$ (Axiom 1 with ϕ in the place of ψ)
(5) $\phi \to \phi$ (MP applied to 3 and 4 above)

□

As we have seen in this proof, writing down a proof can be painful; even for something as simple as $\phi \to \phi$. The next results eases the pain a little.

THEOREM 3.4 (Deduction Theorem). *Let Γ be a set of propositions and let ϕ and ψ be propositions. If $\Gamma \cup \{\phi\} \vdash \psi$, then $\Gamma \vdash \phi \to \psi$.*

PROOF. Let $\phi_1, \dots, \phi_n$ be a proof of ψ from $\Gamma \cup \{\phi\}$; so $\phi_n = \psi$. We prove that $\Gamma \vdash \phi \to \phi_i$ for every $i \in \{1, \dots, n\}$ by induction on i. (In particular, $\Gamma \vdash \phi \to \phi_n$ which is to say that $\Gamma \vdash \phi \to \psi$.)

$\underline{i = 1}$: ϕ_1 is either an axiom, or an element of Γ, or ϕ. In the first two cases, we have the following deduction of $\phi \to \phi_1$ from Γ:

$$\phi_1, \phi_1 \to (\phi \to \phi_1), \phi \to \phi_1.$$

If $\phi_1 = \phi$, then we get $\vdash \phi \to \phi_1$ by the previous lemma.

$\underline{i > 1}$: ϕ_i is either an axiom, or an element of Γ, or ϕ, or obtained from ϕ_j and ϕ_k for some $j, k < i$ by MP. The first three cases are handled in exactly the same way as in the $i = 1$ case. So suppose ϕ_j is $\phi_k \to \phi_i$ with $j, k < i$. By induction, we know that $\Gamma \vdash \phi \to \phi_k$ and $\Gamma \vdash \phi \to (\phi_k \to \phi_i)$. Also being an axiom the proposition

$$(\phi \to (\phi_k \to \phi_i)) \to ((\phi \to \phi_k) \to (\phi \to \phi_i))$$

follows from Γ. Now applying MP twice, we get that $\Gamma \vdash \phi \to \phi_i$, as desired. □

As in the case of logical consequences, we write $\phi_1, \dots, \phi_n \vdash \psi$ rather than $\{\phi_1, \dots, \phi_n\} \vdash \psi$.

COROLLARY 3.5. *Let ϕ, ψ, θ be propositions. Then the following hold:*

(1) $\phi \to \psi, \psi \to \theta \vdash \phi \to \theta$.
(2) $\phi \to (\psi \to \theta), \psi \vdash \phi \to \theta$.

Proof. (1) By Deduction Theorem, it suffices to prove that

$$\phi \to \psi, \psi \to \theta, \phi \vdash \theta.$$

Here is the formal deduction of θ from $\{\phi \to \psi, \psi \to \theta, \phi\}$ without any explanations:

$$\phi \to \psi,\ \psi \to \theta,\ \phi,\ \psi,\ \theta.$$

(2) This time it suffices to prove $\phi \to (\psi \to \theta), \psi, \phi \vdash \theta$. Again here is the formal deduction without explanation:

$$\phi \to (\psi \to \theta),\ \phi,\ \psi \to \theta,\ \psi,\ \theta.$$

$\square$

The converse of the Deduction Theorem is also correct, but it is much easier to prove. So we leave it as an exercise.

Homework 6. Let Γ be a set of propositions and let ϕ and ψ be propositions. Show that if $\Gamma \vdash \phi \to \psi$, then $\Gamma \cup \{\phi\} \vdash \psi$.

Proposition 3.6. *Let ϕ and ψ be propositions. Then the following are theorems of propositional logic.*

(1) $\neg\neg\phi \to \phi$.
(2) $\phi \to \neg\neg\phi$.
(3) $\neg\phi \to (\phi \to \psi)$.
(4) $(\neg\phi \to \neg\psi) \to (\psi \to \phi)$.
(5) $(\psi \to \phi) \to (\neg\phi \to \neg\psi)$.
(6) $\phi \to (\neg\psi \to \neg(\phi \to \psi))$.
(7) $(\phi \to \psi) \to ((\neg\phi \to \psi) \to \psi)$.

Proof. In each of the following, we just write down the formal proof and leave some of the justification to the reader.

(1) $(\neg\phi \to \neg\neg\phi) \to ((\neg\phi \to \neg\phi) \to \phi)$ - Axiom 3
$\neg\phi \to \neg\phi$ - Lemma 3.3
$(\neg\phi \to \neg\neg\phi) \to \phi$ - Corollary 3.5, second part
$\neg\neg\phi \to (\neg\phi \to \neg\neg\phi)$ - Axiom 1
$\neg\neg\phi \to \phi$ - Corollary 3.5, first part
(2) $(\neg\neg\neg\phi \to \neg\phi) \to ((\neg\neg\neg\phi \to \phi) \to \neg\neg\phi)$ - Axiom 3
$\neg\neg\neg\phi \to \neg\phi$ - Part 1 of this
$(\neg\neg\neg\phi \to \phi) \to \neg\neg\phi$ - MP
$\phi \to (\neg\neg\neg\phi \to \phi)$ - Axiom 1
$\phi \to \neg\neg\phi$ - Corollary 3.5, first part
(3) By the Deduction Theorem, it is enough to show that $\phi, \neg\phi \vdash \psi$. Here is that deduction:
$\phi \to (\neg\psi \to \phi)$ - Axiom 1
ϕ
$\neg\psi \to \phi$ - MP
$\neg\phi \to (\neg\psi \to \neg\phi)$ - Axiom 1

$\neg\phi$
$\neg\psi \to \neg\phi$ - MP
$(\neg\psi \to \neg\phi) \to ((\neg\psi \to \phi) \to \psi)$ - Axiom 3
$(\neg\psi \to \phi) \to \psi$ -MP
ψ - MP

(4) By the Deduction Theorem, it is enough to show that $\psi \to \phi$ follows from $\neg\phi \to \neg\psi$. So here that deduction:
$(\neg\phi \to \neg\psi) \to ((\neg\phi \to \psi) \to \phi)$ - Axiom 3
$\neg\phi \to \neg\psi$ - Assumption
$(\neg\phi \to \psi) \to \phi$ - MP
$\psi \to (\neg\phi \to \psi)$ - Axiom 1
$\psi \to \phi$ - Corollary 3.5, first part

(5) Once again, it suffices to show that $\neg\phi \to \neg\psi$ follows from $\psi \to \phi$ by the Deduction Theorem, and here is how it is done:
$\neg\neg\psi \to \psi$ - Part 1
$\psi \to \phi$ - Assumption
$\neg\neg\psi \to \phi$ - Corollary 3.5
$\phi \to \neg\neg\phi$ - Part 2
$\neg\neg\psi \to \neg\neg\phi$ - Corollary 3.5
$(\neg\neg\psi \to \neg\neg\phi) \to (\neg\phi \to \neg\psi)$ - Part 4
$\neg\phi \to \neg\psi$ - MP

(6) First note that $\phi, \phi \to \psi \vdash \psi$ and hence using the Deduction Theorem we get $\phi \to ((\phi \to \psi) \to \psi)$. We also have the following theorem by the previous part:
$$((\phi \to \psi) \to \psi) \to (\neg\psi \to \neg(\phi \to \psi)).$$
Now using Corollary 3.5, we get the desired result:
$$\vdash \phi \to (\neg\psi \to \neg(\phi \to \psi)).$$

(7) We show that ψ follows from $\{\phi \to \psi, \neg\phi \to \psi\}$ as follows:
$(\phi \to \psi) \to (\neg\psi \to \neg\phi)$ - Part 5
$\phi \to \psi$ - Assumption
$\neg\psi \to \neg\phi$ - MP
$(\neg\phi \to \psi) \to (\neg\psi \to \neg\neg\phi)$ - Part 5
$\neg\phi \to \psi$ - Assumption
$\neg\psi \to \neg\neg\phi$ - MP
$(\neg\psi \to \neg\neg\phi) \to ((\neg\psi \to \neg\phi) \to \psi)$ - Axiom 3
$(\neg\psi \to \neg\phi) \to \psi$ - MP
ψ - MP

□

Now we are ready to state and prove the Soundness Theorem.

THEOREM 3.7 (Soundness Theorem). *Let ϕ be a proposition. If $\vdash \phi$, then $\models \phi$.*

PROOF. Suppose that ϕ is a theorem and fix a proof $\phi_1, \ldots, \phi_n = \phi$ of it. We shall prove that each ϕ_i is a tautology, by induction on i. When $i = 1$, the only possibility is that ϕ_i is an axiom, and by an earlier exercise we know that axioms are tautologies. Suppose that $i > 0$ and assume that ϕ_j is a tautology for each $j < i$. Now there are two possibilities: either ϕ_i is an axiom or ϕ_i is obtained from ϕ_j and ϕ_k by MP for $j, k < i$. We are again done if ϕ_i is an axiom. In the latter case we know by the induction hypothesis that ϕ_j and ϕ_k are tautologies, and that ϕ_k is $\phi_j \to \phi_i$. By the second part of Proposition 2.4, if $\psi \to \theta$ and ψ are tautologies, then so is θ. Therefore ϕ_i is a tatutology.

□

Homework 7. Let Γ be a set of propositions and ϕ a proposition. Show that if $\Gamma \vdash \phi$, then $\Gamma \models \phi$.

COROLLARY 3.8. *There are no propositions ϕ such that $\vdash \phi$ and $\vdash \neg\phi$.*

PROOF. Suppose $\vdash \phi$ and $\vdash \neg\phi$, then by Soundness Theorem $\models \phi$ and $\models \neg\phi$, which is not possible by the definition of $\models$. □

DEFINITION 3.9. A set Γ of propositions is called *inconsistent* if there is a proposition ϕ such that $\Gamma \vdash \phi$ and $\Gamma \vdash \neg\phi$; this is denoted as $\Gamma \vdash$. Otherwise, we say that Γ is *consistent*.

With this new notion, Corollary 3.8 says that $\emptyset$ is consistent.

Homework 8. Let Γ be a set of propositions and ϕ a proposition.

(1) Show that Γ is inconsistent if and only if $\Gamma \vdash \psi$ for every ψ.
(2) Show that $\Gamma \cup \{\neg\phi\}$ is inconsistent if and only if $\Gamma \vdash \phi$.

As mentioned before the converse of the Soundness Theorem, the Completeness Theorem, is considerably harder to prove. There are two proofs. One is more constructive: Given a tautology, it allows us to construct a proof of that tautology. The second one is not as constructive. Indeed, it proves the contrapositive: If a proposition is not a theorem, then we may construct a valuation that makes that proposition false. Both of these proofs have advantages over the other. The first one works in a lesser generality. So we give that proof for propositional logic and we present the second proof for first order theories in the next chapter.

The constructive proof is due to László Kalmár, and it heavily depends on the following lemma.

LEMMA 3.10. *Let ϕ be a proposition whose atoms are among $P_1, \ldots, P_n$ and let v be a valuation. Make the following definitions:*

$$\phi' = \begin{cases} \phi & , \textit{if } v(\phi) = 1 \\ \neg\phi & , \textit{if } v(\phi) = 0 \end{cases} \qquad P_i' = \begin{cases} P_i & , \textit{if } v(P_i) = 1 \\ \neg P_i & , \textit{if } v(P_i) = 0 \end{cases}$$

Then $P_1', \dots, P_n' \vdash \phi'$.

Proof. We proceed by induction on the number of connectives in ϕ.

ϕ is P_1: In any case, $\phi' = P_1'$. Then $P_1' \vdash \phi'$.

ϕ is $\neg\psi$: First note that the atoms appearing in ψ are also among $P_1, \dots, P_n$. If $v(\phi) = 1$, then $v(\psi) = 0$. So $\phi' = \phi$ and $\psi' = \neg\psi$. By induction we have

$$P_1', \dots, P_n' \vdash \psi'.$$

So $P_1', \dots, P_n' \vdash \neg\psi$. But this means $P_1', \dots, P_n' \vdash \phi'$.

Now suppose $v(\phi) = 0$. Then $v(\psi) = 1$. So $\phi' = \neg\phi$ and $\psi' = \psi$. Again, by induction we have

$$P_1', \dots, P_n' \vdash \psi.$$

Then using the second part of Proposition 3.6 and applying MP, we get

$$P_1', \dots, P_n' \vdash \neg\neg\psi.$$

This means

$$P_1', \dots, P_n' \vdash \neg\phi.$$

This finishes the case as $\neg\phi = \phi'$.

ϕ is $\psi \to \theta$: Suppose that $v(\phi) = 0$. This happens only when $v(\psi) = 1$ and $v(\theta) = 0$. Then $\phi' = \neg\phi$, $\psi' = \psi$, and $\theta' = \neg\theta$. By induction, we have

$$P_1', \dots, P_n' \vdash \psi \text{ and } P_1', \dots, P_n' \vdash \neg\theta.$$

By the sixth part of Proposition 3.6 and using MP twice, we get

$$P_1', \dots, P_n' \vdash \neg(\psi \to \theta).$$

This means

$$P_1', \dots, P_n' \vdash \phi'.$$

Suppose $v(\phi) = 1$. Then either $v(\psi) = 0$ or $v(\psi) = v(\theta) = 1$.

First assume that $v(\psi) = 0$. Then $\phi' = \phi$ and $\psi' = \neg\psi$. By induction

$$P_1', \dots, P_n' \vdash \neg\psi.$$

Using the third part of Proposition 3.6, we get

$$P_1', \dots, P_n' \vdash \psi \to \theta,$$

which is the same as

$$P_1', \dots, P_n' \vdash \phi'.$$

Finally, suppose that $v(\psi) = v(\theta) = 1$. Then $\phi' = \phi$, $\psi' = \psi$, and $\theta' = \theta$. Then

$$P_1', \dots, P_n' \vdash \theta.$$

We also know that

$$P_1', \dots, P_n' \vdash \theta \to (\psi \to \theta).$$

So by MP we get

$$P_1', \dots, P_n' \vdash \psi \to \theta,$$

as desired. $\square$

Now we are ready to prove the Completeness Theorem for Propositional Logic.

THEOREM 3.11 (Completeness Theorem). *Let ϕ be a proposition. If $\models \phi$, then $\vdash \phi$.*

PROOF. Let the atoms appearing in ϕ be among $P_1, \dots, P_n$.

Let v_1 be a valuation with $v_1(P_n) = 1$. Then $P_n' = P_n$, $v_1(\phi) = 1$ and $\phi' = \phi$. So

$$P_1', \dots, P_{n-1}', P_n \vdash \phi.$$

Now let v_2 be the valuation whose value at P_i is the same as the value v_1 at P_i for each $i \in \{1, \dots, n-1\}$ and $v_2(P_n) = 0$. This time we have

$$P_1', \dots, P_{n-1}', \neg P_n \vdash \phi.$$

Then

$$P_1', \dots, P_{n-1}' \vdash P_n \to \phi \text{ and } P_1', \dots, P_{n-1}' \vdash \neg P_n \to \phi$$

Using the last part of Proposition 3.6, we get

$$P_1', \dots, P_{n-1}' \vdash \phi.$$

Applying the same arguments to P_{n-1} in the place of P_n we get

$$P_1', \dots, P_{n-2}' \vdash \phi.$$

Then continuing this way we get $\vdash \phi$ as desired. $\square$

Homework 9. Let Γ be a set of propositions and ϕ, ψ, θ propositions.

(1) Show that if $\Gamma, \phi \vdash \theta$ and $\Gamma, \psi \vdash \theta$, then $\Gamma, \phi \vee \psi \vdash \theta$.
(2) Show that if $\Gamma, \phi \vdash \psi$ and $\Gamma, \psi \vdash \phi$, then $\Gamma \vdash \phi \leftrightarrow \psi$.

Homework 10. Let Γ be a set of propositions and ϕ a proposition. Show that if $\Gamma \models \phi$, then $\Gamma \vdash \phi$. (*Hint: Imitate the proof of Completeness Theorem.*)

DEFINITION 3.12. A set Γ of propositions is called *satisfiable* if there is a valuation v such that $v(\phi) = 1$ for every $\phi \in \Gamma$.

The following is a straightforward exercise.

Homework 11. Let Γ be a set of propositions. Show that Γ is not satisfiable if and only if there is a proposition ϕ such that $\Gamma \models \phi$ and $\Gamma \models \neg\phi$.

Using this equivalence we may prove the so called Compactness Theorem for Propositional Logic. We will get into more details of the corresponding theorem for first order theories.

