H.-D. Ebbinghaus
J. Flum
W. Thomas

Mathematical Logic

Second Edition

With 13 Illustrations

Springer Science+Business Media, LLC

H.-D. Ebbinghaus
J. Flum
Mathematisches Institut
Universität Freiburg
Albertstrasse 23b
7800 Freiburg
Germany

W. Thomas
Institut für Informatik und
Praktische Mathematik
Universität Kiel
D 24098 Kiel
Germany

Editorial Board

J.H. Ewing
Department of Mathematics
Indiana University
Bloomington, IN 47405
USA

F.W. Gehring
Department of Mathematics
University of Michigan
Ann Arbor, MI 48109
USA

P.R. Halmos
Department of Mathematics
Santa Clara University
Santa Clara, CA 95053
USA

Translated from *Einführung in die mathematische Logik*, published by Wissenschaftliche Buchgesellschaft, Darmstadt, by Ann S. Ferebee, Kohlweg 12, D-6240 Königstein 4, Germany.

Copyright 1978 of the original edition by Wissenschaftliche Buchgesellschaft, Darmstadt, Germany.
(First published in the series: "Die Mathematik. Einführungen in Gegenstand und Ergebnisse ihrer Teilgebiete und Nachbarwissenschaften.")

AMS Subject Classification (1991): 03-01

Library of Congress Cataloging-in-Publication Data
Ebbinghaus, Heinz-Dieter, 1939-
[Einführung in die mathematische Logik.]
Mathematical logic / H.-D. Ebbinghaus, J. Flum, W. Thomas.
p. cm. -- (Undergraduate Texts in Mathematics)
Includes bibliographical references and index.
ISBN 978-1-4757-2357-1 ISBN 978-1-4757-2355-7 (eBook)
DOI 10.1007/978-1-4757-2355-7

1. Logic, Symbolic and mathematical. I. Flum, Jörg.
II. Thomas, Wolfgang, 1947- . III. Title. IV. Series.
QA9.E2213 1994
511.3--dc20 93-50621

Printed on acid-free paper.

© 1994 by Springer Science+Business Media New York
Originally published by Springer-Verlag New York, Inc. in 1994
Softcover reprint of the hardcover 2nd edition 1994

All rights reserved. This work may not be translated or copied in whole or in part without the written permission of the publisher Springer Science+Business Media, LLC , except for brief excerpts in connection with reviews or scholarly analysis. Use in connection with any form of information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed is forbidden.

The use of general descriptive names, trade names, trademarks, etc., in this publication, even if the former are not especially identified, is not to be taken as a sign that such names, as understood by the Trade Marks and Merchandise Marks Act, may accordingly be used freely by anyone.

Production managed by Jim Harbison; manufacturing supervised by Vincent Scelta.
Camera-ready copy provided by the authors using Springer-Verlag's LaTeX macro svsing.sty.

9 8 7 6 5 4 3 2 1

ISBN 978-1-4757-2357-1

Preface

What is a mathematical proof? How can proofs be justified? Are there limitations to provability? To what extent can machines carry out mathematical proofs?

Only in this century has there been success in obtaining substantial and satisfactory answers. The present book contains a systematic discussion of these results. The investigations are centered around first-order logic. Our first goal is Gödel's completeness theorem, which shows that the consequence relation coincides with formal provability: By means of a calculus consisting of simple formal inference rules, one can obtain all consequences of a given axiom system (and in particular, imitate all mathematical proofs).

A short digression into model theory will help us to analyze the expressive power of the first-order language, and it will turn out that there are certain deficiencies. For example, the first-order language does not allow the formulation of an adequate axiom system for arithmetic or analysis. On the other hand, this difficulty can be overcome—even in the framework of first-order logic—by developing mathematics in set-theoretic terms. We explain the prerequisites from set theory necessary for this purpose and then treat the subtle relation between logic and set theory in a thorough manner.

Gödel's incompleteness theorems are presented in connection with several related results (such as Trahtenbrot's theorem) which all exemplify the limitations of machine-oriented proof methods. The notions of computability theory that are relevant to this discussion are given in detail. The concept of computability is made precise by means of the register machine as a computer model.

We use the methods developed in the proof of Gödel's completeness theorem to discuss Herbrand's Theorem. This theorem is the starting point for a detailed description of the theoretical fundamentals of logic programming. The corresponding resolution method is first introduced on the level of propositional logic.

The deficiencies in expressive power of the first-order language are a motivation to look for stronger logical systems. In this context we introduce,

among others, the second-order language and the infinitary languages. For each of them we prove that central facts which hold for the first-order language are no longer valid. Finally, this empirical fact is confirmed by Lindström's theorems, which show that there is no logical system that extends first-order logic and at the same time shares all its advantages.

The book does not require special mathematical knowledge; however, it presupposes an acquaintance with mathematical reasoning as acquired, for example, in the first year of a mathematics or computer science curriculum.

Margit Meßmer prepared the English translation of the extended German edition and the LaTeX–version of the book. We wish to thank her for her efficient and diligent work. For additional LaTeX–editing thanks are due to A. Miller and O. Matz. For helpful suggestions and/or careful proof–reading we also thank U. Bosse, G. Geisler, H. Imhof and J. C. Martinez.

Freiburg and Kiel, July 1993

H.-D. Ebbinghaus
J. Flum
W. Thomas

Contents

Preface v

PART A **1**

I Introduction **3**

§1. An Example from Group Theory 4

§2. An Example from the Theory of Equivalence Relations . . . 5

§3. A Preliminary Analysis 6

§4. Preview 8

II Syntax of First-Order Languages **11**

§1. Alphabets 11

§2. The Alphabet of a First-Order Language 13

§3. Terms and Formulas in First-Order Languages 15

§4. Induction in the Calculus of Terms and in the Calculus of Formulas 19

§5. Free Variables and Sentences 24

III Semantics of First-Order Languages **27**

§1. Structures and Interpretations 28

§2. Standardization of Connectives 31

§3. The Satisfaction Relation 32

§4. The Consequence Relation 33

§5. Two Lemmas on the Satisfaction Relation 40

§6. Some Simple Formalizations 44

§7. Some Remarks on Formalizability 48

§8. Substitution 52

IV A Sequent Calculus **59**

§1. Sequent Rules 60

§2. Structural Rules and Connective Rules 62

§3. Derivable Connective Rules 63

§4. Quantifier and Equality Rules 66

§5. Further Derivable Rules and Sequents 68

§6. Summary and Example 69

§7. Consistency 72

V The Completeness Theorem **75**

§1. Henkin's Theorem 75

§2. Satisfiability of Consistent Sets of Formulas (the Countable Case) 79

§3. Satisfiability of Consistent Sets of Formulas (the General Case) 82

§4. The Completeness Theorem 85

VI The Löwenheim-Skolem and the Compactness Theorem **87**

§1. The Löwenheim-Skolem Theorem 87

§2. The Compactness Theorem 88

§3. Elementary Classes 91

§4. Elementarily Equivalent Structures 94

VII The Scope of First-Order Logic **99**

§1. The Notion of Formal Proof 99

§2. Mathematics Within the Framework of First-Order Logic . 103

§3. The Zermelo-Fraenkel Axioms for Set Theory 107

§4. Set Theory as a Basis for Mathematics 110

VIII Syntactic Interpretations and Normal Forms **115**

§1. Term-Reduced Formulas and Relational Symbol Sets 115

§2. Syntactic Interpretations 118

§3. Extensions by Definitions 125

§4. Normal Forms 128

PART B **135**

IX Extensions of First-Order Logic **137**

§1. Second-Order Logic 138

§2. The System $\mathcal{L}_{\omega_1\omega}$ 142

§3. The System $\mathcal{L}_Q$ 148

X Limitations of the Formal Method **151**

§1. Decidability and Enumerability 152

§2. Register Machines 157

§3. The Halting Problem for Register Machines 163

§4. The Undecidability of First-Order Logic 167

§5. Trahtenbrot's Theorem and the Incompleteness of Second-Order Logic 170

§6. Theories and Decidability 173

§7. Self-Referential Statements and Gödel's Incompleteness Theorems 181

XI Free Models and Logic Programming **189**

§1. Herbrand's Theorem 189

§2. Free Models and Universal Horn Formulas 193

§3. Herbrand Structures 198

§4. Propositional Logic 200

§5. Propositional Resolution 207

§6. First-Order Resolution (without Unification) 218

§7. Logic Programming 226

XII An Algebraic Characterization of Elementary Equivalence **243**

§1. Finite and Partial Isomorphisms 244

§2. Fraïssé's Theorem 249

§3. Proof of Fraïssé's Theorem 251

§4. Ehrenfeucht Games 258

XIII Lindström's Theorems **261**

§1. Logical Systems 261

§2. Compact Regular Logical Systems 264

§3. Lindström's First Theorem 266

§4. Lindström's Second Theorem 272

References **277**

Symbol Index **280**

Subject Index **283**

PART A

I

Introduction

Towards the end of the nineteenth century *mathematical logic* evolved into a subject of its own. It was the works of *Boole*, *Frege*, *Russell*, and *Hilbert*, among others[1], that contributed to its rapid development. Various elements of the subject can already be found in traditional logic, for example, in the works of *Aristotle* or *Leibniz*. However, while traditional logic can be considered as part of philosophy, mathematical logic is more closely related to mathematics. Some aspects of this relation are:

(1) *Motivation and Goals*. Investigations in mathematical logic arose mainly from questions concerning the foundations of mathematics. For example, Frege intended to base mathematics on logical and set-theoretical principles. Russell tried to eliminate contradictions that arose in Frege's system. Hilbert's goal was to show that "the generally accepted methods of mathematics taken as a whole do not lead to a contradiction" (this is known as *Hilbert's program*).

(2) *Methods*. In mathematical logic the methods used are primarily *mathematical*. This is exemplified by the way in which new concepts are formed, definitions are given, and arguments are conducted.

(3) *Applications in Mathematics*. The methods and results obtained in mathematical logic are not only useful for treating foundational problems; they also increase the stock of tools available in mathematics itself. There are applications in many areas of mathematics, such as algebra and topology, but also in various parts of theoretical computer science.

However, these mathematical features do not mean that mathematical logic is of interest solely to mathematics or parts of computer science. For example, the mathematical approach leads to a clarification of concepts and problems that are important in traditional logic and also in other fields, such as epistemology or the philosophy of science. In this sense the restriction to mathematical methods turns out to be very fruitful.

---

[1] Aristotle (384 - 322 B.C.), G. W. Leibniz (1646 - 1716), G. Boole (1815 - 1864), G. Frege (1848 - 1925), D. Hilbert (1862 - 1943), B. Russell (1872 - 1970).

In mathematical logic, as in traditional logic, *deductions* and *proofs* are central objects of investigation. However, it is the methods of deduction and the types of argument as used in *mathematical proofs* which are considered in mathematical logic (cf. (1)). In the investigations themselves, mathematical methods are applied (cf. (2)). This close relationship between the subject and the method of investigation, particularly in the discussion of foundational problems, may create the impression that we are in danger of becoming trapped in a vicious circle. We shall not be able to discuss this problem in detail until Chapter VII, and we ask the reader who is concerned about it to bear with us until then.

§1. An Example from Group Theory

In this and the next section we present two simple mathematical proofs. They illustrate some of the methods of proof used by mathematicians. Guided by these examples, we raise some questions which lead us to the main topics of the book.

We begin with the proof of a theorem from group theory. We therefore require the *axioms of group theory*, which we now state. We use $\circ$ to denote the group multiplication and e to denote the identity element. The axioms may then be formulated as follows:

(G1) For all x, y, z : $\quad (x \circ y) \circ z = x \circ (y \circ z)$.

(G2) For all x : $\quad x \circ e = x$.

(G3) For every x there is a y such that $x \circ y = e$.

A *group* is a triple $(G, \circ^G, e^G)$ which satisfies (G1), (G2), and (G3). Here G is a set, e^G is an element of G, and $\circ^G$ is a binary function on G, i.e., a function defined on all ordered pairs of elements from G, the values of which are also elements of G. The variables x, y, z range over elements of G, $\circ$ refers to $\circ^G$, and e refers to e^G.

As an example of a group we mention *the additive group of the reals* $(\mathbf{R}, +, 0)$, where $\mathbf{R}$ is the set of real numbers, $+$ is the usual addition, and 0 is the real number zero. On the other hand, $(\mathbf{R}, \cdot, 1)$ is not a group (where $\cdot$ is the usual multiplication). For example, the real number 0 violates axiom (G3): there is no real number r such that $0 \cdot r = 1$.

We call triples such as $(\mathbf{R}, +, 0)$ or $(\mathbf{R}, \cdot, 1)$ *structures*. In Chapter III we shall give an exact definition of the notion of "structure."

Now we prove the following simple theorem from group theory:

1.1 Theorem on the Existence of a Left Inverse. *For every x there is a y such that $y \circ x = e$.*

Proof. Let x be chosen arbitrarily. By (G3) we have for suitable y,

$$x \circ y = e. \tag{1}$$

Again from (G3) we get, for this y, an element z such that

$$y \circ z = e. \tag{2}$$

We can now argue as follows:

$$\begin{aligned}
y \circ x &= (y \circ x) \circ e && \text{(by (G2))}\\
&= (y \circ x) \circ (y \circ z) && \text{(from (2))}\\
&= y \circ (x \circ (y \circ z)) && \text{(by (G1))}\\
&= y \circ ((x \circ y) \circ z) && \text{(by (G1))}\\
&= y \circ (e \circ z) && \text{(from (1))}\\
&= (y \circ e) \circ z && \text{(by (G1))}\\
&= y \circ z && \text{(by (G2))}\\
&= e && \text{(from (2)).}
\end{aligned}$$

Since x was arbitrary, we conclude that for every x there is a y such that $y \circ x = e$. □[2]

The proof shows that in every structure where (G1), (G2), and (G3) are satisfied, i.e. in every group, the theorem on the existence of a left inverse holds. A mathematician would also describe this situation by saying that the theorem on the existence of a left inverse *follows from*, or *is a consequence of* the axioms of group theory.

§2. An Example from the Theory of Equivalence Relations

The theory of equivalence relations is based on the following three axioms (xRy is to be read as "x is equivalent to y"):

(E1) For all x: xRx.

(E2) For all x, y: If xRy then yRx.

(E3) For all x, y, z: If xRy and yRz then xRz.

Let A be a nonempty set, and let R^A be a binary relation on A, i.e., $R^A \subset A \times A$. For $(a, b) \in R^A$ we also write aR^Ab. The pair (A, R^A) is another example of a structure. We call R^A an *equivalence relation on* A, and the structure (A, R^A) an *equivalence structure*, if (E1), (E2), and (E3) are satisfied. For example, $(\mathbf{Z}, R_5)$ is an equivalence structure, where $\mathbf{Z}$ is the set of integers and

$$R_5 = \{(a, b) \mid a, b \in \mathbf{Z} \text{ and } b - a \text{ is divisible by } 5\}.$$

[2]From now on, □ denotes the end of a proof.

We now prove a simple theorem about equivalence relations.

2.1 Theorem. *If x and y are both equivalent to a third element, they are equivalent to the same elements.* More formally: *For all x and y, if there is a u such that xRu and yRu, then for all z, xRz if and only if yRz.*

Proof. Let x and y be given arbitrarily; suppose that for some u

(1) $$xRu \text{ and } yRu.$$

From (E2) we then obtain

(2) $$uRx \text{ and } uRy.$$

From xRu and uRy we deduce, using (E3),

(3) $$xRy,$$

and from yRu and uRx we likewise get (using (E3))

(4) $$yRx.$$

Now let z be chosen arbitrarily. If

(5) $$xRz,$$

then, using (E3), we obtain from (4) and (5)

$$yRz.$$

On the other hand, if

(6) $$yRz,$$

then, using (E3), we get from (3) and (6)

$$xRz.$$

Thus the claim is proved for all z. □

As in the previous example, this proof shows that every structure (of the form (A, R^A)) which satisfies the axioms (E1), (E2), and (E3), also satisfies Theorem 2.1, i.e., that 2.1 follows from (E1), (E2), and (E3).

§3. A Preliminary Analysis

We now sketch some aspects which the two examples just given have in common.

In each case one starts from a system Φ of propositions[3] which is taken to be a *system of axioms* for the theory in question (group theory, theory

[3]Here we use the notion of proposition in a naive way. For example, we make no distinction between propositions and formulas.

of equivalence relations). The mathematician is interested in finding the propositions which *follow* from Φ, where the proposition ψ is said to follow from Φ if ψ holds in every structure which satisfies all propositions in Φ. A *proof* of ψ from a system Φ of axioms shows that ψ follows from Φ.

When we think about the scope of methods of mathematical proof, we are led to ask about the *converse*:

(*) Is every proposition ψ which follows from Φ also provable from Φ?

For example, is every proposition which holds in all groups also provable from the group axioms (G1), (G2), and (G3)?

The material developed in Chapters II through V and in Chapter VII yields an essentially positive answer to (*). Clearly it is necessary to make the concepts "proposition", "follows from", and "provable", which occur in (*), more precise. We sketch briefly how we shall do this.

(1) *The Concept "Proposition."* Usually the mathematician uses his everyday language (e.g. English or German) to formulate his propositions. But since sentences in everyday language are not, in general, completely unambiguous in their meaning and structure, one cannot specify them by precise definitions. For this reason we shall introduce a *formal language* L which reflects features of mathematical statements. Like programming languages used today, L will be formed according to fixed rules: Starting with a set of symbols (an "alphabet"), we obtain so-called *formulas* as finite symbol strings built up in a standard way. These formulas correspond to propositions expressed in everyday language. For example, the symbols of L will include $\forall$ (to be read "for all"), $\wedge$ ("and"), $\rightarrow$ ("if ... then"), $\equiv$ ("equal") and variables like x, y and z. Formulas of L will be expressions like

$$\forall x\, x \equiv x, \quad x \equiv y, \quad x \equiv z, \quad \forall x \forall y \forall z((x \equiv y \wedge y \equiv z) \rightarrow x \equiv z).$$

Although the expressive power of L may at first appear to be limited, we shall later see that many mathematical propositions can be formulated in L. We shall even see that L is, in principle, sufficient for all of mathematics. The definition of L will be given in Chapter II.

(2) *The Concept "Follows From" (the Consequence Relation).* Axioms (G1), (G2), (G3) of group theory obtain a meaning when interpreted in structures of the form $(G, \circ^G, e^G)$. In an analogous way we can define the general notion of an L-formula holding in a structure. This enables us (in Chapter III) to define the consequence relation: ψ *follows from* (*is a consequence of*) Φ if and only if ψ holds in every structure where all formulas of Φ hold.

(3) *The Concept "Proof."* A mathematical proof of a proposition ψ from a system Φ of axioms consists of a series of *inferences* which proceeds from axioms of Φ or propositions that have already been proved, to new propositions, and which finally ends with ψ. At each step of a proof the mathematician writes something like "From ... and _ _ _ one obtains

directly that $\sim\sim\sim$," and he expects it to be clear to anyone that the validity of ... and of _ _ _ entails the validity of $\sim\sim\sim$.

An analysis of examples shows that the grounds for accepting such inferences are often closely related to the meaning of *connectives*, such as "and", "or", or "if-then", and *quantifiers*, "for all" or "there exists", which occur there. For example, this is the case in the first step of the proof of 1.1, where we deduce from "*for all* x there is a y such that $x \circ y = e$" that for the given x there is a y such that $x \circ y = e$. Or consider the step from (1) and (2) to (3) in the proof of 2.1, where from the proposition "xRu *and* yRu" we infer the left member of the conjunction, "xRu", and from "uRx *and* uRy" we infer the right member, "uRy", and then using (E3) we conclude (3).

The formal character of the language L makes it possible to represent these inferences as formal operations on symbol strings (the L-formulas). Thus, the inference of "xRu" from "xRu and yRu" mentioned above corresponds to the passage from the L-formula $(xRu \wedge yRu)$ to xRu. We can view this as an application of the following *rule*:

(+) It is allowed to pass from an L-formula of the form $(\varphi \wedge \psi)$ to the L-formula φ.

In Chapter IV we shall give a finite system $\mathfrak{S}$ of rules which, like (+), correspond to elementary inference steps the mathematician uses in his proofs.

A *formal proof* of the L-formula ψ from the L-formulas in Φ (the "axioms") consists then (by definition) of a sequence of formulas in L which ends with ψ, and in which each L-formula is obtained by application of a rule from $\mathfrak{S}$ to the axioms or to preceding formulas in the sequence.

Having introduced the precise notions, one can convince oneself by examples that mathematical proofs can be imitated by formal proofs in L. Moreover, in Chapter V we return to the question (*) at the beginning of this section and answer it positively, showing that if a formula ψ follows from a set Φ of formulas, then there is a proof of ψ from Φ, even a formal proof. This is the content of *Gödel's Completeness Theorem.*

§4. Preview

Gödel's Completeness Theorem forms a bridge between the notion of proof, which is formal in character, and the notion of consequence, which refers to the meaning in structures. In Chapter VI we show how this connection can be used in algebraic investigations.

Once a formal language and an exact notion of proof have been introduced, we have a precise framework for mathematical investigations concerning, for instance, the consistency of mathematics or a justification of rules of inference used in mathematics (Chapters VII and X).

Finally, the formalization of the notion of proof gives the possibility of using a computer to carry out or check proofs. In Chapter X we discuss the scope and the limitations of such machine-oriented methods.

Certain formulas in L can themselves be interpreted in an operational way. For example, one can view an implication of the form "if φ then ψ" as an instruction to go from φ to ψ. This interpretation of L-formulas as programs forms the basis of *logic programming*, which is the starting point of certain computer languages in so-called artificial intelligence. In Chapter XI we develop the fundamentals of this part of "applied" logic.

In formulas of L the variables refer to the *elements* of a structure, for example, to the elements of a group or the elements of an equivalence structure. In a given structure we often call elements of its domain A *first-order objects*, while subsets of A are called *second-order objects*. Since L only has variables for first-order objects (and thus expressions such as "$\forall x$" and "$\exists x$" apply only to the elements of a structure), we call L a *first-order language*.

Unlike L, the so-called *second-order language* also has variables which range over subsets of the domain of a structure. Thus a proposition about a given group which begins "For all subgroups ..." can be directly formulated in the second-order language. We shall investigate this language and others in Chapter IX. In Chapter XIII we shall be able to show that no language with more expressive power than L enjoys both an adequate formal concept of proof and other useful properties of L. From this point of view L is a "best-possible" language; and this fact might explain the dominant role which the first-order language plays in mathematical logic.

II

Syntax of First-Order Languages

In this chapter we introduce the first-order languages. They obey simple, clear formation rules. In later chapters we shall discuss whether, and to what extent, all mathematical propositions can be formalized in such languages.

§1. Alphabets

By an *alphabet* $\mathcal{A}$ we mean a nonempty set of *symbols*. Examples of alphabets are the sets $\mathcal{A}_1 = \{0, 1, 2, \ldots, 9\}$, $\mathcal{A}_2 = \{a, b, c, \ldots, z\}$ (the alphabet of lower-case letters), $\mathcal{A}_3 = \{\circ, \int, a, d, x, f,), (\}$, and $\mathcal{A}_4 = \{c_0, c_1, c_2, \ldots\}$.

We call finite sequences of symbols from an alphabet $\mathcal{A}$ *strings* or *words* over $\mathcal{A}$. $\mathcal{A}^*$ denotes the set of all strings over $\mathcal{A}$. The *length* of a string $\zeta \in \mathcal{A}^*$ is the number of symbols, counting repetitions, occurring in ζ. The empty string is also considered to be a word over $\mathcal{A}$. It is denoted by $\square$, and its length is zero.

Examples of strings over $\mathcal{A}_2$ are

$$softly, \quad xdbxaz.$$

Examples of strings over $\mathcal{A}_3$ are

$$\int f(x)dx, \quad x \circ \int\int a.$$

Suppose $\mathcal{A} = \{|, ||\}$, that is, $\mathcal{A}$ consists of the symbols $a_1 := |$[1] and $a_2 := ||$. Then the string $|||$ over $\mathcal{A}$ can be read in three ways: as $a_1a_1a_1$, as a_1a_2, and as a_2a_1. In the sequel we allow only those alphabets $\mathcal{A}$ where any string over $\mathcal{A}$ can be read in exactly one way. The alphabets $\mathcal{A}_1, \ldots, \mathcal{A}_4$ given above satisfy this condition.

---

[1]Here we write "$a_1 := |$" instead of "$a_1 = |$" in order to make it clear that a_1 is *defined* by the right-hand side of the equation.

We now turn to questions concerning the number of strings over a given alphabet.

We call a set M *countable* if it is not finite and if there is a surjective map α of the set of natural numbers $\mathbf{N} = \{0, 1, 2, \ldots\}$ onto M. We can then represent M as $\{\alpha(n) \mid n \in \mathbf{N}\}$ or, if we write the arguments as indices, as $\{\alpha_n \mid n \in \mathbf{N}\}$. A set M is called *at most countable* if it is finite or countable.

1.1 Lemma. *For a nonempty set M the following are equivalent:*
(a) *M is at most countable.*
(b) *There is a surjective map $\alpha\colon \mathbf{N} \to M$.*
(c) *There is an injective map $\beta\colon M \to \mathbf{N}$.*

Proof.[2] We shall prove (b) from (a), (c) from (b), and (a) from (c).

(b) *from* (a): Let M be at most countable. If M is countable (b) holds by definition. For finite M, say $M = \{a_0, \ldots, a_n\}$ (M is nonempty), we define $\alpha\colon \mathbf{N} \to M$ by

$$\alpha(i) := \begin{cases} a_i & \text{if } 0 \leq i \leq n, \\ a_0 & \text{otherwise.} \end{cases}$$

Clearly, α is surjective.

(c) *from* (b): Let $\alpha\colon \mathbf{N} \to M$ be surjective. We define an injective map $\beta\colon M \to \mathbf{N}$ by setting, for $a \in M$,

$$\beta(a) := \text{ the least } i \text{ such that } \alpha(i) = a.$$

(a) *from* (c): Let $\beta\colon M \to \mathbf{N}$ be injective and suppose M is not finite. We must show that M is countable. To do this we define a surjective map $\alpha\colon \mathbf{N} \to M$ inductively as follows:

$$\begin{aligned} \alpha(0) &= \text{ the } a \in M \text{ with the smallest image under } \beta \text{ in } \mathbf{N}, \\ \alpha(n+1) &= \text{ the } a \in M \text{ with the smallest image under } \beta \text{ greater} \\ &\quad \text{than } \beta(\alpha(0)), \ldots, \beta(\alpha(n)). \end{aligned}$$

Since the images under β are not bounded in $\mathbf{N}$, α is defined for all $n \in \mathbf{N}$, and clearly every $a \in M$ belongs to the range of α. □

With Lemma 1.1 one can easily show that every subset of an at most countable set is at most countable and that, if M_1 and M_2 are at most countable, then so is $M_1 \cup M_2$. The set $\mathbf{R}$ of real numbers is neither finite nor countable: it is *uncountable* (cf. Exercise 1.3).

We shall later show that finite alphabets suffice for representing mathematical statements. Moreover, the symbols may be chosen as "concrete"

[2]The goal of our investigations is, among other things, a discussion of the notion of proof. Therefore the reader may be surprised that we use proofs before we have made precise what a mathematical proof is. As already mentioned in Chapter I, we shall return to this apparent circularity in Chapter VII.

objects so that they can be included on the keyboard of a typewriter. Often, however, one can improve the transparency of an argument by using a countable alphabet such as $\mathcal{A}_4$, and we shall do this frequently. For some mathematical applications of methods in mathematical logic it is also useful to consider uncountable alphabets. The set $\{c_r \mid r \in \mathbf{R}\}$, which contains a symbol c_r for every real number r, is an example of an uncountable alphabet. We shall justify the use of such alphabets in VII.4.

1.2 Lemma. *If $\mathcal{A}$ is an at most countable alphabet, then the set $\mathcal{A}^*$ of strings over $\mathcal{A}$ is countable.*

Proof. Let p_n be the nth prime number: $p_0 = 2$, $p_1 = 3$, $p_2 = 5$, and so on. If $\mathcal{A}$ is finite, say $\mathcal{A} = \{a_0, \ldots, a_n\}$, where $a_0, \ldots, a_n$ are pairwise distinct, or if $\mathcal{A}$ is countable, say $\mathcal{A} = \{a_0, a_1, a_2, \ldots\}$, where the a_i are pairwise distinct, we can define the map $\beta: \mathcal{A}^* \to \mathbf{N}$ by

$$\beta(\square) := 1, \qquad \beta(a_{i_0} \ldots a_{i_r}) := {p_0}^{i_0+1} \cdot \ldots \cdot {p_r}^{i_r+1}.$$

Clearly β is injective and thus $\mathcal{A}^*$ is at most countable (cf. 1.1(c)). Since a_0, a_0a_0, $a_0a_0a_0$, ... are all in $\mathcal{A}^*$ it cannot be finite; hence it is countable. □

1.3 Exercise. Let $\alpha: \mathbf{N} \to \mathbf{R}$ be given. For $a, b \in \mathbf{R}$ such that $a < b$ show that there is a point c in the closed interval $I = [a, b]$ such that $c \notin \{\alpha(n) \mid n \in \mathbf{N}\}$. Conclude from this that I, and hence $\mathbf{R}$ also, are uncountable. (Hint: By induction define a sequence $I = I_0 \supset I_1 \supset \ldots$ of closed intervals such that $\alpha(n) \notin I_{n+1}$ and use the fact that $\bigcap_{n \in \mathbf{N}} I_n \neq \emptyset$.)

1.4 Exercise. (a) Show that if the sets $M_0, M_1, \ldots$ are at most countable then the union $\bigcup_{n \in \mathbf{N}} M_n$ is also at most countable.
(b) Use (a) to give a different proof of Lemma 1.2.

1.5 Exercise. Let M be a set. Show that there is no surjective (and hence no bijective) map from M onto the power set $\mathcal{P}(M) := \{B \mid B \subset M\}$ of M. (Hint: For $\alpha: M \to \mathcal{P}(M)$, the set $\{a \in M \mid a \notin \alpha(a)\}$ is not in the range of α.)

§2. The Alphabet of a First-Order Language

We wish to construct formal languages in which we can formulate, for example, the axioms, theorems, and proofs about groups and equivalence relations which we considered in Chapter I. In that context the connectives, the quantifiers, and the equality relation played an important role. Therefore, we shall include the following symbols in the first-order languages: $\neg$ (for "not"), $\wedge$ (for "and"), $\vee$ (for "or"), $\to$ (for "if-then"), $\leftrightarrow$ (for "if and only if"), $\forall$ (for "for all"), $\exists$ (for "there exists"), $\equiv$ (as symbol for equal-

ity). To these we shall add variables (for elements of groups, elements of equivalence structures, etc.) and, finally, parentheses as auxiliary symbols.

To formulate the axioms for groups we also need certain symbols specific to group theory, e.g. a *binary function symbol*, say $\circ$, to denote the group multiplication, and a symbol, say e, to denote the identity element. We call e a constant symbol, or simply a *constant*. For the axioms of the theory of equivalence relations we need a *binary relation symbol*, say R.