COROLLARY 3.13 (Compactness Theorem for Propositional Logic). *Let Γ be a set of propositions. Show that Γ is satisfiable if and only if every finite subset of Γ is satisfiable.*

PROOF. It is clear that if Γ is satisfiable, then any subset of it is also satisfiable; in particular finite subsets. For the converse direction assume that Γ is not satisfiable. Then using the homework above, there is a proposition ϕ such that $\Gamma \models \phi$ and $\Gamma \models \neg\phi$. Therefore we have $\Gamma \vdash \phi$ and $\Gamma \vdash \neg\phi$ by the Completeness Theorem (the version in Homework 10). Since proofs are finite sequences, there is a finite subset Γ_0 such that $\Gamma_0 \vdash \phi$ and $\Gamma_0 \vdash \neg\phi$ (or we could refer to the second part of Proposition 3.2 for this). Then by Soundness Theorem, we get $\Gamma_0 \models \phi$ and $\Gamma_0 \models \neg\phi$, which is to say that Γ_0 is not satisfiable, using the converse direction of the statement in the homework above. $\square$

CHAPTER 2

First Order Logic

The structure of this chapter will be very similar to the previous one, however it'll be much more technical. Propositional Logic is just a toy we played with to get acquainted with the the ideas. Now we have an idea about the concepts of 'truth' and 'proof', and we will work in an actual mathematical context.

So we will introduce the syntax and semantics of First Order Logic and then we will construct a formal system of proof for First Order Logic.

1. Syntax

Definition 1.1. A *first order alphabet* $\mathcal{A}$ consists of the following:

(1) *Connectives:* $\neg \quad \to$
(2) *Parentheses:* ( )
(3) *Quantifier:* $\forall$
(4) *Variables:* $v_1 \quad v_2 \quad \ldots$
(5) *Equality:* $=$
(6) *Function Symbols:* $f_1^{(m_1)} \quad f_2^{(m_2)} \ldots$
(7) *Relation Symbols:* $R_1^{(n_1)} \quad R_2^{(n_2)} \ldots$
(8) *Constant Symbols:* $c_1 \quad c_2 \quad \ldots$

Function and relation symbols require a little bit of explanation. Each such symbol comes with a positive integer called its *arity*, and we write it on top of the symbol in parentheses. As before, the symbols do not have meaning yet, but later we will construe a function symbol of arity n as a function from n many Cartesian copies of a set to that set. Similarly a relation symbol of arity n will be interpreted as a subset of n many Cartesian copies of a set. When the arity of a function or a relation is 1, then we say that it is *unary*. Similarly, we use the words *binary* and *ternary* for arities 2 and 3.

Next order of business is to determine the well-formed formulas as in Definition 3.1 in the previous chapter. This will be done in several steps.

Definition 1.2. The *terms* of a given first-order alphabet $\mathcal{A}$ are certain expressions of $\mathcal{A}$ that are built as follows:

(1) Variables and constant symbols are terms.
(2) If $t_1, \ldots, t_{m_i}$ are terms, then so is $f_i^{(m_i)}(t_1 t_2 \cdots t_{m_i})$.

(3) Nothing else is a term.

Once we start to give meanings, we will think of terms as basic functions that can be expressed in the alphabet.

DEFINITION 1.3. The *atomic formulas* of $\mathcal{A}$ are expressions that are either of the form $t_1 = t_2$ or of the form $R_i^{(n_i)}(t_1 t_2 \cdots t_{n_i})$ where $t_1, \ldots, t_{n_i}$ are terms of $\mathcal{A}$.

Atomic formulas are like atomic propositions of Propositional Logic and we will construct formulas from them as we have constructed propositions from atomic propositions. There is again one difference: Quantifiers.

DEFINITION 1.4. The *formulas* of a first-order alphabet $\mathcal{A}$ are expressions that are built from atomic formulas as follows:

(1) Atomic formulas are formulas.
(2) If ϕ and ψ are formulas, then so are $(\neg\phi)$, $(\phi \to \psi)$, $(\forall v_i \phi)$.
(3) Nothing else is a formula.

We could have defined formulas in terms of 'construction sequences'; how to do this must be clear to you even though technicalities might be bothersome. That way we could define concepts such as the *construction tree* of a formula, *rank* of a formula and *subformulas* of a formula. We shall use these concepts freely in the rest of this text.

We write *$\mathcal{A}$-formula*, rather than *a formula of the language $\mathcal{A}$*.

Conventions. We keep the parentheses conventions from Propositional Logic. Namely, we omit the outmost parentheses and parentheses around negation are omitted. We also have a convention similar to the negation convention: We omit parentheses around $(\forall v_i\, \phi)$.

As before, we define the shorthand notations as $\vee$,$\wedge$, $\leftrightarrow$. We add the following shorthand: $(\exists v_i \phi)$ is short for $(\neg(\forall v_i(\neg\phi)))$. With the conventions above, this can be written as $\neg\forall v_i \neg\phi$.

We write $f_i^{(m_i)}(t_1, t_2, \cdots, t_{m_i})$ and $R_i^{(n_i)}(t_1, t_2, \cdots, t_{n_i})$ in the places of $f_i^{(m_i)}(t_1 t_2 \cdots t_{m_i})$ and $R_i^{(n_i)}(t_1 t_2 \cdots t_{n_i})$. The reason for this convention is to make it more suitable with the usual notation for functions and relations; that way we hope the readability will be increased as well. An alternative way would be to add commas in the alphabets; so that $f_i^{(n_i)}(t_1, \cdots, t_{n_i})$ would be an expression in a given first-order alphabet.

We use letters x, y, z for variables; possibly with decorations. This way we do not have to distinguish which v_i we are using.

We write a term t as $t(x_1, \ldots, x_n)$ if the variables occurring in t are among $x_1, \ldots, x_n$; if $\vec{x} = (x_1, \ldots, x_n)$ is a tuple of variables, then we simply write $t(\vec{x})$.

Most of the times, we omit the arities of function and relation symbols. Also we use other letters in the neighborhood of f to denote function symbols; again with possible decorations. Sometimes we use familiar function symbols in the familiar way. For instance, in a while $+$ will denote a binary function and we write $x+y$ rather than $+(x, y)$. Similarly we use letters around R to denote relation symbols and familiar relation symbols are used; such as $<$.

Note that what distinguishes alphabets are the function, relation and constant symbols in them; they are called *non-logical symbols* of the alphabet. So when talking about an alphabet, we need to specify only them. As a matter of fact, we will not refer to alphabets any more, but to *languages*. When we say L is a language, we mean that we fix a first-order alphabet and consider terms and formulas in that alphabet; and we write only the non-logical symbols; see examples below.

EXAMPLE 1.5. Let $L_{\mathrm{o}} = \{<\}$ be the language of orderings. This means that there is only one non-logical symbol, which is a binary relation symbol. As mentioned above, we write $x < y$ rather than $<(x, y)$.

EXAMPLE 1.6. We call $L_{\mathrm{ab}} = \{+, -, 0\}$ as the language of abelian groups. It has one binary function symbol, one unary function symbol, and one constant symbol.

EXAMPLE 1.7. Let $\mathbb{Q}$ be the field of rational numbers. The language $L_{\mathbb{Q}\text{-vs}}$ of $\mathbb{Q}$-vector spaces extends L_{ab} by countably many unary function symbols; one for each rational number q. We denote it by s_q.

Variable Occurrences. A variable x may appear in a formula with two different roles; after $\forall$ or not. We would like to set some notation for these and more.

Consider the formula $\forall x\phi$. We say that ϕ is the *scope* of $\forall x$. The formula $\forall x\phi$ might appear as a subformula of a more complicated formula; we still say that ϕ is the scope of $\forall x$. However, $\forall x$ may appear more than once in a formula. In that case, we should make sure scope of which one we are talking about.

DEFINITION 1.8. An occurrence of a variable x in a formula ϕ is said to be *bound* if it is either in $\forall x$ or in the scope of $\forall x$. Otherwise, we say that the occurrence is *free*.

EXAMPLE 1.9. Let $\mathcal{L} = \{f, R, c\}$ where f is a binary function symbol, R is a binary relation symbol, and c is a constant symbol c. Let x, y, z be distinct variables and let ϕ be the following $\mathcal{L}$-formula:

$$R(f(x,c), f(c,y)) \to ((\forall y(f(x,c) = f(y,c))) \to \neg(\forall z(R(f(c,c), f(x,x))))).$$

All four occurrences of the variable x are free. The first occurrence of y is free, but the next two occurrences are bound. Finally, the only occurrence of the variable z is bound.

Consider $\forall y(\phi)$. Then all the occurrences of y become bound. However, there is a strange thing happening: The third y in ϕ seems to be bound (in $\forall y\phi$) by two different quantifiers. This is not a problem. Later when we interpret this formula, we will see that the second $\forall y$ is the one bounding that y and the first one will have no effect on it.

Definition 1.10. A formula in which there are no free occurrences of any variable is called a *sentence*.

We would like to capture the concept of "replacing free occurrences of variables in a formula by a term". In order to do this, we first need to define how to replace variables in a term by other terms.

Definition 1.11. Let $s, t_1, \ldots, t_m$ be terms in a language L, and let $x_1, \ldots, x_m$ be variables. We define $s(t_1/x_1, \ldots, t_m/x_m)$ by induction as follows:

(1) If s is a constant symbol or a variable other than $x_1, \ldots, x_n$, then $s(t_1/x_1, \ldots, t_m/x_m)$ is s.
(2) If s is the variable x_i, then $s(t_1/x_1, \ldots, t_m/x_m)$ is t_i.
(3) If s is $f(s_1, \ldots, s_n)$ for a function symbol f and terms $s_1, \ldots, s_n$, then $s(t_1/x_1, \ldots, t_m/x_m)$ is
$$f(s_1(t_1/x_1, \ldots, t_m/x_m), \ldots, s_n(t_1/x_1, \ldots, t_m/x_m)).$$

Note that the expression $s(t_1/x_1, \ldots, t_m/x_m)$ is really a term by construction.

Definition 1.12. Let ϕ be a formula, $x_1, \ldots, x_m$ distinct variables, and $t_1, \ldots, t_m$ terms. We define $\phi(t_1/x_1, \ldots, t_m/x_m)$ as follows:

(1) If ϕ is $s_1 = s_2$, then $\phi(t_1/x_1, \ldots, t_m/x_m)$ is
$$s_1(t_1/x_1, \ldots, t_m/x_m) = s_2(t_1/x_1, \ldots, t_m/x_m).$$
(2) If ϕ is $R(s_1, \ldots, s_n)$, then $\phi(t_1/x_1, \ldots, t_m/x_m)$ is
$$R(s_1(t_1/x_1, \ldots, t_m/x_m), \ldots, s_n(t_1/x_1, \ldots, t_m/x_m)).$$
(3) If ϕ is $\neg\psi$, then $\phi(t_1/x_1, \ldots, t_m/x_m)$ is
$$\neg\psi(t_1/x_1, \ldots, t_m/x_m).$$
(4) If ϕ is $\psi \to \theta$, then $\phi(t_1/x_1, \ldots, t_m/x_m)$ is
$$\psi(t_1/x_1, \ldots, t_m/x_m) \to \theta(t_1/x_1, \ldots, t_m/x_m).$$
(5) If ϕ is $\forall y\psi$ for some variable other than $x_1, \ldots, x_m$, then $\phi(t_1/x_1, \ldots, t_m/x_m)$ is
$$\forall y\psi(t_1/x_1, \ldots, t_m/x_m).$$

(6) If ϕ is $\forall x_i \psi$ for some $i \in \{1, \dots, m\}$, then $\phi(t_1/x_1, \dots, t_m/x_m)$ is

$$\forall x_i \psi(t_1/x_1, \dots, t_{i-1}/x_{i-1}, x_i/x_i, t_{i+1}/x_{i+1}, \dots, t_m/x_m).$$

Once again, it is clear from the construction that $\phi(t_1/x_1, \dots, t_m/x_m)$ is a formula. The subtlety of this definition is the last part; it basically tells us not to replace the bound occurrences of variables.

Note that there is no obstacle to replace a free occurrence of a variable by another variable which becomes bound. For instance, let ϕ be $\forall y\, R(x)$, where R is a unary relation symbol and x and y are distinct variables. Then the only occurrence of x in ϕ is free and if we consider $\phi(y/x)$, then we get $\forall y\, R(y)$, and clearly this formula is a sentence. The next definition is aiming to care of this situation. That definition looks complicated, but just think it as *'we may replace the variable x by the term t in the formula ϕ without causing problems'*. We will see later what it means to not cause problems.

Definition 1.13. Let x be a variable, t a term, and ϕ a formula. We say that t *is free for x in ϕ* if no free occurrence of x in ϕ lies within the scope of $\forall y$ for some variable y occurring in t.

2. Semantics

Recall that semantics of Propositional Logic was given by 'valuations of atoms'. So what is the corresponding object here? To begin with, what are atoms? Taking them to be the atomic formulas, what does it mean for them to be true or false? Our present setting is much more complicated, but this is the basic question we need to address.

In what follows L is a first-order language.

Definition 2.1. An *L-structure* $\mathcal{M}$ is a set M with the *interpretation* of each of the non-logical symbols of L as follows:

- If f is an n-ary function symbol of L, then $f^{\mathcal{M}} : M^n \to M$ is a function,
- If R is an n-ary relation symbol of L, then $R^{\mathcal{M}}$ is a subset of M^n,
- If c is a constant symbol of L, then $c^{\mathcal{M}}$ is an element of M.

We will call M in this definition the *universe* of $\mathcal{M}$; sometimes the word *domain* is used. Note how we used the same letter in different font for the structure and its universe. We will try to keep doing this in these notes, but there will be some places where it is better use different letters. We generally write

$$\mathcal{M} = (M, f_1^{\mathcal{M}}, \dots, R_1^{\mathcal{M}}, \dots, c_1^{\mathcal{M}}, \dots)$$

to denote a structure with the interpretations of symbols exposed.

EXAMPLE 2.2. Consider $L_o = \{<\}$ from above. One quick remark: Just because the symbol is the ordering symbol, its interpretation does not have to be an ordering. But it will be! The point is symbols are just symbols and we choose them to reflect how they will be interpreted.

An example of an L_o-structure is $\mathcal{R} = (\mathbb{R}, <^{\mathcal{R}})$ where $<^{\mathcal{R}}$ is the usual ordering of real numbers.

In general, we may take any partially ordered set and interpret $<$ as the ordering.

EXAMPLE 2.3. Let $L = \{E\}$, where E is a binary relation symbol. Given a set X, we may, for instance, interpret E as an equivalence relation on X.

EXAMPLE 2.4. Consider the language $L_{\text{ab}} = \{+, -, 0\}$. Any abelian group can be thought of as an L_{ab}-structure. Actually, any group can be seen as an L_{ab}-structure, but it won't really be very natural.

Now we interpret terms in a structure.

DEFINITION 2.5. Let $\mathcal{M}$ be an L-structure and $\vec{a} = (a_1, a_2, \dots)$ be a countable tuple of elements of M. Given an L-term t, we define $t^{\mathcal{M}}(\vec{a})$ by recursion as follows:

(1) If t is v_i, then $t^{\mathcal{M}}(\vec{a}) = a_i$.
(2) If t is c, then $t^{\mathcal{M}}(\vec{a}) = c^{\mathcal{M}}$.
(3) If f is an n-ary function symbol, $t_1, \dots, t_n$ are L-terms, and t is $f(t_1 \cdots t_n)$, then

$$t^{\mathcal{M}}(\vec{a}) = f^{\mathcal{M}}\big(t_1^{\mathcal{M}}(\vec{a}), \dots, t_n^{\mathcal{M}}(\vec{a})\big).$$

Remarks. Even though $\vec{a}$ is an infinite tuple, t contains only finitely many variables; somehow only a finite part of $\vec{a}$ is used. More precisely, suppose that the variables occurring in t are among $v_{i_1}, \dots, v_{i_n}$, and suppose also that $\vec{a}$ and $\vec{b}$ are countable tuples from M such that $a_{i_1} = b_{i_1}, \dots, a_{i_n} = b_{i_n}$. Then $t^{\mathcal{M}}(\vec{a}) = t^{\mathcal{M}}(\vec{b})$.

So if the variables occurring in t are among $x_1, \dots, x_n$ and $\vec{a} = (a_1, \dots, a_n)$ is a tuple from M, then $t^{\mathcal{M}}(\vec{a})$ is the element of M obtained by "substituting a_i in the place of x_i". One may make this mathematically correct, but we think it is clear enough as it is.

In the abstract setting, we keep $\mathcal{M}$ in $t^{\mathcal{M}}(\vec{a})$. However, if we are working in a concrete setting, we drop it. For instance, if we consider the set of integers as an L_{ab}-structure, then we write $m+n$ rather than $m +^{\mathbb{Z}} n$ or $+^{\mathcal{M}}(m, n)$.

Next we define the truth of a formula in a structure.

DEFINITION 2.6. Let $\mathcal{M}$ be an L-structure, $\vec{a} = (a_1, a_2, \dots)$ be a countable tuple of elements of M. We define $\vec{a}$ *satisfying* an L-formula (in $\mathcal{M}$) by recursion as follows:

(1) $\vec{a}$ satisfies $t_1 = t_2$ if $t_1^{\mathcal{M}}(\vec{a}) = t_2^{\mathcal{M}}(\vec{a})$.
(2) $\vec{a}$ satisfies an atomic formula $R(t_1 \cdots t_n)$ if

$$\big(t_1^{\mathcal{M}}(\vec{a}), \ldots, t_n^{\mathcal{M}}(\vec{a})\big) \in R^{\mathcal{M}}.$$

(3) $\vec{a}$ satisfies $\neg\phi$ if it doesn't satisfy ϕ.
(4) $\vec{a}$ satisfies $\phi \to \psi$ if either $\vec{a}$ does not satisfy ϕ or $\vec{a}$ satisfies ψ.
(5) $\vec{a}$ satisfies $\forall v_i \phi$ if every countable sequence $\vec{b}$ that differs from $\vec{a}$ in at most at the i^{th} place satisfies ϕ.

Remarks. As in the case of terms, each formula has only finitely many occurrences of variables. So the 'truth value' of ϕ at $\vec{a}$ depends only a finite part of $\vec{a}$. Let's elaborate on this. Suppose that the variables occurring freely in ϕ are among $v_{i_1}, \ldots, v_{i_n}$, and let $\vec{a}, \vec{a}'$ be two countable tuples from M with $a_j = a'_j$ for $j \in \{i_1, \ldots, i_n\}$. Then one can easily show by induction on the complexity of ϕ that $\vec{a}$ satisfies ϕ if and only if $\vec{a}'$ satisfies ϕ.

So given tuple $\vec{a} = (a_1, \ldots, a_n) \in M^n$ and formula ϕ whose free variables are among $x_1, \ldots, x_n$, we may define $\vec{a}$ satisfying ϕ in $\mathcal{M}$ in a natural way.

Note, in particular, that if σ is a sentence, then either any $\vec{a}$ satisfies it or no $\vec{a}$ satisfies it. So we may talk about a sentence being *true* in $\mathcal{M}$, without referring to any tuple from M.

We denote $\vec{a}$ satisfying ϕ in $\mathcal{M}$ as

$$\mathcal{M} \models \phi(\vec{a}).$$

The *universal closure* of a formula ϕ whose free variables are among $x_1, \ldots, x_n$ is $\forall x_1 \cdots \forall x_n \phi$. (Note that our conventions are in action here.) Clearly, the universal closure of any formula is a sentence. We write $\mathcal{M} \models \phi$ to mean that the universal closure of ϕ is true in $\mathcal{M}$.

Example 2.7. Let L contain a single non-logical symbol R, which is a binary relation symbol.

First consider the L-structure $\mathcal{M}_1 = (\mathbb{Z}, R^{\mathcal{M}_1})$ where $(m, n) \in R^{\mathcal{M}_1}$ if and only if $m - n$ is even.

Let $\phi(x, y)$ be the L-formula

$$\exists z (R(x, z) \wedge R(y, z)).$$

Let's observe for which $(m, n) \in \mathbb{Z}^2$, this formula is satisfied. So suppose $\mathcal{M}_1 \models \phi(m, n)$. This means that there is $k \in \mathbb{Z}$ with $2|m - k$ and $2|n - k$. It follows that $2|m$ if and only if $2|n$. Converse is also correct and hence we get the equivalence:

$$\mathcal{M}_1 \models \phi(m, n) \iff \text{either } m, n \text{ are both even, or they are both odd.}$$

The latter condition means that $2|m - n$. Therefore

$$\mathcal{M}_1 \models \phi(m, n) \iff \mathcal{M}_1 \models R(m, n).$$

This may be re-stated as

$$\mathcal{M}_1 \models \forall x \forall y \big(\phi(x,y) \leftrightarrow R(x,y)\big).$$

Let's take another formula $\psi(x)$ which is

$$\exists y \exists z \big(\neg R(x,y) \wedge \neg R(x,z) \wedge \neg R(y,z)\big).$$

It is clear that $\mathcal{M}_1 \models \forall x \neg \psi(x)$.

Now let $\mathcal{M}_2 = (\mathbb{Z}, R^{\mathcal{M}_2})$ where $(m,n) \in R^{\mathcal{M}_2}$ if and only if $3|m-n$. As in the previous structure we have

$$\mathcal{M}_2 \models \forall x \forall y \big(\phi(x,y) \leftrightarrow R(x,y)\big).$$

However, this time we have $\mathcal{M}_2 \models \forall x \psi(x)$.

Finally, let $\mathcal{M}_3 = (\mathbb{Z}, R^{\mathcal{M}_3})$ where $(m,n) \in R^{\mathcal{M}_2}$ if and only if $m < n$. Then

$$\mathcal{M}_3 \models \forall x \forall y \phi(x,y) \text{ and } \mathcal{M}_3 \models \forall x \psi(x).$$

In the rest of the notes, we sometimes drop the use of language. For instance '$\mathcal{M}$ is a structure' means that we have a language L that we don't want to specify and $\mathcal{M}$ is an L-structure.

DEFINITION 2.8. Let σ be a sentence, ϕ, ψ formulas, and Γ a set of formulas.

(1) We say that σ is *logically valid* if $\mathcal{M} \models \sigma$ for every structure $\mathcal{M}$. We denote this as $\models \sigma$.
(2) We say σ is *satisfiable* if there is a structure $\mathcal{M}$ such that $\mathcal{M} \models \sigma$.
(3) ϕ is *logically valid* (*satisfiable*) if its universal closure is logically valid (satisfiable).
(4) The set Γ is *satisfiable* if there is a structure $\mathcal{M}$ such that $\mathcal{M} \models \theta$ for every $\theta \in \Gamma$; such a structure $\mathcal{M}$ is called a *model* of Γ, and we write $\mathcal{M} \models \Gamma$.
(5) The formula ϕ is a *logical consequence of* Γ if every model of Γ satisfies ϕ. We write $\Gamma \models \phi$ to denote this.
(6) If $\{\phi\} \models \psi$ and $\{\psi\} \models \phi$, we say that ϕ and ψ are *logically equivalent*.

A sentence σ is logically valid if and only if $\neg\sigma$ is not satisfiable. Also a formula ϕ is satisfiable if there is a structure $\mathcal{M}$ such that every countable tuple $\vec{a}$ from M satisfies ϕ.

Note that ϕ and ψ are logically equivalent if and only if $\phi \leftrightarrow \psi$ is logically valid.

EXAMPLE 2.9. Let's return to the language L that contains a single binary relation symbol R and let's consider the concepts in the previous definition. Below, we refer to the structures $\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3$ of Example 2.7.

An example of a logically valid sentence is

$$\forall x\big(\forall z(\exists y R(x,y) \to \neg R(x,z)) \to (R(x,y) \to \forall z \neg R(x,z))\big).$$

We leave the checking to the reader.

Now consider the sentence

$$\forall x \exists y (R(x,y) \wedge R(y,x)).$$

This sentence is satisfiable since $\mathcal{M}_1$ satisfies it. However, it is not logically valid, because $\mathcal{M}_3$ does not satisfy it.

For $n > 0$, let σ_n be the following sentence

$$\exists x_1 \cdots \exists x_n (\bigwedge_{i \neq j} \neg R(x_i = x_j) \wedge \forall y (\bigvee_{i=1}^{n} R(y, x_i))).$$

Here $\bigwedge_{i \neq j} \neg R(x_i = x_j)$ is a certain formula; we think it is clear what it means and hence we do not define what it is. (Similar for $\bigvee_{i=1}^{n} R(y, x_i)$.)

Note that $\mathcal{M}_1 \models \sigma_2$ and $\mathcal{M}_1 \models \neg \sigma_n$ for every $n \neq 2$. Similarly, $\mathcal{M}_2 \models \sigma_3$ and $\mathcal{M}_2 \models \neg \sigma_n$ for $n \neq 3$. Actually, one may show that $\sigma_m \wedge \sigma_n$ with $m \neq n$ is not satisfiable.

Let ϵ be the sentence obtained by connecting the following by conjunctions

- $\forall x R(x,x)$
- $\forall x \forall y (R(x,y) \to R(y,,x))$
- $\forall x \forall y \forall z ((R(x,y) \wedge R(y,z)) \to R(x,z))$.

This sentence epsilon is certainly satisfiable. Indeed, if E is an equivalence relation on a set M, then the structure

$$\mathcal{M} = (M, E)$$

satisfies ϵ; so the universe is M and $R^{\mathcal{M}} = E$. In particular, $\mathcal{M}_1, \mathcal{M}_2$ satisfy ϵ.

For $n > 0$, let τ_n be the sentence

$$\epsilon \wedge \exists x_1 \cdots \exists x_n (\bigwedge_{i,j} R(x_i, x_j) \wedge \bigwedge_{i \neq j} \neg (x_i = x_j)).$$

Clearly, both of $\mathcal{M}_1, \mathcal{M}_2$ satisfy τ_n for any $n > 0$.

We change τ_n a little bit to define ρ_n as follows

$$\epsilon \wedge \exists x_1 \cdots \exists x_n (\bigwedge_{i,j} R(x_i, x_j) \wedge \bigwedge_{i \neq j} \neg (x_i = x_j) \wedge \forall y (R(y, x_1) \to \bigvee_{i=1}^{n} y = x_i)).$$

Now neither of $\mathcal{M}_1$ and $\mathcal{M}_2$ satisfy ρ_n for any $n > 0$.

Homework 12. Let $\Gamma = \{\rho_n : n > 0\}$.

(1) Show that Γ is satisfiable.
(2) Find some logical consequences of Γ that are not already in Γ.

Next proposition gives infinitely many logically valid formulas.

PROPOSITION 2.10. *Let ϕ, ψ be formulas and let x be a variable that does not have a free occurrence in ϕ. Then the formula*

$$\forall x(\phi \to \psi) \to (\phi \to \forall x\psi)$$

is logically valid.

PROOF. Let x be v_i.
Suppose that there is a structure $\mathcal{M}$ such that

$$\mathcal{M} \not\models \forall x(\phi \to \psi) \to (\phi \to \forall x\psi).$$

This means that there is a countable tuple $\vec{a}$ from M that satisfies $\forall x(\phi \to \psi)$ in $\mathcal{M}$ and does not satisfy $\phi \to \forall x\psi$. Therefore $\vec{a}$ satisfies ϕ and does not satisfy $\forall x\psi$. The second part means that there is a countable tuple $\vec{a}\,'$ that differs from $\vec{a}$ at most at the i^{th} component that does not satisfy ψ. Since there are no free occurrences of x in ϕ and $\forall x(\phi \to \psi)$, the tuple $\vec{a}\,'$ satisfies them. Then $\vec{a}\,'$ does not satisfy $\phi \to \psi$, but it also should satisfy it. This contradiction finishes the proof. $\square$

Another large class of logically valid sentences is given in the next homework.

Homework 13. Let Φ be a tautology of Propositional Logic whose atoms are among $P_1, \ldots, P_n$, and let $\phi_1, \ldots, \phi_n$ be L-formulas. Then the expression that is obtained by replacing each occurrence of P_i in Φ by ϕ_i is an L-formula; denote it as $\Phi(\phi_1, \ldots, \phi_n)$ and it is said to be an *instance* of Φ. Show also that $\Phi(\phi_1, \ldots, \phi_n)$ is logically valid.

Next we investigate the effect on the truth of replacing free occurrences of variables in formulas by terms. The following can be proven by using the recursive definition of $\phi(t/x)$ in 1.12.

Homework 14. Let t be a term that is free for the variable v_i in the formula ϕ. Also let $\mathcal{M}$ be a structure and $\vec{a}$ a countable tuple of elements of M. Show that

$$\mathcal{M} \models (\phi(t/v_i))(\vec{a}) \Longleftrightarrow \mathcal{M} \models \phi(\vec{b}),$$

where $b_j = a_j$ for all $j \neq i$ and $b_i = t^{\mathcal{M}}(\vec{a})$.

PROPOSITION 2.11. *Suppose that t is a term that is free for x in ϕ. Then $\forall x\phi \to \phi(t/x)$ is logically valid.*

PROOF. Let x be v_i and take a structure $\mathcal{M}$ and a countable tuple $\vec{a}$ from M. Suppose that $\mathcal{M} \models (\forall x\phi)(\vec{a})$; this means that for every $\vec{b}$ that differs from $\vec{a}$ only possibly at the i^{th} place we have $\mathcal{M} \models \phi(\vec{b})$. In particular, if we take $\vec{b}$ to be the tuple with $b_j = a_j$ for all $j \neq i$ and

$b_i = t^{\mathcal{M}}(\vec{a})$, we have $\mathcal{M} \models \phi(\vec{b})$. Now using Homework 14 above, we get that $\mathcal{M} \models (\phi(t/v_i))(\vec{a})$. This proves $\mathcal{M} \models (\forall x\phi \to \phi(t/x))(\vec{a})$ as desired. $\square$

Propositions 2.10 and 2.11 provide us with a lot of examples of logically valid formulas. As a matter of fact, there are not that many logically valid formulas other than these. We will see what this means when we prove the Completeness Theorem for first-order logic.

A special kind of set of sentences is the *theory* of a structure $\mathcal{M}$:

$$\mathrm{Th}(\mathcal{M}) := \{\sigma : \mathcal{M} \models \sigma\}.$$

This set has the following property: For every L-sentence σ either $\sigma \in \mathrm{Th}(\mathcal{M})$ or $\neg\sigma \in \mathrm{Th}(\mathcal{M})$; actually exactly one of them is in $\mathrm{Th}(\mathcal{M})$. We will come back to this property later when we are discussing basics of model theory.

2.1. Maps Between Structures. Let L be a language and $\mathcal{M}$ and $\mathcal{N}$ be L-structures; as usual the universes of $\mathcal{M}$ and $\mathcal{N}$ are M and N respectively. A function $F : M \to N$ is called a *homomorphism* if the following hold for every n-ary function symbol f, n-ary relation symbol R, constant symbol c and $\vec{a} = (a_1, \ldots, a_n) \in M^n$:

(1) $F(f^{\mathcal{M}}(\vec{a})) = f^{\mathcal{N}}(F(\vec{a}))$,
(2) If $\vec{a} \in R^{\mathcal{M}}$, then $F(\vec{a}) \in R^{\mathcal{N}}$,
(3) $F(c^{\mathcal{M}}) = c^{\mathcal{N}}$.

(Here and later, $F(\vec{a}) = (F(a_1), \ldots, F(a_n))$.)

We sometimes write '$F : \mathcal{M} \to \mathcal{N}$ is a homomorphism'.

Note that we only require that $F(R^{\mathcal{M}}) \subseteq R^{\mathcal{N}}$. Sometimes a homomorphism that has equality here instead of inclusion for every R is called a *strong homomorphism*. However, we do not need this definition except for the next definition: A strong homomorphism that is also injective is called an *embedding*. If an embedding is also surjective, then it is called an *isomorphism*. If there is an isomorphism between two structures $\mathcal{M}$ and $\mathcal{N}$, then we say that they are *isomorphic*, and we denote this as $\mathcal{M} \simeq \mathcal{N}$. An isomorphism of a structure $\mathcal{M}$ with itself is called an *automorphism*.

Homework 15. Let $F : \mathcal{M} \to \mathcal{N}$ be a homomorphism of $\mathcal{L}$-structures and let t be an $\mathcal{L}$-term whose variables are among $x_1, \ldots, x_n$. Show that for any $\vec{a} = (a_1, \ldots, a_n) \in M^n$, we have

$$t^{\mathcal{N}}(F(\vec{a})) = F(t^{\mathcal{M}}(\vec{a})).$$

Homework 16. Show that L-isomorphism gives an equivalence relation on the class of L-structures.