Thus, in addition to the "logical" symbols such as "$\neg$" and "$\wedge$", we shall need a set S of relation symbols, function symbols, and constants which varies from theory to theory. Each such set S of symbols determines a first-order language. We summarize:

2.1 Definition. The *alphabet of a first-order language* contains the following symbols:
(a) $v_0, v_1, v_2, \ldots$ (*variables*);
(b) $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ (*not, and, or, if-then, if and only if*);
(c) $\forall, \exists$ (*for all, there exists*);
(d) $\equiv$ (*equality symbol*);
(e) $), ($ (*parentheses*);
(f) (1) for every $n \geq 1$ a (possibly empty) set of *n-ary relation symbols*;
(2) for every $n \geq 1$ a (possibly empty) set of *n-ary function symbols*;
(3) a (possibly empty) set of *constants*.

$\mathcal{A}$ shall stand for the symbols listed in (a) through (e), and S for the set of symbols from (f). S may be empty. The symbols listed under (f) must, of course, be distinct from each other and from the symbols in $\mathcal{A}$.

S determines a first-order language (cf. Section 3). We call $\mathcal{A}_S := \mathcal{A} \cup S$ the alphabet of this language and S its *symbol set*.

We have already become acquainted with some symbol sets: $S_{\text{gr}} := \{\circ, e\}$ for group theory and $S_{\text{eq}} := \{R\}$ for the theory of equivalence relations. For the theory of ordered groups we could use $\{\circ, e, R\}$, where the binary relation symbol R is now taken to represent the ordering relation. In certain theoretical investigations we shall use the symbol set S_∞, which contains the constants $c_0, c_1, c_2, \ldots$, and for every $n \geq 1$ countably many n-ary relation symbols $R_0^n, R_1^n, R_2^n, \ldots$ and n-ary function symbols $f_0^n, f_1^n, f_2^n, \ldots$.

Henceforth we shall use the letters $P, Q, R, \ldots$ for relation symbols, $f, g, h, \ldots$ for function symbols, $c, c_0, c_1, \ldots$ for constants, and $x, y, z, \ldots$ for variables.

§3. Terms and Formulas in First-Order Languages

Given a symbol set S, we call certain strings over $\mathcal{A}_S$ *formulas* of the first-order language determined by S. For example, if $S = S_{Gr}$ we want the strings

$$e \equiv e, \qquad e \circ v_1 \equiv v_2, \qquad \exists v_1(e \equiv e \wedge v_1 \equiv v_2)$$

to be formulas, but not

$$\equiv \wedge e, \qquad e \vee e.$$

The formulas $e \equiv e$ and $e \circ v_1 \equiv v_2$ have the form of equations. Mathematicians call the strings to the left and to the right of the equality symbol *terms*. Terms are "meaningful" combinations of function symbols, variables and constants (together with commas and parentheses). Clearly, to give a precise definition of formulas and thus, in particular, of equations, we must first specify more exactly what we mean by terms.

In mathematics terms are written in different notations, such as $f(x)$, fx, $x + e$, $g(x, e)$, gxe. We choose a parenthesis-free notation, as with fx and gxe.

To define the notion of term we give *instructions* (or *rules*) which tell us how to generate the terms. (Such a system of rules is often called a *calculus*.)

3.1 Definition. *S-terms* are precisely those strings in $\mathcal{A}_S^*$ which can be obtained by finitely many applications of the following *rules*:

(T1) Every variable is an S-term.
(T2) Every constant in S is an S-term.
(T3) If the strings $t_1, \ldots, t_n$ are S-terms and f is an n-ary function symbol in S, then $ft_1 \ldots t_n$ is also an S-term.

We denote the set of S-terms by T^S.

If f is a unary and g a binary function symbol and $S = \{f, g, c, R\}$, then

$$gv_0fgv_4c$$

is an S-term. First of all, c is an S-term by (T2) and v_0 and v_4 are S-terms by (T1). If we apply (T3) to the S-terms v_4 and c and to the function symbol g, we see that gv_4c is an S-term. Another application of (T3) to the S-term gv_4c and to the function symbol f shows that fgv_4c is an S-term, and a final application of (T3) to the S-terms v_0 and fgv_4c and to the function symbol g shows that gv_0fgv_4c is an S-term.

We say that one can *derive* the string gv_0fgv_4c in the calculus of terms (corresponding to S). The *derivation* just described can be given schematically as follows:

1.	c	(T2)
2.	v_0	(T1)
3.	v_4	(T1)
4.	gv_4c	(T3) applied to 3. and 1. using g
5.	fgv_4c	(T3) applied to 4. using f
6.	gv_0fgv_4c	(T3) applied to 2. and 5. using g.

The string directly following the number at the beginning of each line can be obtained in each case by applying a rule of the calculus of terms; applications of (T3) use terms obtained in preceding lines. The information at the end of each line indicates which rules and preceding terms were used. Clearly, not only the string in the last line, but all strings in preceding lines can be derived and, hence, are S-terms.

The reader should show that the strings $gxgxfy$ and $gxgfxfy$ are S-terms for arbitrary variables x and y. Here we give a derivation to show that the string $\circ x \circ ey$ is an S_{gr}-term.

1.	x	(T1)
2.	y	(T1)
3.	e	(T2)
4.	$\circ ey$	(T3) applied to 3. and 2. using $\circ$
5.	$\circ x \circ ey$	(T3) applied to 1. and 4. using $\circ$.

Mathematicians usually write the term in line 4 as $e \circ y$, and the term in line 5 as $x \circ (e \circ y)$. For easier reading we shall sometimes write terms in this way as well.

Using the notion of term we are now able to give the definition of formulas.

3.2 Definition. *S-formulas* are precisely those strings of $\mathcal{A}_S^*$ which are obtained by finitely many applications of the following rules:

(F1) If t_1 and t_2 are S-terms, then $t_1 \equiv t_2$ is an S-formula.

(F2) If $t_1, \ldots, t_n$ are S-terms and R is an n-ary relation symbol in S, then $Rt_1 \ldots t_n$ is an S-formula.

(F3) If φ is an S-formula, then $\neg\varphi$ is also an S-formula.

(F4) If φ and ψ are S-formulas, then $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, and $(\varphi \leftrightarrow \psi)$ are also S-formulas.

(F5) If φ is an S-formula and x is a variable, then $\forall x\varphi$ and $\exists x\varphi$ are also S-formulas.

S-formulas derived using (F1) and (F2) are called *atomic formulas* because they are not formed by combining other S-formulas. The formula $\neg\varphi$ is called the *negation* of φ, and $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, and $(\varphi \rightarrow \psi)$ are called, respectively, the *conjunction*, *disjunction*, and *implication* of φ and ψ.

We use L^S to denote the set of S-formulas. L^S is the *first-order language associated with the symbol set* S (often called the *language of first-order predicate calculus* associated with S).

Instead of S-terms and S-formulas, we often speak simply of terms and formulas when the reference to S is either clear or unimportant. For terms we use the letters $t, t_0, t_1, \ldots$, and for formulas the letters $\varphi, \psi, \ldots$.

We now give some examples. Let $S = S_{\text{eq}} = \{R\}$. We can express the axioms for the theory of equivalence relations by the following formulas:

$$\forall v_0 R v_0 v_0$$
$$\forall v_0 \forall v_1 (R v_0 v_1 \to R v_1 v_0)$$
$$\forall v_0 \forall v_1 \forall v_2 ((R v_0 v_1 \wedge R v_1 v_2) \to R v_0 v_2).$$

One can verify that these strings really are formulas by giving appropriate derivations (as was done above for terms) in the calculus of S_{eq}-formulas. For the first two formulas we have, for example,

(1)	1.	$R v_0 v_0$	(F2)
	2.	$\forall v_0 R v_0 v_0$	(F5) applied to 1. using $\forall, v_0$
(2)	1.	$R v_0 v_1$	(F2)
	2.	$R v_1 v_0$	(F2)
	3.	$(R v_0 v_1 \to R v_1 v_0)$	(F4) applied to 1., 2. using $\to$
	4.	$\forall v_1 (R v_0 v_1 \to R v_1 v_0)$	(F5) applied to 3. using $\forall, v_1$
	5.	$\forall v_0 \forall v_1 (R v_0 v_1 \to R v_1 v_0)$	(F5) applied to 4. using $\forall, v_0$.

In a similar way the reader should convince him- or herself that, for unary f, binary g, unary P, ternary Q, and variables x, y and z, the following strings are $\{P, Q, f, g\}$-formulas:

(1) $\forall y (P z \to Q x x z)$;
(2) $(P g x f y \to \exists x (x \equiv x \wedge x \equiv x))$;
(3) $\forall z \forall z \exists z Q x y z$.

In spite of its rigor the calculus of formulas has "liberal" aspects: we can quantify over a variable which does not actually occur in the formula in question (as in (1)), we can join two identical formulas by means of a conjunction (as in (2)), or we can quantify several times over the same variable (as in (3)).

For the sake of clarity we shall frequently use an abbreviated or modified notation for terms and formulas. For example, we shall write the S_{eq}-formula $R v_0 v_1$ as $v_0 R v_1$ (compare this with the notation $2 < 3$). Moreover, we shall often omit parentheses if they are not essential in order to avoid ambiguity, e.g. the outermost parentheses surrounding conjunctions, disjunctions, etc. Thus, we may write $\varphi \wedge \psi$ for $(\varphi \wedge \psi)$. In the case of iterated conjunctions or disjunctions we shall agree to associate to the left, e.g. $\varphi \wedge \psi \wedge \chi$ will be

understood to mean $((\varphi \wedge \psi) \wedge \chi)$. Furthermore, $\wedge$ and $\vee$ shall bind more strongly than $\rightarrow$. Thus $\forall x(\varphi \wedge \psi \rightarrow \chi)$ will stand for $\forall x((\varphi \wedge \psi) \rightarrow \chi)$. The reader should always be aware that expressions in the abbreviated form are no longer formulas. Once again we emphasize that we need an exact definition of formulas to have a precise notion of mathematical statement in our analysis of the notion of proof.

Perhaps the following analogy with programming languages will clarify the situation. When writing a program one must be meticulous in following the grammatical rules for the programming language, because a computer can process only a formally correct program. But programmers use an abbreviated notation when devising or discussing programs in order to express themselves more quickly and clearly.

We have used $\equiv$ for the equality symbol in first-order languages in order to make statements of the form $\varphi = x \equiv y$ ("φ is the formula $x \equiv y$") easier to read.

For future use we note the following:

3.3 Lemma. *If S is at most countable, then T^S and L^S are countable.*

Proof. If S is at most countable, then so is $\mathcal{A}_S$ and hence by 1.2 the set $\mathcal{A}_S^*$. Since T^S and L^S are subsets of $\mathcal{A}_S^*$ they are also at most countable. On the other hand T^S and L^S are infinite because T^S contains the variables $v_0, v_1, v_2, \ldots$, and L^S contains the formulas $v_0 \equiv v_0$, $v_1 \equiv v_1$, $v_2 \equiv v_2, \ldots$ (even if $S = \emptyset$). □

With the preceding observations the languages L^S have become the object of investigations. In these investigations we use another language, namely everyday English augmented by some mathematical terminology. In order to emphasize the difference in the present context, the formal language L^S is called the *object language* (since it is the object of the investigations); the language English (the language in which the investigations are carried out) is called the *metalanguage*. In another context, for example in linguistic investigations, everyday English could be an object language. Similarly, first-order languages can play the role of metalanguages in certain set-theoretical investigations (cf. VII.4.3).

Historical Note. G. Frege [9] developed the first comprehensive formal language. He used a two-dimensional system of notation which was so complicated that his language never came into general use. The formal languages used today are based essentially on those introduced by G. Peano [27].

§4. Induction in the Calculus of Terms and in the Calculus of Formulas

Let S be a set of symbols and let $Z \subset \mathcal{A}_S^*$ be a set of strings over $\mathcal{A}_S$. In the case where $Z = T^S$ or $Z = L^S$ we described the elements of Z by means of a calculus. Each rule of such a calculus either says that certain strings belong to Z (e.g., the rules (T1), (T2), (F1), and (F2)), or else permits the passage from certain strings $\zeta_1, \ldots, \zeta_n$ to a new string ζ in the sense that, if $\zeta_1, \ldots, \zeta_n$ all belong to Z, then ζ also belongs to Z. The way such rules work is made clear when we write them schematically, as follows:

$$\frac{\zeta_1, \ldots, \zeta_n}{\zeta}.$$

By allowing $n = 0$, the first sort of rules mentioned above ("premise-free" rules) is included in this scheme. Now we can write the rules for the calculus of terms as follows:

(T1) $\dfrac{}{x}$; (T2) $\dfrac{}{c}$ if $c \in S$;

(T3) $\dfrac{t_1, \ldots, t_n}{ft_1 \ldots t_n}$ if $f \in S$ and f is n-ary.

When we define a set Z of strings by means of a calculus $\mathfrak{C}$ we can then prove assertions about elements of Z by means of *induction over* $\mathfrak{C}$. This principle of proof corresponds to induction over the natural numbers. If one wants to show that all elements of Z have a certain property P, then it is sufficient to show that

(I) for every rule
$$\frac{\zeta_1, \ldots, \zeta_n}{\zeta}$$
of the calculus $\mathfrak{C}$, the following holds: whenever $\zeta_1, \ldots, \zeta_n$ are derivable in $\mathfrak{C}$ and have the property P ("*induction hypothesis*"), then ζ also has the property P.

Hence in the case $n = 0$ we must show that ζ has the property P.

This principle of proof is evident: in order to show that all strings derivable in $\mathfrak{C}$ have the property P, we show that everything derivable by means of a "premise-free" rule (i.e., $n = 0$ in (I)) has the property P, and that P is preserved under the application of the remaining rules. This method can also be justified using the principle of complete induction for natural numbers. For this purpose, one defines, in an obvious way, the length of a derivation in $\mathfrak{C}$ (cf. the examples of derivations in Section 3), and then argues as follows: If the condition (I) is satisfied for P, one shows by induction on m that every string which has a derivation of length m has the

property P. Since every element of Z has a derivation of some finite length, P must hold for all elements of Z.

In the special case where $\mathfrak{C}$ is the calculus of terms or the calculus of formulas, we call the proof procedure outlined above *proof by induction on terms* or *on formulas*, respectively. In order to show that all S-terms have a certain property P it is sufficient to show:

(T1)′ Every variable has the property P.

(T2)′ Every constant in S has the property P.

(T3)′ If the S-terms $t_1, \ldots, t_n$ have the property P, and if $f \in S$ is n-ary, then $ft_1 \ldots t_n$ also has the property P.

In the case of the calculus of formulas the corresponding conditions are

(F1)′ Every S-formula of the form $t_1 \equiv t_2$ has the property P.

(F2)′ Every S-formula of the form $Rt_1 \ldots t_n$ has the property P.

(F3)′ If the S-formula φ has the property P, then $\neg\varphi$ also has the property P.

(F4)′ If the S-formulas φ and ψ have the property P, then $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, and $(\varphi \leftrightarrow \psi)$ also have the property P.

(F5)′ If the S-formula φ has the property P and if x is a variable, then $\forall x\varphi$ and $\exists x\varphi$ also have the property P.

We now give some applications of this method of proof.

4.1. (a) *For all symbol sets S, the empty string $\square$ is neither an S-term nor an S-formula.*

(b) (1) *$\circ$ is not an S_{gr}-term.*

(2) *$\circ\circ v_1$ is not an S_{gr}-term.*

(c) *For all symbol sets S, every S-formula contains the same number of right parentheses) as of left parentheses (.*

Proof. (a) Let P be the property on $\mathcal{A}_S^*$ which holds for a string ζ iff[3] ζ is nonempty. We show by induction on terms that every S-term has the property P, and leave the proof for formulas to the reader.

(T1)′, (T2)′: Terms of the form x or c (with $c \in S$) are nonempty.

(T3)′: Every term formed according to (T3) begins with a function symbol, and hence is nonempty. (Note that we do not need to use the induction hypothesis.)

(b) We leave (1) to the reader. To prove (2), let P be the property on $\mathcal{A}_{S_{\mathrm{gr}}}^*$ which holds for a string ζ over $\mathcal{A}_{S_{\mathrm{gr}}}$ iff ζ is distinct from $\circ\circ v_1$. We show by

[3]Throughout "iff" is an abbreviation for "if and only if".

induction on terms that every S_{gr}-term is distinct from $\circ\circ v_1$. The reader will notice that we start using a more informal presentation of inductive proofs.

$t = x$, $t = e$: t is distinct from the string $\circ\circ v_1$.

$t = \circ t_1 t_2$: If $\circ t_1 t_2 = \circ\circ v_1$ then, by (a), we would have $t_1 = \circ$ and $t_2 = v_1$. But $t_1 = \circ$ contradicts (1).

(c) First, one shows by induction on terms that no S-term contains a left or right parenthesis. Then one considers the property P over $\mathcal{A}_S^*$, which holds for a string ζ over $\mathcal{A}_S$ iff ζ has the same number of left parentheses as right parentheses, and one shows by induction on formulas that every S-formula has the property P. We give some cases here as examples:

$\varphi = t_1 \equiv t_2$, where t_1 and t_2 are S-terms: By the observation above there are no parentheses in φ, thus P holds for φ.

$\varphi = \neg\psi$, where ψ has the property P by induction hypothesis: Since φ does not contain any parentheses except those in ψ, φ also has the property P.

$\varphi = (\psi \wedge \chi)$, where P holds for ψ and χ by induction hypothesis: Since φ contains one left parenthesis and one right parenthesis in addition to the parentheses in ψ and χ, the property P also holds for φ.

$\varphi = \forall x\psi$, where ψ has the property P by induction hypothesis: The proof here is the same as in the case $\varphi = \neg\psi$. □

Next, we want to show that terms and formulas have a unique decomposition into their constituents. We refer to a fixed symbol set S. The following two lemmas contain some preliminary results needed for this purpose.

4.2 Lemma. (a) *For all terms t and t', t is not a proper initial segment of t' (i.e. there is no ζ distinct from $\square$ such that $t\zeta = t'$).*
(b) *For all formulas φ and φ', φ is not a proper initial segment of φ'.*

We confine ourselves to the *proof of* (a), and consider the property P, which holds for a string η iff

(*) for all terms t', t' is not a proper initial segment of η and η is not a proper initial segment of t'.

Using induction on terms, we show that all terms t have the property P.

$t = x$: Let t' be an arbitrary term. By 4.1(a), t' cannot be a proper initial segment of x, for then t' would have to be the empty string $\square$. On the other hand, one can easily show by induction on terms that x is the only term which begins with the variable x. Therefore, t cannot be a proper initial segment of t'.

$t = c$: The argument is similar.

$t = ft_1 \dots t_n$ and $(*)$ holds for $t_1, \dots, t_n$: Let t' be an arbitrary fixed term. We show that t' cannot be a proper initial segment of t. Otherwise there would be a ζ such that

$$(1) \qquad \zeta \neq \square,\ t = t'\zeta.$$

Since t' begins with f (for t begins with f), t' cannot be a variable or a constant, thus t' must have been generated using (T3). Therefore it has the form $ft'_1 \dots t'_n$ for suitable terms $t'_1, \dots t'_n$. From (1) we have

$$(2) \qquad ft_1 \dots t_n = ft'_1 \dots t'_n\zeta,$$

and from this, cancelling the symbol f, we obtain

$$(3) \qquad t_1 \dots t_n = t'_1 \dots t'_n\zeta.$$

Therefore t_1 is an initial segment of t'_1 or vice versa. Since t_1 satisfies $(*)$ by induction hypothesis, neither of these can be a proper initial segment of the other. Thus $t_1 = t'_1$. Cancelling t_1 on both sides of (3) we obtain

$$(4) \qquad t_2 \dots t_n = t'_2 \dots t'_n\zeta.$$

Repeatedly applying the argument leading from (3) to (4) we finally obtain

$$\square = \zeta.$$

This contradicts (1). Therefore t' cannot be a proper initial segment of t. The proof that t cannot be a proper initial segment of t' is analogous. □

Applying 4.2 one obtains in a similar way

4.3 Lemma. (a) *If $t_1, \dots, t_n$ and $t'_1, \dots, t'_m$ are terms, and if $t_1 \dots t_n = t'_1 \dots t'_m$, then $n = m$ and $t_i = t'_i$ for $1 \leq i \leq n$.*
(b) *If $\varphi_1, \dots, \varphi_n$ and $\varphi'_1, \dots, \varphi'_m$ are formulas, and if $\varphi_1 \dots \varphi_n = \varphi'_1 \dots \varphi'_m$, then $n = m$ and $\varphi_i = \varphi'_i$ for $1 \leq i \leq n$.*

Using 4.2 and 4.3 one can then easily prove

4.4 Theorem. (a) *Every term is either a variable or a constant or a term of the form $ft_1 \dots t_n$. In the last case the function symbol f and the terms $t_1, \dots, t_n$ are uniquely determined.*
(b) *Every formula is of the form*
(1) $t_1 \equiv t_2$ *or* (2) $Rt_1 \dots t_n$ *or* (3) $\neg\varphi$ *or* (4) $(\varphi \wedge \psi)$ *or* (5) $(\varphi \vee \psi)$ *or* (6) $(\varphi \rightarrow \psi)$ *or* (7) $(\varphi \leftrightarrow \psi)$ *or* (8) $\forall x\varphi$ *or* (9) $\exists x\varphi$,
where the cases (1) – (9) *are mutually exclusive and where the following are uniquely determined: the terms t_1, t_2 in case* (1), *the relation symbol R and the terms $t_1, \dots, t_n$ in case* (2), *the formula φ in case* (3), *the formulas φ and ψ in* (4), (5), (6), (7), *and the variable x and the formula φ in* (8) *and* (9).

Theorem 4.4 asserts that a term or a formula has a unique decomposition into its constituents. Thus, as we shall now show, we can give *inductive definitions on terms* or *formulas*. For example, to define a function for all terms it will be sufficient

(T1)″ to assign a value to each variable;

(T2)″ to assign a value to each constant;

(T3)″ for every n-ary f and for all terms $t_1, \ldots, t_n$ to assign a value to the term $ft_1 \ldots t_n$ assuming that values have already been assigned to the terms $t_1, \ldots, t_n$.

Each term is assigned exactly one value by (T1)″ through (T3)″. We show this by means of induction on terms as follows.

$t = x$: By 4.4(a) t is not a constant and does not begin with a function symbol. Therefore, it is assigned a value only by an application of (T1)″. Thus t is assigned exactly one value.

$t = c$: The argument is analogous to the preceding case.

$t = ft_1 \ldots t_n$, and each of the terms $t_1, \ldots, t_n$ has been assigned exactly one value: To assign a value to t we can only use (T3)″, by 4.4(a). Since, again by 4.4(a), the t_i are uniquely determined, t is assigned a unique value. □

We now give some examples of inductive definitions.

4.5 Definition. (a) The function var (more precisely, var_S), which associates with each S-term the set of variables occurring in it, can be defined as follows:

$$\begin{aligned} \mathrm{var}(x) &:= \{x\} \\ \mathrm{var}(c) &:= \emptyset \\ \mathrm{var}(ft_1 \ldots t_n) &:= \mathrm{var}(t_1) \cup \ldots \cup \mathrm{var}(t_n). \end{aligned}$$

(b) The function SF, which assigns to each formula the set of its subformulas, can be defined by induction on formulas as follows:

$$\begin{aligned} \mathrm{SF}(t_1 \equiv t_2) &:= \{t_1 \equiv t_2\} \\ \mathrm{SF}(Rt_1 \ldots t_n) &:= \{Rt_1 \ldots t_n\} \\ \mathrm{SF}(\neg\varphi) &:= \{\neg\varphi\} \cup \mathrm{SF}(\varphi) \\ \mathrm{SF}((\varphi * \psi)) &:= \{(\varphi * \psi)\} \cup \mathrm{SF}(\varphi) \cup \mathrm{SF}(\psi) \\ &\quad\ \text{for } * = \wedge, \vee, \rightarrow, \leftrightarrow \\ \mathrm{SF}(\forall x\varphi) &:= \{\forall x\varphi\} \cup \mathrm{SF}(\varphi) \\ \mathrm{SF}(\exists x\varphi) &:= \{\exists x\varphi\} \cup \mathrm{SF}(\varphi). \end{aligned}$$

In these examples the set-theoretical notation allows a concise formulation. A means of defining the preceding notions by calculi is indicated in the following exercise.

4.6 Exercise. (a) Let the calculus $\mathfrak{C}_v$ consist of the following rules:

$$\frac{}{x \quad x}\ ; \qquad \frac{y \quad t_i}{y \quad ft_1 \ldots t_n} \text{ if } f \in S \text{ is } n\text{-ary and } i \in \{1, \ldots, n\}.$$

Show that, for all variables x and all S-terms t, xt is derivable in $\mathfrak{C}_v$ iff $x \in \mathrm{var}(t)$.

(b) Give a result for SF analogous to the result for var in (a).

4.7 Exercise. Alter the calculus of formulas by omitting the parentheses in 3.2 (F4), e.g. by writing simply "$\varphi \wedge \psi$" instead of "$(\varphi \wedge \psi)$". So, for example, $\chi := \exists v_0 P v_0 \wedge Q v_1$ is a $\{P, Q\}$-formula in this new sense. Show that the analogue of 4.4 no longer holds, and that the corresponding definition of SF can yield both $\mathrm{SF}(\chi) = \{\chi, Pv_0 \wedge Qv_1, Pv_0, Qv_1\}$ and $\mathrm{SF}(\chi) = \{\chi, \exists v_0 Pv_0, Pv_0, Qv_1\}$, so that SF is no longer a well-defined function.

4.8 Exercise (Parenthesis-Free, or So-Called *Polish Notation* for Formulas). Let S be a symbol set and let $\mathcal{A}'$ be the set of symbols given in 2.1(a)–(d). Let $\mathcal{A}'_S := \mathcal{A}' \cup S$. Define S-formulas in Polish notation (S-P-formulas) to be all strings over $\mathcal{A}'_S$ which can be obtained by finitely many applications of the rules (F1), (F2), (F3), and (F5) from 3.2, and the rule (F4)':

(F4)' If φ and ψ are S-P-formulas then $\wedge\varphi\psi$, $\vee\varphi\psi$, $\rightarrow\varphi\psi$, and $\leftrightarrow\varphi\psi$ are also S-P-formulas.

Prove the analogues of 4.3(b) and 4.4(b) for S-P-formulas.

4.9 Exercise. Let $n \geq 1$ and let $t_1, \ldots, t_n \in T^S$. Show that at each place in the word $t_1 \ldots t_n$ exactly one term starts, i.e. if $0 \leq i <$ length of $t_1 \ldots t_n$, there are uniquely determined $\xi, \eta \in \mathcal{A}^*_S$ and $t \in T^S$ such that length of $\xi = i$ and $t_1 \ldots t_n = \xi t \eta$.

§5. Free Variables and Sentences

Let x, y and z be distinct variables. Consider the atomic subformulas of the $\{R\}$-formula

$$\varphi := \exists x(R\underline{y}\underline{z} \wedge \forall y(\neg \underline{\underline{y}} \equiv \underline{\underline{x}} \vee R\underline{\underline{y}}\underline{z})).$$

The occurrences of the variables y and z marked with single underlining are not quantified, i.e., not in the scope of a corresponding quantifier. Such occurrences are called *free*, and, as we shall see later, the variables there act as *parameters*. The occurrences of the variables x and y marked with double underlining shall be called *bound* occurrences. (Thus the variable y has both free and bound occurrences in φ.)

We give a definition by induction on formulas of the set of *free variables in a formula* φ; we denote this set by $\mathrm{free}(\varphi)$. Again, we fix a symbol set S.

5.1 Definition.

$$\begin{array}{lcl}
\mathrm{free}(t_1 \equiv t_2) & := & \mathrm{var}(t_1) \cup \mathrm{var}(t_2) \\
\mathrm{free}(Pt_1 \ldots t_n) & := & \mathrm{var}(t_1) \cup \cdots \cup \mathrm{var}(t_n) \\
\mathrm{free}(\neg\varphi) & := & \mathrm{free}(\varphi) \\
\mathrm{free}((\varphi * \psi)) & := & \mathrm{free}(\varphi) \cup \mathrm{free}(\psi) \text{ for } * = \wedge, \vee, \rightarrow, \leftrightarrow \\
\mathrm{free}(\forall x\varphi) & := & \mathrm{free}(\varphi) \setminus \{x\} \\
\mathrm{free}(\exists x\varphi) & := & \mathrm{free}(\varphi) \setminus \{x\}.
\end{array}$$

The reader should use this definition to determine the set of free variables in the formula φ at the beginning of this section ($S = \{R\}$). We do this here for a simpler example. Again, let x, y, and z be distinct variables.

$$\begin{array}{lcl}
\mathrm{free}((Ryx \rightarrow \forall y \neg y \equiv z)) & = & \mathrm{free}(Ryx) \cup \mathrm{free}(\forall y \neg y \equiv z) \\
& = & \{x, y\} \cup (\{y, z\} \setminus \{y\}) \\
& = & \{x, y, z\}.
\end{array}$$

Formulas without free variables ("parameter-free" formulas) are called *sentences*. For example, $\exists v_0 \neg v_0 \equiv v_0$ is a sentence.

Finally, we denote by L_n^S the set of S-formulas in which the variables occurring free are among $v_0, \ldots, v_{n-1}$:

$$L_n^S := \{\varphi \mid \varphi \text{ is an } S\text{-formula and } \mathrm{free}(\varphi) \subset \{v_0, \ldots, v_{n-1}\}\}.$$

In particular L_0^S is the set of S-sentences.

5.2 Exercise. Show that the following calculus $\mathfrak{C}_{nf}$ permits to derive precisely those strings of the form $x\varphi$ for which $\varphi \in L^S$ and for which x does not occur free in φ.

$$\frac{}{x \quad t_1 \equiv t_2} \quad \text{if } t_1, t_2 \in T^S \text{ and } x \notin \mathrm{var}(t_1) \cup \mathrm{var}(t_2);$$

$$\frac{}{x \quad Rt_1 \ldots t_n} \quad \text{if } R \in S \text{ is } n\text{-ary, } t_1, \ldots, t_n \in T^S \text{ and } x \notin \mathrm{var}(t_1) \cup \cdots \cup \mathrm{var}(t_n);$$

$$\frac{x \quad \varphi}{x \quad \neg\varphi}; \qquad \frac{\begin{array}{ll} x & \varphi \\ x & \psi \end{array}}{x \quad (\varphi * \psi)} \quad \text{for } * = \wedge, \vee, \rightarrow, \leftrightarrow;$$

$$\frac{}{x \quad \forall x\varphi}; \qquad \frac{}{x \quad \exists x\varphi};$$

$$\frac{x \quad \varphi}{x \quad \forall y\varphi}; \qquad \frac{x \quad \varphi}{x \quad \exists y\varphi}.$$

III

Semantics of First-Order Languages

Let R be a binary relation symbol. The $\{R\}$-formula

$$\forall v_0 R v_0 v_0 \tag{1}$$

is, at present, merely a string of symbols to which no meaning is attached. The situation changes if we specify a domain for the variable v_0 and if we interpret the binary relation symbol R as a binary relation over this domain. There are, of course, many possible choices for such a domain and relation.

For example, suppose we choose $\mathbf{N}$ for the domain, take "$\forall v_0$" to mean "for all $n \in \mathbf{N}$" and interpret R as the divisibility relation $R^{\mathbf{N}}$ on $\mathbf{N}$. Then clearly (1) becomes the (true) statement

$$\textit{for all } n \in \mathbf{N},\ R^{\mathbf{N}} nn,$$

i.e. the statement

every natural number is divisible by itself.

We say that the formula $\forall v_0 R v_0 v_0$ *holds in* $(\mathbf{N}, R^{\mathbf{N}})$.

But if we choose the set $\mathbf{Z}$ of integers as the domain and interpret R as the "smaller-than" relation $R^{\mathbf{Z}}$ on $\mathbf{Z}$, then (1) becomes the (false) statement

$$\textit{for all } a \in \mathbf{Z},\ R^{\mathbf{Z}} aa,$$

i.e. the statement

for every integer a, $a < a$.

We say that the formula $\forall v_0 R v_0 v_0$ *does not hold in* $(\mathbf{Z}, R^{\mathbf{Z}})$.

If we consider the formula

$$\exists v_0 (R v_1 v_0 \wedge R v_0 v_2),$$

in $(\mathbf{Z}, R^{\mathbf{Z}})$, we must also interpret the free variables v_1 and v_2 as elements of $\mathbf{Z}$. If we interpret v_1 as 5 and v_2 as 8 we obtain the (true) statement

there is an integer a such that $5 < a$ and $a < 8$.

If we interpret v_1 as 5 and v_2 as 6 we get the (false) statement

there is an integer a such that $5 < a$ *and* $a < 6$.

The central aim of this chapter is to give a rigorous formulation of the notion of interpretation and precisely define when an interpretation yields a true (or false) statement. This allows us to define in an exact way the consequence relation, which we mentioned in Chapter I.

The definitions of "term," "formula," "free occurrence," etc., given in Chapter II, involve only formal (i.e. grammatical) properties of symbol strings. We call these concepts *syntactic*. On the other hand, the concepts introduced in this chapter depend on the *meanings* of symbol strings also (for example, on the meaning in structures, as in the case above). Such concepts are called *semantic concepts*.

§1. Structures and Interpretations

If A is a set and $n \geq 1$, an *n-ary function on A* is a map whose domain is the set A^n of n-tuples of elements from A, and whose values lie in A. By an *n-ary relation* $\mathfrak{R}$ *on A* we mean a subset of A^n. Instead of writing $(a_1, \ldots, a_n) \in \mathfrak{R}$, we shall often write $\mathfrak{R}a_1 \ldots a_n$, and we shall say that the relation $\mathfrak{R}$ holds for $a_1, \ldots, a_n$. According to this definition, the divisibility relation on $\mathbf{N}$ is the set

$$\{(n, m) \mid n, m \in \mathbf{N} \text{ and there is } k \in \mathbf{N} \text{ with } n \cdot k = m\},$$

and the relation "smaller-than" on $\mathbf{Z}$ is the set

$$\{(a, b) \mid a, b \in \mathbf{Z} \text{ and } a < b\}.$$

In the examples given earlier, the structures $(\mathbf{N}, R^{\mathbf{N}})$ and $(\mathbf{Z}, R^{\mathbf{Z}})$ were determined by the domains $\mathbf{N}$ and $\mathbf{Z}$ and by the binary relations $R^{\mathbf{N}}$ and $R^{\mathbf{Z}}$ as interpretations of the symbol R. We call $(\mathbf{N}, R^{\mathbf{N}})$ and $(\mathbf{Z}, R^{\mathbf{Z}})$ $\{R\}$-structures, thereby specifying the set of interpreted symbols, in this case $\{R\}$. Consider once more the symbol set $S_{\text{gr}} = \{\circ, e\}$ of group theory. If we take the real numbers $\mathbf{R}$ as the domain and interpret $\circ$ as the addition $+$ over $\mathbf{R}$ and e as the element 0 of $\mathbf{R}$, then we obtain the S_{gr}-structure $(\mathbf{R}, +, 0)$. In general an S-structure $\mathfrak{A}$ is determined by specifying:

(a) a domain A,
(b) (1) an n-ary relation on A for every n-ary relation symbol in S,
(2) an n-ary function on A for every n-ary function symbol in S,
(3) an element of A for every constant in S.

We combine the separate parts of (b) by a map with domain S and define:

1.1 Definition. An *S-structure* is a pair $\mathfrak{A} = (A, \mathfrak{a})$ with the following properties:

(a) A is a nonempty set, the *domain* or *universe* of $\mathfrak{A}$.

(b) $\mathfrak{a}$ is a map defined on S satisfying:

(1) for every n-ary relation symbol R in S, $\mathfrak{a}(R)$ is an n-ary relation on A,

(2) for every n-ary function symbol f in S, $\mathfrak{a}(f)$ is an n-ary function on A,

(3) for every constant c in S, $\mathfrak{a}(c)$ is an element of A.

Instead of $\mathfrak{a}(R)$, $\mathfrak{a}(f)$, and $\mathfrak{a}(c)$, we shall frequently write $R^{\mathfrak{A}}$, $f^{\mathfrak{A}}$, and $c^{\mathfrak{A}}$, or simply R^A, f^A, and c^A. For structures $\mathfrak{A}, \mathfrak{B}, \ldots$ we shall use $A, B, \ldots$ to denote their domains. Instead of writing an S-structure in the form $\mathfrak{A} = (A, \mathfrak{a})$, we shall often replace $\mathfrak{a}$ by a list of its values. For example, we write an $\{R, f, g\}$-structure as $\mathfrak{A} = (A, R^{\mathfrak{A}}, f^{\mathfrak{A}}, g^{\mathfrak{A}})$.

In investigations of arithmetic the symbol sets

$$S_{\mathrm{ar}} := \{+, \cdot, 0, 1\} \quad \text{and} \quad S_{\mathrm{ar}}^{<} := \{+, \cdot, 0, 1, <\}$$

play a special role, where $+$ and $\cdot$ are binary function symbols, 0 and 1 are constants, and $<$ is a binary relation symbol. Henceforth, we shall use $\mathfrak{N}$ to denote the S_{ar}-structure $(\mathbf{N}, +^{\mathbf{N}}, \cdot^{\mathbf{N}}, 0^{\mathbf{N}}, 1^{\mathbf{N}})$, where $+^{\mathbf{N}}$ and $\cdot^{\mathbf{N}}$ are the usual addition and multiplication on $\mathbf{N}$ and $0^{\mathbf{N}}$ and $1^{\mathbf{N}}$ are the numbers zero and one, respectively.

$$\mathfrak{N}^{<} := (\mathbf{N}, +^{\mathbf{N}}, \cdot^{\mathbf{N}}, 0^{\mathbf{N}}, 1^{\mathbf{N}}, <^{\mathbf{N}}),$$

where $<^{\mathbf{N}}$ denotes the usual ordering on $\mathbf{N}$, is an example of an $S_{\mathrm{ar}}^{<}$-structure. Similarly we set

$$\mathfrak{R} := (\mathbf{R}, +^{\mathbf{R}}, \cdot^{\mathbf{R}}, 0^{\mathbf{R}}, 1^{\mathbf{R}}) \quad \text{and} \quad \mathfrak{R}^{<} := (\mathbf{R}, +^{\mathbf{R}}, \cdot^{\mathbf{R}}, 0^{\mathbf{R}}, 1^{\mathbf{R}}, <^{\mathbf{R}}).$$

We shall often omit the superscripts $^{\mathbf{N}}, {}^{\mathbf{R}}, \ldots$ from $+^{\mathbf{N}}, +^{\mathbf{R}}, \ldots, <^{\mathbf{N}}, <^{\mathbf{R}}$. It will, however, be clear from the context whether, for example, $+$ is intended to denote the function symbol, the addition on $\mathbf{N}$, or the addition on $\mathbf{R}$.

The interpretation of variables is given by a so-called assignment.

1.2 Definition. An *assignment* in an S-structure $\mathfrak{A}$ is a map $\beta\colon \{v_n \mid n \in \mathbf{N}\} \to A$ of the set of variables into the domain A.

Now we can give a precise definition of the notion of interpretation:

1.3 Definition. An *S-interpretation* $\mathfrak{I}$ is a pair $(\mathfrak{A}, \beta)$ consisting of an S-structure $\mathfrak{A}$ and an assignment β in $\mathfrak{A}$.

When the particular symbol set S in question is either clear or unimportant, we shall simply speak of structures and interpretations instead of S-structures and S-interpretations.

If β is an assignment in $\mathfrak{A}$, $a \in A$ and x is a variable, then let $\beta\frac{a}{x}$ be the assignment in $\mathfrak{A}$ which maps x to a and agrees with β on all variables distinct from x:

$$\beta\tfrac{a}{x}(y) := \begin{cases} \beta(y) & \text{if } y \neq x \\ a & \text{if } y = x \end{cases}$$

For $\mathfrak{I} = (\mathfrak{A}, \beta)$ let $\mathfrak{I}\frac{a}{x} := (\mathfrak{A}, \beta\frac{a}{x})$.

In the introduction to this chapter we gave some examples showing how an S-formula can be read in everyday language once an S-interpretation has been given. It is useful to practice reading formulas under interpretations.

For example, if $S = S_{\text{ar}}^{<}$, and the interpretation $\mathfrak{I} = (\mathfrak{A}, \beta)$ is given by

$$(*) \qquad \mathfrak{A} = (\mathbf{N}, +, \cdot, 0, 1, <) \quad \text{and} \quad \beta(v_n) = 2n \text{ for } n \geq 0,$$

then the formula $v_2 \cdot (v_1 + v_2) \equiv v_4$ (actually: $\cdot v_2 + v_1 v_2 \equiv v_4$) reads "$4 \cdot (2 + 4) = 8$", and the formula $\forall v_0 \exists v_1\, v_0 < v_1$ (actually: $\forall v_0 \exists v_1 {<} v_0 v_1$) reads "for every natural number there is a larger natural number."

1.4 Exercise. Let $\mathfrak{I}$ be the interpretation defined above in $(*)$. How do the following formulas read with this interpretation?

(a) $\exists v_0\, v_0 + v_0 \equiv v_1$;
(b) $\exists v_0\, v_0 \cdot v_0 \equiv v_1$;
(c) $\exists v_1\, v_0 \equiv v_1$;
(d) $\forall v_0 \exists v_1\, v_0 \equiv v_1$;
(e) $\forall v_0 \forall v_1 \exists v_2 (v_0 < v_2 \wedge v_2 < v_1)$.

1.5 Exercise. Let A be a finite nonempty set and S a finite symbol set. Show that there are only finitely many S-structures with A as the domain.

1.6 Exercise. For S-structures $\mathfrak{A} = (A, \mathfrak{a})$ and $\mathfrak{B} = (B, \mathfrak{b})$ let $\mathfrak{A} \times \mathfrak{B}$, the *direct product of* $\mathfrak{A}$ *and* $\mathfrak{B}$, be the S-structure with domain

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

which is determined by the following conditions:

for n-ary R in S and $(a_1, b_1), \ldots, (a_n, b_n) \in A \times B$,

$$R^{\mathfrak{A}\times\mathfrak{B}}(a_1, b_1) \ldots (a_n, b_n) \quad \text{iff} \quad R^{\mathfrak{A}} a_1 \ldots a_n \text{ and } R^{\mathfrak{B}} b_1 \ldots b_n;$$

for n-ary f in S and $(a_1, b_1), \ldots, (a_n, b_n) \in A \times B$,

$$f^{\mathfrak{A}\times\mathfrak{B}}((a_1, b_1), \ldots, (a_n, b_n)) := (f^{\mathfrak{A}}(a_1, \ldots, a_n), f^{\mathfrak{B}}(b_1, \ldots, b_n));$$

and for $c \in S$,

$$c^{\mathfrak{A}\times\mathfrak{B}} := (c^{\mathfrak{A}}, c^{\mathfrak{B}}).$$

Show the following:

(a) If the S_{gr}-structures $\mathfrak{A}$ and $\mathfrak{B}$ are groups, then $\mathfrak{A} \times \mathfrak{B}$ is also a group.
(b) If $\mathfrak{A}$ and $\mathfrak{B}$ are equivalence structures, then $\mathfrak{A} \times \mathfrak{B}$ is also an equivalence structure.
(c) If the S_{ar}-structures $\mathfrak{A}$ and $\mathfrak{B}$ are fields, then $\mathfrak{A} \times \mathfrak{B}$ is not a field.

§2. Standardization of Connectives

When we define the notion of satisfaction in the next section we shall refer to the meaning of the connectives "not", "and", "or", "if-then" and "if and only if". In ordinary language their meanings vary. For example, "or" is sometimes used in an inclusive sense and at other times in the exclusive sense "either-or". However, for our purposes it is useful to fix a standard meaning: We shall always use "or" in the inclusive sense, that is, a compound proposition whose constituents are connected by "or" is true (has the *truth-value* T) iff at least one of the constituents is true; it is false (has the *truth-value* F) iff both constituents are false. For example, we specify in 3.2 that a formula $(\varphi \vee \psi)$ is assigned the truth-value T under an interpretation $\mathfrak{I}$ if and only if φ is assigned the truth-value T under $\mathfrak{I}$ *or* ψ is assigned the truth-value T under $\mathfrak{I}$. Because of our fixed standard meaning we have that $(\varphi \vee \psi)$ is assigned the truth-value T under $\mathfrak{I}$ if and only if at least one of the formulas φ, ψ is assigned T under $\mathfrak{I}$.

According to our convention, the truth-value of a proposition compounded by "or" depends only on the truth-value of its constituents. Thus we can use a function

$$\dot{\vee}\colon \{T, F\} \times \{T, F\} \to \{T, F\}$$

to capture the meaning of "or"; the table of values ("truth-table") is as follows:

		$\dot{\vee}$
T	T	T
T	F	T
F	T	T
F	F	F

We proceed in a similar way with the connectives "and", "if-then", "if and only if" and "not". The truth-tables for the functions $\dot{\wedge}, \dot{\to}, \dot{\leftrightarrow}$ and $\dot{\neg}$ are:

		$\dot{\wedge}$	$\dot{\to}$	$\dot{\leftrightarrow}$
T	T	T	T	T
T	F	F	F	F
F	T	F	T	F
F	F	F	T	T

	$\dot{\neg}$
T	F
F	T

These conventions correspond to mathematical practice.

Connectives for which the truth-values of compound propositions depend only on the truth-values of the constituents are called *extensional*. Thus we use the connectives "not", "and", "or", "if-then" and "if and only if" extensionally. In colloquial speech, however, these connectives are often not used extensionally. Consider, for example, the statements "John fell

ill and the doctor gave him a prescription," and "The doctor gave John a prescription and he fell ill." By contrast with the extensional case, the truth-values of these compound statements also depend on the temporal relation expressed by the order of the two components (*intensional* usage).

When we restrict ourselves to using the connectives extensionally, we sacrifice certain expressive possibilities of informal language. Experience shows, however, that this restriction is unimportant as far as the formalization of *mathematical* assertions is concerned. Furthermore, we will show in XI.4 that all other extensional connectives can be defined from the connectives we have chosen.

2.1 Exercise. Show for arbitrary $x, y \in \{T, F\}$:
(a) $\dot{\rightarrow}(x,y) = \dot{\vee}(\dot{\neg}(x), y)$;
(b) $\dot{\wedge}(x,y) = \dot{\neg}(\dot{\vee}(\dot{\neg}(x), \dot{\neg}(y)))$;
(c) $\dot{\leftrightarrow}(x,y) = \dot{\wedge}(\dot{\rightarrow}(x,y), \dot{\rightarrow}(y,x))$.

§3. The Satisfaction Relation

The satisfaction relation makes precise the notion of a formula being true under an interpretation. Again we fix a symbol set S. By "term", "formula" or "interpretation" we always mean "S-term", "S-formula" or "S-interpretation". As a preliminary step we associate with every interpretation $\mathfrak{I} = (\mathfrak{A}, \beta)$ and every term t an element $\mathfrak{I}(t)$ from the domain A. We define $\mathfrak{I}(t)$ by induction on terms.

3.1 Definition. (a) For a variable x let $\mathfrak{I}(x) := \beta(x)$.
(b) For a constant $c \in S$ let $\mathfrak{I}(c) := c^{\mathfrak{A}}$.
(c) For an n-ary function symbol $f \in S$ and terms $t_1, \ldots, t_n$ let

$$\mathfrak{I}(ft_1 \ldots t_n) := f^{\mathfrak{A}}(\mathfrak{I}(t_1), \ldots, \mathfrak{I}(t_n)).$$

As an illustration, if $S = S_{\mathrm{gr}}$ and $\mathfrak{I} = (\mathfrak{A}, \beta)$ with $\mathfrak{A} = (\mathbf{R}, +, 0)$ and $\beta(v_0) = 2, \beta(v_2) = 6$, then $\mathfrak{I}(v_0 \circ (e \circ v_2)) = \mathfrak{I}(v_0) + \mathfrak{I}(e \circ v_2) = 2 + (0+6) = 8$.

Now, using induction on formulas φ, we give a definition of the relation $\mathfrak{I}$ *is a model of* φ, where $\mathfrak{I}$ is an arbitrary interpretation. If $\mathfrak{I}$ is a model of φ we also say that $\mathfrak{I}$ *satisfies* φ or that φ *holds in* $\mathfrak{I}$, and we write $\mathfrak{I} \models \varphi$.

3.2 Definition of the Satisfaction Relation. For all interpretations $\mathfrak{I} = (\mathfrak{A}, \beta)$ we let

$$\mathfrak{I} \models t_1 \equiv t_2 \qquad :\text{iff}^{\,1} \qquad \mathfrak{I}(t_1) = \mathfrak{I}(t_2)$$

[1] For "iff" see the footnote on p. 20; a colon in front of "iff" indicates that the left-hand side is defined by the right-hand side.

$$
\begin{array}{lll}
\mathfrak{I} \models Rt_1 \ldots t_n & \text{:iff} & R^{\mathfrak{A}}\mathfrak{I}(t_1)\ldots\mathfrak{I}(t_n) \quad \text{(i.e. } R^{\mathfrak{A}} \text{ holds} \\
 & & \text{for } \mathfrak{I}(t_1),\ldots,\mathfrak{I}(t_n)) \\
\mathfrak{I} \models \neg\varphi & \text{:iff} & \text{not } \mathfrak{I} \models \varphi \\
\mathfrak{I} \models (\varphi \wedge \psi) & \text{:iff} & \mathfrak{I} \models \varphi \text{ and } \mathfrak{I} \models \psi \\
\mathfrak{I} \models (\varphi \vee \psi) & \text{:iff} & \mathfrak{I} \models \varphi \text{ or } \mathfrak{I} \models \psi \\
\mathfrak{I} \models (\varphi \rightarrow \psi) & \text{:iff} & \text{if } \mathfrak{I} \models \varphi \text{ then } \mathfrak{I} \models \psi \\
\mathfrak{I} \models (\varphi \leftrightarrow \psi) & \text{:iff} & \mathfrak{I} \models \varphi \text{ if and only if } \mathfrak{I} \models \psi \\
\mathfrak{I} \models \forall x\varphi & \text{:iff} & \text{for all } a \in A,\ \mathfrak{I}\frac{a}{x} \models \varphi \\
\mathfrak{I} \models \exists x\varphi & \text{:iff} & \text{there is an } a \in A \text{ such that } \mathfrak{I}\frac{a}{x} \models \varphi.
\end{array}
$$

(For the definition of $\mathfrak{I}\frac{a}{x}$ see Section 1.)

Given a set Φ of S-formulas, we say that $\mathfrak{I}$ is a model of Φ and write $\mathfrak{I} \models \Phi$ if $\mathfrak{I} \models \varphi$ for all $\varphi \in \Phi$.

By going through the individual steps of Definition 3.2 the reader should convince himself that $\mathfrak{I} \models \varphi$ if and only if φ becomes a true statement under the interpretation $\mathfrak{I}$. The steps in 3.2 involving quantifiers are illustrated by the following example. Again, let $S = S_{\mathrm{gr}}$ and $\mathfrak{I} = (\mathfrak{A}, \beta)$ with $\mathfrak{A} = (\mathbf{R}, +, 0)$ and $\beta(x) = 9$ for all x. Then we have

$$
\begin{array}{lll}
\mathfrak{I} \models \forall v_0\, v_0 \circ e \equiv v_0 & & \\
\quad \text{iff} & \text{for all } r \in \mathbf{R}: & \mathfrak{I}\frac{r}{v_0} \models v_0 \circ e \equiv v_0 \\
\quad \text{iff} & \text{for all } r \in \mathbf{R}: & r + 0 = r.
\end{array}
$$

3.3 Exercise. Let P be a unary relation symbol and f be a binary function symbol. For each of the formulas

$$\forall v_1 f v_0 v_1 \equiv v_0, \quad \exists v_0 \forall v_1 f v_0 v_1 \equiv v_1, \quad \exists v_0 (P v_0 \wedge \forall v_1 P f v_0 v_1)$$

find an interpretation which satisfies the formula and one which does not satisfy it.

3.4 Exercise. A formula which does not contain $\neg, \rightarrow$, or $\leftrightarrow$ is called *positive*. Show that for every positive S-formula there is an S-interpretation which satisfies it. (Hint: One can, for example, use a domain consisting of one element.)

§4. The Consequence Relation

Using the notion of satisfaction we can state exactly when a formula is a consequence of a set of formulas. Again, we assume a symbol set S is given.

4.1 Definition of the Consequence Relation. Let Φ be a set of formulas and φ a formula.

φ is a consequence of Φ (written: $\Phi \models \varphi$) :iff every interpretation which is a model of Φ is also a model of φ.[2]

Instead of "$\{\psi\} \models \varphi$" we shall also write "$\psi \models \varphi$".

We have already sketched some examples of the consequence relation in Chapter I. Now we can formulate I.1.1 (existence of a left inverse in groups) as follows:

$$\Phi_{\mathrm{gr}} \models \forall v_0 \exists v_1\, v_1 \circ v_0 \equiv e,$$

where

$$\Phi_{\mathrm{gr}} = \{\forall v_0 \forall v_1 \forall v_2\, (v_0 \circ v_1) \circ v_2 \equiv v_0 \circ (v_1 \circ v_2),\\ \forall v_0\, v_0 \circ e \equiv v_0,\ \forall v_0 \exists v_1\, v_0 \circ v_1 \equiv e\}.$$

To show that a formula φ is *not* a consequence of a set of formulas Φ, it is sufficient to give an interpretation which satisfies every formula in Φ but fails to satisfy φ. For example, one shows

$$\text{(1)} \qquad \text{not } \Phi_{\mathrm{gr}} \models \forall v_0 \forall v_1\, v_0 \circ v_1 \equiv v_1 \circ v_0$$

by giving as an interpretation a nonabelian group $\mathfrak{G}$ with an arbitrary assignment of variables to elements of $\mathfrak{G}$. Analogously, one can use an abelian group to show

$$\text{(2)} \qquad \text{not } \Phi_{\mathrm{gr}} \models \neg \forall v_0 \forall v_1 v_0 \circ v_1 \equiv v_1 \circ v_0.$$

With (1) and (2) we see that

$$\text{not } \Phi \models \varphi$$

does not necessarily imply

$$\Phi \models \neg\varphi.$$

In Chapter I it became clear, both by examples and in an informal way, that when φ can be proved from a system of axioms Φ then φ is a consequence of Φ. There we raised the question as to what extent the consequences of a system of axioms can be obtained by mathematical proofs. The precise definitions of concepts given in this and the next chapter lay the foundation for a rigorous discussion of this question. In Chapter V we obtain the fundamental result that the consequence relation $\Phi \models \varphi$ can always be established by means of a mathematical proof. We shall see that such a proof consists of elementary steps which, moreover, can be described in a purely formal way (that is, syntactically).

Using the notion of consequence we are now able to define the notions of *validity*, *satisfiability* and *logical equivalence*.

---

[2]We use the symbol $\models$ for both the satisfaction relation ($\mathfrak{I} \models \varphi$) and for the consequence relation ($\Phi \models \varphi$). The symbol preceding "$\models$" (either for an interpretation, such as $\mathfrak{I}$, or for a set of formulas, such as Φ) determines the meaning.

4.2 Definition. A formula φ is *valid* (written: $\models \varphi$) :iff $\emptyset \models \varphi$.

Thus a formula is valid if and only if it holds under all interpretations. For example, all formulas of the form $(\varphi \vee \neg\varphi)$ or $\exists x\, x \equiv x$ are valid.

4.3 Definition. A formula φ is *satisfiable* (written: Sat φ) iff there is an interpretation which is a model of φ. A set of formulas Φ is *satisfiable* (written: Sat Φ) iff there is an interpretation which is a model of all the formulas in Φ.

4.4 Lemma. *For all Φ and all φ,*

$$\Phi \models \varphi \quad \textit{iff} \quad \textit{not} \ \text{Sat}\ \Phi \cup \{\neg\varphi\}.$$

In particular, φ is valid iff $\neg\varphi$ is not satisfiable.

Proof. $\Phi \models \varphi$

iff every interpretation which is a model of Φ is also a model of φ

iff there is no interpretation which is a model of Φ but not a model of φ

iff there is no interpretation which is a model of $\Phi \cup \{\neg\varphi\}$

iff not Sat $\Phi \cup \{\neg\varphi\}$. □

4.5 Definition. Two formulas φ and ψ are *logically equivalent* (written: $\varphi \dashv\models \psi$) :iff $\varphi \models \psi$ and $\psi \models \varphi$.

Thus the formulas φ and ψ are logically equivalent iff they are valid under the same interpretations, that is, iff $\models \varphi \leftrightarrow \psi$.

It is immediately evident from the definition of the notion of satisfaction, together with the truth-tables for connectives, that the following formulas are logically equivalent:

$$(+) \qquad \begin{array}{lll} \varphi \wedge \psi & \text{and} & \neg(\neg\varphi \vee \neg\psi) \\ \varphi \to \psi & \text{and} & \neg\varphi \vee \psi \\ \varphi \leftrightarrow \psi & \text{and} & \neg(\varphi \vee \psi) \vee \neg(\neg\varphi \vee \neg\psi) \\ \forall x\varphi & \text{and} & \neg\exists x \neg\varphi. \end{array}$$

Therefore, we can dispense with the connectives $\wedge$, $\to$ and $\leftrightarrow$, and the quantifier $\forall$. More precisely, we define a map * by induction on formulas, which associates with every formula φ a formula φ^* such that φ^* is logically equivalent to φ and does not contain $\wedge, \to, \leftrightarrow$, or $\forall$:

$$\begin{array}{ll} \varphi^* & := \varphi \ \text{ if } \varphi \text{ is atomic} \\ (\neg\varphi)^* & := \neg\varphi^* \\ (\varphi \vee \psi)^* & := \varphi^* \vee \psi^* \\ (\varphi \wedge \psi)^* & := \neg(\neg\varphi^* \vee \neg\psi^*) \\ (\varphi \to \psi)^* & := \neg\varphi^* \vee \psi^* \\ (\varphi \leftrightarrow \psi)^* & := \neg(\varphi^* \vee \psi^*) \vee \neg(\neg\varphi^* \vee \neg\psi^*) \end{array}$$

$$(\exists x\varphi)^* \quad := \exists x\varphi^*$$
$$(\forall x\varphi)^* \quad := \neg\exists x\neg\varphi^*.$$

Using (+) one can easily prove that * has the desired properties.

In general, a formula φ is easier to read than the corresponding φ^*, as is clear from (+). But because of the logical equivalence of φ and φ^* we do not lose expressive power when we exclude the symbols $\wedge, \rightarrow, \leftrightarrow$, and $\forall$ from our first-order languages. This simplifies our investigations of the languages; in particular, proofs by induction on formulas will be shorter. Thus we make the following conventions:

(1) *In the sequel we restrict ourselves to formulas in which only the connectives* $\neg$ *and* $\vee$ *and the quantifier* $\exists$ *occur*; i.e. in the common alphabet $\mathcal{A}$ (cf. II.2.1) of the first-order languages we omit the symbols $\wedge, \rightarrow, \leftrightarrow$, and $\forall$. In the definition II.3.2 of formulas we restrict cases (F4) and (F5) to the introduction of formulas of the form $(\varphi \vee \psi)$ and $\exists x\varphi$, respectively. Finally, in the definition of the notion of satisfaction we eliminate the cases corresponding to $\wedge, \rightarrow, \leftrightarrow$, and $\forall$.

(2) Nevertheless we shall sometimes retain the symbols $\wedge, \rightarrow, \leftrightarrow, \forall$ when writing formulas. Such "formulas φ in the old style" should now be understood as abbreviations for φ^*; for example, $\forall x(Px \wedge Qx)$ should be understood as an abbreviation for $\neg\exists x\neg\neg(\neg Px \vee \neg Qx)$.

We close this section with a lemma which gives an exact formulation of the – intuitively clear – fact that the satisfaction relation between an S-formula φ and an S-interpretation $\mathfrak{I}$ depends only on the interpretation of the *symbols of* S *occurring in* φ, and on the *variables occurring free in* φ.

4.6 Coincidence Lemma. *Let* $\mathfrak{I}_1 = (\mathfrak{A}_1, \beta_1)$ *be an* S_1*-interpretation and* $\mathfrak{I}_2 = (\mathfrak{A}_2, \beta_2)$ *be an* S_2*-interpretation, both with the same domain, i.e.* $A_1 = A_2$. *Put* $S := S_1 \cap S_2$.

(a) *Let* t *be an* S*-term. If* $\mathfrak{I}_1$ *and* $\mathfrak{I}_2$ *agree*[3] *on the* S*-symbols occurring in* t *and on the variables occurring in* t, *then* $\mathfrak{I}_1(t) = \mathfrak{I}_2(t)$.

(b) *Let* φ *be an* S*-formula. If* $\mathfrak{I}_1$ *and* $\mathfrak{I}_2$ *agree on the* S*-symbols and on the variables occurring free in* φ, *then* $\mathfrak{I}_1 \models \varphi$ *iff* $\mathfrak{I}_2 \models \varphi$.

Proof. (a) We use induction on S-terms.

$t = x$: By hypothesis, $\beta_1(x) = \beta_2(x)$ and therefore

$$\mathfrak{I}_1(x) = \beta_1(x) = \beta_2(x) = \mathfrak{I}_2(x).$$

$t = c$: Similarly.

$t = ft_1 \ldots t_n$ ($f \in S$ n-ary and $t_1, \ldots, t_n \in T^S$):

$$\mathfrak{I}_1(ft_1 \ldots t_n) = f^{\mathfrak{A}_1}(\mathfrak{I}_1(t_1), \ldots, \mathfrak{I}_1(t_n))$$

[3] $\mathfrak{I}_1$ and $\mathfrak{I}_2$ agree on $k \in S$ or on x if $k^{\mathfrak{A}_1} = k^{\mathfrak{A}_2}$ or $\beta_1(x) = \beta_2(x)$, respectively.

$$\begin{aligned} &= f^{\mathfrak{A}_1}(\mathfrak{I}_2(t_1), \ldots, \mathfrak{I}_2(t_n)) \quad \text{(by induction hypothesis)} \\ &= f^{\mathfrak{A}_2}(\mathfrak{I}_2(t_1), \ldots, \mathfrak{I}_2(t_n)) \quad \text{(by hypothesis, } f^{\mathfrak{A}_1} = f^{\mathfrak{A}_2}) \\ &= \mathfrak{I}_2(ft_1 \ldots t_n). \end{aligned}$$

(b) We use induction on S-formulas.