Homework 17. Show that the set of automorphisms of a structure forms a group with composition as the group operation.

Homework 18. Show that structures $\mathcal{M}_1$, $\mathcal{M}_2$, and $\mathcal{M}_3$ from Example 2.7 are not isomorphic to each other.

Suppose that $\mathcal{M}$ and $\mathcal{N}$ are structures such that $M \subseteq N$. If the inclusion map $M \to N$ is an embedding, then we say that $\mathcal{M}$ is a *substructure* of $\mathcal{N}$ and we denote this as $\mathcal{M} \subseteq \mathcal{N}$.

EXAMPLE 2.12. Let's study these concepts in structures of more algebraic content. Let $L_{\mathrm{r}} = \{+, \cdot, -, 0, 1\}$ and $L_{\mathrm{or}} = \{+, \cdot, -, <, 0, 1\}$ where $+, \cdot$ are binary function symbols, $-$ is a unary function symbol, $<$ is binary relation symbol, and $0, 1$ are constant symbols.

We may interpret $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$, and $\mathbb{C}$ as L_{r}-structures in a natural way, and all excepts $\mathbb{C}$ are naturally L_{or}-structures. Moreover, the smaller ones are substructures of bigger ones. However, they are very different structures in many ways. This difference will be captured by the next definition.

Homework 19. Suppose that $\mathcal{M} \subseteq \mathcal{N}$, and let σ be a sentence that does not have any occurrence of $\forall$. Show that $\mathcal{M} \models \sigma$ if and only if $\mathcal{N} \models \sigma$. (When we say '$\sigma$ does not contain $\forall$', we mean this in the original form of σ as an expression in the alphabet. So no conventions or shorthands are used; in particular there is no occurrence of $\exists$ either.)

2.2. Elementary Equivalence. Two L-structures $\mathcal{M}$ and $\mathcal{N}$ are *elementarily equivalent* if for every L-sentence σ, we have

$$\mathcal{M} \models \sigma \Longleftrightarrow \mathcal{N} \models \sigma.$$

This is denoted as $\mathcal{M} \equiv \mathcal{N}$.

We leave a few things about this concept as exercises. This concept will be revisited in the chapter on model theory.

Homework 20. Show that elementary equivalence is an equivalence relation on the class of L-structures.

PROPOSITION 2.13. *If $\mathcal{M} \simeq \mathcal{N}$, then $\mathcal{M} \equiv \mathcal{N}$.*

PROOF. HW... $\square$

PROPOSITION 2.14. *Let $\mathcal{M}$ and $\mathcal{N}$ be L-structures. Then $\mathcal{M} \equiv \mathcal{N}$ if and only if $\mathcal{M}$ is a model of* $\mathrm{Th}(\mathcal{N})$.

PROOF. It is clear that if $\mathcal{M} \equiv \mathcal{N}$, then $\mathcal{M} \models \mathrm{Th}(\mathcal{N})$. Suppose conversely that $\mathcal{M}$ is a model of $\mathrm{Th}(\mathcal{N})$. Let σ be a sentence. If $\mathcal{N} \models \sigma$, then $\sigma \in \mathrm{Th}(\mathcal{N})$, hence $\mathcal{M} \models \sigma$. If $\mathcal{N} \not\models \sigma$, then $\mathcal{N} \models \neg\sigma$ and hence $\neg\sigma \in \mathrm{Th}(\mathcal{N})$ and $\mathcal{M} \models \neg\sigma$. So $\mathcal{M} \not\models \sigma$. $\square$

Homework 21. Show that none of the pairs of the structures in the previous examples are elementarily equivalent.

3. Formal First-Order Theories

Let L be a first-order language. We will define formal theories whose alphabet is the alphabet of L and whose well-formed formulas are L-formulas. We will call them L-theories. We need to introduce the axioms and inference rules. There are many different choices of axioms and inference rules; we adapt Hilbert-style deductive system.

3.1. Axioms. Axioms of an L-theory T are as follows:

(A1) For every pair ϕ, ψ of L-formulas we have the axiom:

$$\phi \to (\psi \to \phi)$$

(A2) For every triple ϕ, ψ, θ of L-formulas we have the axiom:

$$(\phi \to (\psi \to \theta)) \to ((\phi \to \psi) \to (\phi \to \theta))$$

(A3) For every pair ϕ, ψ of L-formulas we have the axiom:

$$(\neg\phi \to \neg\psi) \to ((\neg\phi \to \psi) \to \phi)$$

(A4) If t is a term that is free for a variable x in a formula ϕ, then the following is an axiom:

$$\forall x\phi \to \phi(t/x)$$

(A5) If ϕ is an L-formula that has no free occurrence of a variable x, then the following is an axiom:

$$\forall x(\phi \to \psi) \to (\phi \to \forall x\psi)$$

(E1) $\forall x\, x = x$

(E2) If $\phi(x, y)$ is an L-formula, then the following is an axiom:

$$\forall x \forall y(x = y \to (\phi(x, x) \to \phi(x, y)))$$

(PA) We also have non-logical axioms called *proper axioms*, which are certain L-sentences. (There might be none, or there might be infinitely many of them.)

3.2. Inference Rules. We have two inference rules:

(MP) Modus Ponens: ψ follows from $\phi \to \psi$ and ϕ.

(Gen_x) Generalization: $\forall x\phi$ follows from ϕ.

Recall that we have defined the concepts of *proof*, *theorem*, and *deductive consequence* in the very general setting of formal theories. First-order theories are formal theories, hence all those concepts make sense. In particular, $\vdash_T \phi$ means that ϕ is a theorem of T.

Remarks. The first three axioms are familiar from Propositional Logic; indeed they are instances of axioms of Propositional Logic. Therefore they are all logically valid by Homework 13.

The axioms A4 and A5 are new and they deal with quantifiers. We know that they are logically valid by Proposition 2.11 and Proposition 2.10.

Homework 22. Show that any instance of a tautology of Propositional Logic is a theorem of any first-order theory.

The axioms E1 and E2 are intended to capture the fact that the symbol $=$ is always interpreted as equality. It is trivial to check that they are logically valid.

Homework 23. Write down the sentence stating that $=$ is an equivalence relation and show that it is a theorem of every first-order theory.

The proper axioms could be any set of formulas. If they are logically valid, then there is little reason to add them, because the Completeness Theorem for first-order theories will tell us that logically valid formulas are theorems that can be obtained only from non-proper axioms. On the other extreme, if they are not satisfied, then the theory can prove any formula; another consequence of the Completeness Theorem. So it is most interesting if the set of proper axioms has a model, but not every structure is a model of it.

The first inference rule, Modus Ponens (MP), is again familiar to us from the study of Propositional Logic. One may again show that MP applied to to logically valid formulas, it yields a logically valid formula. More generally, we have the following:

Homework 24. Let $\mathcal{M}$ be an $L-$structure and let ϕ and ψ be L-formulas. Suppose $\mathcal{M} \models \phi \to \psi$ and $\mathcal{M} \models \phi$. Show that $\mathcal{M} \models \psi$.

The inference rule, Generalization is also new. It reflects that satisfiability of a formula being defined as the satisfiability of its universal closure.

Homework 25. Let T be a first-order theory in a language L, ϕ an L-formula, and x a variable. Show that $\vdash_T \phi$ if and only if $\vdash_T \forall x \phi$.

3.3. Predicate Logic. Let L be a first-order language. The first example of an L-theory is the one with no proper axioms. It is generally referred to as the *Predicate Logic (in L)*. Let's denote this theory by T_L; note that it only depends on L.

Actually, predicate logic is more than an example. Suppose that T is an L-theory, and suppose that Γ is the set of its proper axioms. Then it is clear that $\vdash_T \phi$ if and only if $\Gamma \vdash_{T_L} \phi$ for every L-formula ϕ. Therefore we stop referring to first-order theories and just work with provability in the Predicate Logic. If the language L is clear from the set up, then we use $\vdash$ in the place of $\vdash_{T_L}$.

The next homework can be proven by putting together bits and pieces from the last few pages.

Homework 26. (Soundness Theorem) Let L be a first-order language, Γ a set of formulas, and ϕ an L-formula. Show that if $\Gamma \vdash_{T_L} \phi$, then $\Gamma \models \phi$. Conclude that theorems of T_L are logically valid.

Before starting to deduce theorems, let's prove a version of Deduction Theorem for first-order theories. First, an example illustrating that we cannot have the Deduction Theorem in form as simple as in the case of Propositional Logic.

Example 3.1. Let $L = \{+\}$ where $+$ is a binary function symbol. Let ϕ be the L-formula $v_1 + v_1 = v_1$. Then using generalization we have $\phi \vdash_{T_L} \forall v_1 \phi$. Let $\mathcal{M}$ be the L-structure whose universe is $\mathbb{Z}$ and $+$ is interpreted as the usual addition. Then, taking for instance $\vec{a} = (0, 0, \dots)$ we see that ϕ is satisfied at $\vec{a}$, and it is clear that $\mathcal{M} \not\models \forall v_1 \phi$. Therefore $\mathcal{M} \not\models \phi \to \forall v_1 \phi$. Hence by Homework 26 above, we have $\nvdash_{T_L} \phi \to \forall v_1 \phi$. So the Deduction Theorem cannot be correct as in the exact the same form in Propositional Logic. We need the following definition for the correct statement.

Definition 3.2. Let T be a first order theory, Γ be a set of formulas and $\phi \in \Gamma$ such that $\Gamma \setminus \{\phi\} \nvdash_T \phi$. Also let $\phi_1, \dots, \phi_n$ be a proof from Γ in T. We define ϕ_i *depending on ϕ in this proof* by induction on i as follows:

- Either ϕ_i is ϕ,
- or ϕ_i is obtained by applying MP to two earlier formulas in the proof such that at least one of those formulas depend on ϕ,
- or ϕ_i is obtained by applying Generalization to an earlier formula in the proof that depends on ϕ.

If $\Gamma \setminus \{\phi\} \vdash_T \phi$, then we say that *no proof from Γ (in T) depends on ϕ.*

PROPOSITION 3.3. *Let T be a first-order theory, Γ a set of formulas and ϕ, ψ formulas. Suppose that there is proof in T of ψ from $\Gamma \cup \{\phi\}$ that does not depend on ϕ. Then $\Gamma \vdash_T \psi$.*

PROOF. If $\Gamma \setminus \{\phi\} \vdash_T \phi$, then the result is clear. So assume $\Gamma \setminus \{\phi\} \nvdash_T \phi$, and let $\phi_1, \ldots, \phi_n = \psi$ be a proof of ψ from $\Gamma \cup \{\phi\}$ that does not depend on ϕ. We proceed by induction on n. If $n = 1$, then $\phi_1 = \psi$ and $\phi_1 \neq \phi$. So we have $\Gamma \vdash_T \psi$. By the inductive hypothesis, we know that $\Gamma \vdash_T \phi_i$ for each $i = 1, \ldots, n-1$. We also know that ϕ_n is either an axiom or in Γ or equal to ϕ or is obtained from previous ones by MP or Generalization. By assumption ϕ_n cannot be ϕ and in all the other cases, we have $\Gamma \vdash_T \phi_n$. □

THEOREM 3.4 (Deduction Theorem). *Let T be a first-order theory, Γ a set of formulas and ϕ, ψ formulas. Suppose that there is a proof of ψ from $\Gamma \cup \{\phi\}$ that does not contain an application of Gen_x to a formula that depends on ϕ, where x is a variable occurring in ϕ freely. Then $\Gamma \vdash_T \phi \to \psi$.*

PROOF. Let $\phi_1, \ldots, \phi_n$ be a proof of ψ from $\Gamma \cup \{\phi\}$ as in the statement. We prove, by induction on i that $\Gamma \vdash_T \phi \to \phi_i$ for $i = 1, \ldots, n$.

If $\phi_i = \phi$, then $\Gamma \vdash_T \phi \to \phi_i$ as $\phi \to \phi_i$ is an instance of a tautology of a proposition of Propositional Logic. If ϕ_i is an axiom or an element of Γ, then $\Gamma \vdash_T \phi \to \phi_i$ using the axiom $\phi_i \to (\phi \to \phi_i)$. This also covers the case of $i = 1$.

If ϕ_i is obtained by ϕ_j and ϕ_k by MP, then we get that $\Gamma \vdash_T \phi \to \phi_i$ using the inductive hypothesis and Axiom (A2).

The final case is that ϕ_i is $\forall x \phi_j$ for some $j < i$. Then by induction hypothesis, we have $\Gamma \vdash_T \phi \to \phi_j$.

There are two cases: Either ϕ_j does not depend on ϕ or x does not occur freely in ϕ.

If ϕ_j does not depend on ϕ, then $\Gamma \vdash_T \phi_j$ by the previous proposition and by Gen_x we have $\Gamma \vdash_T \forall x \phi_j$. This means that $\Gamma \vdash_T \phi_i$. Using (A1) and MP, we get $\Gamma \vdash_T \phi \to \phi_i$ as desired.

Suppose x does not occur freely in ϕ. Then we may use (A5) to deduce

$$\Gamma \vdash_T \forall x(\phi \to \phi_j) \to (\phi \to \forall x \phi_j).$$

We also have $\Gamma \vdash_T \forall x(\phi \to \phi_j)$ by Generalization. Therefore we have $\Gamma \vdash_T \phi \to \forall x \phi_j$ using MP. This means $\Gamma \vdash_T \phi \to \phi_i$. □

EXAMPLE 3.5. Let ϕ be an L-formula. Consider the L-formula $\forall x \forall y \phi \to \forall y \forall x \phi$. Here is a deduction of $\forall y \forall x \phi$ from $\forall x \forall y \phi$ (in the Predicate Logic of L):

(1) $\forall x \forall y \phi$ - Hypothesis

(2) $\forall x \forall y \phi \to \forall y \phi$ - (A4) (t is x)
(3) $\forall y \phi$ - MP
(4) $\forall y \phi \to \phi$ - (A4) (t is y)
(5) ϕ - MP
(6) $\forall x \phi$ - Gen_x
(7) $\forall y \forall x \phi$ - Gen_y

In this proof Generalization is applied with the variables x and y, both of which are bound in $\forall x \forall y \phi$. Hence Deduction Theorem can be applied to deduce that $\vdash_{T_L} \forall x \forall y \phi \to \forall y \forall x \phi$.

Remark. Analyzing the proof of the Deduction Theorem, we get some extra information. In the proof of $\phi \to \psi$ from Γ that is constructed has an application of Gen_x to a formula that depends on ϕ only when the same happens in the proof of ψ from $\Gamma \cup \{\phi\}$. Let's see this in an example.

Example 3.6. Consider the following deduction of $\forall x \psi$ (in T_L) from $\Gamma = \{\forall x(\phi \to \psi), \forall x \phi\}$:

(1) $\forall x(\phi \to \psi) \to (\phi \to \psi)$ - (A4)
(2) $\forall x(\phi \to \psi)$ - Hypothesis
(3) $\phi \to \psi$ - MP
(4) $\forall x \phi$ - Hypothesis
(5) $\forall x \phi \to \phi$ - (A4)
(6) ϕ - MP
(7) ψ - MP
(8) $\forall x \psi$ - Gen_x

In this proof, every formula after step 4 depends on $\forall x \phi$. However, the application of Generalization after that is only to the variable x, which does not have a free occurrence in $\forall x \phi$. So using the Deduction Theorem, we get

$$\forall x(\phi \to \psi) \vdash_{T_L} \forall x \phi \to \forall x \psi.$$

According to the remark above this new proof uses Generalization applied only to x. Once again, since x has no free occurrence in $\forall x(\phi \to \psi)$, using the Deduction Theorem we get

$$\vdash_{T_L} \forall x(\phi \to \psi) \to (\forall x \phi \to \forall x \psi).$$

We record the following obvious yet very useful consequence of the Deduction Theorem.

Corollary 3.7. *Let T be a first-order theory, Γ a set of formulas, σ a sentence, and ψ a formula. Suppose $\Gamma \cup \{\sigma\} \vdash_T \psi$. Then $\Gamma \vdash_T \sigma \to \psi$.*

We record two arguments that is used very frequently as "rules". (Recall that we simply write $\vdash$ in the place of $\vdash_{T_L}$.)

PROPOSITION 3.8. *Let ϕ be a formula, and let t be a term that is free for a variable x in ϕ.*

(1) *(Particularization A4)* $\forall x\phi \vdash \phi(t/x)$.
(2) *(Existential E4)* $\phi(t/x) \vdash \exists x\phi$.

PROOF. The following is a proof of the first one:

$$\forall x\phi \to \phi(t/x), \forall x\phi, \phi(t/x).$$

For the second one, it suffices to show that $\vdash \phi(t/x) \to \exists x\phi$. The formula

$$(\forall x\neg\phi \to \neg\phi(t/x)) \to (\phi(t/x) \to \neg\forall x\neg\phi)$$

is an instance of a tautology of Propositional Logic; hence is a theorem. The formula $\forall x\neg\phi \to \neg\phi(t/x)$ is (A4). So applying MP, we get $\phi(t/x) \to \neg\forall x\neg\phi$, which is $\phi(t/x) \to \exists x\phi$. □

It is important to note that neither of the proofs of these two rules use Generalization. So in many cases, we may use the Deduction Theorem to conclude more; as illustrated in the next example.

EXAMPLE 3.9. Let's show that $\vdash \forall x\phi \to \exists x\phi$. First note that the following is a deduction of $\exists x\phi$ from $\forall x\phi$:

$$\forall x\phi, \phi(\text{Rule A4}), \exists x\phi(\text{Rule E4}).$$

The Deduction Theorem is applicable to this proof; hence we get $\vdash \forall x\phi \to \exists x\phi$.

COROLLARY 3.10. *Let ϕ be a formula and let x be a variable. Then $\forall x\phi \vdash \phi$ and $\phi \vdash \exists x\phi$.*

PROOF. The variable x is always free for itself in any formula. □

COROLLARY 3.11. *Let Γ be a set of formulas and ϕ a formula. Then $\Gamma \vdash \phi$ if and only if $\Gamma \vdash \hat{\phi}$, where $\hat{\phi}$ is the universal closure of ϕ.*

PROOF. If $\Gamma \vdash \phi$, then we get $\Gamma \vdash \hat{\phi}$ by applying Generalization a few times. If $\Gamma \vdash \hat{\phi}$, then we get $\Gamma \vdash \phi$ by applying the previous corollary several times. □

COROLLARY 3.12. *For every formula ϕ, we have $\phi \vdash \hat{\phi}$ and $\hat{\phi} \vdash \phi$, where $\hat{\phi}$ is the universal closure of ϕ.*

PROOF. Apply the previous corollary twice. □

COROLLARY 3.13. *Let Γ be a set of formulas, ϕ a formula, $x_1, \ldots, x_n$ distinct variables and let $t_1, \ldots, t_n$ be terms whose variables do not occur bound in ϕ. If $\Gamma \vdash \phi$, then $\Gamma \vdash \phi(t_1/x_1, \ldots, t_n/x_n)$.*

PROOF. Exercise. (It is not totally trivial!) □

The next proposition collects some deduction rules together.

PROPOSITION 3.14. *Let ϕ, ψ, θ be formulas.*

(1) $\neg\neg\phi \vdash \phi$ *(Negation Elimination)*
(2) $\phi \vdash \neg\neg\phi$ *(Negation Introduction)*
(3) $\phi \wedge \psi \vdash \phi$ *(Conjunction Elimination)*
(4) $\phi, \psi \vdash \phi \wedge \psi$ *(Conjunction Introduction)*
(5) $\phi \vee \psi, \neg\psi \vdash \psi$ *(Disjunction Elimination)*
(6) $\neg(\phi \vee \psi) \vdash \neg\phi \wedge \neg\psi$ *(De Morgan 1)*
(7) $\neg(\phi \wedge \psi) \vdash \neg\phi \vee \neg\psi$ *(De Morgan 2)*
(8) $\phi \to \theta, \psi \to \theta, \phi \vee \psi \vdash \theta$
(9) $\phi \vdash \phi \vee \psi$ *(Disjunction Introduction)*
(10) $\phi \to \psi, \neg\psi \vdash \neg\phi$ *(Conditional Elimination 1)*
(11) $\phi \to \neg\psi, \psi \vdash \neg\phi$ *(Conditional Elimination 2)*
(12) $\neg\phi \to \psi, \neg\psi \vdash \phi$ *(Conditional Elimination 3)*
(13) $\neg\phi \to \neg\psi, \psi \vdash \phi$ *(Conditional Elimination 4)*
(14) $\neg(\phi \to \psi) \vdash \phi$ *(Conditional Elimination 5)*
(15) $\neg(\phi \to \psi) \vdash \neg\psi$ *(Conditional Elimination 6)*
(16) $\psi, \neg\psi \vdash \phi \to \psi$*(Conditional Introduction)*
(17) $\phi \to \psi \vdash \neg\psi \to \neg\phi$ *(Contrapositive 1)*
(18) $\neg\psi \to \neg\phi \vdash \phi \to \psi$ *(Contrapositive 2)*
(19) $\phi \leftrightarrow \psi, \phi \vdash \psi$ *(Biconditional Elimination 1)*
(20) $\phi \to \psi, \neg\phi \vdash \neg\psi$ *(Biconditional Elimination 2)*
(21) $\phi \leftrightarrow \psi, \psi \vdash \phi$ *(Biconditional Elimination 3)*
(22) $\phi \to \psi, \neg\psi \vdash \neg\phi$ *(Biconditional Elimination 4)*
(23) $\phi \to \psi, \psi \to \phi \vdash \phi \leftrightarrow \psi$ *(Biconditional Introduction)*
(24) $\phi \leftrightarrow \psi \vdash \neg\phi \leftrightarrow \neg\psi$ *(Biconditional Negation 1)*
(25) $\neg\phi \leftrightarrow \neg\psi \vdash \phi \leftrightarrow \psi$ *(Biconditional Negation 2)*
(26) *Suppose that there is a proof of $\psi \wedge \neg\psi$ from $\Gamma \cup \{\neg\phi\}$ that has no application of Generalization to a free variable of ϕ. Then $\Gamma \vdash \phi$. (Proof by Contradiction)*

PROOF. HW... □

LEMMA 3.15. *Let ϕ be a formula, and let $x_1, \ldots, y_1, \ldots, y_n$ be distinct variables. Then*

$$x_1 = y_1, \ldots, x_n = y_n \vdash (\phi(x_1, \ldots, x_n) \to \phi(y_1/x_1, \ldots, y_n/x_n)).$$

PROOF. We first show by induction on i that

$$(*) \quad x_1 = y_1, \ldots, x_n = y_n \vdash \phi(x_1, \ldots, x_n) \to \phi(y_1/x_1, \ldots, y_i/x_i).$$

The case that $i = 1$ follows from axiom E2 and Rule A4.

For the inductive step, by combining axiom E2 and Rule A4, we note that

$$x_{i+1} = y_{i+1} \vdash \phi(y_1/x_1, \ldots, y_i/x_i) \to \phi(y_1/x_1, \ldots, y_i/x_i, y_{i+1}/x_{i+1}).$$

Now using the tautology $((P \to Q) \wedge (Q \to R)) \to (P \to R)$, we get $(*)$ for $i + 1$.

□

COROLLARY 3.16. *Let ϕ be a formula, and let $x_1, \ldots, y_1, \ldots, y_n$ be distinct variables. Then*

$$(x_1 = y_1 \wedge \cdots \wedge x_n = y_n) \to (\phi(x_1, \ldots, x_n) \to \phi(y_1/x_1, \ldots, y_n/x_n))$$

is a theorem of Predicate Logic.

PROOF. By Conjugation Elimination applied to the lemma above, we get that

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \vdash \phi(x_1, \ldots, x_n) \to \phi(y_1/x_1, \ldots, y_n/x_n)$$

Note that this proof does not use Generalization. Hence we may apply the Deduction Theorem to conclude that

$$\vdash (x_1 = y_1 \wedge \cdots \wedge x_n = y_n) \to (\phi(x_1, \ldots, x_n) \to \phi(y_1/x_1, \ldots, y_n/x_n)).$$

$\square$

CHAPTER 3

Completeness Theorem and Its Applications

We have all the necessary tools to prove the Completeness Theorem. We state two version of it; the first one being the version we are used to. After proving their equivalence, we prove the second version.

Most of the proof of Completeness Theorem is adapted from [**1**].

1. Two Versions of Completeness Theorem

THEOREM 1.1 (Completeness Theorem – Version 1). *For a set Γ of formulas and a formula ϕ, if $\Gamma \vdash \phi$, then $\Gamma \models \phi$.*

In order to state the second version, we need the concept of consistency and some fact about it.

DEFINITION 1.2. A set Γ of formulas is *inconsistent* if there is a formula ϕ such that $\Gamma \vdash \phi$ and $\Gamma \vdash \neg\phi$; this is denoted as $\Gamma \vdash$. Otherwise it is called *consistent*.

LEMMA 1.3. *A set Γ of formulas is inconsistent if and only if $\Gamma \vdash \psi$ for every ψ.*

PROOF. It is clear that if Γ proves every formula, then it proves a formula and its negation. In order to prove the other implication, let $\Gamma \vdash \phi$, $\Gamma \vdash \neg\phi$, and let ψ be an arbitrary formula. Since it is an axiom, Γ proves the formula $(\neg\psi \to \phi) \to ((\neg\psi \to \phi) \to \psi)$. Similarly, Γ proves $\phi \to (\neg\psi \to \phi)$ and $\neg\phi \to (\neg\psi \to \neg\phi)$. Now by using Modus Ponens a few times, we get $\Gamma \vdash \psi$. $\square$

PROPOSITION 1.4. *Let Γ be set of formulas and ϕ a formula. Then $\Gamma \vdash \phi$ if and only if $\Gamma \cup \{\neg\phi\}$ is inconsistent.*

PROOF. Using Corollaries 3.11 and 3.12, it suffices to prove that $\Gamma \vdash \hat{\phi}$ if and only if $\Gamma \cup \{\neg\hat{\phi}\}$ is inconsistent, where $\hat{\phi}$ is the universal closure of ϕ.

Suppose that $\Gamma \vdash \hat{\phi}$. Then $\Gamma \cup \{\neg\hat{\phi}\} \vdash \hat{\phi}$. But we also have $\Gamma \cup \{\neg\hat{\phi}\} \vdash \neg\hat{\phi}$. Therefore $\Gamma \cup \{\neg\hat{\phi}\}$ is inconsistent.

Conversely, suppose $\Gamma \cup \{\neg\phi\}$ is inconsistent. Then by the previous lemma, $\Gamma \cup \{\neg\hat{\phi}\} \vdash \hat{\phi}$. Using the Deduction Theorem, we get that $\Gamma \vdash \neg\hat{\phi} \to \hat{\phi}$. We also have

$$\Gamma \vdash (\neg\hat{\phi} \to \neg\hat{\phi}) \to ((\neg\hat{\phi} \to \hat{\phi}) \to \hat{\phi})$$

and

$$\Gamma \vdash \neg\hat{\phi} \to \neg\hat{\phi}.$$

Now using MP twice, we get $\Gamma \vdash \hat{\phi}$. □

It is clear from Soundness Theorem (see Homework 26) that a satisfiable set Γ of formulas is consistent. Indeed, for a model $\mathcal{M}$ of Γ it is not possible that $\mathcal{M} \models \phi$ and $\mathcal{M} \models \neg\phi$. The second version of Completeness Theorem is the converse of this.

THEOREM 1.5 (Completeness Theorem – Version 2). *Every consistent set of formulas is satisfiable.*

We are going to prove this second version in detail, but first we prove that the versions are equivalent to each other.

PROPOSITION 1.6. *The following are equivalent:*

(1) *For every set Γ of formulas and formula ϕ, if $\Gamma \models \phi$, then $\Gamma \vdash \phi$.*
(2) *Every consistent set of formulas is satisfiable.*

PROOF. Suppose that condition (1) holds and let Γ be a set of formulas that does not have a model. Then $\Gamma \models \phi$ holds for every ϕ and hence $\Gamma \vdash \phi$ for every ϕ. So Γ is inconsistent.

Conversely, let's assume that every consistent set of formulas have a model. Suppose that $\Gamma \nvdash \phi$. Then by Proposition 1.4 we have that $\Gamma \cup \{\neg\phi\}$ is consistent. Suppose that $\mathcal{M} \models \Gamma \cup \{\neg\phi\}$. Then $\mathcal{M} \models \Gamma$, but $\mathcal{M} \not\models \phi$. Hence $\Gamma \not\models \phi$. □

2. Proof of the Completeness Theorem - V2

The idea of the proof is a great one and it is also simple. We take a consistent set Γ of formulas. We need to construct a model of Γ. The underlying set of that model will be the set of terms without variables modulo the following equivalence relation:

$$t_1 \sim t_2 \quad \Longleftrightarrow \quad \Gamma \vdash t_1 = t_2.$$

We still need to define the interpretation of symbols in the language. We do that in a similar way; for instance

$$(t_1/\sim, \ldots, t_n/\sim) \in R^{\mathcal{M}} \Longleftrightarrow \Gamma \vdash R(t_1, \ldots, t_n).$$

There are a lot of little details to check. For instance, we need to check that $\sim$ indeed gives an equivalence relation on the set of terms without variables and that $R^{\mathcal{M}}$ as above is well-defined. We do this below, in addition to giving the precise definitions. Note how this construction is in parallel with constructing free groups or abelian groups or groups with prescribed torsion.

LEMMA 2.1. *Let t_1, t_2, t_3 be terms. Then we have the following.*

(1) $\vdash t_1 = t_1$.
(2) $t_1 = t_2 \vdash t_2 = t_1$.
(3) $t_1 = t_2, t_2 = t_3 \vdash t_1 = t_3$.

PROOF. We begin by proving that the following are theorems of predicate logic:

$$x = x$$

$$x = y \to y = x$$

and

$$x = y \to (y = z \to x = z).$$

The first one is just Rule A4 applied to Axiom E1. Let $\phi(x, y)$ be $y = x$. Then rule A4 applied to axiom E2 twice gives

$$\vdash x = y \to (x = x \to y = x).$$

Then using MP, we get

$$x = y \vdash x = x \to y = x.$$

Then by the first part, we have $x = y \vdash y = x$. Hence using the Deduction Theorem, we get $\vdash x = y \to y = x$.

For the last part, let $\phi(x, y)$ be $y = z$. Then applying Rule A4 twice, we get

$$\vdash x = y \to (x = z \to \ y = z).$$

Let ψ be $x = y \to (x = z \to \ y = z)$. Then $\psi(y/x, x/y)$ is $y = x \to (y = z \to \ x = z)$. Then using the second part, we get

$$x = y \vdash y = z \to \ x = z.$$

Hence

$$\vdash x = y \to (y = z \to \ x = z).$$

The proof is concluded by applying Corollary 3.13 to appropriate formulas and terms. $\square$

LEMMA 2.2. *Let Γ be a set of formulas, $s_1, \ldots, , s_n, t_1, \ldots, t_n$ terms, and R is an n-ary relation symbol. Suppose that $\Gamma \vdash s_i = t_i$ for every i, and $\Gamma \vdash R(s_1, \ldots, s_n)$. Then $\Gamma \vdash R(t_1, \ldots, t_n)$.*

PROOF. Let $x_1, \ldots, x_n, y_1, \ldots, y_n$ be distinct variables not appearing in any of the terms s_i, t_i. By Corollary 3.16, we have

$$\vdash (x_1 = y_1 \wedge \cdots \wedge x_n = y_n) \to (R(x_1, \ldots, x_n) \to R(y_1/x_1, \ldots, y_n/x_n)).$$

Using Rule A4 several times we get

$$\vdash (s_1 = t_1 \wedge \cdots \wedge s_n = t_n) \to (R(s_1, \ldots, s_n) \to R(t_1, \ldots, t_n)).$$

By using Conjunction Introduction and MP several times we get the desired result. $\square$

LEMMA 2.3. *Let Γ be a set of formulas, $s_1, \ldots, , s_n, t_1, \ldots, t_n$ terms, and f is an n-ary function symbol. Suppose that $\Gamma \vdash s_i = t_i$ for every i. Then $\Gamma \vdash f(s_1, \ldots, s_n) = f(t_1, \ldots, t_n)$.*

Proof. Let $x_1, \dots, x_n, y_1, \dots, y_n$ be distinct variables not appearing in any of the terms s_i, t_i. By Lemma 3.15, we have that the formula

$$f(x_1, \dots, x_n) = f(x_1, \dots, x_n) \to f(x_1, \dots, x_n) = f(y_1, \dots, y_n).$$

follows from $x_1 = y_1, \dots, x_n = y_n$. Applying Rule A4 to axiom E1 with $t = f(x_1, \dots, x_n)$, we have $\vdash f(x_1, \dots, x_n) = f(x_1, \dots, x_n)$. Then using MP, we get

$$x_1 = y_1, \dots, x_n = y_n \vdash f(x_1, \dots, x_n) = f(y_1, \dots, y_n)$$

Then using Rule A4 several times we get the desired result. □

Now we are ready to make the following definition.