$\varphi = Rt_1 \ldots t_n$ ($R \in S$ n-ary, $t_1, \ldots, t_n \in T^S$):

$$\begin{aligned} \mathfrak{I}_1 \models Rt_1 \ldots t_n \quad &\text{iff} \quad R^{\mathfrak{A}_1}\mathfrak{I}_1(t_1) \ldots \mathfrak{I}_1(t_n) \\ &\text{iff} \quad R^{\mathfrak{A}_1}\mathfrak{I}_2(t_1) \ldots \mathfrak{I}_2(t_n) \quad \text{(by (a))} \\ &\text{iff} \quad R^{\mathfrak{A}_2}\mathfrak{I}_2(t_1) \ldots \mathfrak{I}_2(t_n) \quad \text{(by hypothesis, } R^{\mathfrak{A}_1} = R^{\mathfrak{A}_2}) \\ &\text{iff} \quad \mathfrak{I}_2 \models Rt_1 \ldots t_n. \end{aligned}$$

$\varphi = t_1 \equiv t_2$: Similarly.

$\varphi = \neg\psi$: $\mathfrak{I}_1 \models \neg\psi$

iff not $\mathfrak{I}_1 \models \psi$

iff not $\mathfrak{I}_2 \models \psi$ (by induction hypothesis)

iff $\mathfrak{I}_2 \models \neg\psi$.

$\varphi = (\psi \vee \chi)$: Similarly.

$\varphi = \exists x\psi$: $\mathfrak{I}_1 \models \exists x\psi$

iff there is an $a \in A_1$ such that $\mathfrak{I}_1\frac{a}{x} \models \psi$

iff there is an $a \in A_2 (= A_1)$ such that $\mathfrak{I}_2\frac{a}{x} \models \psi$

(by induction hypothesis applied to ψ, $\mathfrak{I}_1\frac{a}{x}$ and $\mathfrak{I}_2\frac{a}{x}$; note that, because $\text{free}(\psi) \subset \text{free}(\varphi) \cup \{x\}$, the interpretations $\mathfrak{I}_1\frac{a}{x}$ and $\mathfrak{I}_2\frac{a}{x}$ agree on all symbols occurring in ψ and all variables occurring free in ψ)

iff $\mathfrak{I}_2 \models \exists x\psi$. □

In particular, the Coincidence Lemma says that, for an S-formula φ and an S-interpretation $\mathfrak{I} = (\mathfrak{A}, \beta)$, the validity of φ under $\mathfrak{I}$ depends only on the assignments for the *finitely many* variables occurring free in φ (and, of course, on the interpretation of the symbols of S in $\mathfrak{A}$). If these variables are among $v_0, \ldots, v_{n-1}$, i.e. if $\varphi \in L_n^S$, it is, at most, the β-values $a_i = \beta(v_i)$ for $i = 0, \ldots, n-1$ which are significant. Thus, instead of $(\mathfrak{A}, \beta) \models \varphi$, we shall often use the more suggestive notation

$$\mathfrak{A} \models \varphi[a_0, \ldots, a_{n-1}].$$

(Similarly, for an S-term t such that $\text{var}(t) \subset \{v_0, \ldots, v_{n-1}\}$ we shall write $t^{\mathfrak{A}}[a_0, \ldots, a_{n-1}]$ instead of $\mathfrak{I}(t)$.)

If φ is a sentence, i.e. $\varphi \in L_0^S$, we can choose $n = 0$ and write

$$\mathfrak{A} \models \varphi,$$

without even mentioning an assignment. In that case we say that $\mathfrak{A}$ *is a model of* φ. For a set of sentences Φ, $\mathfrak{A} \models \Phi$ means that $\mathfrak{A} \models \varphi$ for every $\varphi \in \Phi$.

4.7 Definition. Let S and S' be symbol sets such that $S \subset S'$; let $\mathfrak{A} = (A, \mathfrak{a})$ be an S-structure, and $\mathfrak{A}' = (A', \mathfrak{a}')$ be an S'-structure. We call $\mathfrak{A}$ a *reduct* (more precisely: the S-*reduct*) of $\mathfrak{A}'$ (and conversely $\mathfrak{A}'$ an *expansion* of $\mathfrak{A}$) iff $A = A'$ and $\mathfrak{a}$ and $\mathfrak{a}'$ agree on S. We write $\mathfrak{A} = \mathfrak{A}'|_S$.

The ordered field $\mathfrak{R}^<$ of real numbers as an $S_{\mathrm{ar}}^<$-structure is an expansion of the field $\mathfrak{R}$ of real numbers as S_{ar}-structure: $\mathfrak{R} = \mathfrak{R}^<|_{S_{\mathrm{ar}}}$.

If $\mathfrak{A} = \mathfrak{A}'|_S$, then it follows from the Coincidence Lemma that for every S-formula φ whose free variables are among $v_0, \ldots, v_{n-1}$, and for all $a_0, \ldots, a_{n-1} \in A$,

$$\mathfrak{A} \models \varphi[a_0, \ldots, a_{n-1}] \quad \text{iff} \quad \mathfrak{A}' \models \varphi[a_0, \ldots, a_{n-1}].$$

To see that this holds we choose $\beta\colon \{v_m \mid n \in \mathbf{N}\} \to A$ so that $\beta(v_i) = a_i$ for $i < n$, and we apply the Coincidence Lemma for $\mathfrak{I}_1 = (\mathfrak{A}, \beta)$ and $\mathfrak{I}_2 = (\mathfrak{A}', \beta)$; $\mathfrak{I}_1$ and $\mathfrak{I}_2$ agree on the symbols occurring in φ and on the variables occurring free in φ.

The definitions of interpretation, consequence, and satisfiability refer to a fixed symbol set S. Using the Coincidence Lemma we can remove this reference to S. Let us consider, for example, the notion of satisfiability. If Φ is a set of S-formulas and $S' \supset S$, then Φ is also a set of S'-formulas. As a set of S-formulas, Φ is satisfiable if there is an S-interpretation which satisfies it, and as a set of S'-formulas it is satisfiable if there is an S'-interpretation which satisfies it. We have

4.8. *Φ is satisfiable with respect to S iff Φ is satisfiable with respect to S'.*

Proof. If $\mathfrak{I}' = (\mathfrak{A}', \beta')$ is an S'-interpretation such that $\mathfrak{I}' \models \Phi$, then by 4.6 the S-interpretation $(\mathfrak{A}'|_S, \beta')$ is a model of Φ. On the other hand, if $\mathfrak{I} = (\mathfrak{A}, \beta)$ is an S-interpretation which satisfies Φ, we choose an S'-structure $\mathfrak{A}'$ such that $\mathfrak{A}'|_S = \mathfrak{A}$. (The symbols in $S' \setminus S$ can be interpreted arbitrarily.) By 4.6 the S-interpretation $(\mathfrak{A}', \beta)$ is then a model of Φ. $\square$

4.9 Exercise. For arbitrary formulas φ, ψ and χ show that:
(a) $(\varphi \vee \psi) \models \chi$ iff $\varphi \models \chi$ and $\psi \models \chi$.
(b) $\models (\varphi \to \psi)$ iff $\varphi \models \psi$.

4.10 Exercise. (a) Show that: $\exists x \forall y \varphi \models \forall y \exists x \varphi$.
(b) Show that $\forall y \exists x Rxy \models \exists x \forall y Rxy$ does not hold.

4.11 Exercise. Prove:
(a) $\forall x(\varphi \wedge \psi) \mathrel{=\!\!|\!\models} (\forall x \varphi \wedge \forall x \psi)$.
(b) $\exists x(\varphi \vee \psi) \mathrel{=\!\!|\!\models} (\exists x \varphi \vee \exists x \psi)$.
(c) $\forall x(\varphi \vee \psi) \mathrel{=\!\!|\!\models} (\varphi \vee \forall x \psi)$, if $x \notin \mathrm{free}(\varphi)$.

(d) $\exists x(\varphi \wedge \psi) \mathrel{=\!\!|\!\models} (\varphi \wedge \exists x\psi)$, if $x \notin \mathrm{free}(\varphi)$.
(e) Show that one cannot do without the assumption "$x \notin \mathrm{free}(\varphi)$" in (c) and (d).

4.12 Exercise. Let φ and ψ be formulas such that $\varphi \mathrel{=\!\!|\!\models} \psi$. Let χ' be obtained from the formula χ by replacing all subformulas of the form φ by ψ.
(a) Define the map $'$ by induction on formulas.
(b) Show that for all χ, $\chi \mathrel{=\!\!|\!\models} \chi'$.

4.13 Exercise. Prove the analogue of 4.8 for the consequence relation.

4.14 Exercise. A set Φ of sentences is called *independent* if there is no $\varphi \in \Phi$ such that $\Phi \setminus \{\varphi\} \models \varphi$. Show that the set Φ_{gr} of group axioms and the set of axioms for equivalence relations (cf. p. 17) are independent.

4.15 Exercise (cf. Exercise 1.6). Let I be a nonempty set. For every $i \in I$, let $\mathfrak{A}_i$ be an S-structure. We write $\prod_{i\in I} \mathfrak{A}_i$ for the *direct product* of the structures $\mathfrak{A}_i$, that is, the S-structure $\mathfrak{A}$ with domain

$$\textstyle\prod_{i\in I} A_i := \{g \mid g : I \to \bigcup_{i\in I} A_i, \text{ and for all } i \in I\text{: } g(i) \in A_i\},$$

which is determined by the following conditions (for $g \in \prod_{i\in I} A_i$ we also write $\langle g(i) \mid i \in I\rangle$):

For n-ary $R \in S$ and $g_1, \ldots, g_n \in \prod_{i\in I} A_i$,

$$R^{\mathfrak{A}} g_1 \ldots g_n \quad \text{:iff} \quad R^{\mathfrak{A}_i} g_1(i) \ldots g_n(i) \text{ for all } i \in I;$$

for n-ary $f \in S$ and $g_1, \ldots, g_n \in \prod_{i\in I} A_i$,

$$f^{\mathfrak{A}}(g_1, \ldots, g_n) := \langle f^{\mathfrak{A}_i}(g_1(i), \ldots, g_n(i)) \mid i \in I\rangle;$$

and for $c \in S$,

$$c^{\mathfrak{A}} := \langle c^{\mathfrak{A}_i} \mid i \in I\rangle.$$

Show: If t is an S-term with $\mathrm{var}(t) \subset \{v_0, \ldots, v_{n-1}\}$ and if $g_0, \ldots, g_{n-1} \in \prod_{i\in I} A_i$ then the following holds:

$$t^{\mathfrak{A}}[g_0, \ldots, g_{n-1}] = \langle t^{\mathfrak{A}_i}[g_0(i), \ldots, g_{n-1}(i)] \mid i \in I\rangle.$$

4.16 Exercise. Formulas which are derivable in the following calculus are called *Horn formulas* (after the American logician A. Horn):

(1) $\dfrac{}{(\neg\varphi_1 \vee \ldots \vee \neg\varphi_n \vee \varphi)}$ if $n \in \mathbf{N}$ and $\varphi_1, \ldots, \varphi_n, \varphi$ are atomic;

(2) $\dfrac{}{\neg\varphi_0 \vee \ldots \vee \neg\varphi_n}$ if $n \in \mathbf{N}$ and $\varphi_0, \ldots, \varphi_n$ are atomic;

(3) $\dfrac{\varphi, \psi}{(\varphi \wedge \psi)}$; (4) $\dfrac{\varphi}{\forall x\varphi}$; (5) $\dfrac{\varphi}{\exists x\varphi}$.

Horn formulas without free variables are called *Horn sentences.*
Show: If φ is a Horn sentence and if every $\mathfrak{A}_i$ is a model of φ then $\prod_{i\in I}\mathfrak{A}_i \models \varphi$. (Hint: State and prove the corresponding result for Horn formulas.)

Historical Note. The precise development of semantics is due essentially to A. Tarski [32]. The notion of logical consequence was already present in work of B. Bolzano [5].

§5. Two Lemmas on the Satisfaction Relation

Now we come to results about isomorphic structures and substructures.

5.1 Definition. Let $\mathfrak{A}$ and $\mathfrak{B}$ be S-structures.

(a) A map $\pi: A \to B$ is called an *isomorphism of* $\mathfrak{A}$ *onto* $\mathfrak{B}$ (written: $\pi: \mathfrak{A} \cong \mathfrak{B}$) :iff

(1) π is a bijection of A onto B.

(2) For n-ary $R \in S$ and $a_1, \ldots, a_n \in A$,

$$R^{\mathfrak{A}} a_1 \ldots a_n \quad \text{iff} \quad R^{\mathfrak{B}} \pi(a_1) \ldots \pi(a_n).$$

(3) For n-ary $f \in S$ and $a_1, \ldots, a_n \in A$,

$$\pi(f^{\mathfrak{A}}(a_1, \ldots, a_n)) = f^{\mathfrak{B}}(\pi(a_1), \ldots, \pi(a_n)).$$

(4) For $c \in S$, $\pi(c^{\mathfrak{A}}) = c^{\mathfrak{B}}$.

(b) $\mathfrak{A}$ and $\mathfrak{B}$ are said to be *isomorphic* (written: $\mathfrak{A} \cong \mathfrak{B}$) iff there is an isomorphism $\pi: \mathfrak{A} \cong \mathfrak{B}$.

For example, the S_{gr}-structure $(\mathbf{N}, +, 0)$ is isomorphic to the S_{gr}-structure $(G, +^G, 0)$ consisting of the even natural numbers with ordinary addition $+^G$. The map $\pi: \mathbf{N} \to G$ with $\pi(n) = 2n$ is an isomorphism of $(\mathbf{N}, +, 0)$ onto $(G, +^G, 0)$.

The following lemma shows that isomorphic structures cannot be distinguished by means of first-order sentences.

5.2 Isomorphism Lemma. *If* $\mathfrak{A}$ *and* $\mathfrak{B}$ *are isomorphic* S*-structures then for all* S*-sentences* φ

$$\mathfrak{A} \models \varphi \quad \textit{iff} \quad \mathfrak{B} \models \varphi.$$

Proof. Let $\pi: \mathfrak{A} \cong \mathfrak{B}$. For the intended proof by induction it is convenient to show not only that the same S-sentences hold in $\mathfrak{A}$ and $\mathfrak{B}$, but also that the same S-formulas hold if one uses corresponding assignments: With every assignment β in $\mathfrak{A}$ we associate the assignment $\beta^\pi := \pi \circ \beta$ in $\mathfrak{B}$, and for the corresponding interpretations $\mathfrak{I} = (\mathfrak{A}, \beta)$ and $\mathfrak{I}^\pi := (\mathfrak{B}, \beta^\pi)$ we shall show:

(i) For every S-term t: $\quad \pi(\mathfrak{I}(t)) = \mathfrak{I}^\pi(t)$.

(ii) For every S-formula φ: $\mathfrak{I} \models \varphi$ iff $\mathfrak{I}^\pi \models \varphi$.

This will complete the proof.

(i) can easily be proved by induction on terms. (ii) is proved by induction on formulas φ simultaneously for all assignments β in $\mathfrak{A}$. We only treat the case of atomic formulas and the steps involving $\neg$ and $\exists$.

$\varphi = t_1 \equiv t_2$: $\mathfrak{I} \models t_1 \equiv t_2$
iff $\mathfrak{I}(t_1) = \mathfrak{I}(t_2)$
iff $\pi(\mathfrak{I}(t_1)) = \pi(\mathfrak{I}(t_2))$ (since $\pi\colon A \to B$ is injective)
iff $\mathfrak{I}^\pi(t_1) = \mathfrak{I}^\pi(t_2)$ (by (i))
iff $\mathfrak{I}^\pi \models t_1 \equiv t_2$.

$\varphi = Rt_1 \dots t_n$: $\mathfrak{I} \models Rt_1 \dots t_n$
iff $R^{\mathfrak{A}}\mathfrak{I}(t_1) \dots \mathfrak{I}(t_n)$
iff $R^{\mathfrak{B}}\pi(\mathfrak{I}(t_1)) \dots \pi(\mathfrak{I}(t_n))$ (because $\pi\colon \mathfrak{A} \cong \mathfrak{B}$)
iff $R^{\mathfrak{B}}\mathfrak{I}^\pi(t_1) \dots \mathfrak{I}^\pi(t_n)$ (by (i))
iff $\mathfrak{I}^\pi \models Rt_1 \dots t_n$.

$\varphi = \neg\psi$: $\mathfrak{I} \models \neg\psi$
iff not $\mathfrak{I} \models \psi$
iff not $\mathfrak{I}^\pi \models \psi$ (by induction hypothesis)
iff $\mathfrak{I}^\pi \models \neg\psi$.

$\varphi = \exists x\psi$: $\mathfrak{I} \models \exists x\psi$
iff there is an $a \in A$ such that $\mathfrak{I}\frac{a}{x} \models \psi$
iff there is an $a \in A$ such that $\left(\mathfrak{I}\frac{a}{x}\right)^\pi \models \psi$ (by induction hypothesis)
iff there is an $a \in A$ such that $\mathfrak{I}^\pi\frac{\pi(a)}{x} \models \psi$ (since $\left(\mathfrak{I}\frac{a}{x}\right)^\pi = \mathfrak{I}^\pi\frac{\pi(a)}{x}$)
iff there is $b \in B$ such that $\mathfrak{I}^\pi\frac{b}{x} \models \psi$ (since $\pi\colon A \to B$ is surjective)
iff $\mathfrak{I}^\pi \models \exists x\psi$. □

From this proof we infer

5.3 Corollary. *If $\pi\colon \mathfrak{A} \cong \mathfrak{B}$ then for $\varphi \in L_n^S$ and $a_0, \dots, a_{n-1} \in A$,*

$$\mathfrak{A} \models \varphi[a_0, \dots, a_{n-1}] \quad \textit{iff} \quad \mathfrak{B} \models \varphi[\pi(a_0), \dots, \pi(a_{n-1})]. \qquad \square$$

Isomorphic structures cannot be distinguished in L_0^S. Conversely, one could ask whether S-structures in which the same S-sentences are satisfied are isomorphic. In Chapter VI we shall see that this is not always the case. For example, there are structures not isomorphic to the S_{ar}-structure $\mathfrak{N}$ of natural numbers in which the same first-order sentences hold.

In the rational numbers every number is divisible by 2. Therefore we have

$$(\mathbf{Q}, +, 0) \models \forall v_0 \exists v_1\, v_1 + v_1 \equiv v_0.$$

In the integers this is no longer true:

$$\text{not } (\mathbf{Z}, +, 0) \models \forall v_0 \exists v_1\, v_1 + v_1 \equiv v_0.$$

So sentences might no longer hold when passing to substructures. We finish this section by introducing the notion of substructure, and we shall give a class of sentences which are preserved by substructures.

5.4 Definition. Let $\mathfrak{A}$ and $\mathfrak{B}$ be S-structures. Then $\mathfrak{A}$ is called a *substructure* of $\mathfrak{B}$ (written: $\mathfrak{A} \subset \mathfrak{B}$) if

(a) $A \subset B$;

(b) (1) for n-ary $R \in S$, $R^{\mathfrak{A}} = R^{\mathfrak{B}} \cap A^n$
(that is, for all $a_1, \ldots, a_n \in A$, $R^{\mathfrak{A}} a_1 \ldots a_n$ iff $R^{\mathfrak{B}} a_1 \ldots a_n$);

(2) for n-ary $f \in S$, $f^{\mathfrak{A}}$ is the restriction of $f^{\mathfrak{B}}$ to A^n;

(3) for $c \in S$, $c^{\mathfrak{A}} = c^{\mathfrak{B}}$.

For example, $(\mathbf{Z}, +, 0)$ is a substructure of $(\mathbf{Q}, +, 0)$, and $(\mathbf{N}, +, 0)$ is a substructure of $(\mathbf{Z}, +, 0)$ (although $(\mathbf{N}, +, 0)$ is not a subgroup of $(\mathbf{Z}, +, 0)$).

If $\mathfrak{A} \subset \mathfrak{B}$, then A is *S-closed* (in $\mathfrak{B}$), that is, A is not empty, for n-ary $f \in S$, $a_1, \ldots, a_n \in A$ implies that $f^{\mathfrak{B}}(a_1, \ldots, a_n) \in A$, and $c^{\mathfrak{B}} \in A$ for $c \in S$.

Conversely, every subset X of B which is S-closed in $\mathfrak{B}$ is the domain of exactly one substructure of $\mathfrak{B}$: In fact, the conditions in 5.4(b) determine exactly one structure with domain X. We denote this substructure by $[X]^{\mathfrak{B}}$ and call it the *substructure generated by X in $\mathfrak{B}$*.

For example, the set $\{2n \mid n \in \mathbf{N}\}$ of the even, non-negative integers is S_{gr}-closed in $(\mathbf{Z}, +, 0)$, but the set $\{2n + 1 \mid n \in \mathbf{N}\}$ is not S_{gr}-closed (3+3 is even!).

A formula which does not contain any quantifiers is called *quantifier-free*.

5.5 Lemma. *Let $\mathfrak{A}$ and $\mathfrak{B}$ be S-structures with $\mathfrak{A} \subset \mathfrak{B}$ and let $\beta\colon \{v_n \mid n \in \mathbf{N}\} \to A$ be an assignment in $\mathfrak{A}$. Then the following holds for every S-term t:*

$$(\mathfrak{A}, \beta)(t) = (\mathfrak{B}, \beta)(t);$$

and for every quantifier-free S-formula φ:

$$(\mathfrak{A}, \beta) \models \varphi \quad \textit{iff} \quad (\mathfrak{B}, \beta) \models \varphi.$$

The easy proof is left to the reader. It follows, for example, from the proof of the Isomorphism Lemma by leaving out the parts referring to the existential quantifier, and by choosing the identity for the map $\pi\colon A \to B$, i.e. the map with $\pi(a) = a$ for all $a \in A$.

If $\mathfrak{B}$ is a group and $\mathfrak{A}$ a substructure of $\mathfrak{B}$, the associative law $\varphi := \forall v_0 \forall v_1 \forall v_2\, (v_0 \circ v_1) \circ v_2 \equiv v_0 \circ (v_1 \circ v_2)$ holds also in $\mathfrak{A}$, since $(a \circ^{\mathfrak{B}} b) \circ^{\mathfrak{B}} c = a \circ^{\mathfrak{B}} (b \circ^{\mathfrak{B}} c)$ holds even for all elements $a, b, c \in B$ (and $\circ^{\mathfrak{B}}$ on A agrees with $\circ^{\mathfrak{A}}$). The sentence φ is universal in the sense of the following definition.

5.6 Definition. The formulas which are derivable by means of the following calculus are called *universal* formulas:

(i) $\dfrac{}{\varphi}$ if φ is quantifier-free; (ii) $\dfrac{\varphi, \psi}{(\varphi * \psi)}$ for $* = \wedge, \vee$;

(iii) $\dfrac{\varphi}{\forall x \varphi}$.

From the proof of VIII.4.4 one can see that every universal formula is logically equivalent to a formula of the form $\forall x_1 \ldots \forall x_n \psi$ with quantifier-free ψ.

5.7 Substructure Lemma. *Let $\mathfrak{A}$ and $\mathfrak{B}$ be S-structures with $\mathfrak{A} \subset \mathfrak{B}$ and let $\varphi \in L_n^S$ be universal. Then the following holds for all $a_0, \ldots, a_{n-1} \in A$:*

$$\textit{If } \mathfrak{B} \models \varphi[a_0, \ldots, a_{n-1}] \textit{ then } \mathfrak{A} \models \varphi[a_0, \ldots, a_{n-1}].$$

Proof. Let $\mathfrak{A} \subset \mathfrak{B}$. We show by induction on universal formulas that for all assignments β in $\mathfrak{A}$,

$$(*) \qquad \text{If } (\mathfrak{B}, \beta) \models \varphi \text{ then } (\mathfrak{A}, \beta) \models \varphi.$$

(Then the lemma follows immediately if, for given $a_0, \ldots, a_{n-1} \in A$, we choose an assignment β in $\mathfrak{A}$ with $\beta(v_i) = a_i$ for $i < n$.)

For quantifier-free φ, $(*)$ holds by 5.5. For $\varphi = (\psi \wedge \chi)$ and for $\varphi = (\psi \vee \chi)$ the claim follows immediately from the induction hypothesis. Now let $\varphi = \forall x \psi$, and let $(*)$ hold for ψ. If $(\mathfrak{B}, \beta) \models \forall x \psi$, we get successively:

for all $b \in B$, $(\mathfrak{B}, \beta\frac{b}{x}) \models \psi$;

for all $a \in A$, $(\mathfrak{B}, \beta\frac{a}{x}) \models \psi$ (since $A \subset B$);

for all $a \in A$, $(\mathfrak{A}, \beta\frac{a}{x}) \models \psi$ (by induction hypothesis);

$(\mathfrak{A}, \beta) \models \forall x \psi$ (by definition of the satisfaction relation). □

5.8 Corollary. *If $\mathfrak{A}$ is a substructure of $\mathfrak{B}$ then the following holds for every universal sentence φ:*

$$\textit{If } \mathfrak{B} \models \varphi \textit{ then } \mathfrak{A} \models \varphi. \qquad \square$$

The substructure $(\mathbf{N}, +, 0)$ of the group $(\mathbf{Z}, +, 0)$ is itself not a group. Therefore the corollary shows that there cannot be a system of axioms for group theory in $L^{S_{\mathrm{gr}}}$ consisting only of universal sentences. If however, we add a unary function symbol $^{-1}$ to S_{gr} for the inverse map and put $S_{\mathrm{grp}} := \{\circ, ^{-1}, e\}$, then the system of axioms

$$\begin{aligned}\Phi_{\mathrm{grp}} := \{&\forall v_0 \forall v_1 \forall v_2\, (v_0 \circ v_1) \circ v_2 \equiv v_0 \circ (v_1 \circ v_2),\\ &\forall v_0\, v_0 \circ e \equiv v_0, \quad \forall v_0\, v_0 \circ v_0{}^{-1} \equiv e\}\end{aligned}$$

consists only of universal sentences. For groups as S_{grp}-structures, substructures and subgroups coincide.

5.9 Exercise. Let S be a finite symbol set and let $\mathfrak{A}$ be a finite S-structure. Show that there is an S-sentence $\varphi_{\mathfrak{A}}$ the models of which are precisely the S-structures isomorphic to $\mathfrak{A}$.

5.10 Exercise. Show:
(a) The relation $<$ ("less-than") is *elementarily definable* in $(\mathbf{R}, +, \cdot, 0)$, i.e. there is a formula $\varphi \in L_2^{\{+,\cdot,0\}}$ such that for all $a, b \in \mathbf{R}$, $(\mathbf{R}, +, \cdot, 0) \models \varphi[a, b]$ iff $a < b$.
(b) The relation $<$ is not elementarily definable in $(\mathbf{R}, +, 0)$. (Hint: Work with a suitable *automorphism* of $(\mathbf{R}, +, 0)$, i.e. with a suitable isomorphism of $(\mathbf{R}, +, 0)$ onto itself.)

5.11 Exercise. The formulas which are derivable by means of the following calculus are called *existential* formulas:

(i) $\dfrac{}{\varphi}$ if φ is quantifier-free; (ii) $\dfrac{\varphi, \psi}{(\varphi * \psi)}$ for $* = \wedge, \vee$;

(iii) $\dfrac{\varphi}{\exists x \varphi}$.

Show:
(a) The negation of a universal sentence is logically equivalent to an existential sentence, and the negation of an existential sentence is logically equivalent to a universal sentence.
(b) If $\mathfrak{A} \subset \mathfrak{B}$ and φ is an existential sentence, then $\mathfrak{A} \models \varphi$ implies $\mathfrak{B} \models \varphi$.

§6. Some Simple Formalizations

As we already saw in Section 4, the axioms for group theory can be formulated, or as we often say, *formalized*, in first-order language. Another example of formalization is the cancellation law for group theory:

$$\varphi := \forall v_0 \forall v_1 \forall v_2 (v_0 \circ v_2 \equiv v_1 \circ v_2 \rightarrow v_0 \equiv v_1).$$

To say that the cancellation law holds in a group $\mathfrak{G}$ means that $\mathfrak{G} \models \varphi$, and to say that it holds in all groups means that $\Phi_{\mathrm{gr}} \models \varphi$.

The statement "there is no element of order two" can be formalized as

$$\psi := \neg \exists v_0 (\neg v_0 \equiv e \wedge v_0 \circ v_0 \equiv e).$$

The observation that there is no element of order two in $(\mathbf{Z}, +, 0)$ thus means that $(\mathbf{Z}, +, 0)$ is a model of ψ.

For applications of our results it is helpful to have a certain proficiency in formalization. The following examples should serve this purpose. As the exact choice of variables is unimportant (for example, instead of using the formula φ above we could have used

$$\forall v_{17}\forall v_8\forall v_1(v_{17} \circ v_1 \equiv v_8 \circ v_1 \to v_{17} \equiv v_8)$$

to formalize the cancellation law) we shall denote the variables simply by $x, y, z \ldots$, where distinct letters stand for *distinct* variables.

6.1 Equivalence Relations. The three defining properties of an equivalence relation can be formalized with the aid of a single binary relation symbol R as follows:

$$\forall x Rxx,$$
$$\forall x\forall y(Rxy \to Ryx),$$
$$\forall x\forall y\forall z((Rxy \wedge Ryz) \to Rxz).$$

The theorem mentioned in I.2,

> *If x and y are both equivalent to a third element then they are equivalent to the same elements,*

can be reformulated as

> *For all x, y, if there is an element u such that x is equivalent to u and y is equivalent to u, then for all z, x is equivalent to z iff y is equivalent to z,*

and then formalized as

$$\forall x\forall y(\exists u(Rxu \wedge Ryu) \to \forall z(Rxz \leftrightarrow Ryz)).$$

6.2 Continuity. Let ρ be a unary function on $\mathbf{R}$ and let Δ be the binary distance function on $\mathbf{R}$, that is, $\Delta(r_0, r_1) = |r_0 - r_1|$ for $r_0, r_1 \in \mathbf{R}$. Using the function symbols f (for ρ) and d (for Δ) we can treat $(\mathbf{R}, +, \cdot, 0, 1, <, \rho, \Delta)$ as an $S^{<}_{\mathrm{ar}} \cup \{f, d\}$-structure. The continuity of ρ on $\mathbf{R}$ can be stated as follows:

(∗) For all x and for all $\epsilon > 0$ there is a $\delta > 0$ such that for all y, if $\Delta(x, y) < \delta$ then $\Delta(\rho(x), \rho(y)) < \epsilon$.

Concerning the "restricted" quantifiers "for all $\epsilon > 0$" and "there is a $\delta > 0$" that appear in (∗) it is useful to observe that a statement of the form

for all x such that ..., we have _ _ _

can be formalized

$$\forall x(\ldots \to _\ _\ _),$$

and a statement of the type

there is an x with ... such that _ _ _

can be formalized

$$\exists x(\ldots \wedge _\ _\ _).$$

Thus, using the variables u and v for ϵ and δ we can give the following formalization of (∗):

$$\forall x\forall u(0 < u \to \exists v(0 < v \wedge \forall y(dxy < v \to dfxfy < u))).$$

6.3 Cardinality Statements. The sentence

$$\varphi_{\geq 2} := \exists v_0 \exists v_1 \neg v_0 \equiv v_1$$

is a formalization of "there are at least two elements." More precisely, for all S and all S-structures $\mathfrak{A}$,

$$\mathfrak{A} \models \varphi_{\geq 2} \quad \text{iff} \quad A \text{ contains at least two elements.}$$

In a similar way, for $n \geq 3$, the sentence

$$\varphi_{\geq n} := \exists v_0 \ldots \exists v_{n-1}(\neg v_0 \equiv v_1 \wedge \ldots \wedge \neg v_0 \equiv v_{n-1} \wedge \ldots \wedge \neg v_{n-2} \equiv v_{n-1})$$

states that there are at least n elements, and the sentences $\neg\varphi_{\geq n}$ and $\varphi_{\geq n} \wedge \neg\varphi_{\geq n+1}$ say that there are fewer than n elements and exactly n elements, respectively. If we now put

$$\Phi_\infty := \{\varphi_{\geq n} \mid n \geq 2\},$$

then the models of Φ_∞ are precisely the infinite structures, that is, for all S and all S-structures $\mathfrak{A}$,

$$\mathfrak{A} \models \Phi_\infty \quad \text{iff} \quad A \text{ contains infinitely many elements.}$$

For later use, we state some further systems of axioms for different theories.

6.4 The Theory of Orderings. A structure $\mathfrak{A} = (A, <^{\mathfrak{A}})$ is called an *ordering* if it is a model of the following sentences:

$$\Phi_{\text{ord}} \begin{cases} \forall x \neg x < x \\ \forall x \forall y \forall z((x < y \wedge y < z) \rightarrow x < z) \\ \forall x \forall y (x < y \vee x \equiv y \vee y < x). \end{cases}$$

$(\mathbf{R}, <^{\mathbf{R}})$ and $(\mathbf{N}, <^{\mathbf{N}})$ are examples of orderings. If $\mathbf{C}$ denotes the set of complex numbers and $<^{\mathbf{C}}$ is defined by

$$z_1 <^{\mathbf{C}} z_2 \quad \text{:iff} \quad z_1, z_2 \in \mathbf{R} \text{ and } z_1 <^{\mathbf{R}} z_2$$

then $(\mathbf{C}, <^{\mathbf{C}})$ is not an ordering because the third axiom in Φ_{ord} is violated. If for a structure $\mathfrak{A} = (A, <^{\mathfrak{A}})$ we set

$$\text{field} <^{\mathfrak{A}} := \{a \in A \mid \text{for some } b \in A,\ a <^{\mathfrak{A}} b \text{ or } b <^{\mathfrak{A}} a\},$$ [4]

then, for $(\mathbf{C}, <^{\mathbf{C}})$, field $<^{\mathbf{C}} = \mathbf{R}$ and $(\text{field} <^{\mathbf{C}}, <^{\mathbf{C}})$ is an ordering. We say that $\mathfrak{A} = (A, <^{\mathfrak{A}})$ is a *partially defined ordering* (also: *partial ordering*[5]) on A if $(\text{field} <^{\mathfrak{A}}, <^{\mathfrak{A}})$ is an ordering. So the partial orderings are exactly the models of

$$\Phi_{\text{pord}} \begin{cases} \exists x \exists y\, x < y \\ \forall x \neg x < x \\ \forall x \forall y \forall z((x < y \wedge y < z) \rightarrow x < z) \\ \forall x \forall y((\exists u(x < u \vee u < x) \wedge \exists v(y < v \vee v < y)) \\ \qquad\qquad \rightarrow (x < y \vee x \equiv y \vee y < x)). \end{cases}$$

[4] Of course not to be confused with the notion of field as introduced in 6.5.

[5] In the literature *partial ordering* sometimes has a different meaning.

6.5 The Theory of Fields. We take $S_{\mathrm{ar}} = \{+, \cdot, 0, 1\}$ to be the underlying symbol set. An S_{ar}-structure is a *field* if it satisfies the following sentences:

$$\Phi_{\mathrm{fd}} \begin{cases} \forall x \forall y \forall z (x+y)+z \equiv x+(y+z) & \forall x\, x+0 \equiv x \\ \forall x \forall y \forall z (x \cdot y) \cdot z \equiv x \cdot (y \cdot z) & \forall x\, x \cdot 1 \equiv x \\ \forall x \exists y\, x+y \equiv 0 & \forall x (\neg x \equiv 0 \to \exists y\, x \cdot y \equiv 1) \\ \forall x \forall y\, x+y \equiv y+x & \forall x \forall y\, x \cdot y \equiv y \cdot x \\ \neg\, 0 \equiv 1 & \\ \forall x \forall y \forall z\, x \cdot (y+z) \equiv (x \cdot y)+(x \cdot z). & \end{cases}$$

Ordered fields are $S_{\mathrm{ar}}^{<}$-structures which satisfy the following sentences:

$$\Phi_{\mathrm{ofd}} \begin{cases} \text{the sentences in } \Phi_{\mathrm{fd}} \text{ and } \Phi_{\mathrm{ord}} \\ \forall x \forall y \forall z (x < y \to x+z < y+z) \\ \forall x \forall y \forall z ((x < y \wedge 0 < z) \to x \cdot z < y \cdot z). \end{cases}$$

6.6 The Theory of Graphs. Let $S = \{R\}$ with a binary relation symbol R. S-structures $\mathfrak{G} = (G, R^{\mathfrak{G}})$ which are models of

$$\begin{aligned} \Phi_{\mathrm{dgph}} &:= \{\forall x \neg Rxx\} \text{ and} \\ \Phi_{\mathrm{gph}} &:= \{\forall x \neg Rxx, \forall x \forall y (Rxy \leftrightarrow Ryx)\} \end{aligned}$$

are called *directed graphs* and *graphs*, respectively. One can visualize a (directed) graph $\mathfrak{G} = (G, R^{\mathfrak{G}})$ by thinking of two different points a, b of G with $R^{\mathfrak{G}}ab$ as being connected by a line (an arrow) going from a to b. Such a pair of points (a, b) is called a (*directed*) *edge* of $\mathfrak{G}$ and the elements of G are called *vertices* of $\mathfrak{G}$.

6.7 Exercise. Formalize the following statements using the symbol set of 6.2:
(a) Every positive real number has a positive square root.
(b) If ρ is strictly monotone then ρ is injective.
(c) ρ is uniformly continuous on $\mathbf{R}$.
(d) For all x, if ρ is differentiable at x then ρ is continuous at x.

6.8 Exercise. Let $S_{\mathrm{eq}} = \{R\}$. Formalize:
(a) R is an equivalence relation with at least two equivalence classes.
(b) R is an equivalence relation with an equivalence class containing more than one element.

6.9 Exercise. Use 4.16 to show:
(a) If, for every $i \in I$, the structure $\mathfrak{A}_i$ is a group then $\prod_{i \in I} \mathfrak{A}_i$ is a group.
(b) Neither the theory of orderings nor the theory of fields can be axiomatized by Horn sentences.

6.10 Exercise. A set M of natural numbers is called a *spectrum* if there is a symbol set S and an S-sentence φ such that

$$M = \{n \in \mathbf{N} \mid \varphi \text{ has a model containing exactly } n \text{ elements}\}.$$

Show:
(a) Every finite subset of $\{1, 2, 3, \ldots\}$ is a spectrum.
(b) For every $m \geq 1$, the set of numbers > 0 which are divisible by m is a spectrum.
(c) The set of squares > 0 is a spectrum.
(d) The set of nonprime numbers > 0 is a spectrum.
(e) The set of prime numbers is a spectrum.

§7. Some Remarks on Formalizability

In the preceding section we had a number of examples showing how mathematical statements can be formalized by first-order formulas. However, the process of formalization is not always as simple as it was in those cases. In this section we discuss some typical difficulties which can arise.

7.1 Partial Functions. When we defined the notion of structure we stipulated that function symbols be interpreted by total functions, i.e. in the case of an n-ary function symbol, by a function which is defined on all n-tuples of elements of the domain. If, for example, in the field of real numbers, we regard division on $\mathbf{R}$ as a function then we do not have a structure in our sense (because the quotient is undefined if the divisor is zero). The following are possible solutions to this difficulty:

(1) The division function can be extended to a total function. For example, one can define $\frac{r}{0} := 0$ for all $r \in \mathbf{R}$ and take this into consideration when formulating statements about the division function.

(2) Instead of the division function, one can consider its graph, that is, the ternary relation $\{(a, b, c) \in \mathbf{R} \mid b \neq 0 \text{ and } \frac{a}{b} = c\}$[6]. In VIII.1. we shall describe how statements about functions can be translated into statements about their graphs. The remarks made there for total functions can easily be modified to cover the case of partial functions.

(3) One can introduce first-order languages which also include partial functions. However, this approach leads to a more complicated logical system without yielding anything essentially new, as we see from (1) and (2).

7.2 Many-Sorted Structures. The structures we have hitherto considered have only one domain and, in this sense, consist of elements of only one *sort*. On the other hand, some important structures in mathematics contain elements of different sorts. Planes in affine spaces consist of points

[6]Note that this notion of graph is different from the one in 6.6; there graphs are special structures.

and lines, and vector spaces consist of vectors and scalars. Taking vector spaces as an example, we give two possibilities for treating many-sorted structures.

(1) *Many-Sorted Languages.* We regard a vector space $\mathfrak{V}$ as a "structure with two domains" (a so-called *two-sorted structure*):

$$\mathfrak{V} = (F, V, +^F, \cdot^F, 0^F, 1^F, \circ^V, e^V, *^{F,V}),$$

where F is the set of scalars, $(F, +^F, \cdot^F, 0^F, 1^F)$ is the field of scalars, V is the set of vectors, $(V, \circ^V, e^V)$ is the additive group of vectors, and $*^{F,V}$ is the multiplication of scalars and vectors defined on $F \times V$.

In order to describe such two-sorted structures we introduce a two-sorted language, that is, a language built up in the same way as the languages we have used so far, but having two sorts of variables, namely $u_0, u_1, u_2, \ldots$ (for elements of the first domain, in the case above, scalars) and $w_0, w_1, w_2, \ldots$ (for elements of the second domain, in the case above, vectors). A quantified variable always ranges over the corresponding domain. To illustrate this we formalize some of the axioms for vector spaces.

(α) Associativity of scalar addition:
$\forall u_0 \forall u_1 \forall u_2\, (u_0 + u_1) + u_2 \equiv u_0 + (u_1 + u_2)$.

(β) Associativity of vector addition:
$\forall w_0 \forall w_1 \forall w_2\, (w_0 \circ w_1) \circ w_2 \equiv w_0 \circ (w_1 \circ w_2)$.

(γ) Associativity of scalar multiplication of vectors:
$\forall u_0 \forall u_1 \forall w_0 (u_0 \cdot u_1) * w_0 \equiv u_0 * (u_1 * w_0)$.

(2) *Sort Reduction.* It is also possible to use our one-sorted first-order languages to treat many-sorted structures, namely, by a so-called *sort reduction*. We demonstrate this method briefly for the case of vector spaces. Let $\underline{F}$ and $\underline{V}$ be two new unary relation symbols. We regard a vector space as a $\{\underline{F}, \underline{V}, +, \cdot, 0, 1, \circ, e, *\}$-structure

$$\mathfrak{V} = (F \cup V, \underline{F}^{\mathfrak{V}}, \underline{V}^{\mathfrak{V}}, +^{\mathfrak{V}}, \cdot^{\mathfrak{V}}, 0^{\mathfrak{V}}, 1^{\mathfrak{V}}, \circ^{\mathfrak{V}}, e^{\mathfrak{V}}, *^{\mathfrak{V}})$$

with $\underline{F}^{\mathfrak{V}} := F$, $\underline{V}^{\mathfrak{V}} := V$, where the functions $+^{\mathfrak{V}}, \cdot^{\mathfrak{V}}, \circ^{\mathfrak{V}}, *^{\mathfrak{V}}$ are arbitrary extensions of $+^F, \cdot^F, \circ^V, *^{F,V}$ to $(F \cup V) \times (F \cup V)$. The introduction of the "sort symbols" $\underline{F}$ and $\underline{V}$ enables us to speak of scalars and vectors. We exemplify this by reformulating the many-sorted vector axioms given above:

(α) $\forall x \forall y \forall z ((\underline{F}x \wedge \underline{F}y \wedge \underline{F}z) \rightarrow (x + y) + z \equiv x + (y + z))$.

(β) $\forall x \forall y \forall z ((\underline{V}x \wedge \underline{V}y \wedge \underline{V}z) \rightarrow (x \circ y) \circ z \equiv x \circ (y \circ z))$.

(γ) $\forall x \forall y \forall z ((\underline{F}x \wedge \underline{F}y \wedge \underline{V}z) \rightarrow (x \cdot y) * z \equiv x * (y * z))$.

Since in (α) for example, all quantifiers are "relativized" to $\underline{F}$, it makes no difference how the extension $+^{\mathfrak{V}}$ of $+^F$ is chosen.

7.3 Limits of Formalizability. The question of the limits of formalizability, which is ultimately the question of the expressive power of first-order languages, will be treated in detail in Chapter VI and in VII.2. Here we discuss two examples.

(1) *Torsion Groups.* A group $\mathfrak{G}$ is called a *torsion group* if every element of $\mathfrak{G}$ has finite order, i.e. if for every $a \in G$ there is an $n \geq 1$ such that $a^n = e^G$. An ad hoc formalization of this property would be

$$\forall x(x \equiv e \vee x \circ x \equiv e \vee (x \circ x) \circ x \equiv e \vee \ldots).$$

However, in first-order logic we may not form infinitely long disjunctions. Indeed, we shall later show that there is no set of first-order formulas whose models are precisely the torsion groups.

(2) *Peano's Axioms.* We consider the question whether there is a set of S_{ar}-sentences the models of which are the structures isomorphic to

$$\mathfrak{N} = (\mathbf{N}, +, \cdot, 0, 1).$$

For simplicity we start our discussion with the structure $\mathfrak{N}_\sigma = (\mathbf{N}, \sigma, 0)$, where σ is the successor function on $\mathbf{N}$ ($\sigma(n) = n + 1$ for $n \in \mathbf{N}$). $\mathfrak{N}_\sigma$ is a $\{\boldsymbol{\sigma}, 0\}$-structure, with $\boldsymbol{\sigma}$ ("successor") a unary function symbol. The results can easily be extended to $\mathfrak{N}$, cf. Exercise 7.5.

$\mathfrak{N}_\sigma$ satisfies the so-called *Peano axiom system*:

(α) 0 is not a value of the successor function σ.

(β) σ is injective.

(γ) (the so-called *induction axiom*) For every subset X of $\mathbf{N}$: if $0 \in X$ and if $\sigma(n) \in X$ whenever $n \in X$, then $X = \mathbf{N}$.

Axioms (α) and (β) may be easily formalized in $L^{\{\boldsymbol{\sigma}, 0\}}$ by

(P1) $\forall x \neg \boldsymbol{\sigma} x \equiv 0$;

(P2) $\forall x \forall y(\boldsymbol{\sigma} x \equiv \boldsymbol{\sigma} y \rightarrow x \equiv y)$.

The induction axiom (γ) is a statement about arbitrary subsets of $\mathbf{N}$. For an "ad hoc" formalization of this axiom we would need to quantify over variables for subsets of the domain. In such a language, (γ) could be formalized as follows:

(P3) $\forall X((X0 \wedge \forall x(Xx \rightarrow X\boldsymbol{\sigma} x)) \rightarrow \forall y Xy)$.

(P3) is a so-called *second-order* formula (cf. IX.1). The following theorem shows that (P1) – (P3) characterize the structure $\mathfrak{N}_\sigma$ up to isomorphism, i.e. $\mathfrak{N}_\sigma$ is, up to isomorphism, the only model of (P1) – (P3).

7.4 Dedekind's Theorem. *Every structure* $\mathfrak{A} = (A, \boldsymbol{\sigma}^A, 0^A)$ *which satisfies* (P1) – (P3) *is isomorphic to* $\mathfrak{N}_\sigma$.

In VI.4 we shall show that no set of first-order $\{\boldsymbol{\sigma}, 0\}$-sentences has (up to isomorphism) just $\mathfrak{N}_\sigma$ as a model. Thus the induction axiom cannot be formalized in the first-order language $L^{\{\boldsymbol{\sigma},0\}}$.

The *proof of Dedekind's Theorem* depends essentially on the fact that in structures $\mathfrak{A}$ which satisfy (P3), the following kind of *proofs by induction in* $\mathfrak{A}$ can be given: In order to show that every element of the domain A has a certain property P, one verifies that 0^A has the property P and that if an element a has the property P then $\boldsymbol{\sigma}^A(a)$ does also.

Suppose $\mathfrak{A} = (A, \boldsymbol{\sigma}^A, 0^A)$ is a structure which satisfies (P1) – (P3). The isomorphism $\pi\colon \mathfrak{N}_\sigma \cong \mathfrak{A}$ we need must have the following properties:

$$\begin{array}{lrcll} \text{(i)} & \pi(0^{\mathbf{N}}) & = & 0^A & \\ \text{(ii)} & \pi(\boldsymbol{\sigma}^{\mathbf{N}}(n)) & = & \boldsymbol{\sigma}^A(\pi(n)) & \text{for all } n \in \mathbf{N}, \end{array}$$

that is,

$$\begin{array}{lrcll} \text{(i)}' & \pi(0) & = & 0^A & \\ \text{(ii)}' & \pi(n+1) & = & \boldsymbol{\sigma}^A(\pi(n)) & \text{for all } n \in \mathbf{N}. \end{array}$$

We define π by induction on n, taking (i)$'$ and (ii)$'$ to be the defining clauses. Then the compatibility conditions for an isomorphism are trivially satisfied and we only have to show that π is a bijective map from $\mathbf{N}$ onto A.

Surjectivity of π: By induction in $\mathfrak{A}$ ($\mathfrak{A}$ satisfies (P3)) we prove that every element of A lies in the range of π. By (i)$'$, 0^A is in the range of π. Further, if a is in the range of π, say $a = \pi(n)$, then $\boldsymbol{\sigma}^A(a) = \boldsymbol{\sigma}^A(\pi(n))$. Hence, by (ii)$'$, $\boldsymbol{\sigma}^A(a) = \pi(n+1)$, and it follows that $\boldsymbol{\sigma}^A(a)$ is also in the range of π.

Injectivity of π: By induction on n we prove

$$(*) \qquad \text{For all } m \in \mathbf{N}, \text{ if } m \neq n \text{ then } \pi(m) \neq \pi(n).$$

$n = 0$: If $m \neq 0$, say $m = k+1$, then $\pi(m) = \pi(k+1) = \boldsymbol{\sigma}^A(\pi(k))$, and since $\mathfrak{A}$ satisfies the axiom (P1), $\boldsymbol{\sigma}^A(\pi(k)) \neq 0^A$. Hence, by (i)$'$, $\pi(m) \neq \pi(0)$.

Induction step: Suppose that $(*)$ has been proved for n and suppose $m \neq n+1$. If $m = 0$, we argue as in the case $n = 0$ that $\pi(m) = 0^A \neq \pi(n+1)$. If $m \neq 0$, say $m = k+1$, then $k \neq n$ and so, by induction hypothesis, $\pi(k) \neq \pi(n)$. By injectivity of $\boldsymbol{\sigma}^A$ ($\mathfrak{A}$ satisfies (P2)!) it follows that $\boldsymbol{\sigma}^A(\pi(k)) \neq \boldsymbol{\sigma}^A(\pi(n))$; hence from (ii)$'$ we have $\pi(k+1) \neq \pi(n+1)$, i.e. $\pi(m) \neq \pi(n+1)$. □

7.5 Exercise. Let Π be the following set of second-order S_{ar}-sentences:

$$\begin{array}{l} \forall x \neg x + 1 \equiv 0 \\ \forall x \forall y (x + 1 \equiv y + 1 \to x \equiv y) \\ \forall X ((X0 \wedge \forall x (Xx \to Xx + 1)) \to \forall y Xy) \\ \forall x\, x + 0 \equiv x \\ \forall x \forall y\, x + (y + 1) \equiv (x + y) + 1 \end{array}$$

$\forall x\, x \cdot 0 \equiv 0$
$\forall x \forall y\, x \cdot (y+1) \equiv (x \cdot y) + x.$

Show:
(a) If $\mathfrak{A} = (A, +^A, \cdot^A, 0^A, 1^A)$ is a model of Π and if $\boldsymbol{\sigma}^A \colon A \to A$ is given by $\boldsymbol{\sigma}^A(a) = a +^A 1^A$, then $(A, \boldsymbol{\sigma}^A, 0^A)$ satisfies the axioms (P1) – (P3).
(b) $\mathfrak{N} = (\mathbf{N}, +, \cdot, 0, 1)$ is characterized by Π up to isomorphism.

§8. Substitution

In this section we define how to substitute a term t for a variable x in a formula φ at the places where x occurs free, thus obtaining a formula ψ. We wish to define the substitution so that ψ expresses the same about t as φ does about x. First, we give an example to illustrate our objective and to show why a certain care is necessary. Let

$$\varphi := \exists z\, z + z \equiv x.$$

In $\mathfrak{N}$ the formula φ says that x is even; more precisely

$$(\mathfrak{N}, \beta) \models \varphi \quad \text{iff} \quad \beta(x) \text{ is even.}$$

If we replace the variable x by y in φ we obtain the formula $\exists z\, z + z \equiv y$, which states that y is even. But if we replace the variable x by z, we obtain the formula $\exists z\, z + z \equiv z$, which no longer says that z is even; in fact, this formula is valid in $\mathfrak{N}$ regardless of the assignment for z (because $0+0=0$). In this case the meaning is altered because at the place where x occurred free, the variable z gets bound. On the other hand, we obtain a formula which expresses the same about z as φ does about x, if we proceed as follows: First, we introduce a new bound variable u in φ, and then in the formula $\exists u\, u + u \equiv x$ thus obtained we replace x by z. It is immaterial which variable u (distinct from x and z) we choose. However, for certain technical purposes it is useful to make a fixed choice.

In the preceding example we replaced only one variable, but in our exact definition we specify the procedure for simultaneously replacing several variables: With a given formula φ, *pairwise distinct variables* $x_0, \ldots, x_r$ and arbitrary terms $t_0, \ldots, t_r$, we associate a formula $\varphi \frac{t_0 \ldots t_r}{x_0 \ldots x_r}$, which is said to be obtained from φ by *simultaneously substituting* $t_0, \ldots, t_r$ for $x_0, \ldots, x_r$. The reader should note that x_i has to be replaced by t_i only if

$$x_i \in \text{free}(\varphi) \quad \text{and} \quad x_i \neq t_i.$$

In the following inductive definition this is explicitly taken into account in the quantifier step; in the other steps it follows immediately.

It will become apparent that it is convenient to first introduce a simultaneous substitution for terms. Let S be a fixed symbol set.

8.1 Definition.

(a) $x\frac{t_0\ldots t_r}{x_0\ldots x_r} := \begin{cases} x & \text{if } x \neq x_0,\ldots,x \neq x_r \\ t_i & \text{if } x = x_i \end{cases}$

(b) $c\frac{t_0\ldots t_r}{x_0\ldots x_r} := c$

(c) $[ft'_1\ldots t'_n]\frac{t_0\ldots t_r}{x_0\ldots x_r} := ft'_1\frac{t_0\ldots t_r}{x_0\ldots x_r}\ldots t'_n\frac{t_0\ldots t_r}{x_0\ldots x_r}$.

For easier reading we use square brackets here and in what follows.

8.2 Definition.

(a) $[t'_1 \equiv t'_2]\frac{t_0\ldots t_r}{x_0\ldots x_r} := t'_1\frac{t_0\ldots t_r}{x_0\ldots x_r} \equiv t'_2\frac{t_0\ldots t_r}{x_0\ldots x_r}$

(b) $[Rt'_1\ldots t'_n]\frac{t_0\ldots t_r}{x_0\ldots x_r} := Rt'_1\frac{t_0\ldots t_r}{x_0\ldots x_r}\ldots t'_n\frac{t_0\ldots t_r}{x_0\ldots x_r}$

(c) $[\neg\varphi]\frac{t_0\ldots t_r}{x_0\ldots x_r} := \neg[\varphi\frac{t_0\ldots t_r}{x_0\ldots x_r}]$

(d) $(\varphi \vee \psi)\frac{t_0\ldots t_r}{x_0\ldots x_r} := \left(\varphi\frac{t_0\ldots t_r}{x_0\ldots x_r} \vee \psi\frac{t_0\ldots t_r}{x_0\ldots x_r}\right)$

(e) Suppose $x_{i_1},\ldots,x_{i_s}$ $(i_1 < \ldots < i_s)$ are exactly the variables x_i among the $x_0,\ldots,x_r$, such that

$$x_i \in \text{free}(\exists x\varphi) \text{ and } x_i \neq t_i.$$

In particular, $x \neq x_{i_1},\ldots,x \neq x_{i_s}$. Then set

$$[\exists x\varphi]\frac{t_0\ldots t_r}{x_0\ldots x_r} := \exists u\left[\varphi\frac{t_{i_1}\ldots t_{i_s}u}{x_{i_1}\ldots x_{i_s}x}\right],$$

where u is the variable x if x does not occur in $t_{i_1},\ldots,t_{i_s}$; otherwise u is the first variable in the list $v_0, v_1, v_2, \ldots$ which does not occur in $\varphi, t_{i_1},\ldots,t_{i_s}$.

(By introducing the variable u we ensure that no variable occurring in $t_{i_1},\ldots,t_{i_s}$ falls within the scope of a quantifier. In case there is no x_i such that $x_i \in \text{free}(\exists x\varphi)$ and $x_i \neq t_i$, we have $s = 0$, and from (e) we obtain

$$[\exists x\varphi]\frac{t_0\ldots t_r}{x_0\ldots x_r} = \exists x\left[\varphi\frac{x}{x}\right]$$

which is $\exists x\varphi$, as we shall see in 8.4(b).)

Examples. For binary P and f we have

(1) $[Pv_0fv_1v_2]\frac{v_2v_0v_1}{v_1v_2v_3} = Pv_0fv_2v_0$.

(2) $[\exists v_0Pv_0fv_1v_2]\frac{v_4\ fv_1v_1}{v_0\quad v_2} = \exists v_0\left[Pv_0fv_1v_2\frac{fv_1v_1\ v_0}{v_2\quad v_0}\right]$
$= \exists v_0Pv_0fv_1fv_1v_1$.

(3) $[\exists v_0Pv_0fv_1v_2]\frac{v_0v_2v_4}{v_1v_2v_0} = \exists v_3\left[Pv_0fv_1v_2\frac{v_0v_3}{v_1v_0}\right] = \exists v_3Pv_3fv_0v_2$.

At the places where x_i occurred free in φ, we now find in $\varphi\frac{t_0\ldots t_r}{x_0\ldots x_r}$ the term t_i. Hence, if $\text{free}(\varphi) \subset \{x_0,\ldots,x_r\}$ then we expect that $\varphi\frac{t_0\ldots t_r}{x_0\ldots x_r}$ will hold for an interpretation $\mathfrak{I} = (\mathfrak{A},\beta)$ iff φ holds in $\mathfrak{A}$, provided we use the assignments $\mathfrak{I}(t_0)$ for x_0, ..., $\mathfrak{I}(t_r)$ for x_r. An exact formulation of this property is given in the following "Substitution Lemma" 8.3. Later we shall frequently refer to this lemma, whereas we shall rarely return to the technical details of definition 8.2[7].

Before stating the lemma we generalize the definition of $\mathfrak{I}\frac{a}{x}$: Let $x_0,\ldots,x_r$ be pairwise distinct and suppose $\mathfrak{I} = (\mathfrak{A},\beta)$ is an interpretation, and $a_0,\ldots,a_r \in A$; then let $\beta\frac{a_0\ldots a_r}{x_0\ldots x_r}$ be the assignment in $\mathfrak{A}$ with

$$\beta\frac{a_0\ldots a_r}{x_0\ldots x_r}(y) := \begin{cases} \beta(y) & \text{if } y \neq x_0,\ldots,y \neq x_r \\ a_i & \text{if } y = x_i \end{cases}$$

and

$$\mathfrak{I}\frac{a_0\ldots a_r}{x_0\ldots x_r} := \left(\mathfrak{A},\beta\frac{a_0\ldots a_r}{x_0\ldots x_r}\right).$$

8.3 Substitution Lemma. (a) *For every term t,*

$$\mathfrak{I}\left(t\frac{t_0\ldots t_r}{x_0\ldots x_r}\right) = \mathfrak{I}\frac{\mathfrak{I}(t_0)\ldots\mathfrak{I}(t_r)}{x_0\ldots x_r}(t).$$

(b) *For every formula φ,*

$$\mathfrak{I} \models \varphi\frac{t_0\ldots t_r}{x_0\ldots x_r} \quad \textit{iff} \quad \mathfrak{I}\frac{\mathfrak{I}(t_0)\ldots\mathfrak{I}(t_r)}{x_0\ldots x_r} \models \varphi.$$

Proof. We proceed by induction on terms and formulas in accordance with the definitions 8.1 and 8.2. We treat some typical cases.

$t = x$: If $x \neq x_0,\ldots,x \neq x_r$ then, by 8.1(a),

$$x\frac{t_0\ldots t_r}{x_0\ldots x_r} = x$$

and therefore

$$\mathfrak{I}\left(x\frac{t_0\ldots t_r}{x_0\ldots x_r}\right) = \mathfrak{I}(x) = \mathfrak{I}\frac{\mathfrak{I}(t_0)\ldots\mathfrak{I}(t_r)}{x_0\ldots x_r}(x).$$

If $x = x_i$ then

$$x\frac{t_0\ldots t_r}{x_0\ldots x_r} = t_i$$

and hence

$$\mathfrak{I}\left(x\frac{t_0\ldots t_r}{x_0\ldots x_r}\right) = \mathfrak{I}(t_i) = \mathfrak{I}\frac{\mathfrak{I}(t_0)\ldots\mathfrak{I}(t_r)}{x_0\ldots x_r}(x_i) = \mathfrak{I}\frac{\mathfrak{I}(t_0)\ldots\mathfrak{I}(t_r)}{x_0\ldots x_r}(x).$$

$\varphi = Rt'_1\ldots t'_n$: $\mathfrak{I} \models [Rt'_1\ldots t'_n]\frac{t_0\ldots t_r}{x_0\ldots x_r}$

---

[7]Like the Substitution Lemma, the subsequent results of this section are intuitively clear. The proofs are straightforward but lengthy, and may be skipped by a reader already familiar with proofs by induction on terms and formulas.

iff $\mathfrak{I}(R)$ holds for $\mathfrak{I}\left(t_1'\frac{t_0\dots t_r}{x_0\dots x_r}\right),\dots$ (by 8.2(b))

iff $\mathfrak{I}(R)$ holds for $\mathfrak{I}\frac{\mathfrak{I}(t_0)\dots\mathfrak{I}(t_r)}{x_0\dots x_r}(t_1'),\dots$ (by (a))

iff $\mathfrak{I}\frac{\mathfrak{I}(t_0)\dots\mathfrak{I}(t_r)}{x_0\dots x_r}(R)$ holds for $\mathfrak{I}\frac{\mathfrak{I}(t_0)\dots\mathfrak{I}(t_r)}{x_0\dots x_r}(t_1'),\dots$

iff $\mathfrak{I}\frac{\mathfrak{I}(t_0)\dots\mathfrak{I}(t_r)}{x_0\dots x_r}\models Rt_1'\dots t_n'$.