Definition 2.4. Let L be a language that contains at least one constant symbol and let Γ be a consistent set of L-formulas. We define an L-structure $\mathcal{M}_\Gamma$ as follows:

- The universe is $M_\Gamma := \mathrm{Term_L} / \sim$, where $\mathrm{Term_L}$ is the set of L-terms that do not contain variables and $s \sim t$ means $\Gamma \vdash s = t$.
- For an n-ary relation symbol R and $(t_1/\sim, \dots, t_n/\sim) \in M^n$:
  $$(t_1/\sim, \dots, t_n/\sim) \in R^{\mathcal{M}_\Gamma} \iff \Gamma \vdash R(t_1, \dots, t_n).$$
- For an n-ary function symbol f and $(t_1/\sim, \dots, t_n/\sim) \in M^n$:
  $$f^{\mathcal{M}_\Gamma}(t_1/\sim, \dots, t_n/\sim) = f(t_1, \dots, t_n)/\sim .$$

Homework 27. Let L, Γ, $\mathcal{M}_\Gamma$ be as in the definition above.

(1) Let $t \in \mathrm{Term_L}$. Show that $t^{\mathcal{M}_\Gamma} = t/\sim$.
(2) Let $t_1, t_2 \in \mathrm{Term_L}$. Show that $\Gamma \vdash t_1 = t_2$ if and only if $\mathcal{M}_\Gamma \models t_1 = t_2$.
(3) Let R an n-ary relation symbol and $t_1, \dots, t_n \in \mathrm{Term_L}$. Show that $\Gamma \vdash R(t_1, \dots, t_n)$ if and only if $\mathcal{M}_\Gamma \models R(t_1, \dots, t_n)$.
(4) Suppose that ϕ and ψ are formulas with no occurrences of $\forall$, whose free variables are among $x_1, \dots, x_n$. Also let $t_1, \dots, t_n \in \mathrm{Term_L}$ such that
   - $\Gamma \vdash \phi(t_1/x_1, \dots, t_n/x_n) \iff \mathcal{M}_\Gamma \models \phi(t_1/x_1, \dots, t_n/x_n)$, and
   - $\Gamma \vdash \psi(t_1/x_1, \dots, t_n/x_n) \iff \mathcal{M}_\Gamma \models \psi(t_1/x_1, \dots, t_n/x_n)$.

   Show that
   $$\Gamma \vdash \neg\phi(t_1/x_1, \dots, t_n/x_n) \iff \mathcal{M}_\Gamma \models \neg\phi(t_1/x_1, \dots, t_n/x_n)$$
   and
   $$\Gamma \vdash (\phi \to \psi)(t_1/x_1, \dots, t_n/x_n) \iff \mathcal{M}_\Gamma \models (\phi \to \psi)(t_1/x_1, \dots, t_n/x_n).$$

The kind of formulas considered in the second and third parts of this homework are the atomic sentences of L. So using also the last part, we may conclude that for any sentence σ with no occurrences of $\forall$, we have

$$\Gamma \vdash \sigma \Longleftrightarrow \mathcal{M}_\Gamma \models \sigma.$$

If this could have been proven for all sentences, then $\mathcal{M}_\Gamma$ would be a model of Γ. (Note that we may assume that Γ consists of sentences, by replacing each element with its universal closure.)

EXAMPLE 2.5. Let $L = \{+, -, 0\}$, where $+$ is a binary function symbol, $-$ is a unary function symbol, and 0 is a constant symbol. Let Γ consist of the following L-sentences:

- $\forall x \forall y \forall z((x + y) + z = x + (y + z))$
- $\forall x(x + 0 = x \wedge 0 + x = x)$
- $\forall x(x + (-x) = 0)$
- $\exists x \neg(x = 0)$.

It is clear that Γ has a model; as a matter of fact its models are exactly nontrivial abelian groups. In particular, it is consistent. Let's try to understand $\mathcal{M}_\Gamma$. The elements of $\mathrm{Term_L}$ are certain expressions in 0, $+$, $-$, and parentheses. It takes a moments thought to see that all those terms are equivalent to each other. So M_Γ consists of only one element, namely $0/\sim$. The functions are interpreted as follows:

$$0/\sim +^{\mathcal{M}_\Gamma} 0/\sim = 0/\sim,$$
$$-0/\sim = 0/\sim,$$
$$0^{\mathcal{M}_\Gamma} = 0/\sim$$

So $\mathcal{M}_\Gamma$ is the trivial abelian group. Hence even though it is a model of the first three axioms, $\mathcal{M}_\Gamma$ is not a model of Γ. the problem about the last axiom is that it is an existential formula and it does not have a Γ-witness as will be defined below.

If L does not contain any constant symbols, then $\mathrm{Term_L}$ is the empty set. Hence the structure $\mathcal{M}_\Gamma$ does not exist in that case. However, we may always add a constant into L to obtain a slightly larger language. Then Γ is still a set of formulas in that language and the next result states that the concept of an L-formula ϕ following from Γ remains the same in that language.

LEMMA 2.6. *Let L be a language and c a constant symbol not in L. Put $L_c = L \cup \{c\}$. Also let Γ be a set of L-formulas and $\phi(x)$ an L-formula such that $\Gamma \vdash_{T_{L_c}} \phi(c/x)$. Then $\Gamma \vdash_{T_L} \phi(x)$.*

PROOF. We simply write $\phi(c)$ rather than $\phi(c/x)$. Let $\psi_1, \ldots, \psi_k$ be a proof of $\phi(c)$ from Γ; so $\psi_k = \phi(c)$. Let y be a variable not appearing in this proof and for each i, let θ_i be the expression obtained by all occurrences of c in ψ_i by y. It is easy to see that each θ_i is an L-formula; note that $\theta_k = \phi(y/x)$. If ψ_i is an axiom (in the language L_c), then θ_i is an axiom (in the language L). If $\psi_i \in \Gamma$, then $\theta_i = \psi_i$. It is also clear that if ψ_i follows from ψ_j and ψ_l by MP, then θ_i follows from θ_j and θ_l by MP. Similarly if ψ_i follows from ψ_j by Generalization, then θ_i follows from θ_j by Generalization. Therefore $\theta_1, \dots, \theta_k$ is a proof of $\phi(y/x)$ from Γ in the language L. So $\Gamma \vdash_{T_L} \phi(y/x)$ and hence $\Gamma \vdash_{T_L} \phi(x)$. $\square$

In a while we will add more than one constant symbol to the language.

PROPOSITION 2.7 (Lindenbaum's Lemma). *Let* Γ *be a consistent set of sentences. Then there is a consistent set* Γ' *of sentences containing* Γ *such that for every sentence* σ*, either* $\Gamma' \vdash \sigma$ *or* $\Gamma' \vdash \neg\sigma$*.*

PROOF. Let $\mathfrak{P}$ be the collection of all consistent sets of sentences containing Γ and order it with set inclusion. It is non-empty as $\Gamma \in \mathfrak{P}$. If $\Gamma_1 \subseteq \Gamma_2 \subseteq \cdots$ is a chain of elements of $\mathfrak{P}$, then $\Gamma^* = \bigcup_{n>0} \Gamma_n$ is also an element of $\mathfrak{P}$, because if a formula ϕ has a proof from Γ^*, then it has a proof from Γ_n for some $n > 0$. So any chain in $\mathfrak{P}$ has an upper bound in $\mathfrak{P}$. Therefore, using Zorn's lemma we get that $\mathfrak{P}$ has a maximal element, say Γ'.

Suppose that $\Gamma' \nvdash \sigma$. Then by Proposition 1.4, $\Gamma' \cup \{\neg\sigma\}$ is consistent. Therefore $\neg\sigma \in \Gamma'$ by the maximality of Γ'. In particular, $\Gamma' \vdash \neg\sigma$ $\square$

DEFINITION 2.8. A set Γ of sentences is called *complete* if it is consistent and for every sentence σ either $\Gamma \vdash \sigma$ or $\Gamma \vdash \neg\sigma$.

So Lindenbaum's Lemma could be read as 'any consistent set of sentences can be extended to a complete set of sentences'.

DEFINITION 2.9. Let Γ be a set of sentences and let σ be a sentence of the form $\exists x\phi$. A variable-free term t is called a Γ-*witness* if $\Gamma \vdash \phi(t/x)$. The set Γ *has witnesses* if there is a Γ-witness for every sentence of the form $\exists x\phi$ with $\Gamma \vdash \exists x\phi$.

For instance, the last axiom of Example 2.5 is of the form above, but there is no Γ-witness for it. We need to go around this problem, so that $\mathcal{M}_\Gamma$ becomes a model of Γ. We shall do this by extending Γ.

The key to end the proof of Completeness Theorem will be the next result.

PROPOSITION 2.10. *Let* Γ *be a consistent set of sentences. Then the following are equivalent.*

(1) *For every sentence* σ*,* $\Gamma \vdash \sigma$ *if and only if* $\mathcal{M}_\Gamma \models \sigma$*.*
(2) Γ *is complete and has witnesses.*

PROOF. First assume (1). Let σ be a sentence. Then either $\mathcal{M}_\Gamma \models \sigma$ or $\mathcal{M}_\Gamma \models \neg\sigma$. Then either $\Gamma \vdash \sigma$ or $\Gamma \vdash \neg\sigma$. So Γ is complete. Now suppose that $\exists x\phi$ is a sentence with $\Gamma \vdash \exists x\phi$. By assumption, $\mathcal{M}_\Gamma \models \exists x\phi$. This mean that there is $t \in \mathrm{Term}_L$ such that $\mathcal{M}_\Gamma \models \phi(t/\sim)$. Using the assumption on the opposite direction, we get $\Gamma \vdash \phi(t/x)$.

Now assume (2). Let σ be a sentence. We would like to show that

$$\Gamma \vdash \sigma \Longleftrightarrow \mathcal{M}_\Gamma \models \sigma.$$

We proceed by induction on the complexity of σ. The case that σ is atomic is covered by the second and the third part of Homework 27, and the cases that σ is $\neg\tau$ and $\tau \to \rho$ are taken care of by the last part of the same homework. So let's assume that σ is of the form $\forall x\phi$. Note that the only possible free occurrence of a variable in ϕ can be of x.

Suppose $\Gamma \nvdash \forall x\phi$. Then $\Gamma \vdash \exists x\neg\phi$. Then by assumption, there is $t \in \mathrm{Term}_L$ such that $\Gamma \vdash \neg\phi(t/x)$. By induction this means that $\mathcal{M}_\Gamma \models \neg\phi(t/\sim)$. Therefore $\mathcal{M}_\Gamma \models \exists x\neg\phi$, which means $\mathcal{M}_\Gamma \not\models \forall x\phi$.

Conversely, suppose that $\mathcal{M}_\Gamma \not\models \forall x\phi$. So $\mathcal{M}_\Gamma \models \exists x\neg\phi$ and there is $t \in \mathrm{Term}_L$ with $\mathcal{M}_\Gamma \models \neg\phi(t/\sim)$. Once again, by the induction assumption, we have that $\Gamma \vdash \neg\phi(t)$. Then using Rule E4, we have $\Gamma \vdash \exists x\neg\phi$. Then $\Gamma \nvdash \forall x\phi$. □

Homework 28. Suppose that Γ proves sentences $\exists x_1\phi_1, \ldots, \exists x_n\phi_n$. Let $L' = L \cup \{c_1, \ldots, c_n\}$, where $c_1, \ldots, c_n$ are pairwise distinct constant symbols not in L and consider the set $\Gamma' = \Gamma \cup \{\phi_1(c_1/x_1), \ldots, \phi_n(c_n/x_n)\}$ of L'-sentences. Show that Γ' is consistent. (*You might want to use Lemma 2.6.*)

Next we generalize this homework to infinitely many sentences of the form $\exists x\phi$; indeed to all sentences of that form. Let Δ be the set of all sentences σ of the form $\exists x\phi$ such that $\Gamma \vdash \sigma$ and let

$$L' = L \cup \{c_\sigma : \sigma \in \Delta\},$$

where each c_σ is a new constant symbol and for distinct σ, τ we have $c_\sigma \neq c_\tau$. Finally, let

$$\Gamma' = \Gamma \cup \{\phi(c_\sigma) : \sigma \in \Delta \text{ and } \sigma \text{ is } \exists x\phi\}.$$

LEMMA 2.11. *The set Γ' is consistent.*

PROOF. Suppose that Γ' proves ϕ and $\neg\phi$. Then there are $\sigma_1, \ldots, \sigma_k$ from Δ such that $\Gamma \cup \{\phi(c_{\sigma_1}, \ldots, \phi(c_{\sigma_k})\}$ proves both ϕ and $\neg\phi$. However, by the previous homework, the set $\Gamma \cup \{\phi(c_{\sigma_1}, \ldots, \phi(c_{\sigma_k})\}$ is consistent. This contradiction finishes the proof. □

Using the idea of this proof, one may prove the following result.

LEMMA 2.12. *Let $L_1 \subseteq L_2 \subseteq \ldots$ be a countable chain of first-order languages, also for each $n >$ let Γ_n be a consistent set of L_n-sentences such that $\Gamma_m \subseteq \Gamma_n$ for $m < n$. Put*

$$L^* = \bigcup_{n>0} L_n \text{ and } \Gamma^* = \bigcup_{n>0} \Gamma_n.$$

Then Γ' is consistent.

PROOF. HW... □

Now we are ready to prove the second version of the Completeness Theorem.

PROOF OF COMPLETENESS THEOREM - V2. We will extend Γ to a complete set of sentences which also has witnesses. We do this in countably many steps: the odd steps are for completeness and the even steps are for witnesses. More precisely, by induction, we define a language L_n and a consistent set Γ_n of L_n-sentences for each $n \in \mathbb{N}$.

Let $L_0 = L$ and let $\Gamma_0 = \Gamma$.

Suppose $L_0 \subseteq L_1 \subseteq \cdots \subseteq L_n$ and $\Gamma_0 \subseteq \Gamma_1 \subseteq \cdots \subseteq \Gamma_n$ are already constructed. First let n be odd. Then let $L_{n+1} = L_n$ and let Γ_{n+1} be a complete set of L_{n+1}-sentences extending Γ_n, whose existence is guaranteed by Proposition 2.7. If n is even, then let $L_{n+1} = L'_n$ and $\Gamma_{n+1} = \Gamma'_n$ which are given by Lemma 2.11 and the definitions preceding it, with L and Γ replaced by L_n and Γ_n respectively.

Let $L^* = \bigcup_{n>0} L_n$ and $\Gamma^* = \bigcup_{n>0} \Gamma_n$. Then Γ^* is consistent by the previous lemma. We claim that it is also complete and has witnesses.

For completeness, let σ be an L^*-sentence. So σ is an L_{2n+1} sentence for some n. Since Γ_{2n+1} is complete, either $\Gamma_{2n+1} \vdash \sigma$ or $\Gamma_{2n+1} \vdash \neg\sigma$. Therefore either $\Gamma^* \vdash \sigma$ or $\Gamma^* \vdash \neg\sigma$.

Now let σ be an L^*-sentence of the form $\exists x \phi$ such that $\Gamma^* \vdash \sigma$. Then σ is an L_{2n}-sentence for some n and $\Gamma_{2n} \vdash \sigma$. Then $\Gamma_{2n} \vdash \phi(c_\sigma)$ and hence $\Gamma^* \vdash \phi(c_\sigma)$.

By Proposition 2.10, we have that for every L^*-sentence σ we have

$$\Gamma^* \vdash \sigma \iff \mathcal{M}_{\Gamma^*} \models \sigma.$$

Note that we may consider $\mathcal{M}_{\Gamma^*}$ as an L-structure, let's call that structure $\mathcal{M}$. Now for an L-sentence $\sigma \in \Gamma$, we have $\sigma \in \Gamma^*$. Hence $\mathcal{M}_{\Gamma^*} \models \sigma$ and consequently $\mathcal{M} \models \sigma$. Thus $\mathcal{M}$ is a model of Γ. □

Homework 29. Show that a consistent set Γ of sentences is complete if and only if any two models of Γ are elementarily equivalent.