$\varphi=\exists x\psi$: As in 8.2(e), let $x_{i_1},\dots,x_{i_s}$ be exactly those variables x_i for which $x_i\in\text{free}(\exists x\psi)$ and $x_i\neq t_i$. Then, for u chosen as in 8.2(e),

$\mathfrak{I}\models[\exists x\psi]\frac{t_0\dots t_r}{x_0\dots x_r}$

iff $\mathfrak{I}\models\exists u\left[\psi\frac{t_{i_1}\dots t_{i_s}\ u}{x_{i_1}\dots x_{i_s}\ x}\right]$

iff there is an $a\in A$ such that $\mathfrak{I}\frac{a}{u}\models\psi\frac{t_{i_1}\dots t_{i_s}\ u}{x_{i_1}\dots x_{i_s}\ x}$

iff there is an $a\in A$ such that $\left[\mathfrak{I}\frac{a}{u}\right]\frac{\mathfrak{I}\frac{a}{u}(t_{i_1})\dots\mathfrak{I}\frac{a}{u}(t_{i_s})\ \mathfrak{I}\frac{a}{u}(u)}{x_{i_1}\dots x_{i_s}\qquad x}\models\psi$ (by induction hypothesis)

iff there is an $a\in A$ such that $\left[\mathfrak{I}\frac{a}{u}\right]\frac{\mathfrak{I}(t_{i_1})\dots\mathfrak{I}(t_{i_s})\ a}{x_{i_1}\dots x_{i_s}\qquad x}\models\psi$ (by the Coincidence Lemma, since u does not occur in $t_{i_1},\dots,t_{i_s}$)

iff there is an $a\in A$ such that $\mathfrak{I}\frac{\mathfrak{I}(t_{i_1})\dots\mathfrak{I}(t_{i_s})\ a}{x_{i_1}\dots x_{i_s}\qquad x}\models\psi$ (since $u=x$ or u does not occur in ψ (Coincidence Lemma!))

iff there is an $a\in A$ such that $\left[\mathfrak{I}\frac{\mathfrak{I}(t_{i_1})\dots\mathfrak{I}(t_{i_s})}{x_{i_1}\dots x_{i_s}}\right]\frac{a}{x}\models\psi$ (note that $x\neq x_{i_1},\dots,x\neq x_{i_s}$)

iff $\mathfrak{I}\frac{\mathfrak{I}(t_{i_1})\dots\mathfrak{I}(t_{i_s})}{x_{i_1}\dots x_{i_s}}\models\exists x\psi$

iff $\mathfrak{I}\frac{\mathfrak{I}(t_0)\dots\mathfrak{I}(t_r)}{x_0\dots x_r}\models\exists x\psi$ (since for $i\neq i_1,\dots,i\neq i_s$, $x_i\notin\text{free}(\exists x\psi)$ or $x_i=t_i$). □

In the following lemmas we collect several "syntactic" properties of substitution.

8.4 Lemma. (a) *For every permutation π of the numbers $0,\dots,r$,*

$$\varphi\frac{t_0\dots t_r}{x_0\dots x_r}=\varphi\frac{t_{\pi(0)}\dots t_{\pi(r)}}{x_{\pi(0)}\dots x_{\pi(r)}}.$$

(b) *If* $0 \le i \le r$ *and* $x_i = t_i$, *then*

$$\varphi\frac{t_0 \dots t_r}{x_0 \dots x_r} = \varphi\frac{t_0 \dots t_{i-1}\ t_{i+1} \dots t_r}{x_0 \dots x_{i-1}\ x_{i+1} \dots x_r}.$$

In particular, $\varphi\frac{x}{x} = \varphi$.

(c) *For every variable* y,

(i) *if* $y \in \mathrm{var}\left(t\frac{t_0 \dots t_r}{x_0 \dots x_r}\right)$ *then* $(y \in \mathrm{var}(t_0) \cup \dots \cup \mathrm{var}(t_r))$ *or* $(y \in \mathrm{var}(t)$ *and* $y \ne x_0, \dots, y \ne x_r)$;

(ii) *if* $y \in \mathrm{free}\left(\varphi\frac{t_0 \dots t_r}{x_0 \dots x_r}\right)$ *then* $(y \in \mathrm{var}(t_0) \cup \dots \cup \mathrm{var}(t_r))$ *or* $(y \in \mathrm{free}(\varphi)$ *and* $y \ne x_0, \dots, y \ne x_r)$.

Proof. By induction, using 8.1 and 8.2. We give two typical cases of (c).

$t = x$: In case $x \ne x_0, \dots, x \ne x_r$ we have $x\frac{t_0 \dots t_r}{x_0 \dots x_r} = x$. Suppose $y \in \mathrm{var}\left(x\frac{t_0 \dots t_r}{x_0 \dots x_r}\right)$, then $y = x$ and so $(y \in \mathrm{var}(x)$ and $y \ne x_0, \dots, y \ne x_r)$. In case $x = x_i$ we have $x_i\frac{t_0 \dots t_r}{x_0 \dots x_r} = t_i$. Suppose $y \in \mathrm{var}\left(x_i\frac{t_0 \dots t_r}{x_0 \dots x_r}\right)$, then $y \in \mathrm{var}(t_i)$, therefore $y \in \mathrm{var}(t_0) \cup \dots \cup \mathrm{var}(t_r)$.

$\varphi = \exists x\psi$: Let $s, i_1, \dots, i_s$ and u be as in Definition 8.2.(e) and let

$$y \in \mathrm{free}\left([\exists x\psi]\frac{t_0 \dots t_r}{x_0 \dots x_r}\right) = \mathrm{free}\left(\exists u\left[\psi\frac{t_{i_1} \dots t_{i_s}\ u}{x_{i_1} \dots x_{i_s}\ x}\right]\right).$$

Then $y \ne u$ and

$$y \in \mathrm{free}\left(\psi\frac{t_{i_1} \dots t_{i_s}\ u}{x_{i_1} \dots x_{i_s}\ x}\right);$$

thus, by induction hypothesis, $y \ne u$ and $(y \in \mathrm{var}(t_{i_1}) \cup \dots \cup \mathrm{var}(t_{i_s}) \cup \{u\}$ or $y \in \mathrm{free}(\psi)$, $y \ne x_{i_1}, \dots, y \ne x_{i_s}, y \ne x)$. Since for $i \ne i_1, \dots, i \ne i_s$ we have $x_i \notin \mathrm{free}(\psi)$ or $x_i = t_i$, it follows that $y \in \mathrm{var}(t_0) \cup \dots \cup \mathrm{var}(t_r)$ or $y \in \mathrm{free}(\exists x\psi), y \ne x_0, \dots, y \ne x_r$. □

8.5 Corollary. *Suppose* $\mathrm{free}(\varphi) \subset \{x_0, \dots, x_r\}$, *where we continue to assume that* $x_0, \dots, x_r$ *are distinct. Then, for terms* $t_0, \dots, t_r$ *such that* $\mathrm{var}(t_i) \subset \{v_0, \dots, v_{n-1}\}$, *the formula* $\varphi\frac{t_0 \dots t_r}{x_0 \dots x_r}$ *is in* L_n^S. *In particular,* $\varphi\frac{c_0 \dots c_r}{x_0 \dots x_r}$ *is a sentence.* □

We call the number of connectives and quantifiers occurring in a formula φ the *rank* of φ, written $\mathrm{rk}(\varphi)$. More precisely:

8.6 Definition.

$$\begin{aligned}
\mathrm{rk}(\varphi) &:= 0 \text{ if } \varphi \text{ is atomic}\\
\mathrm{rk}(\neg\varphi) &:= \mathrm{rk}(\varphi) + 1\\
\mathrm{rk}(\varphi \vee \psi) &:= \mathrm{rk}(\varphi) + \mathrm{rk}(\psi) + 1\\
\mathrm{rk}(\exists x\varphi) &:= \mathrm{rk}(\varphi) + 1.
\end{aligned}$$

From the definition of substitution one immediately obtains:

8.7 Lemma. $\mathrm{rk}\left(\varphi\frac{t_0\ldots t_r}{x_0\ldots x_r}\right) = \mathrm{rk}(\varphi)$. □

The quantifier "there exists exactly one" can be conveniently formulated with the use of substitution. Let φ be a formula, x a variable, and y the first variable which is different from x and does not occur free in φ. Then we write $\exists^{=1}x\varphi$ ("there is exactly one x such that φ") for $\exists x(\varphi\wedge\forall y(\varphi\frac{y}{x}\to x\equiv y))$. It can easily be shown that for every interpretation $\mathfrak{I}=(\mathfrak{A},\beta)$,

$$\mathfrak{I}\models\exists^{=1}x\varphi \quad \text{iff} \quad \text{there is exactly one } a\in A \text{ such that } \mathfrak{I}\tfrac{a}{x}\models\varphi.$$

8.8 Exercise. For $n\geq 1$ give a similar definition of the quantifiers "there exist at most n" and "there exist exactly n."

8.9 Exercise. Let P and f be binary and set $x=v_0, y=v_1, u=v_2, v=v_3$, and $w=v_4$. Show, using Definition 8.2, that

(a) $\exists x\exists y(Pxu\wedge Pyv)\frac{u\ u\ u}{x\ y\ v} = \exists x\exists y(Pxu\wedge Pyu)$,

(b) $\exists x\exists y(Pxu\wedge Pyv)\frac{v\ fuv}{u\ \ v} = \exists x\exists y(Pxv\wedge Pyfuv)$,

(c) $\exists x\exists y(Pxu\wedge Pyv)\frac{u\ x\ fuv}{x\ u\ \ v} = \exists w\exists y(Pwx\wedge Pyfuv)$,

(d) $[\forall x\exists y(Pxy\wedge Pxu)\vee\exists ufuu\equiv x]\frac{x\ fxy}{x\ \ u} =$
$\forall v\exists w(Pvw\wedge Pvfxy)\vee\exists ufuu\equiv x.$

8.10 Exercise. Show that if $x_0,\ldots,x_r\notin\mathrm{var}(t_0)\cup\ldots\cup\mathrm{var}(t_r)$ then

$$\varphi\frac{t_0\ldots t_r}{x_0\ldots x_r} \mathrel{=\!\!|\!\models} \forall x_0\ldots\forall x_r(x_0\equiv t_0\wedge\ldots\wedge x_r\equiv t_r\to\varphi).$$

8.11 Exercise. Give a calculus for which the derivable strings are exactly those of the form $tx_0\ldots x_rt_0\ldots t_rt\frac{t_0\ldots t_r}{x_0\ldots x_r}$ or $\varphi x_0\ldots x_rt_0\ldots t_r\varphi\frac{t_0\ldots t_r}{x_0\ldots x_r}$. Hint: For (a) and (c) in 8.1 one can choose the following rules:

$$\frac{}{x\quad x_0\ldots x_r\quad t_0\ldots t_r\quad x}\qquad \text{if } x\neq x_0,\ldots,x\neq x_r;$$

$$\frac{}{x\quad x_0\ldots x_r\quad t_0\ldots t_r\quad t_i}\qquad \text{if } x=x_i;$$

$$\frac{\begin{array}{cccc} t'_1 & x_0\ldots x_r & t_0\ldots t_r & s'_1\\ \vdots & \vdots & \vdots & \vdots\\ t'_n & x_0\ldots x_r & t_0\ldots t_r & s'_n\end{array}}{ft'_1\ldots t'_n\quad x_0\ldots x_r\quad t_0\ldots t_r\quad fs'_1\ldots s'_n}\qquad \text{if } f\in S \text{ and } f \text{ is } n\text{-ary.}$$

IV

A Sequent Calculus

In Chapter I we discussed the way a mathematician proceeds to develop a particular mathematical theory: In order to obtain an overview of the theory, he tries to find out which propositions follow from its axioms. To show that a proposition follows from the axioms, he supplies a proof. Now that we have an exact definition of the notion of consequence, we are sufficiently equipped to give a more thorough discussion of the goals and methods in mathematics. If S is a symbol set and Φ is a set of S-sentences, we let $\Phi^{\models}$ be the set of S-sentences which are consequences of Φ. A mathematical proof of an S-sentence φ from the axioms in Φ shows that φ belongs to $\Phi^{\models}$. For example, consider the set Φ_{gr} of axioms for groups, where $S = S_{\mathrm{gr}}$. The proof of I.1.1 then shows us that the S_{gr}-sentence $\forall x \exists y\, y \circ x \equiv e$ belongs to $\Phi_{\mathrm{gr}}^{\models}$. However, in view of the goals of the mathematician and the scope of his methods, a central question is whether *every* sentence in $\Phi^{\models}$ can be proved from the axioms in Φ. In order to answer this we must analyze the notion of proof. But even if we limit ourselves to statements which can be formulated in first-order logic, we encounter difficulties at the very outset of such an attempt. The difficulties arise from the fact that mathematicians do not have an exact notion of proof. A mathematician does not learn what a proof is from a list of permissible inferences; rather he gets acquainted with this notion by doing concrete proofs in the course of his mathematical education. Furthermore, the collection of commonly accepted methods of proof is continually being expanded by the addition of new variants. Last, but not least, the development of new theories often includes the invention of new proof techniques.

In view of this situation we shall not attempt to give an exact description of the whole spectrum of mathematical arguments. Rather we shall look at some concrete proofs and try to abstract from them certain basic constituents. From these constituents we shall build up a precise notion of proof. It will turn out that they are sufficient to reconstruct all types of mathematical arguments. Thus, we proceed as we did when we introduced the precise notion of mathematical statement, where instead of trying to give an exact description we used the first-order languages to give a clearly defined framework. In the case of first-order languages we shall merely be able to make it plausible that, in spite of their limited expressive power,

these languages are in principle sufficient for the purpose of mathematics (cf. VII.2). By contrast, we can really prove that every sentence in $\Phi^{\models}$ is provable from sentences in Φ in the precise sense.

How can we single out basic constituents of mathematical deductions? If we analyze the proofs in Chapter I, for example, we see that those steps which are directly related to the meaning of connectives, the quantifiers, and the equality symbol seem very elementary. We mention three examples. In a proof one can proceed from statements φ and ψ, which have already been obtained, to the conjunction $(\varphi \wedge \psi)$; similarly one can proceed from Pt to $\exists x Px$, and from Px and $x \equiv t$ to Pt. We can represent these rules by the following schemes:

$$(*) \qquad \frac{\varphi, \psi}{(\varphi \wedge \psi)}, \quad \frac{Pt}{\exists x Px}, \quad \frac{Px,\ x \equiv t}{Pt}.$$

Written in this way, these constituents of proofs can be regarded as syntactic operations on strings of symbols. Adhering consistently to this point of view, we shall set up a list of deduction rules (in Sections 2 and 4), in this way obtaining a *calculus* $\mathfrak{S}$. We shall motivate its form in Section 1. In Section 6 (with a preview in Section 1) we shall give the fundamental definition for the notion of a formula φ being *formally provable* from a set Φ of formulas. This definition will be based on the notion of derivability in $\mathfrak{S}$. Formal provability is the syntactic counterpart of the semantic notion of consequence.

Throughout this chapter we fix a symbol set S.

§1. Sequent Rules

A mathematical proof proceeds from one statement to the next until it finally arrives at the assertion of the theorem in question. The individual statements depend on certain hypotheses. These can either be hypotheses of the theorem or additional hypotheses temporarily assumed in the course of the proof. For example, if one wants to prove an intermediate claim φ by contradiction one adds $\neg\varphi$ to the hypotheses; if a contradiction results then φ has been proved, and the additional assumption $\neg\varphi$ is dropped.

This observation leads us to describe a stage in a proof by listing the corresponding assumptions and the respective claim. If we call a nonempty list (sequence) of formulas a *sequent*, then we can use sequents to describe "stages in a proof". For instance, the "stage" with assumptions $\varphi_1, \ldots, \varphi_n$ and claim φ is rendered by the sequent $\varphi_1 \ldots \varphi_n\ \varphi$. The sequence $\varphi_1 \ldots \varphi_n$ is called the *antecedent* and φ the *succedent* of the sequent $\varphi_1 \ldots \varphi_n\ \varphi$. From II.4.3 it follows that the formulas which constitute a sequent are

uniquely determined. In particular, the antecedent and the succedent are well-defined.

In terms of sequents, the indirect proof sketched above can be represented schematically as follows:

$$(+) \qquad \begin{array}{lll} \varphi_1 \ldots \varphi_n & \neg\varphi & \psi \\ \varphi_1 \ldots \varphi_n & \neg\varphi & \neg\psi \\ \hline \varphi_1 \ldots \varphi_n & & \varphi \end{array}$$

Thus (+) describes the following argument: If under the assumptions φ_1, $\ldots, \varphi_n$ and (the additional assumption) $\neg\varphi$ one can obtain both the formula ψ and its negation $\neg\psi$ (that is, a contradiction), then from the assumptions $\varphi_1, \ldots, \varphi_n$ one can infer φ.

In the following we shall use the letters $\Gamma, \Delta, \ldots$ to denote (possibly empty) sequences of formulas. Then we can write sequents as $\Gamma\varphi\psi$, $\Delta\psi$, ... and the scheme (+) in the form

$$(++) \qquad \begin{array}{lll} \Gamma & \neg\varphi & \psi \\ \Gamma & \neg\varphi & \neg\psi \\ \hline \Gamma & & \varphi \end{array}$$

(As in (+), we use spaces between elements in a sequent merely for easier reading.)

According to the concepts we have developed so far, each step in a proof leads from certain "stages" already attained to a new one and hence, from sequents to a new sequent. Thus it seems natural to represent deduction rules such as (++) as rules of a calculus $\mathfrak{S}$, which operates on sequents (*sequent calculus*). Our conception of $\mathfrak{S}$ is based upon [16]. For comparison the reader can find calculi of a different nature in [30].

Before listing the rules of $\mathfrak{S}$ in the next section, some further remarks will be helpful.

If, in the calculus $\mathfrak{S}$, there is a derivation of the sequent $\Gamma\varphi$, then we write $\vdash \Gamma\varphi$ and say that $\Gamma\varphi$ is *derivable*.

1.1 Definition. A formula φ is *formally provable* or *derivable* from a set Φ of formulas (written: $\Phi \vdash \varphi$) if and only if there are finitely many formulas $\varphi_1, \ldots, \varphi_n$ in Φ such that $\vdash \varphi_1 \ldots \varphi_n\ \varphi$.

A sequent $\Gamma\varphi$ is called *correct* if $\Gamma \models \varphi$, more precisely, if $\{\psi \mid \psi$ is a member of $\Gamma\} \models \varphi$. Since the rules of $\mathfrak{S}$ are modelled after usual mathematical inferences, it will turn out that they are *correct*, i.e. when applied to correct sequents they yield a correct sequent. As a result, every formula which is derivable from Φ also follows from Φ. We convince ourselves of the correctness of each rule as soon as we introduce it.

§2. Structural Rules and Connective Rules

We divide the rules of the sequent calculus $\mathfrak{S}$ into the following categories: *structural rules* (2.1, 2.2), *connective rules* (2.3, 2.4, 2.5, 2.6), *quantifier rules* (4.1, 4.2) and *equality rules* (4.3, 4.4). We start with the two structural rules.

2.1 Antecedent Rule (Ant).

$$\frac{\Gamma \quad \varphi}{\Gamma' \quad \varphi} \quad \textit{if every member of } \Gamma \textit{ is also a member of } \Gamma' \textit{ (briefly: if } \Gamma \subset \Gamma').$$

Note that a formula which occurs more than once in Γ need only occur once in Γ'.

2.2 Assumption Rule (Assm).

$$\frac{}{\Gamma \quad \varphi} \quad \textit{if } \varphi \textit{ is a member of } \Gamma.$$

Correctness. (Ant): If a sequent $\Gamma\varphi$ is correct and $\Gamma \subset \Gamma'$, then since $\Gamma \models \varphi$, also $\Gamma' \models \varphi$.
(Assm) is correct since $\Phi \models \varphi$ always holds for $\varphi \in \Phi$. □

(Assm) reflects the trivial fact that one can conclude φ from a set of assumptions which includes φ. (Ant) expresses the fact that one can re-order or add to assumptions.

Now we state the connective rules. (Remember that we restricted ourselves to the connectives $\neg$ and $\vee$; cf. (1) on page 36.) The first rule is concerned with negation and incorporates the commonly used method of *proof by cases*. In order to conclude φ from Γ one first considers the case where a condition ψ holds and then treats the case where $\neg\psi$ holds. That is, one first has ψ and then $\neg\psi$ as an additional assumption. We can translate this argument into a rule for sequents as follows:

2.3 Proof by Cases Rule (PC).

$$\frac{\begin{array}{lll} \Gamma & \psi & \varphi \\ \Gamma & \neg\psi & \varphi \end{array}}{\begin{array}{ll} \Gamma & \varphi \end{array}}$$

Correctness. Suppose $\Gamma\psi \models \varphi$ and $\Gamma\neg\psi \models \varphi$ hold. We must show that $\Gamma \models \varphi$. Let $\mathfrak{I}$ be any interpretation such that $\mathfrak{I} \models \Gamma$, i.e. $\mathfrak{I} \models \chi$ for every member χ of Γ. Either $\mathfrak{I} \models \psi$ or $\mathfrak{I} \models \neg\psi$. If $\mathfrak{I} \models \psi$ then, since $\Gamma\psi \models \varphi$, it follows that $\mathfrak{I} \models \varphi$. If $\mathfrak{I} \models \neg\psi$, one obtains the same result because $\Gamma\neg\psi \models \varphi$. □

As the second rule concerning negation we take the schema (++) given in Section 1:

2.4 Contradiction Rule (Ctr).

$$\begin{array}{lll} \Gamma & \neg\varphi & \psi \\ \Gamma & \neg\varphi & \neg\psi \\ \hline \Gamma & & \varphi \end{array}$$

Correctness. Let $\Gamma\neg\varphi \models \psi$ and $\Gamma\neg\varphi \models \neg\psi$. Then there is no interpretation satisfying $\Gamma\neg\varphi$; hence any interpretation satisfying Γ must satisfy φ, i.e. $\Gamma\varphi$ is correct. □

2.5 $\vee$-Rule for the Antecedent ($\vee$A).

$$\begin{array}{lll} \Gamma & \varphi & \chi \\ \Gamma & \psi & \chi \\ \hline \Gamma & (\varphi \vee \psi) & \chi \end{array}$$

The proof that this rule is correct is similar to that for (PC).

2.6 $\vee$-Rules for the Succedent ($\vee$S).

(a) $\dfrac{\Gamma \quad \varphi}{\Gamma \quad (\varphi \vee \psi)}$ (b) $\dfrac{\Gamma \quad \varphi}{\Gamma \quad (\psi \vee \varphi)}$

Correctness. Suppose $\Gamma \models \varphi$ and let $\mathfrak{I} \models \Gamma$. Then $\mathfrak{I} \models \varphi$ and hence both $\mathfrak{I} \models (\varphi \vee \psi)$ and $\mathfrak{I} \models (\psi \vee \varphi)$. □

2.7 Exercise. Decide whether the following rules are correct:

(a) $\begin{array}{lll} \Gamma & \varphi_1 & \psi_1 \\ \Gamma & \varphi_2 & \psi_2 \\ \hline \Gamma & (\varphi_1 \vee \varphi_2) & (\psi_1 \vee \psi_2) \end{array}$ (b) $\begin{array}{lll} \Gamma & \varphi_1 & \psi_1 \\ \Gamma & \varphi_2 & \psi_2 \\ \hline \Gamma & (\varphi_1 \vee \varphi_2) & (\psi_1 \wedge \psi_2) \end{array}$

§3. Derivable Connective Rules

Using the rules of $\mathfrak{S}$ which we have formulated so far, we can derive a number of sequents. In our first example we show that all sequents of the form $(\varphi \vee \neg\varphi)$ are derivable. Our notation is similar to that used for derivations in previous calculi (cf. II.3).

	1.	φ	φ	(Assm)
	2.	φ	$(\varphi \vee \neg\varphi)$	($\vee$S) applied to 1.
(*)	3.	$\neg\varphi$	$\neg\varphi$	(Assm)
	4.	$\neg\varphi$	$(\varphi \vee \neg\varphi)$	($\vee$S) applied to 3.
	5.		$(\varphi \vee \neg\varphi)$	(PC) applied to 2. and 4.

Let us consider the rule (TND) ("Tertium non datur")

$$\frac{}{(\varphi \vee \neg\varphi)}\,,$$

which is not a rule of $\mathfrak{S}$. If we add (TND) to $\mathfrak{S}$ we do not enlarge the set of derivable sequents. For if we are given a derivation of a sequent which uses rules of $\mathfrak{S}$ together with (TND), we can insert lines 1. – 4. of (*) directly before every sequent $(\varphi \vee \neg\varphi)$, which originally was introduced by (TND). In this way we obtain a derivation in $\mathfrak{S}$.

Rules for sequents, whose use in a derivation can be eliminated by a derivation schema like (*), and which, therefore, do not enlarge the set of derivable sequents, will be called *derivable rules*. Thus (TND) is a derivable rule. The use of such derivable rules contributes to the transparency of derivations in the sequent calculus. In the remainder of this section we give some useful examples, also including derivable rules with premises.

3.1 Second Contradiction Rule (Ctr').

$$\frac{\begin{array}{ll}\Gamma & \psi\\ \Gamma & \neg\psi\end{array}}{\begin{array}{ll}\Gamma & \varphi\end{array}}$$

Justification. (The justification shows that the rule is derivable. In the present case we have to show how one can use rules of $\mathfrak{S}$ to obtain the sequent $\Gamma\varphi$ from the "premises" $\Gamma\psi$ and $\Gamma\neg\psi$.)

1.	Γ		ψ	premise
2.	Γ		$\neg\psi$	premise
3.	Γ	$\neg\varphi$	ψ	(Ant) applied to 1.
4.	Γ	$\neg\varphi$	$\neg\psi$	(Ant) applied to 2.
5.	Γ		φ	(Ctr) applied to 3. and 4.

3.2 Chain Rule (Ch).

$$\frac{\begin{array}{lll}\Gamma & & \varphi\\ \Gamma & \varphi & \psi\end{array}}{\begin{array}{lll}\Gamma & & \psi\end{array}}$$

Justification.

1.	Γ		φ	premise
2.	Γ	φ	ψ	premise
3.	Γ	$\neg\varphi$	φ	(Ant) applied to 1.
4.	Γ	$\neg\varphi$	$\neg\varphi$	(Assm)
5.	Γ	$\neg\varphi$	ψ	(Ctr') applied to 3. and 4.
6.	Γ		ψ	(PC) applied to 2. and 5.

3.3 Contrapositon Rules (Cp).

(a) $\dfrac{\begin{array}{lll}\Gamma & \varphi & \psi\end{array}}{\begin{array}{lll}\Gamma & \neg\psi & \neg\varphi\end{array}}$ (b) $\dfrac{\begin{array}{lll}\Gamma & \neg\varphi & \neg\psi\end{array}}{\begin{array}{lll}\Gamma & \psi & \varphi\end{array}}$

(c) $\dfrac{\begin{array}{lll}\Gamma & \neg\varphi & \psi\end{array}}{\begin{array}{lll}\Gamma & \neg\psi & \varphi\end{array}}$ (d) $\dfrac{\begin{array}{lll}\Gamma & \varphi & \neg\psi\end{array}}{\begin{array}{lll}\Gamma & \psi & \neg\varphi\end{array}}$

Justification of (a).

1.	Γ	φ		ψ	premise
2.	Γ	$\neg\psi$	φ	ψ	(Ant) applied to 1.
3.	Γ	$\neg\psi$	φ	$\neg\psi$	(Assm)
4.	Γ	$\neg\psi$	φ	$\neg\varphi$	(Ctr′) applied to 2. and 3.
5.	Γ	$\neg\psi$	$\neg\varphi$	$\neg\varphi$	(Assm)
6.	Γ	$\neg\psi$		$\neg\varphi$	(PC) applied to 4. and 5.

3.4.

$$\frac{\begin{array}{ll}\Gamma & (\varphi \vee \psi)\\ \Gamma & \neg\varphi\end{array}}{\begin{array}{ll}\Gamma & \psi\end{array}}$$

Justification.

1.	Γ		$(\varphi \vee \psi)$	premise
2.	Γ		$\neg\varphi$	premise
3.	Γ	φ	$\neg\varphi$	(Ant) applied to 2.
4.	Γ	φ	φ	(Assm)
5.	Γ	φ	ψ	(Ctr′) applied to 4. and 3.
6.	Γ	ψ	ψ	(Assm)
7.	Γ	$(\varphi \vee \psi)$	ψ	($\vee$A) applied to 5. and 6.
8.	Γ		ψ	(Ch) applied to 1. and 7.

3.5 "Modus ponens".

$$\frac{\begin{array}{ll}\Gamma & (\varphi \to \psi)\\ \Gamma & \varphi\end{array}}{\begin{array}{ll}\Gamma & \psi\end{array}}, \quad \text{that is,} \quad \frac{\begin{array}{ll}\Gamma & (\neg\varphi \vee \psi)\\ \Gamma & \varphi\end{array}}{\begin{array}{ll}\Gamma & \psi\end{array}}$$

The justification is analogous to the one given for 3.4.

3.6 Exercise. Show that the following rules are derivable.

(a1) $\dfrac{\begin{array}{ll}\Gamma & \varphi\end{array}}{\begin{array}{ll}\Gamma & \neg\neg\varphi\end{array}}$ (a2) $\dfrac{\begin{array}{ll}\Gamma & \neg\neg\varphi\end{array}}{\begin{array}{ll}\Gamma & \varphi\end{array}}$

(b) $\dfrac{\begin{array}{ll}\Gamma & \varphi\\ \Gamma & \psi\end{array}}{\begin{array}{ll}\Gamma & (\varphi \wedge \psi)\end{array}}$ (c) $\dfrac{\begin{array}{lll}\Gamma & \varphi & \psi\end{array}}{\begin{array}{ll}\Gamma & (\varphi \to \psi)\end{array}}$

(d1) $\dfrac{\begin{array}{ll}\Gamma & (\varphi \wedge \psi)\end{array}}{\begin{array}{ll}\Gamma & \varphi\end{array}}$ (d2) $\dfrac{\begin{array}{ll}\Gamma & (\varphi \wedge \psi)\end{array}}{\begin{array}{ll}\Gamma & \psi\end{array}}$

§4. Quantifier and Equality Rules

Now we give two sequent rules of $\mathfrak{S}$ which involve the existential quantifier. The first is a generalization of a scheme already mentioned in the introduction to this chapter.

4.1 Rule for $\exists$-Introduction in the Succedent ($\exists$S).

$$\frac{\Gamma \quad \varphi\frac{t}{x}}{\Gamma \quad \exists x\varphi}$$

($\exists$S) says that we can conclude $\exists x\varphi$ from Γ if we have already obtained the "witness" t for this existence claim.

Correctness. Suppose $\Gamma \models \varphi\frac{t}{x}$. Let $\mathfrak{I}$ be an interpretation such that $\mathfrak{I} \models \Gamma$. By assumption, we have $\mathfrak{I} \models \varphi\frac{t}{x}$. Therefore, by the Substitution Lemma, $\mathfrak{I}\frac{\mathfrak{I}(t)}{x} \models \varphi$ and hence $\mathfrak{I} \models \exists x\varphi$. □

The second $\exists$-rule is more complicated, but it incorporates a method of argument that is frequently used. The aim is to prove a claim ψ from assumptions $\varphi_1, \ldots, \varphi_n, \exists x\varphi$. (On our formal level: to achieve a derivation of the sequent

$$(*) \qquad \varphi_1 \ldots \varphi_n \ \exists x\varphi \ \psi$$

in the sequent calculus.) According to the hypothesis $\exists x\varphi$, one assumes one has an example – denoted by a new variable y – which "satisfies φ" and uses it to prove ψ. (In the sequent calculus this corresponds to a derivation of

$$(**) \qquad \varphi_1 \ldots \varphi_n \ \varphi\frac{y}{x} \ \psi,$$

where y is not free in $(*)$.) Then one regards ψ as having been proved from $\varphi_1, \ldots, \varphi_n, \exists x\varphi$[1]. We can reproduce this argument in the sequent calculus by a rule which allows us to proceed from $(**)$ to $(*)$:

4.2 Rule for $\exists$-Introduction in the Antecedent ($\exists$A).

$$\frac{\Gamma \quad \varphi\frac{y}{x} \quad \psi}{\Gamma \quad \exists x\varphi \quad \psi} \qquad \textit{if } y \textit{ is not free in } \Gamma \ \exists x\varphi \ \psi.$$

Correctness. Suppose $\Gamma \ \varphi\frac{y}{x} \models \psi$ and y is not free in $\Gamma \ \exists x\varphi \ \psi$. Let the interpretation $\mathfrak{I} = (\mathfrak{A}, \beta)$ be a model of $\Gamma \ \exists x\varphi$. We must show that $\mathfrak{I} \models \psi$. First, there is an $a \in A$ such that $\mathfrak{I}\frac{a}{x} \models \varphi$. Using the Coincidence Lemma we can conclude $(\mathfrak{I}\frac{a}{y})\frac{a}{x} \models \varphi$ (for $x = y$ this is clear; for $x \neq y$ note that $y \notin \text{free}(\varphi)$ since otherwise $y \in \text{free}(\exists x\varphi)$ contrary to the assumption).

[1] Cf. the proof of I.1.1 with the use of y in line (1).

Because $\mathfrak{I}\frac{a}{y}(y) = a$ we have $(\mathfrak{I}\frac{a}{y})\frac{\mathfrak{I}\frac{a}{y}(y)}{x} \models \varphi$ and hence by the Substitution Lemma, $\mathfrak{I}\frac{a}{y} \models \varphi\frac{y}{x}$. From $\mathfrak{I} \models \Gamma$ and $y \notin \text{free}(\Gamma)$ we get $\mathfrak{I}\frac{a}{y} \models \Gamma$, again by the Coincidence Lemma; since $\Gamma\ \varphi\frac{y}{x} \models \psi$ we obtain $\mathfrak{I}\frac{a}{y} \models \psi$ and therefore $\mathfrak{I} \models \psi$ because $y \notin \text{free}(\psi)$. □

The condition on y made in (∃A) is essential. For example, the sequent $[x \equiv fy]\frac{y}{x}\ y \equiv fy$ is correct; however, the sequent $\exists x\, x \equiv fy\ \ y \equiv fy$, which we could obtain by applying (∃A) while ignoring this extra condition, is no longer correct. This can be verified, say, by an interpretation with domain $\mathbf{N}$, which interprets f as the successor function $n \mapsto n+1$ and y as 0.

From a formula $\varphi\frac{t}{x}$ it is not, in general, possible to recover either φ or t. For instance, the formula Rfy can be written as $Rx\ \frac{fy}{x}$ or as $Rfx\frac{y}{x}$. Therefore, in applications of the rules (∃S) and (∃A), we shall explicitly mention φ and t or φ and y if they are not clear from the notation.

The last two rules of $\mathfrak{S}$ arise from two basic properties of the equality relation.

4.3 Reflexivity Rule for Equality (≡).

$$\frac{}{t \equiv t}$$

4.4 Substitution Rule for Equality (Sub).

$$\frac{\Gamma \qquad \varphi\frac{t}{x}}{\Gamma \quad t \equiv t' \quad \varphi\frac{t'}{x}}$$

Correctness. (≡): trivial. (Sub): Suppose $\Gamma \models \varphi\frac{t}{x}$ and suppose $\mathfrak{I}$ satisfies $\Gamma\ t \equiv t'$. Then $\mathfrak{I} \models \varphi\frac{t}{x}$ and hence, by the Substitution Lemma, $\mathfrak{I}\frac{\mathfrak{I}(t)}{x} \models \varphi$; therefore since $\mathfrak{I}(t) = \mathfrak{I}(t')$ we have $\mathfrak{I}\frac{\mathfrak{I}(t')}{x} \models \varphi$. A further application of the Substitution Lemma yields finally that $\mathfrak{I} \models \varphi\frac{t'}{x}$. □

4.5 Exercise. Decide whether the following rules are correct:

$$\frac{\varphi \qquad \psi}{\exists x\varphi \quad \exists x\psi}\ ; \qquad \frac{\Gamma \quad \varphi \qquad \psi}{\Gamma \quad \forall x\varphi \quad \exists x\psi}\ ;$$

$$\frac{\Gamma \quad \varphi\frac{fy}{x}}{\Gamma \quad \forall x\varphi}$$ if f is unary, and f and y do not occur in $\Gamma\ \forall x\varphi$.

§5. Further Derivable Rules and Sequents

Since $\varphi\frac{x}{x} = \varphi$, we obtain from 4.1 and 4.2 (for $t = x$ and $y = x$) the following derivable rules:

5.1.

(a) $\dfrac{\Gamma \quad \varphi}{\Gamma \quad \exists x\varphi}$ (b) $\dfrac{\Gamma \quad \varphi \qquad \psi}{\Gamma \quad \exists x\varphi \quad \psi}$ *if x is not free in Γ ψ.*

A corresponding special case of (Sub) is

5.2.

$$\frac{\Gamma \qquad\qquad \varphi}{\Gamma \quad x \equiv t \quad \varphi\frac{t}{x}}\ .$$

We conclude with some derivable rules dealing with the *symmetry* and the *transitivity* of the equality relation and its *compatibility* with functions and relations.

5.3.

(a) $\dfrac{\Gamma \quad t_1 \equiv t_2}{\Gamma \quad t_2 \equiv t_1}$; (b) $\dfrac{\begin{array}{ll}\Gamma & t_1 \equiv t_2\\ \Gamma & t_2 \equiv t_3\end{array}}{\Gamma \quad t_1 \equiv t_3}$.

5.4. (a) *For n-ary $R \in S$:*

$$\frac{\begin{array}{ll}\Gamma & Rt_1 \ldots t_n\\ \Gamma & t_1 \equiv t'_1\\ \vdots & \\ \Gamma & t_n \equiv t'_n\end{array}}{\Gamma \quad Rt'_1 \ldots t'_n}\ ;$$

(b) *For n-ary $f \in S$:*

$$\frac{\begin{array}{ll}\Gamma & t_1 \equiv t'_1\\ \vdots & \\ \Gamma & t_n \equiv t'_n\end{array}}{\Gamma \quad ft_1 \ldots t_n \equiv ft'_1 \ldots t'_n}\ .$$

Justification of 5.3 and 5.4. Let x be a variable occurring neither in any of the terms nor in Γ.

5.3(a):

1.	Γ		$t_1 \equiv t_2$	premise
2.	Γ		$t_1 \equiv t_1$	($\equiv$) and (Ant)
3.	Γ	$t_1 \equiv t_2$	$t_2 \equiv t_1$	(Sub) applied to 2. with $t_1 \equiv t_1 = [x \equiv t_1]\frac{t_1}{x}$
4.	Γ		$t_2 \equiv t_1$	(Ch) applied to 1. and 3.

5.3(b):

1.	Γ		$t_1 \equiv t_2$	premise
2.	Γ		$t_2 \equiv t_3$	premise
3.	Γ	$t_2 \equiv t_3$	$t_1 \equiv t_3$	(Sub) applied to 1. with $t_1 \equiv t_2 = [t_1 \equiv x]\frac{t_2}{x}$
4.	Γ		$t_1 \equiv t_3$	(Ch) applied to 2. and 3.

5.4(a) (The justification for 5.4(b) is similar): W.l.o.g let $n = 2$.

1.	Γ		Rt_1t_2	premise
2.	Γ		$t_1 \equiv t_1'$	premise
3.	Γ		$t_2 \equiv t_2'$	premise
4.	Γ	$t_1 \equiv t_1'$	$Rt_1't_2$	(Sub) applied to 1. with $Rt_1t_2 = [Rxt_2]\frac{t_1}{x}$
5.	Γ		$Rt_1't_2$	(Ch) applied to 2. and 4.
6.	Γ	$t_2 \equiv t_2'$	$Rt_1't_2'$	(Sub) applied to 5. with $Rt_1't_2 = [Rt_1'x]\frac{t_2}{x}$
7.	Γ		$Rt_1't_2'$	(Ch) applied to 3. and 6.

5.5 Exercise. Show that the following rules are derivable:

(a1) $\dfrac{\Gamma \quad \forall x\varphi}{\Gamma \quad \varphi\frac{t}{x}}$, that is, $\dfrac{\Gamma \quad \neg\exists x\neg\varphi}{\Gamma \quad \varphi\frac{t}{x}}$ (a2) $\dfrac{\Gamma \quad \forall x\varphi}{\Gamma \quad \varphi}$

(b1) $\dfrac{\Gamma \quad \varphi\frac{t}{x} \quad \psi}{\Gamma \quad \forall x\varphi \quad \psi}$ (b2) $\dfrac{\Gamma \quad \varphi\frac{y}{x}}{\Gamma \quad \forall x\varphi}$ if y is not free in $\Gamma\ \forall x\varphi$

(b3) $\dfrac{\Gamma \quad \varphi \quad \psi}{\Gamma \quad \forall x\varphi \quad \psi}$ (b4) $\dfrac{\Gamma \quad \varphi}{\Gamma \quad \forall x\varphi}$ if x is not free in Γ.

§6. Summary and Example

For the reader's convenience, we list all the rules of $\mathfrak{S}$ together.

(Assm) $\dfrac{}{\Gamma \quad \varphi}$ if $\varphi \in \Gamma$ (Ant) $\dfrac{\Gamma \quad \varphi}{\Gamma' \quad \varphi}$ if $\Gamma \subset \Gamma'$

(PC) $\dfrac{\begin{array}{lll}\Gamma & \psi & \varphi \\ \Gamma & \neg\psi & \varphi\end{array}}{\Gamma \quad \varphi}$ (Ctr) $\dfrac{\begin{array}{lll}\Gamma & \neg\varphi & \psi \\ \Gamma & \neg\varphi & \neg\psi\end{array}}{\Gamma \quad \varphi}$

($\vee$A) $\dfrac{\begin{array}{lll}\Gamma & \varphi & \chi \\ \Gamma & \psi & \chi\end{array}}{\Gamma \quad (\varphi \vee \psi) \quad \chi}$ ($\vee$S) $\dfrac{\Gamma \quad \varphi}{\Gamma \quad (\varphi \vee \psi)}$, $\dfrac{\Gamma \quad \varphi}{\Gamma \quad (\psi \vee \varphi)}$

($\exists$A) $\dfrac{\Gamma \quad \varphi\frac{y}{x} \quad \psi}{\Gamma \quad \exists x\varphi \quad \psi}$ if y is not free in $\Gamma\ \exists x\varphi\ \psi$

($\exists$S) $\dfrac{\Gamma \quad \varphi\frac{t}{x}}{\Gamma \quad \exists x\varphi}$

($\equiv$) $\dfrac{}{t \equiv t}$ (Sub) $\dfrac{\Gamma \quad \varphi\frac{t}{x}}{\Gamma \quad t \equiv t' \quad \varphi\frac{t'}{x}}$

In 1.1 we defined a formula φ to be *derivable* (*formally provable*) from Φ (written: $\Phi \vdash \varphi$) if there are formulas $\varphi_1, \ldots, \varphi_n$ in Φ such that $\vdash \varphi_1 \ldots \varphi_n\ \varphi$. From this definition we immediately obtain:

6.1 Lemma. *For all Φ and φ, $\Phi \vdash \varphi$ if and only if there is a finite subset Φ_0 of Φ such that $\Phi_0 \vdash \varphi$.* □

We have already more or less proved the correctness of $\mathfrak{S}$:

6.2 Theorem on the Correctness of $\mathfrak{S}$. *For all Φ and φ, if $\Phi \vdash \varphi$ then $\Phi \models \varphi$.*

Proof. Suppose $\Phi \vdash \varphi$. Then for a suitable Γ from Φ (that is, a Γ whose members are formulas from Φ) we have $\vdash \Gamma\varphi$. As we showed, every rule without premises yields only correct sequents, and the other rules of $\mathfrak{S}$ always lead from correct sequents to correct sequents. Thus, by induction over $\mathfrak{S}$, we see that every derivable sequent is correct, hence also $\Gamma\varphi$. Therefore $\Gamma \models \varphi$ and so $\Phi \models \varphi$. □

We shall prove the converse of 6.2, namely "if $\Phi \models \varphi$ then $\Phi \vdash \varphi$", in the next chapter. In particular, it will follow that if φ is *mathematically* provable from Φ, and hence $\Phi \models \varphi$, then φ is also *formally* provable from Φ. However, because of the elementary character of the rules for sequents, a formal proof is in general considerably longer than the corresponding mathematical proof. As an example we give here a formal proof of the theorem

$$\forall x \exists y\, y \circ x \equiv e$$

(existence of a left inverse) from the group axioms

$$\begin{aligned} \varphi_0 &:= \forall x \forall y \forall z (x \circ y) \circ z \equiv x \circ (y \circ z), \\ \varphi_1 &:= \forall x\, x \circ e \equiv x, \\ \varphi_2 &:= \forall x \exists y\, x \circ y \equiv e. \end{aligned}$$

The reader should compare the formal proof below with the mathematical proof of the same theorem in I.1.1. The "chain of equations" given there corresponds to the underlined formulas in the derivation up to line 23. For simplicity we shall write "xy" instead of "$x \circ y$" and we put $\Gamma := \varphi_0\ \varphi_1\ \varphi_2$.

1. Γ		$\forall x\, xe \equiv x$	(Assm)
2. Γ		$(yx)e \equiv yx$	5.5(a1) applied to 1. with $t = yx$
3. Γ		$\underline{yx \equiv (yx)e}$	5.3(a) appl. to 2.
4. Γ	$e \equiv yz$	$yx \equiv (yx)(yz)$	(Sub) applied to 3.
5. Γ	$yz \equiv e$	$e \equiv yz$	5.3(a) and (Ant)
6. Γ	$yz \equiv e$	$\underline{yx \equiv (yx)(yz)}$	(Ant) and (Ch) appl. to 5. and 4.
7. Γ	$yz \equiv e$	$\forall x \forall y \forall z (xy)z \equiv x(yz)$	(Assm)

8.	$\Gamma \quad yz \equiv e$	$\forall u \forall v (yu)v \equiv y(uv)$	5.5(a1) applied to 7. with $t = y$
9.	$\Gamma \quad yz \equiv e$	$\forall w (yx)w \equiv y(xw)$	5.5(a1) applied to 8. with $t = x$
10.	$\Gamma \quad yz \equiv e$	$(yx)(yz) \equiv y(x(yz))$	5.5(a1) applied to 9. with $t = yz$
11.	$\Gamma \quad yz \equiv e$	$\underline{yx \equiv y(x(yz))}$	5.3(b) applied to 6. and 10.
12.	$\Gamma \quad yz \equiv e \; x(yz) \equiv (xy)z$	$yx \equiv y((xy)z)$	(Sub) appl. to 11.
13.	$\Gamma \quad yz \equiv e$	$(xy)z \equiv x(yz)$	5.5(a2) appl. three times to 7.
14.	$\Gamma \quad yz \equiv e$	$x(yz) \equiv (xy)z$	5.3(a) appl. to 13.
15.	$\Gamma \quad yz \equiv e$	$\underline{yx \equiv y((xy)z)}$	(Ch) applied to 14. and 12.
16.	$\Gamma \quad yz \equiv e \; xy \equiv e$	$\underline{yx \equiv y(ez)}$	(Sub) appl. to 15.
17.	$\Gamma \quad yz \equiv e \; xy \equiv e$	$(ye)z \equiv y(ez)$	with 5.5(a1) from φ_0 as for 10.
18.	$\Gamma \quad yz \equiv e \; xy \equiv e$	$y(ez) \equiv (ye)z$	5.3(a) appl. to 17.
19.	$\Gamma \quad yz \equiv e \; xy \equiv e$	$\underline{yx \equiv (ye)z}$	5.3(b) applied to 16. and 18.
20.	$\Gamma \quad yz \equiv e \; xy \equiv e \; ye \equiv y$	$yx \equiv yz$	(Sub) appl. to 19.
21.	$\Gamma \quad yz \equiv e \; xy \equiv e$	$ye \equiv y$	5.5(a1) applied to 1. with $t = y$ and (Ant)
22.	$\Gamma \quad yz \equiv e \; xy \equiv e$	$\underline{yx \equiv yz}$	(Ch) applied to 21. and 20.
23.	$\Gamma \quad xy \equiv e \; yz \equiv e$	$\underline{yx \equiv e}$	(Sub) and (Ant) applied to 22.
24.	$\Gamma \quad xy \equiv e \; yz \equiv e$	$\exists y \, yx \equiv e$	($\exists$S) applied to 23.
25.	$\Gamma \quad xy \equiv e \; \exists z \, yz \equiv e$	$\exists y \, yx \equiv e$	($\exists$A) applied to 24.
26.	$\Gamma \quad xy \equiv e \; \forall y \exists z \, yz \equiv e$	$\exists y \, yx \equiv e$	5.5(b3) appl. to 25.
27.	$xy \equiv e$	$xy \equiv e$	(Assm)
28.	$xy \equiv e$	$\exists z \, xz \equiv e$	($\exists$S) applied to 27.
29.	$\exists y \, xy \equiv e$	$\exists z \, xz \equiv e$	($\exists$A) applied to 28.
30.	$\forall x \exists y \, xy \equiv e$	$\exists z \, xz \equiv e$	5.5(b3) appl. to 29.
31.	φ_2	$\forall y \exists z \, yz \equiv e$	5.5(b2) appl. to 30.
32.	$\Gamma \quad xy \equiv e$	$\exists y \, yx \equiv e$	(Ant), (Ch) applied to 31. and 26.
33.	$\Gamma \quad \forall x \exists y \, xy \equiv e$	$\exists y \, yx \equiv e$	($\exists$A) and 5.5(b3) applied to 32.
34.	Γ	$\forall x \exists y \, yx \equiv e$	(Ant) and 5.5(b4) applied to 33.

§7. Consistency

The semantic concept $\models$ of the consequence relation corresponds to the syntactic concept $\vdash$ of derivability. As a syntactic counterpart to satisfiability we define the concept of *consistency*.

7.1 Definition. (a) Φ is *consistent* (written: Con Φ) if and only if there is no formula φ such that $\Phi \vdash \varphi$ and $\Phi \vdash \neg\varphi$.
(b) Φ is *inconsistent* (written: Inc Φ) if and only if Φ is not consistent (that is, if there is a formula φ such that $\Phi \vdash \varphi$ and $\Phi \vdash \neg\varphi$).

7.2 Lemma. *For a set of formulas Φ the following are equivalent:*
(a) Inc Φ.
(b) *For all φ:* $\Phi \vdash \varphi$.

Proof. (a) follows immediately from (b). Suppose, on the other hand, that Inc Φ holds, i.e. $\Phi \vdash \psi$ and $\Phi \vdash \neg\psi$ for some formula ψ. Let φ be an arbitrary formula. We show $\Phi \vdash \varphi$.

First of all, there exist Γ_1 and Γ_2, which consist of formulas from Φ, and derivations

$$\begin{array}{c} \vdots \\ \Gamma_1\psi \end{array} \quad \text{and} \quad \begin{array}{c} \equiv \\ \equiv \\ \Gamma_2\neg\psi \end{array} .$$

By using them, we obtain the following derivation:

$$\begin{array}{llll}
 & & \vdots & \\
m. & & \Gamma_1 & \psi \\
 & & \equiv & \\
n. & & \Gamma_2 & \neg\psi \\
(n+1). & \Gamma_1 & \Gamma_2 & \psi \quad \text{(Ant) applied to } m. \\
(n+2). & \Gamma_1 & \Gamma_2 & \neg\psi \quad \text{(Ant) applied to } n. \\
(n+3). & \Gamma_1 & \Gamma_2 & \varphi \quad \text{(Ctr}'\text{) applied to } (n+1)., (n+2).
\end{array}$$

Thus we see that $\Phi \vdash \varphi$. □

7.3 Corollary. *For a set of formulas Φ the following are equivalent:*
(a) Con Φ.
(b) *There is a formula φ which is not derivable from Φ.* □

Since $\Phi \vdash \varphi$ if and only if $\Phi_0 \vdash \varphi$ for a suitable finite subset Φ_0 of Φ, we obtain:

7.4 Lemma. *For all Φ,* Con Φ *if and only if* Con Φ_0 *for all finite subsets Φ_0 of Φ.* □

7.5 Lemma. *Every satisfiable set of formulas is consistent.*

Proof. Suppose Inc Φ. Then for a suitable φ both $\Phi \vdash \varphi$ and $\Phi \vdash \neg\varphi$; hence, by the theorem on the correctness of $\mathfrak{S}$, $\Phi \models \varphi$ and $\Phi \models \neg\varphi$. But then Φ cannot be satisfiable. □

Later we shall need:

7.6 Lemma. *For all* Φ *and* φ *with* Con Φ *the following holds:*
(a) $\Phi \vdash \varphi$ *iff* Inc $\Phi \cup \{\neg\varphi\}$.
(b) $\Phi \vdash \neg\varphi$ *iff* Inc $\Phi \cup \{\varphi\}$.
(c) Con $\Phi \cup \{\varphi\}$ *or* Con $\Phi \cup \{\neg\varphi\}$.

Proof. (a): If $\Phi \vdash \varphi$ then $\Phi \cup \{\neg\varphi\} \vdash \varphi$; since $\Phi \cup \{\neg\varphi\} \vdash \neg\varphi$, $\Phi \cup \{\neg\varphi\}$ is inconsistent. Conversely, let $\Phi \cup \{\neg\varphi\}$ be inconsistent. Then for a suitable Γ consisting of formulas from Φ, there is a derivation of the sequent $\Gamma\ \neg\varphi\ \varphi$. From this we obtain the following derivation:

$$\begin{array}{llll} & \vdots & & \\ \Gamma & \neg\varphi & \varphi & \\ \Gamma & \varphi & \varphi & \text{(Assm)} \\ \Gamma & & \varphi & \text{(PC).} \end{array}$$

This shows that $\Phi \vdash \varphi$.

(b): In the proof of (a) interchange the roles of φ and $\neg\varphi$.

(c): If neither Con $\Phi \cup \{\varphi\}$ nor Con $\Phi \cup \{\neg\varphi\}$, i.e. if Inc $\Phi \cup \{\varphi\}$ and Inc $\Phi \cup \{\neg\varphi\}$, then (by (b) and (a)) $\Phi \vdash \neg\varphi$ and $\Phi \vdash \varphi$. Hence Φ is inconsistent, a contradiction to the assumption Con Φ. □

In this chapter we have referred to a fixed symbol set S. Thus, when we spoke of formulas we understood them to be S-formulas, and when discussing the sequent calculus $\mathfrak{S}$ we actually referred to the particular calculus $\mathfrak{S}_S$ corresponding to the symbol set S. In some cases it is necessary to treat several symbol sets simultaneously. Then we insert indices for the sake of clarity. To be specific, we use the more precise notation $\Phi \vdash_S \varphi$ to indicate that there is a derivation in $\mathfrak{S}_S$ (consisting of S-formulas) whose last sequent is of the form $\Gamma\varphi$, where Γ consists of formulas from Φ. Similarly, we write $\mathrm{Con}_S\ \Phi$ if there is no S-formula φ such that $\Phi \vdash_S \varphi$ and $\Phi \vdash_S \neg\varphi$ [2].

[2]The reader should note that for two symbol sets S and S' with $S \subset S'$, and for $\Phi \subset L^S$ and $\varphi \in L^S$, it is conceivable that $\Phi \vdash_{S'} \varphi$ but not $\Phi \vdash_S \varphi$, for it could be that formulas from $L^{S'} \setminus L^S$ are used in *every* derivation of φ from Φ in $\mathfrak{S}_{S'}$, and that (later on in the proof) these formulas are then eliminated from the sequents, say by application of the rules (Ctr), (PC) or ($\exists$S). We shall show later on that this cannot happen.

In the next chapter we shall need:

7.7 Lemma. *For $n \in \mathbf{N}$, let S_n be symbol sets such that*

$$S_0 \subset S_1 \subset S_2 \subset \ldots,$$

and let Φ_n be sets of S_n-formulas such that $\mathrm{Con}_{S_n}\ \Phi_n$ *and*

$$\Phi_0 \subset \Phi_1 \subset \Phi_2 \subset \ldots.$$

Let $S = \bigcup_{n \in \mathbf{N}} S_n$ and $\Phi = \bigcup_{n \in \mathbf{N}} \Phi_n$. Then $\mathrm{Con}_S\ \Phi$.

Proof. Assume the hypotheses of the theorem, and suppose $\mathrm{Inc}_S\ \Phi$. Then, by 7.4, $\mathrm{Inc}_S\ \Psi$ must hold for a suitable *finite* subset Ψ of Φ. There is a k such that $\Psi \subset \Phi_k$ and hence $\mathrm{Inc}_S\ \Phi_k$; in particular, $\Phi_k \vdash_S v_0 \equiv v_0$ and $\Phi_k \vdash_S \neg v_0 \equiv v_0$. Suppose we are given S-derivations for these two formulas. Since they contain only a finite number of symbols, all the formulas occurring there are actually contained in some L^{S_m}. We may assume that $m \geq k$. Then both derivations are derivations in the S_m-sequent calculus, and therefore $\mathrm{Inc}_{S_m}\ \Phi_k$. Since $\Phi_k \subset \Phi_m$ we obtain $\mathrm{Inc}_{S_m}\ \Phi_m$, which contradicts the hypotheses of the theorem. □

7.8 Exercise. Define $(\exists\forall)$ to be the rule

$$\frac{}{\Gamma \quad \exists x\varphi \quad \forall x\varphi}\ .$$

(a) Determine whether $(\exists\forall)$ is a derivable rule.
(b) Let $\mathfrak{S}'$ be obtained from the calculus of sequents $\mathfrak{S}$ by adding the rule $(\exists\forall)$. Is every sequent derivable in $\mathfrak{S}'$?

V

The Completeness Theorem

The subject of this chapter is a proof of the completeness of the sequent calculus, i.e. the statement:

(*) For all Φ and φ: If $\Phi \models \varphi$ then $\Phi \vdash \varphi$.

In order to verify (*) we show:

(**) Every consistent set of formulas is satisfiable.

From this, (*) can be proved as follows: We assume for Φ and φ that $\Phi \models \varphi$, but not $\Phi \vdash \varphi$. Then $\Phi \cup \{\neg\varphi\}$ is consistent but not satisfiable (cf. III.4.4 and IV.7.6(a)), a contradiction to (**).

To establish (**) we have to find a model for any consistent set Φ of formulas. In Section 1 we shall see that there is a natural way to do this if Φ is *negation complete* and if it *contains witnesses.* Then we reduce the general case to this one: in Section 2 for at most countable symbol sets, and in Section 3 for arbitrary symbol sets. Unless stated otherwise, we refer to a fixed symbol set S.

§1. Henkin's Theorem

Let Φ be a consistent set of formulas. In order to find an interpretation $\mathfrak{I} = (\mathfrak{A}, \beta)$ satisfying Φ, we have at our disposal only the "syntactical" information given by the consistency of Φ. Hence, we shall try to obtain a model using syntactical objects as far as possible. A first idea is to take as domain A the set T^S of all S-terms, to define β by

$$\beta(v_i) := v_i \ (i \in \mathbf{N})$$

and to interpret, for instance, a unary function symbol f by

$$f^{\mathfrak{A}}(t) := ft$$

and a unary relation symbol R by

$$R^{\mathfrak{A}} := \{t \in A \mid \Phi \vdash Rt\}.$$

Then, for a variable x we have $\mathfrak{I}(fx) = f^{\mathfrak{A}}(\beta(x)) = fx$. Here a first difficulty arises concerning the equality symbol: If y is a variable different from x then $fx \neq fy$, hence $\mathfrak{I}(fx) \neq \mathfrak{I}(fy)$. If we choose Φ such that $\Phi \vdash fx \equiv fy$ (e.g. $\Phi = \{fx \equiv fy\}$), then $\mathfrak{I}$ is not a model of Φ. (Namely, by IV.6.2 it follows that $\Phi \models fx \equiv fy$, and with $\mathfrak{I} \models \Phi$ we would have $\mathfrak{I}(fx) = \mathfrak{I}(fy)$.)

We overcome this difficulty by defining an equivalence relation on terms and then using the equivalence classes rather than the individual terms as elements of the domain of $\mathfrak{I}$.

Let Φ be a set of formulas. We define an interpretation $\mathfrak{I}^{\Phi} = (\mathfrak{T}^{\Phi}, \beta^{\Phi})$. For this purpose we first introduce a binary relation $\sim$ on the set T^S of S-terms by

1.1. $t_1 \sim t_2$ *:iff* $\Phi \vdash t_1 \equiv t_2$.

1.2 Lemma. (a) *$\sim$ is an equivalence relation.*
(b) *$\sim$ is compatible with the symbols in S in the following sense:*

If $t_1 \sim t_1', \ldots, t_n \sim t_n'$ then for n-ary $f \in S$

$$ft_1 \ldots t_n \sim ft_1' \ldots t_n'$$

and for n-ary $R \in S$

$$\Phi \vdash Rt_1 \ldots t_n \quad \textit{iff} \quad \Phi \vdash Rt_1' \ldots t_n'.$$

The *proof* uses the rule ($\equiv$) and IV.5.3, 5.4. We give two cases as examples:

(1) $\sim$ is symmetric: Suppose $t_1 \sim t_2$, that is $\Phi \vdash t_1 \equiv t_2$. By IV.5.3(a) we obtain $\Phi \vdash t_2 \equiv t_1$, i.e. $t_2 \sim t_1$.