3. Compactness Theorem

Our first application of Completeness Theorem is the Compactness Theorem, which has tons of applications itself.

THEOREM 3.1 (Compactness Theorem). *Let Γ be a set of L-sentences. Then there Γ has a model if and only if every finite subset of Γ has a model.*

Proof. It is clear that if Γ has a model, then that model is a model of any subset of Γ. So let's assume that every finite subset of Γ has a model and show that Γ itself has a model.

Suppose not. Then by the second version of the Completeness Theorem, Γ is inconsistent. This means that there is an L-sentence σ such that $\Gamma \vdash \sigma$ and $\Gamma \vdash \neg\sigma$. In both cases, the proofs involves only finitely many hypotheses from Γ. Therefore there is a finite subset of Γ which is inconsistent. But then using the second version of the Completeness Theorem on the reverse direction, we get that that finite subset of Γ does not have a model, contradicting the assumption. Hence Γ has a model. $\square$

Remark. There is a possible confusion about the Compactness Theorem akin to the usual confusion in the proof of infinitude of primes that is attributed to Euclid. Namely, the models of finite subsets of Γ might not be models of Γ. Actually, generally they are not, otherwise we probably do not need the Compactness Theorem. The next example illustrates this; none of the models of the finite parts work for our purposes.

Example 3.2. Consider the structure $\mathcal{R} = (\mathbb{R}, +, \cdot, -, <, 0, 1)$ in the language $L_{\text{or}} = \{+, \cdot, -, <, 0, 1\}$ of *ordered rings*. We will show that there is $\mathcal{R}^*$ that is elementarily equivalent to $\mathcal{R}$ and there is $\alpha \in R^*$ which is positive and less than any $1/n$ for any $n > 0$. Obviously, there is not such element in $\mathbb{R}$, and it might sound like this contradicts elementary equivalence of $\mathcal{R}$ and $\mathcal{R}^*$. However, it only shows that containing such an element cannot be expressed with an L_{or}-sentence.

Let $L^* = L_{\text{or}} \cup \{c\}$ where c is a new constant symbol. Consider the following set of $L_{\text{or},c}$-sentences:

$$\Gamma := \text{Th}(\mathcal{R}) \cup \{0 < c\} \cup \{c < \frac{1}{n} : n > 0\}.$$

Note that if $\mathcal{R}_c^* = (R^*, +, \cdot, -, <, 0, 1, c^{\mathcal{R}_c})$ is a model of Γ, then the L_{or}-structure $\mathcal{R}^* = (R^*, +, \cdot, -, <, 0, 1)$ satisfies the condition we ask for with $\alpha = c^{\mathcal{R}_c}$.

So we need to show that any finite subset Δ of Γ has a model. Write $\Delta = \Delta_1 \cup \Delta_2$, where $\Delta_1 \subseteq \text{Th}(\mathcal{R})$ and $\Delta_2 \subseteq \{0 < c\} \cup \{c < \frac{1}{n} : n > 0\}$. Since Δ_2 is finite, it is subset of $\{0 < c\} \cup \{c < \frac{1}{1}, c < \frac{1}{2}, \ldots, c < \frac{1}{N}\}$ for some $N > 0$. Now it is clear that $(\mathbb{R}, +, \cdot, -, <, 0, 1, \frac{1}{N+1})$ is a model of Δ.

Note that the universe of the models of each finite subset of Γ is $\mathbb{R}$. So none of those structures can be a model of Γ.

Another application of the Compactness Theorem is that we can find arbitrarily large models of a consistent sets of sentences.

THEOREM 3.3 (Upward Löwenheim-Skolem Theorem). *Let* Γ *be a set of sentences that has an infinite model, and let* κ *be a cardinal number. Then there is a model* $\mathcal{M}$ *of* Γ *with* $|M| \geq \kappa$.

PROOF. Let C be a set of cardinality κ disjoint from L. We add elements of C to L as constant symbols to get the language L'. Consider the following set of L'-sentences:

$$\Gamma' = \Gamma \cup \{\neg c = d : c, d \in C, c \neq d\}.$$

Clearly it suffices to show that Γ' has a model. So let Δ be a finite subset of Γ'. Then only finitely many sentences of the form $\neg c = d$ appear in Δ. Taking an infinite model of Γ, we may interpret the new constants in a way that all those sentences are correct. So Δ has a model and consequently so does Γ'. □

4. Downward Löwenheim-Skolem Theorem and Vaught's Test

Here we are going to show that any consistent set of sentences has a models as small as possible; the lower limit being the cardinality of the language. Combining that the last result of the previous section, we get that any consistent set of sentences has a model of any cardinality larger than or equal to the cardinality of the language.

THEOREM 4.1 (Downward Löwenheim-Skolem). *Suppose that* Γ *is a consistent set of* L*-sentences. Then there is a model of* Γ *of cardinality at most* $|L|$.

PROOF. We analize the proof of the completeness theorem. Each of the languages L_n constructed in that proof have the same cardinality as $|L|$, because at each step either we do not extend the language or we extend it by a set of constants of the same cardinality. Therefore the language L^* has the same cardinality as L.

The model $\mathcal{M}_{\Gamma^*}$ of Γ^* constructed in that proof has cardinality at most the cardinality of the set of L^*-terms, which is the same the cardinality of L^*. Therefore the cardinality of $\mathcal{M}_{\Gamma^*}$ is at most $|L|$ as desired. □

Putting the two Löwenheim-Skolem Theorems together we get the following consequence.

COROLLARY 4.2. *Let* Γ *be a set of* L*-sentences that has an infinite model and let* $\kappa \geq |L|$ *be a cardinal number. Then* Γ *has a model of cardinality* κ.

PROOF. Let C a set of cardinality κ disjoint from L, and let $L' = L \cup C$, where elements of C are added as constant symbols. Note that L' has cardinality κ. Then by the proof of Theorem 3.3, there is a model of

$$\Gamma' = \Gamma \cup \{\neg c = d : c, d \in C, c \neq d\}.$$

So Γ' is consistent and applying Theorem 4.1, Γ' has a model $\mathcal{M}$ of cardinality at most $|L'| = \kappa$. However models of Γ' has at least κ many elements. Therefore $\mathcal{M}$ has cardinality κ. $\square$

Corollary 4.3 (Vaught's Test). *Let κ be a cardinal number and L a language of cardinality at most κ. Also let Γ be a consistent set of L-sentences that does not have finite models. Suppose that any two models of Γ of cardinality is κ are isomorphic. Then any two models of Γ are elementarily equivalent.*

Proof. Let $\mathcal{M}$ and $\mathcal{N}$ be models of Γ. By Proposition 2.14, we need to show that $\mathrm{Th}(\mathcal{M}) = \mathrm{Th}(\mathcal{N})$. Being consistent theories both $\mathrm{Th}(\mathcal{M})$ and $\mathrm{Th}(\mathcal{M})$ have models of cardinality κ by the previous corollary; say $\mathcal{M}'$ and $\mathcal{N}'$ respectively. By assumption $\mathcal{M}' \simeq \mathcal{N}'$. So $\mathcal{M}' \equiv \mathcal{N}'$ by Proposition 2.13. Therefore

$$\mathcal{M} \equiv \mathcal{M}' \equiv \mathcal{N}' \equiv \mathcal{N}.$$

$\square$

Homework 30. Let $L = \{<\}$, where $<$ is a binary relation. Show that $(\mathbb{R}, <) \equiv (\mathbb{Q}, <)$ where $<$ is interpreted as the natural ordering in both structures.

Homework 31. Let $L = \{+, -, 0\}$, where $+$ is a binary function symbol, $-$ is a unary function symbol, and 0 is a constant symbol. Show that the L-structures $(\mathbb{R}, +, -, 0)$, $(\mathbb{Q}, +, -, 0)$, and $(\mathbb{C}, +, -, 0)$ are elementarily equivalent to each other.

CHAPTER 4

Incompleteness Theorem – A Very Brief Introduction

This last part aims to give some idea about the famous Incompleteness Theorem of Gödel. It is far from being complete in many respects, but the biggest gap is that most of the proofs are omitted. The outline of this part is also borrowed from [**1**], and the details of proofs can be found there.

We start by presenting a simpler version of Incompleteness Theorem. For that purpose, we define the language $L(\underline{\mathrm{N}}) = \{s, +, \cdot, <, 0\}$, where s is a unary function symbol, $+$ and $\cdot$ are binary function symbols, $<$ is a binary relation symbol, and 0 is a constant symbol. The $L(\underline{\mathrm{N}})$-structure we are interested in is $\mathcal{N} = (\mathbb{N}, s, +, \cdot, <, 0)$, where s is the successor function, namely $s(n) = n + 1$, and the other symbols are interpreted in the most natural ways.

Theorem 0.1 (Incompleteness Theorem - Simple Version). *Suppose that* $\Gamma \subseteq \mathrm{Th}(\mathcal{N})$ *is a* computable *set of* $L(\underline{\mathrm{N}})$*-sentences. Then there is* $\sigma \in \mathrm{Th}(\mathcal{N})$ *such that* $\Gamma \not\vdash \sigma$*. In particular,* $\mathrm{Th}(\mathcal{N})$ *is not computable.*

In this statement, the word *computable* is the only concept we have not covered yet. Indeed, we shall spend quite some time understanding this term.

1. Computable Functions

The class of *computable functions* $\mathbb{N}^n \to \mathbb{N}$ (for various $n > 0$) is obtained by applying the following recursively:

(R1) (Basic Functions) $+ : \mathbb{N}^2 \to \mathbb{N}$, $\cdot : \mathbb{N}^2 \to \mathbb{N}$, $\chi_{\leq} : \mathbb{N}^2 \to \mathbb{N}$, $\pi_i^n : \mathbb{N}^n \to \mathbb{N}$ are computable functions. (Here and later, for a subset R of $\mathbb{N}^n$, the function χ_R is the characteristic function of R; that is it gets value 1 at elements of R and 0 otherwise. The π_i^n is the projection of $\mathbb{N}^n$ onto the i^{th} coordinate.)

(R2) (Composition) Suppose that $G : \mathbb{N}^m \to \mathbb{N}$ and $H_1, \ldots, H_m : \mathbb{N}^n \to \mathbb{N}$ are computable functions, then $G(H_1, \ldots, H_n)$ is a computable function.

(R3) (Smallest Element Satisfying a Certain Property) Let $G : \mathbb{N}^{n+1} \to \mathbb{N}$ be a computable function such that for every $a \in \mathbb{N}^n$, there is $x \in \mathbb{N}$ with $G(a, x) = 0$. Suppose that $F(a)$ is the smallest $x \in \mathbb{N}$ such that $G(a, x) = 0$. Then $F : \mathbb{N}^n \to \mathbb{N}$ is a computable function. (We write $\mu x(\dots)$ to denote the smallest element x such that ... happens.)

We say $R \subseteq \mathbb{N}^n$ is *computable* of χ_R is a computable function.

EXAMPLE 1.1. Constant functions are computable. For instance, $\mathbb{N}^n \to \ a \mapsto 0$ is $\mu x(\pi_{n+1}^{n+1}(a, x) = 0)$. Other constant functions can be constructed in a recursive way.

EXAMPLE 1.2. If $P, Q \subseteq \mathbb{N}^n$ are computable then so are $P \cap Q$, $P \cup Q$, $\mathbb{N}^n \setminus P$. We sometimes denote these sets as $P \wedge Q$, $P \vee Q$, $\neg P$. This way we can define the sets $P \to Q$ and $P \leftrightarrow Q$, which are also computable.

EXAMPLE 1.3. All the relations $<, \leq, >, \geq, =, \neq$ are computable. For instance $\chi_{\geq}(x, y) := \chi_{\leq}(y, x)$ and $\chi_{=}$ is the product of $\chi_{\leq}$ and $\chi_{\geq}$.

EXAMPLE 1.4 (Definition by Cases). Let $G_1, \dots, G_t$ be computable functions on $\mathbb{N}^n$, and let $R_1, \dots, R_t$ be definable subsets of $\mathbb{N}^n$ such that

$$R_1 \vee \dots \vee R_t = \mathbb{N}^n \text{ and } R_i \wedge R_j = \emptyset \text{ for all } i \neq j.$$

Define $G := G_1 \cdot \chi_{R_1} + \dots + G_t \cdot \chi_{R_t}$. Then G is clearly a computable function. We say that G is *defined by cases*; here the cases are R_i's and for a given i, G is defined to be G_i in the case of R_i.

EXAMPLE 1.5. Let $R \subseteq \mathbb{N}^{n+1}$ be computable. If for each $\in \mathbb{N}^n$, there is x with $R(a, x)$, then $\mu x(R(a, x))$ is a computable function of $a \in \mathbb{N}^n$. Given $a \in \mathbb{N}^n$ and $b \in \mathbb{N}$ define $\mu x_{<b}(R(a, x))$ to be the smallest x with $R(a, x) \wedge x < b$ if such x exists, and to be b is no such x exists. Then $\mu x_{<b}(R(a, x))$ is computable. The sets

$$\exists \boldsymbol{x}_{<b} R(a, b) := \{a \in \mathbb{N}^n : R(a, x) \text{ for some } x < b\}$$
$$\forall \boldsymbol{x}_{<b} R(a, b) := \{a \in \mathbb{N}^n : R(a, x) \text{ for all } x < b\}$$

are also computable.

EXAMPLE 1.6. The function $\dot{-} : \mathbb{N}^2 \to \mathbb{N}$ defined as

$$a \dot{-} b := \begin{cases} a - b & \text{if } a \geq b \\ 0 & \text{if } a < b. \end{cases}$$

is computable.

EXAMPLE 1.7 (Pairing Function). The function Pair : $\mathbb{N}^2 \to \mathbb{N}$ defined as

$$\text{Pair}(a, b) = \frac{(a + b)(a + b + 1)}{2} + a$$

is clearly a computable function. What is less clear is that it is indeed a bijection between $\mathbb{N}^2$ and $\mathbb{N}$. This function will be referred to many times later, because it will be useful in reducing many problems in $\mathbb{N}^n$ down to $\mathbb{N}$.

We may define the following functions using Pair: Suppose that $a = \mathrm{Pair}(b, c)$, then

$$\mathrm{Left}(a) := b, \text{ and } \mathrm{Right}(a) := c.$$

Note that both of these are really functions since Pair is a bijection. It is easy to see that they are both computable, and that $\mathrm{Left}(a) < a$ and $\mathrm{Right}(a) < a$ for $a \neq 0$.

Example 1.8. The relation R defined as

$$R(a, b, c) \iff a \equiv b \mod c$$

is computable. Just note that

$$R(a, b, c) \iff \exists \boldsymbol{x}_{<a+1}(a = xc + b) \vee \exists \boldsymbol{x}_{<b+1}(b = xc + a).$$

Example 1.9 (Gödel's β-function). Define

$$\beta(a, i) := \mu x(x \equiv \mathrm{Left}(a) \mod 1 + (i + 1)\,\mathrm{Right}(a)).$$

Clearly, $\beta : \mathbb{N}^2 \to \mathbb{N}$ is computable. Note that $\beta(a, i)$ is the remainder when we divide $\mathrm{Left}(a)$ by $1 + (i + 1)\,\mathrm{Right}(a)$.

It is easy to see that $\beta(a, i) \leq a \dot{-} 1$ for all a. A less straightforward result is the following.

Proposition 1.10. *Let $a_0, a_1, \ldots, a_n \in \mathbb{N}$. Then there is $a \in \mathbb{N}$ such that $\beta(a, i) = a_i$ for all i.*

Proof. Take $N \in \mathbb{N}$ such that $N > a_i$ for all i and every prime divisor of n divides N. Then $1 + N, 1 + 2N, \ldots, 1 + (n + 1)N$ are pairwise relatively prime. By Chinese Remainder Theorem, there is $M \in \mathbb{N}$ such that $M \equiv a_i \mod 1 + (i + 1)N$ for all i. Now it is easy to see that $a = \mathrm{Pair}(M, N)$ works. □

Example 1.11. Consider the computable function $f : \mathbb{N} \to \mathbb{N}$ defined as follows:

$$f(n) := \mu x(\beta(x, 0) = 1 \wedge \forall \boldsymbol{i}_{<n} \beta(x, i + 1) = 2\beta(x, i)).$$

Let $a_i = 2^i$ for $i = 0, 1, \ldots, n$. Then by the proposition above, there is $a \in \mathbb{N}$ such that $\beta(a, i) = a_i$ for each i. However, it is also clear that $f(n) = a$ as well. So

$$2^n = \beta(a, n) = \beta(f(n), n)$$

for all n. Hence the function $n \mapsto 2^n$ is computable.

Using the β-function we can may attach a natural number value to finite sequences as follows.

Definition 1.12. Given $n > 0$, we define $\langle \cdot \rangle : \mathbb{N}^n \to \mathbb{N}$ by

$$\langle a_1, \ldots, a_n \rangle = \mu\, x(\beta(x, 0) = n \wedge \forall \boldsymbol{i}_{\leq n}(\beta(x, i) = a_i)).$$

Clearly, this is a computable function. Note that we use exact the same notation for all $n > 0$. By definition $\beta(\langle a_1, \ldots, a_n\rangle, 0)$ gives us the *length* of the sequence.

Homework 32. Show that the subset of $\mathbb{N}$ consisting of $\langle a_1, \ldots, a_n\rangle$ for various $n > 0$ and $a_1, \ldots, a_n \in \mathbb{N}$ is computable.

Definition 1.13. Let $A : \mathbb{N}^n \to \mathbb{N}$ and $B : \mathbb{N}^{n+2} \to \mathbb{N}$ be given. Define $F : \mathbb{N}^{n+1} \to \mathbb{N}$ be defined as follows:

$$F(a, 0) = A(a), \quad F(a, b+1) = B(a, b, F(a, b)).$$

This F is said to be defined by *primitive recursion* from A and B.

It not totally trivial, but still not hard to see that if A, B are computable, then so is F. For instance, let $A(a) = 1$ and $B(a, b, c) = a \cdot c$. Then the corresponding F sends (a, b) to a^b. Hence $(a, b) \mapsto a^b$ is computable.

2. Representable Functions

Now we make the connection with the concept of computable functions and first-order logic.

Definition 2.1. Let $\underline{\mathrm{N}}$ be the following set of $L(\underline{\mathrm{N}})$-sentences:

- $\forall x\, S(x) \neq 0$
- $\forall x \forall y\, S(x) = S(y) \to x = y$
- $\forall x\, x + 0 = x$
- $\forall x \forall y\, x + S(y) = S(x + y)$
- $\forall x\, x \cdot 0 = 0$
- $\forall x \forall y\, x \cdot S(y) = x \cdot y + x$
- $\forall x\, x \not< 0$
- $\forall x \forall y (x < S(y) \to (x < y \vee x = y))$
- $\forall x \forall y (x < y \vee x = y \vee y < x)$

Let $\mathcal{M} \models \underline{\mathrm{N}}$. Then there is an embedding $\iota : \mathcal{N} \to \mathcal{M}$ given by $n \mapsto S^n(0)$. This embedding has the property that $a < \iota(n)$, then $a = \iota(m)$ for some $m < n$. It follows that if $a \notin \iota(\mathbb{N})$, then $\iota(n) < a$ for all $n \in \mathbb{N}$.

Any language containing $L(\underline{\mathrm{N}})$ is called a *numerical language.* So we can express $\underline{\mathrm{N}}$ in any numerical language.

Definition 2.2. Let Γ be a set of L-sentences, where L is a numerical language. We say that $R \subseteq \mathbb{N}^m$ is Γ-*representable* if there is an L-formula $\phi(x_1, \ldots, x_m)$ such that for all $(a_1, \ldots, a_m) \in \mathbb{N}^n$ we have

$$(a_1, \ldots, a_m) \in R \Longrightarrow \Gamma \vdash \phi(S^{a_1}(0), \ldots, S^{a_m}(0)),$$

$$(a_1, \ldots, a_m) \notin R \Longrightarrow \Gamma \vdash \neg\phi(S^{a_1}(0), \ldots, S^{a_m}(0)).$$

A function $F : \mathbb{N}^m \to \mathbb{N}$ is *Γ-representable* if there is an L-formula $\phi(x_1, \dots, x_m, y)$ such that for every $(a_1, \dots, a_m) \in \mathbb{N}^m$ we have

$$\Gamma \vdash \phi(S^{a_1}(0), \dots, S^{a_m(0)}, y) \leftrightarrow y = S^{F(a_1,\dots,a_m)}(0).$$

The next theorem makes the connection with computability. It is a very important result for the rest of these notes, but we do not present a proof.

THEOREM 2.3. *Let $F : \mathbb{N}^m \to \mathbb{N}$ be given. Then F is $\underline{\mathrm{N}}$-representable if and only if F is computable.*

It is straightforward to show that computable functions are $\underline{\mathrm{N}}$-representable, but the proof of the other implication is quite lengthy and technical.

3. Gödel Numbering

Recall that the version of Incompleteness Theorem we have stated above has the assumption that a set of sentences is computable. So far we defined the term computable only for subsets of $\mathbb{N}^n$. Now we make the connection: we enumerate formulas in a countable language in a way that we could start to see sets of formulas as subsets of $\mathbb{N}^n$ for some n.

Let L be a countable first-order alphabet. (Here we return to using the word 'alphabet', because it will be important to enumerate the logical symbols. However, we still use the letter L to denote it.) As a matter of fact, we assume that L has only finitely many nonlogical symbols. We first define an injective map, called $\mathcal{S}$ from L into $\mathbb{N}$ as follows:

$$\mathcal{S}(v_i) = 2i, \mathcal{S}(\to) = 1, \mathcal{S}(\neg) = 3, \mathcal{S}(() = 5,$$
$$\mathcal{S}()) = 7, \mathcal{S}(\forall) = 9, \mathcal{S}(=) = 11,$$

and we assign the rest of the odd numbers to the non-logical symbols in L.

We would like to extend $\mathcal{S}$ to all formulas. As usual, we do this by induction on the complexity of formulas. We first do it for terms:

$$\ulcorner v_i \urcorner = \mathcal{S}(v_i),$$
$$\ulcorner c \urcorner = \mathcal{S}(c),$$
$$\ulcorner f(t_1, \dots, f_n) \urcorner = \langle \mathcal{S}(f), \mathcal{S}((), \ulcorner t_1 \urcorner, \dots, \ulcorner t_n \urcorner, \mathcal{S}()) \rangle.$$

We extend this definition to formulas in the same way. We present how to do it for atomic formulas anyway.

$$\ulcorner R(t_1, \dots, t_n) \urcorner = \langle \mathcal{S}(R), \mathcal{S}((), \ulcorner t_1 \urcorner, \dots, \ulcorner t_n \urcorner, \mathcal{S}()) \rangle,$$
$$\ulcorner t_1 = t_2 \urcorner = \langle \mathcal{S}(=), \ulcorner t_1 \urcorner, \ulcorner t_2 \urcorner \rangle.$$

This way, we assigned a natural number value to every L-formula; this is called *Gödel Number* of the formula. The set $\{\ulcorner v_i \urcorner : i = 1, \dots\}$ of

Gödel numbers of variables is simply the set of positive even numbers; hence it is indeed $\underline{\mathbb{N}}$-representable and this computable. Consider the set $\{\ulcorner t \urcorner : t \text{ is and } L\text{-term}\}$ of L-terms. Then it is represented by the formula saying that it is either the Gödel number of a variable, or is the Gödel number of one of the finitely many constants, or that it is the Gödel number of $f(t_1, \ldots, t_n)$ where the Gödel number of the terms t_i are less than it. The key here is that we only need to check for smaller Gödel numbers.

Now we have a meaning for a set Γ of L-sentences being computable: Γ is *computable* if $\ulcorner \Gamma \urcorner := \{\ulcorner \sigma \urcorner : \sigma \in \Gamma\}$ is a computable subset of $\mathbb{N}$. Therefore we can formally make sense of the statement of the Incompleteness Theorem from above.

It is easy to see that if Γ is computable, then so is the set

$$\mathrm{Proof}_\Gamma := \{\langle \ulcorner \phi_1 \urcorner, \ldots, \ulcorner \phi_t \urcorner \rangle : \phi_1, \ldots, \phi_t \text{ is a proof from } \Gamma\}.$$

In order to give a more detailed version of the Incompleteness Theorem, we introduce some new notions. We say that a consistent set T of L-sentences is *proof-closed*[1] if for every L-sentence σ if $T \vdash \sigma$, then $\sigma \in T$. For instance, $\mathrm{Th}(\mathcal{M})$ is a proof-closed set of sentences. In general, given a nonempty class $\mathcal{K}$ of L-structures, the set

$$\mathrm{Th}(\mathcal{K}) := \{\sigma : \mathcal{M} \models \sigma \text{ for every } \mathcal{M} \in \mathcal{K}\}$$

is proof-closed.

Given a consistent set Γ of L-sentence, the set $\tilde{\Gamma} := \{\sigma : \Gamma \vdash \sigma\}$ is also proof-closed. If $T = \tilde{\Gamma}$ for some consistent Γ, then we say that Γ *axiomatizes* T. If a proof-closed T is axiomatized by a computable set, then we say that T is *computably axiomatized*.

We generally use the word *decidable* for proof-closed sets of sentences in the place of computable. An important point is that a computably axiomatized proof-closed set is not necessarily decidable since it is not necessarily correct that there is a computable bijection between $\ulcorner \tilde{\Gamma} \urcorner$ and Proof_Γ. However, they are still not too unrelated, as we shall see in a bit.

Definition 3.1. Let $R \subseteq \mathbb{N}^n$. If there is a computable $Q \subseteq \mathbb{N}^{n+1}$ such that for every $a \in \mathbb{N}^n$ we have

$$a \in R \Longleftrightarrow \text{there is } x \in \mathbb{N} \text{ with } (a, x) \in Q,$$

we say that R is *recursively enumerable (r.e.)*.

[1]The more common word is *theory*, but we already have another meaning for that word.

In other words $R \subseteq \mathbb{N}^n$ is recursively enumerable if it is the projection onto the first n coordinates of a computable subset of $\mathbb{N}^{n+1}$.

Suppose that Γ is computable. We know that Proof_Γ is computable. Then it is more or less clear using this that $\tilde{\Gamma}$ is recursively enumerable. However, it still might not be decidable. The next result will tell us when it is.

PROPOSITION 3.2. *Let $R \subseteq \mathbb{N}^n$ be recursively enumerable such that $\neg R$ is also recursively enumerable. Then R is computable.*

PROOF. ... $\square$

COROLLARY 3.3. *Let T be computably axiomatized complete set of L-sentences. Then T is decidable.*

4. Incompleteness Theorem

Let L the numerical language with finitely many non-logical symbols. Before stating the Incompleteness Theorem, we would like to mention Church's theorem without proof.

THEOREM 4.1 (Church). *Let T be a proof-closed set of L-sentences containing $\underline{\mathrm{N}}$. Then T is not decidable.*

The following consequence of Church's theorem is similar to the version of the Incompleteness Theorem from the introduction.

COROLLARY 4.2. *There are no complete computably axiomatized L-theories containing $\underline{\mathrm{N}}$.*

For instance, $\tilde{\underline{\mathrm{N}}}$ is clearly computably axiomatized as $\underline{\mathrm{N}}$ is finite, and it is contained in $\mathrm{Th}(\mathcal{N})$. Then by the corollary above $\tilde{\underline{\mathrm{N}}}$ is not complete and in particular it is not $\mathrm{Th}(\mathcal{N})$.

THEOREM 4.3 (Gödel's Incompleteness Theorem). *Let $\Gamma \supseteq \underline{\mathrm{N}}$ be a consistent computable set of L-sentences. There is an $L(\underline{\mathrm{N}})$-formula $\phi(x)$ such that $\underline{\mathrm{N}} \vdash \phi(S^m(0))$ for each $m \in \mathbb{N}$, and $\Gamma \nvdash \forall x\phi(x)$.*

Note that if we managed to find a ϕ as above, then $\mathcal{N} \models \forall x\phi(x)$ since $\mathcal{N} \models \underline{\mathrm{N}}$. Therefore taking $\sigma := \forall x\phi(x)$ will prove the version in the introduction.

We need some notation to continue with the proof of this result.

DEFINITION 4.4. Let $\mathrm{Pr} \subseteq \mathbb{N}^2$ be defined as the set of (m, n) such that m the Gödel number of the of a proof from Γ of a sentence whose Gödel number is n.

It is clear that $\mathrm{Pr} \subseteq \mathbb{N}^2$ is $\underline{\mathrm{N}}$-representable; let $pr(x, y)$ represent it. It also follows that $\mathrm{Pr} \subseteq \mathbb{N}^2$ is computable.

FACT 4.5. There is a computable function $\mathrm{Sub} : \mathbb{N}^3 \to \mathbb{N}$ such that for every pair t, s of L-terms, for every L-formula ϕ and every variable x we have:

$$\mathrm{Sub}(\ulcorner t \urcorner, \ulcorner x \urcorner, \ulcorner s \urcorner) = \ulcorner t(s/x) \urcorner,$$
$$\mathrm{Sub}(\ulcorner \phi \urcorner, \ulcorner x \urcorner, \ulcorner s \urcorner) = \ulcorner \phi(s/x) \urcorner.$$

FACT 4.6. The function $\mathrm{Num} : \mathbb{N} \to \mathbb{N}$ defined as $\mathrm{Num}(m) = \ulcorner S^m(0) \urcorner$ is computable.

LEMMA 4.7 (Fixed Point Lemma). *Let $\Gamma \subseteq \underline{\mathrm{N}}$ and let $\rho(y)$ be an L-formula. Then there is an L-sentence σ such that*

$$\Gamma \vdash \sigma \leftrightarrow \rho(S^{\ulcorner \sigma \urcorner}(0)).$$

PROOF. Using the facts above, the function sending $(a, b) \in \mathbb{N}^2$ to $\mathrm{Sub}(a, \ulcorner x \urcorner, \mathrm{Num}(b))$ is computable. So it is $\underline{\mathrm{N}}$-representable; say $sub(x_1, x_2, y)$ represents it. This means that for every $(a, b) \in \mathbb{N}^2$ we have:

$$\underline{\mathrm{N}} \vdash sub(S^a(0), S^b(0), y) \leftrightarrow y = S^{\mathrm{Sub}(a, \ulcorner x \urcorner, \ulcorner S^b(0) \urcorner)}(0).$$

Let $\theta(x) := \exists y(sub(x, x, y) \wedge \rho(y))$ and put $m := \ulcorner \theta(x) \urcorner$. Also let σ be $\theta(S^m(0))$ and put $n := \ulcorner \sigma \urcorner$. Note that $n = \mathrm{Sub}(m, \ulcorner x \urcorner, \mathrm{Num}(m))$. Therefore

$$\underline{\mathrm{N}} \vdash sub(S^m(0), S^m(0), y) \leftrightarrow y = S^n(0).$$

So

$$\Gamma \vdash \sigma \leftrightarrow \exists y(y = S^n(0) \wedge \rho(y)).$$

Then $\Gamma \vdash \sigma \leftrightarrow \rho(S^n(0))$, as promised. $\square$

Now we are ready to prove the Incompleteness Theorem.

PROOF OF GÖDEL'S INCOMPLETENESS THEOREM. Let $\rho(y)$ be

$$\forall x \neg pr(x, y)$$

and let σ be the corresponding sentence given by the Fixed Point Theorem. This means

$$\Gamma \vdash \sigma \leftrightarrow \forall x \neg pr(x, S^{\ulcorner \sigma \urcorner}(0)).$$

Suppose $\Gamma \vdash \sigma$. Then $(m, \ulcorner \sigma \urcorner) \in \mathrm{Pr}$, where m is the Gödel number of a proof of σ from Γ. We also have $\Gamma \vdash \forall x \neg pr(x, S^{\ulcorner \sigma \urcorner}(0))$; in particular $\Gamma \vdash \neg pr(S^m(0), S^{\ulcorner \sigma \urcorner}(0))$. This is a contradiction. So $\Gamma \nvdash \sigma$. Putting $\phi(x)$ be $\neg pr(x, S^{\ulcorner \sigma \urcorner}(0))$, we get $\Gamma \nvdash \forall x \phi(x)$. We also have $\underline{\mathrm{N}} \vdash \phi(S^m(0))$, by the defining property of pr. $\square$

Bibliography

[1] L. van den Dries. Mathematical logic - lecture notes. available at `https://faculty.math.illinois.edu/%7evddries/main2.pdf`.

[2] Herbert B. Enderton. *A mathematical introduction to logic*. Harcourt/Academic Press, Burlington, MA, second edition, 2001.

[3] Martin Hils and François Loeser. *A first journey through logic*, volume 89 of *Student Mathematical Library*. American Mathematical Society, Providence, RI, 2019.