(2) Let f be an n-ary function symbol from S, and assume $t_1 \sim t_1', \ldots, t_n \sim t_n'$, i.e. $\Phi \vdash t_1 \equiv t_1', \ldots, \Phi \vdash t_n \equiv t_n'$. Then by IV.5.4(b), $\Phi \vdash ft_1 \ldots t_n \equiv ft_1' \ldots t_n'$, i.e. $ft_1 \ldots t_n \sim ft_1' \ldots t_n'$. □

Let $\overline{t}$ be the equivalence class of t:

$$\overline{t} := \{t' \in T^S \mid t \sim t'\},$$

and let T^{Φ} (more precisely: $T^{\Phi,S}$) be the set of equivalence classes:

$$T^{\Phi} := \{\overline{t} \mid t \in T^S\}.$$

The set T^{Φ} is not empty. We define the S-structure $\mathfrak{T}^{\Phi}$ over T^{Φ}, the so-called *term structure* corresponding to Φ, by the following clauses:

1.3. *For n-ary $R \in S$,*

$$R^{\mathfrak{T}^{\Phi}}\overline{t_1} \ldots \overline{t_n} \quad \text{:iff} \quad \Phi \vdash Rt_1 \ldots t_n.$$

1.4. *For n-ary $f \in S$,*

$$f^{\mathfrak{T}^{\Phi}}(\overline{t_1}, \ldots, \overline{t_n}) := \overline{ft_1 \ldots t_n}.$$

1.5. *For* $c \in S$, $c^{\mathfrak{T}^\Phi} := \overline{c}$.

By 1.2(b) the conditions in 1.3 and 1.4 are independent of the choice of the representatives $t_1, \ldots, t_n$ of $\overline{t_1}, \ldots, \overline{t_n}$, hence $R^{\mathfrak{T}^\Phi}$ and $f^{\mathfrak{T}^\Phi}$ are well defined.

Finally, we fix an assignment β^Φ by

1.6. $\beta^\Phi(x) := \overline{x}$.

We call $\mathfrak{I}^\Phi := \left(\mathfrak{T}^\Phi, \beta^\Phi\right)$ the *term interpretation* associated with Φ.

1.7 Lemma. (a) *For all* t, $\mathfrak{I}^\Phi(t) = \overline{t}$.
(b) *For every atomic formula* φ,

$$\mathfrak{I}^\Phi \models \varphi \quad \textit{iff} \quad \Phi \vdash \varphi.$$

(c) *For every formula* φ *and pairwise distinct variables* $x_1, \ldots, x_n$,
(i) $\mathfrak{I}^\Phi \models \exists x_1 \ldots \exists x_n \varphi$ *iff there are* $t_1, \ldots, t_n \in T^S$ *with*

$$\mathfrak{I}^\Phi \models \varphi \frac{t_1 \ldots t_n}{x_1 \ldots x_n}.$$

(ii) $\mathfrak{I}^\Phi \models \forall x_1 \ldots \forall x_n \varphi$ *iff for all terms* $t_1, \ldots, t_n \in T^S$,

$$\mathfrak{I}^\Phi \models \varphi \frac{t_1 \ldots t_n}{x_1 \ldots x_n}.$$

Proof. (a) By induction on terms. The assertion holds for $t = x$ by 1.6 and for $t = c$ by 1.5. If $t = ft_1 \ldots t_n$ then

$$\begin{aligned} \mathfrak{I}^\Phi(ft_1 \ldots t_n) &= f^{\mathfrak{T}^\Phi}\left(\mathfrak{I}^\Phi(t_1), \ldots, \mathfrak{I}^\Phi(t_n)\right) \\ &= f^{\mathfrak{T}^\Phi}(\overline{t_1}, \ldots, \overline{t_n}) \quad \text{(by induction hypothesis)} \\ &= \overline{ft_1 \ldots t_n} \quad \text{(by 1.4).} \end{aligned}$$

(b) $\mathfrak{I}^\Phi \models t_1 \equiv t_2$ iff $\mathfrak{I}^\Phi(t_1) = \mathfrak{I}^\Phi(t_2)$
iff $\overline{t_1} = \overline{t_2}$ (by (a))
iff $t_1 \sim t_2$
iff $\Phi \vdash t_1 \equiv t_2$.

$\mathfrak{I}^\Phi \models Rt_1 \ldots t_n$ iff $R^{\mathfrak{T}^\Phi}\overline{t_1} \ldots \overline{t_n}$
iff $\Phi \vdash Rt_1 \ldots t_n$ (by 1.3).

(c) (i) $\mathfrak{I}^\Phi \models \exists x_1 \ldots \exists x_n \varphi$
iff there are $a_1, \ldots, a_n \in T^\Phi$ with $\mathfrak{I}^\Phi \frac{a_1 \ldots a_n}{x_1 \ldots x_n} \models \varphi$
iff there are $t_1, \ldots, t_n \in T^S$ with $\mathfrak{I}^\Phi \frac{\overline{t_1} \ldots \overline{t_n}}{x_1 \ldots x_n} \models \varphi$
(since $T^\Phi = \{\overline{t} \mid t \in T^S\}$)
iff there are $t_1, \ldots, t_n \in T^S$ with $\mathfrak{I}^\Phi \frac{\mathfrak{I}^\Phi(t_1) \ldots \mathfrak{I}^\Phi(t_n)}{x_1 \ldots x_n} \models \varphi$ (by (a))
iff there are $t_1, \ldots, t_n \in T^S$ with $\mathfrak{I}^\Phi \models \varphi \frac{t_1 \ldots t_n}{x_1 \ldots x_n}$
(by the Substitution Lemma).

(ii) follows easily from (i). □

By part (b) of the previous lemma, $\mathfrak{I}^\Phi$ is a model of the atomic formulas in Φ, but not in general of all formulas in Φ: If, for instance, $S = \{R\}$ and $\Phi = \{\exists x Rx\}$, then, by part (c) of the lemma, if $\mathfrak{I}^\Phi \models \Phi$, there should be a term t such that $\exists x Rx \vdash Rt$; so in our case there should be a variable y such that $\exists x Rx \vdash Ry$, and this can easily be refuted (see also Exercise 1.12). We will be able to show that $\mathfrak{I}^\Phi$ is a model of Φ only if Φ satisfies certain closure conditions, as pointed out for $\exists$ in the example just given. These conditions are made precise in the following definition.

1.8 Definition. (a) Φ is *negation complete* iff for every formula φ,

$$\Phi \vdash \varphi \quad \text{or} \quad \Phi \vdash \neg\varphi.$$

(b) Φ *contains witnesses* iff for every formula of the form $\exists x\varphi$ there exists a term t such that $\Phi \vdash (\exists x\varphi \to \varphi\frac{t}{x})$.

The following lemma shows that for a consistent set Φ which is negation complete and contains witnesses, there is a parallelism between the property of being derivable from Φ and the inductive definition of the satisfaction relation. This will allow us to show that the term interpretation $\mathfrak{I}^\Phi$ is a model of Φ.

1.9 Lemma. *Suppose that Φ is consistent and negation complete and that it contains witnesses. Then the following holds for all φ and ψ:*
(a) $\Phi \vdash \neg\varphi$ *iff not* $\Phi \vdash \varphi$.
(b) $\Phi \vdash (\varphi \vee \psi)$ *if and only if* $\Phi \vdash \varphi$ *or* $\Phi \vdash \psi$.
(c) $\Phi \vdash \exists x\varphi$ *if and only if there is a term t with* $\Phi \vdash \varphi\frac{t}{x}$.

Proof. (a) Since Φ is negation complete, we have $\Phi \vdash \varphi$ or $\Phi \vdash \neg\varphi$; and since Φ is consistent, $\Phi \vdash \neg\varphi$ iff not $\Phi \vdash \varphi$.

(b) First let $\Phi \vdash (\varphi \vee \psi)$. If not $\Phi \vdash \varphi$, then $\Phi \vdash \neg\varphi$ (since Φ is negation complete), and IV.3.4 gives $\Phi \vdash \psi$. The other direction follows immediately by the $\vee$-rules ($\vee$S) for the succedent.

(c) Let $\Phi \vdash \exists x\varphi$. Since Φ contains witnesses, there is a term t with $\Phi \vdash (\exists x\varphi \to \varphi\frac{t}{x})$; using Modus ponens, IV.3.5, we get $\Phi \vdash \varphi\frac{t}{x}$. Conversely let $\Phi \vdash \varphi\frac{t}{x}$ for a term t. Then the rule ($\exists$S) of the $\exists$-introduction in the succedent gives $\Phi \vdash \exists x\varphi$. □

1.10 Henkin's Theorem. *Let Φ be a consistent set of formulas which is negation complete and contains witnesses. Then for all φ,*

$$(*) \qquad \mathfrak{I}^\Phi \models \varphi \quad \textit{iff} \quad \Phi \vdash \varphi.$$

Proof. We show $(*)$ by induction on the number of connectives and quantifiers in φ, in other words, by induction on $\mathrm{rk}(\varphi)$ (cf. III.8.6). If $\mathrm{rk}(\varphi) = 0$, then φ is atomic, and 1.7(b) shows that $(*)$ holds. The induction step splits into three separate cases.

(1) $\varphi = \neg\psi$: $\mathfrak{I}^\Phi \models \neg\psi$
iff not $\mathfrak{I}^\Phi \models \psi$
iff not $\Phi \vdash \psi$ (by induction hypothesis)
iff $\Phi \vdash \neg\psi$ (by 1.9(a)).

(2) $\varphi = (\psi \vee \chi)$: $\mathfrak{I}^\Phi \models (\psi \vee \chi)$
iff $\mathfrak{I}^\Phi \models \psi$ or $\mathfrak{I}^\Phi \models \chi$
iff $\Phi \vdash \psi$ or $\Phi \vdash \chi$ (by induction hypothesis)
iff $\Phi \vdash (\psi \vee \chi)$ (by 1.9(b)).

(3) $\varphi = \exists x\psi$: $\mathfrak{I}^\Phi \models \exists x\psi$
iff there is a t with $\mathfrak{I}^\Phi \models \psi\frac{t}{x}$ (by 1.7(c)(i))
iff there is a t with $\Phi \vdash \psi\frac{t}{x}$ (by induction hypothesis, since $\mathrm{rk}(\psi\frac{t}{x}) = \mathrm{rk}(\psi) < \mathrm{rk}(\varphi)$; cf. III.8.7)
iff $\Phi \vdash \exists x\psi$ (by 1.9(c)). □

1.11 Corollary. *If Φ is a consistent set which is negation complete and contains witnesses, then $\mathfrak{I}^\Phi \models \Phi$ (and hence Φ is satisfiable).* □

1.12 Exercise. (a) Let $S := \{R\}$ with unary R and let $\Phi := \{\exists xRx\} \cup \{\neg Ry \mid y \text{ is a variable}\}$. Show:
(i) Φ is satisfiable and therefore consistent.
(ii) For no term $t \in T^S$, $\Phi \vdash Rt$.
(iii) If $\mathfrak{I} = (\mathfrak{A}, \beta)$ is a model of Φ, then $A \setminus \{\mathfrak{I}(t) \mid t \in T^S\}$ is nonempty.
(b) Again, let $S = \{R\}$ with unary R and let x and y be distinct variables. For $\Phi = \{Rx \vee Ry\}$ show:
(i) Not $\Phi \vdash Rx$ and not $\Phi \vdash \neg Rx$, i.e. Φ is not negation complete.
(ii) Not $\mathfrak{I}^\Phi \models \Phi$.

1.13 Exercise. Fix a symbol set S. Find $\mathfrak{I}^\Phi$ for an inconsistent set Φ. Does $\mathfrak{I}^\Phi$ depend on the inconsistent set Φ?

§2. Satisfiability of Consistent Sets of Formulas (the Countable Case)

By 1.11, every consistent set of formulas which is negation complete and contains witnesses, is satisfiable. We now prove that any consistent set of formulas is satisfiable, by showing how to extend it to a consistent set of formulas which is negation complete and contains witnesses. In this section we settle the case of symbol sets which are at most countable.

In the following let S be at most countable. First we treat the case where only finitely many variables occur free in the consistent set Φ of formulas, i.e. where $\mathrm{free}(\Phi) := \bigcup_{\varphi\in\Phi} \mathrm{free}(\varphi)$ is finite. We need two lemmas.

2.1 Lemma. *Let $\Phi \subset L^S$ be consistent and let* $\mathrm{free}(\Phi)$ *be finite. Then there is a consistent set Ψ such that $\Phi \subset \Psi \subset L^S$ and Ψ contains witnesses.*

2.2 Lemma. *Let $\Psi \subset L^S$ be consistent. Then there is a consistent, negation complete set Θ with $\Psi \subset \Theta \subset L^S$.*

2.1 and 2.2 enable us to extend a consistent set Φ of formulas with finitely many free variables in two stages to a consistent set of formulas which is negation complete and contains witnesses. First of all, we extend Φ to Ψ according to 2.1, and then Ψ to Θ according to 2.2. Θ is consistent and negation complete; it contains witnesses because Ψ does already. Hence by 1.11, Θ is satisfiable, and since $\Phi \subset \Theta$, Φ is also satisfiable. We summarize:

2.3 Corollary. *Let Φ be consistent, and let* $\mathrm{free}(\Phi)$ *be finite. Then Φ is satisfiable.* □

Proof of Lemma 2.1. By II.3.3, L^S is countable. Let $\exists x_0\varphi_0$, $\exists x_1\varphi_1$, ... be a list of all formulas in L^S which begin with an existential quantifier. Inductively we define formulas $\psi_0, \psi_1, \ldots$, which we add to Φ. For each n, ψ_n is a "witness formula" for $\exists x_n\varphi_n$.

Suppose ψ_m is already defined for $m < n$. Since $\mathrm{free}(\Phi)$ is finite, only finitely many variables occur free in $\Phi \cup \{\psi_m \mid m < n\} \cup \{\exists x_n\varphi_n\}$. Let y_n be the variable with smallest index distinct from these. We set

$$\psi_n := (\exists x_n\varphi_n \to \varphi_n\tfrac{y_n}{x_n}).$$

Now let

$$\Psi := \Phi \cup \{\psi_0, \psi_1, \ldots\}.$$

Then $\Phi \subset \Psi$ and Ψ clearly contains witnesses. It remains to be shown that Ψ is consistent. For this purpose put

$$\Phi_n := \Phi \cup \{\psi_m \mid m < n\}.$$

Then $\Phi_0 \subset \Phi_1 \subset \Phi_2 \subset \ldots$ and $\Psi = \bigcup_{n\in\mathbf{N}} \Phi_n$. By IV.7.7 (for $S = S_0 = S_1 = \ldots$) the proof will be complete if we can show that each Φ_n is consistent. We proceed by induction on n.

Since $\Phi_0 = \Phi$, Con Φ_0 holds by hypothesis. For the induction step we assume that Φ_n is consistent. Suppose, for a contradiction, that $\Phi_{n+1} = \Phi_n \cup \{\psi_n\}$ is inconsistent. Then for every φ there exists Γ over Φ_n such that $\vdash \Gamma\ \psi_n\ \varphi$, i.e.

$$\vdash \Gamma\ (\neg\exists x_n\varphi_n \vee \varphi_n\tfrac{y_n}{x_n})\ \varphi.$$

Thus, there is a derivation

$$\begin{array}{lll} & & \vdots \\ m. & \Gamma & (\neg\exists x_n\varphi_n \vee \varphi_n\frac{y_n}{x_n}) \quad \varphi\ . \end{array}$$

If φ is a sentence, we can extend this derivation as follows:

(m+1).	Γ	$\neg\exists x_n\varphi_n$	$\neg\exists x_n\varphi_n$	(Assm)
(m+2).	Γ	$\neg\exists x_n\varphi_n$	$\left(\neg\exists x_n\varphi_n \vee \varphi_n\frac{y_n}{x_n}\right)$	($\vee$S) applied to (m+1).
(m+3).	Γ	$\neg\exists x_n\varphi_n$	φ	(Ch) applied to (m+2). and m. (with (Ant))
		$\vdots$		
(l).	Γ	$\varphi_n\frac{y_n}{x_n}$	φ	(analogously)
(l+1).	Γ	$\exists x_n\varphi_n$	φ	($\exists$A) applied to (l). (y_n does not occur free in Γ $\exists x_n\varphi_n$ φ)
(l+2).	Γ		φ	(PC) applied to (l+1). and (m+3).

For $\varphi = \exists v_0\, v_0 \equiv v_0$ and $\varphi = \neg\exists v_0\, v_0 \equiv v_0$, this gives $\Phi_n \vdash \exists v_0\, v_0 \equiv v_0$ and $\Phi_n \vdash \neg\exists v_0\, v_0 \equiv v_0$, respectively. Hence Inc Φ_n which contradicts the induction hypothesis. □

Proof of Lemma 2.2. Suppose Ψ is consistent and let $\varphi_0, \varphi_1, \varphi_2, \ldots$ be an enumeration of L^S. We define sets of formulas Θ_n inductively as follows:

$$\Theta_0 := \Psi$$

and

$$\Theta_{n+1} := \begin{cases} \Theta_n \cup \{\varphi_n\} & \text{if Con } \Theta_n \cup \{\varphi_n\}, \\ \Theta_n & \text{otherwise,} \end{cases}$$

and we set

$$\Theta := \bigcup_{n\in\mathbf{N}} \Theta_n.$$

First of all, $\Psi \subset \Theta$. Clearly all Θ_n are consistent, and hence by IV.7.7, Θ is consistent as well. Finally, Θ is negation complete. For if $\varphi \in L^S$, say $\varphi = \varphi_n$, and not $\Theta \vdash \neg\varphi$, then Con $\Theta \cup \{\varphi\}$ (by IV.7.6(b)) and therefore Con $\Theta_n \cup \{\varphi\}$. So $\Theta_{n+1} = \Theta_n \cup \{\varphi\}$, hence $\varphi \in \Theta$ and therefore $\Theta \vdash \varphi$. □

Now we drop the assumption that free(Φ) is finite.

2.4 Theorem. *If S is at most countable and $\Phi \subset L^S$ is consistent, then Φ is satisfiable.*

Proof. We reduce 2.4 to 2.3 by replacing the free variables by new constants. Let $c_0, c_1, \ldots$ be distinct constants which do not belong to S, and set

$$S' := S \cup \{c_0, c_1, \ldots\}.$$

For $\varphi \in L^S$ denote by $n(\varphi)$ the smallest n with free(φ) $\subset \{v_0, \ldots, v_{n-1}\}$. Let

$$\varphi' := \varphi\frac{c_0 \cdots c_{n(\varphi)-1}}{v_0 \cdots v_{n(\varphi)-1}} \qquad \text{and} \qquad \Phi' := \{\varphi' \mid \varphi \in \Phi\}.$$

First (by III.8.5), free(Φ') $= \emptyset$, i.e.

(1) $\qquad \Phi'$ is a set of S'-sentences.

Now it will suffice to show that

$$\text{(2)} \qquad \mathrm{Con}_{S'}\ \Phi',$$

for then we know from the special case proved in 2.3 that Φ' is satisfiable, say by some interpretation $\mathfrak{I}' = (\mathfrak{A}', \beta')$. Since Φ' is a set of sentences (cf. (1)), we can (by the Coincidence Lemma) choose β' such that $\beta'(v_n) = c_n^{\mathfrak{A}'}$, i.e. $\mathfrak{I}'(v_n) = \mathfrak{I}'(c_n)$ for all $n \in \mathbf{N}$. Then (using the Substitution Lemma) for $\varphi \in \Phi$ we have $\mathfrak{I}' \models \varphi$, since $\mathfrak{I}' \models \varphi\frac{c_0 \cdots c_{n(\varphi)-1}}{v_0 \cdots v_{n(\varphi)-1}}$. Hence $\mathfrak{I}'$ is a model of Φ, i.e. Φ is satisfiable. We prove (2) by showing that every finite subset Φ_0' of Φ' is satisfiable, and thus, by IV.7.5, consistent (with respect to S'). Let $\Phi_0' = \{\varphi_1', \ldots, \varphi_n'\}$, where $\varphi_1, \ldots, \varphi_n \in \Phi$. Since $\{\varphi_1, \ldots, \varphi_n\}$ is a subset of Φ, it is consistent (with respect to S), and since only finitely many variables occur free therein, it is satisfiable (cf. 2.3). Choose an S-interpretation $\mathfrak{I} = (\mathfrak{A}, \beta)$ such that

$$(*) \qquad \mathfrak{I} \models \{\varphi_1, \ldots, \varphi_n\}$$

and expand $\mathfrak{A}$ to an S'-structure $\mathfrak{A}'$ with $c_i^{\mathfrak{A}'} = \mathfrak{I}(v_i)$ for $i \in \mathbf{N}$. For this new S'-interpretation $\mathfrak{I}' = (\mathfrak{A}', \beta)$ the Substitution Lemma yields for $\varphi \in L^S$:

$$\mathfrak{I} \models \varphi \quad \text{iff} \quad \mathfrak{I}' \models \varphi\frac{c_0 \cdots c_{n(\varphi)-1}}{v_0 \cdots v_{n(\varphi)-1}}.$$

By $(*)$, $\mathfrak{I}'$ is a model of Φ_0'. □

The following exercise shows that the assumption "free(Φ) is finite" in 2.1 is necessary.

2.5 Exercise. Let S be arbitrary and let $\Phi = \{v_0 \equiv t \mid t \in T^S\} \cup \{\exists v_0 \exists v_1 \neg v_0 \equiv v_1\}$. Show that Con Φ holds and that there is no consistent set in L^S which includes Φ and contains witnesses.

§3. Satisfiability of Consistent Sets of Formulas (the General Case)

In this section we no longer assume that S is countable. In Section 2 the set Φ we started with was consistent and free(Φ) was finite. We extended Φ to a consistent set containing witnesses by adding a formula $(\exists x\varphi \to \varphi\frac{y}{x})$ with a "new" variable y for each formula of the form $\exists x\varphi$. If Φ is uncountable we run out of variables. We solve this problem by adding constants to the symbol set which will play the role of the variables. The claims corresponding to Lemma 2.1 and 2.2 are:

3.1 Lemma. *Assume $\Phi \subset L^S$ with $\mathrm{Con}_S\ \Phi$. Then there is an $S' \supset S$ and a set Ψ such that $\Phi \subset \Psi \subset L^{S'}$ and $\mathrm{Con}_{S'}\ \Psi$, and Ψ contains witnesses with respect to S' (i.e. for every formula of the form $\exists x\varphi \in L^{S'}$ there is a term $t \in T^{S'}$ such that $\Psi \vdash (\exists x\varphi \to \varphi\frac{t}{x})$).*

3.2 Lemma. *Assume $\Psi \subset L^S$ with $\mathrm{Con}_S\ \Psi$. Then there is a set Θ such that $\Psi \subset \Theta \subset L^S$ and Θ is consistent and negation complete with respect to S.*

As we obtained 2.3 from 2.1 and 2.2, we likewise have from 3.1 and 3.2 the following:

3.3 Corollary. *If $\Phi \subset L^S$ and $\mathrm{Con}_S\ \Phi$, then Φ is satisfiable.* □

The following argument will lead to a proof of 3.1.

Let S be an arbitrary symbol set. Associate with every $\varphi \in L^S$ a constant c_φ which is not in S. For $\varphi \neq \psi$ let $c_\varphi \neq c_\psi$. We set

$$S^* := S \cup \{c_{\exists x\varphi} \mid \exists x\varphi \in L^S\}$$

and

$$W(S) := \left\{(\exists x\varphi \to \varphi\tfrac{c_{\exists x\varphi}}{x}) \mid \exists x\varphi \in L^S\right\}.$$

3.4. *For $\Phi \subset L^S$, if $\mathrm{Con}_S\ \Phi$ then $\mathrm{Con}_{S^*}\ \Phi \cup W(S)$.*

Proof. Suppose $\mathrm{Con}_S\ \Phi$ holds. We show that every finite subset Φ_0^* of $\Phi \cup W(S)$ is consistent with respect to S^* by proving that it is satisfiable. Let

$$\Phi_0^* = \Phi_0 \cup \left\{(\exists x_1\varphi_1 \to \varphi_1\tfrac{c_1}{x_1}), \ldots, (\exists x_n\varphi_n \to \varphi_n\tfrac{c_n}{x_n})\right\},$$

where $\Phi_0 \subset \Phi$ and $\exists x_1\varphi_1, \ldots, \exists x_n\varphi_n \in L^S$. Here c_i stands for $c_{\exists x_i\varphi_i}$.

First, using 2.3, we show that Φ_0 is satisfiable; then, from a model $\mathfrak{I}$ of Φ_0 we get a model of Φ_0^* by a suitable interpretation of the constants.

For a suitable finite (hence at most countable) subset $S_0 \subset S$, we have $\Phi_0 \cup \{\exists x_1\varphi_1, \ldots, \exists x_n\varphi_n\} \subset L^{S_0}$. Furthermore, since $\mathrm{Con}_S\ \Phi$ holds, so does $\mathrm{Con}_S\ \Phi_0$, and hence of course $\mathrm{Con}_{S_0}\ \Phi_0$. Because $\mathrm{free}(\Phi_0)$ is finite, it follows from 2.3 that Φ_0 is satisfiable.

Let $\mathfrak{I} = (\mathfrak{A}, \beta)$ be an S-interpretation which satisfies Φ_0 and fix an element a in A. For $1 \leq i \leq n$ we choose $a_i \in A$ so that

$$(*) \qquad \mathfrak{I}\tfrac{a_i}{x_i} \models \varphi_i, \text{ if } \mathfrak{I} \models \exists x_i\varphi_i,$$

and $a_i = a$ otherwise. We extend $\mathfrak{A}$ to an S^*-structure $\mathfrak{A}^*$ as follows: For $1 \leq i \leq n$ let

$$c_i^{\mathfrak{A}^*} := a_i,$$

and interpret the remaining constants of the form $c_{\exists x\varphi}$ by a. Let $\mathfrak{I}^* = (\mathfrak{A}^*, \beta)$. Since no constant $c_{\exists x\varphi}$ occurs in Φ_0, it follows from $\mathfrak{I} \models \Phi_0$ that $\mathfrak{I}^* \models \Phi_0$. Furthermore,

$$\mathfrak{I}^* \models \exists x_i\varphi_i \to \varphi_i\tfrac{c_i}{x_i}$$

(and this shows that Φ_0^* is satisfiable). In fact, if $\mathfrak{I}^* \models \exists x_i \varphi_i$ then $\mathfrak{I}^* \frac{a_i}{x_i} \models \varphi_i$ by $(*)$. Since $a_i = \mathfrak{I}^*(c_i)$ it follows by the Substitution Lemma that $\mathfrak{I}^* \models \varphi_i \frac{c_i}{x_i}$. □

Proof of Lemma 3.1. Let $\Phi \subset L^S$ and suppose $\mathrm{Con}_S\ \Phi$. We define a symbol set S' and $\Psi \subset L^{S'}$ with the following properties:

(a) $S \subset S'$ and $\Phi \subset \Psi$.

(b) $\mathrm{Con}_{S'}\ \Psi$.

(c) Ψ contains witnesses.

For this purpose we define symbol sets S_n and sets Φ_n of formulas by induction on n:

$$\begin{aligned} S_0 &:= S \quad \text{and} \quad S_{n+1} := (S_n)^* \\ \Phi_0 &:= \Phi \quad \text{and} \quad \Phi_{n+1} := \Phi_n \cup W(S_n). \end{aligned}$$

(For the definitions of $(S_n)^*$ and $W(S_n)$ see the definitions before 3.4.)

From the construction it follows that

$$\begin{aligned} &S = S_0 \subset S_1 \subset S_2 \subset \ldots, \\ &\Phi_n \subset L^{S_n} \text{ for } n \in \mathbf{N}, \\ &\Phi = \Phi_0 \subset \Phi_1 \subset \Phi_2 \subset \ldots . \end{aligned}$$

We set $S' := \bigcup_{n \in \mathbf{N}} S_n$ and $\Psi := \bigcup_{n \in \mathbf{N}} \Phi_n$. Then (a) holds. Using 3.4 one can easily show $\mathrm{Con}_{S_n}\ \Phi_n$ by induction on n, and hence, by IV.7.7, that $\mathrm{Con}_{S'}\ \Psi$. Therefore, (b) also holds. Finally, Ψ contains witnesses. In fact, let $\exists x \varphi \in L^{S'}$. Then, for a suitable n, $\exists x \varphi \in L^{S_n}$. Thus for some constant $c \in S_{n+1}$, the formula $\left(\exists x \varphi \to \varphi \frac{c}{x}\right)$ is an element of $W(S_n)$ and, hence, an element of Ψ. □

Proof of Lemma 3.2. In the proof of 2.2 we made essential use of the countability of L^S. For arbitrary S we no longer have this property at our disposal. We resort to *Zorn's Lemma*, which we now state in a form suited for our purposes. The reader can find a proof of this lemma in books on set theory, e.g. in [23], [24].

Let M be a set and let $\mathfrak{U}$ be a nonempty set of subsets of M. $\mathfrak{V}$ is called a *chain* in $\mathfrak{U}$, if $\mathfrak{V} \subset \mathfrak{U}$, $\mathfrak{V} \neq \emptyset$, and if for $V_1, V_2 \in \mathfrak{V}$ we have $V_1 \subset V_2$ or $V_2 \subset V_1$. Then Zorn's Lemma says

3.5. *If for every chain $\mathfrak{V}$ in $\mathfrak{U}$ the union $\bigcup_{V \in \mathfrak{V}} V$ belongs to $\mathfrak{U}$, then there is at least one maximal element in $\mathfrak{U}$, i.e. an element U_0 for which there is no $U_1 \in \mathfrak{U}$ such that $U_0 \subsetneq U_1$.*

Now, let $\Psi \subset L^S$ and $\mathrm{Con}_S\ \Psi$. Set $M := L^S$ and

$$\mathfrak{U} := \{\Phi \mid \Psi \subset \Phi \subset L^S \text{ and } \mathrm{Con}_S\ \Phi\}.$$

Clearly, $\Psi \in \mathfrak{U}$ and so $\mathfrak{U}$ is not empty. Let $\mathfrak{V}$ be a chain in $\mathfrak{U}$. $\Theta_1 := \bigcup_{\Phi \in \mathfrak{V}} \Phi$ is an element of $\mathfrak{U}$, since $\Psi \subset \Theta_1 \subset L^S$ and $\mathrm{Con}_S\ \Theta_1$. The consistency of

Θ_1 can be proved as follows: If Θ_0 is a finite subset of Θ_1, say $\Theta_0 = \{\varphi_1, \ldots, \varphi_n\}$, then there are $\Phi_1, \ldots, \Phi_n \in \mathfrak{V}$ with $\varphi_i \in \Phi_i$ for $1 \leq i \leq n$. Since $\mathfrak{V}$ is a chain, we can number the Φ_i such that $\Phi_1 \subset \Phi_2 \subset \ldots \subset \Phi_n$. Thus $\Theta_0 \subset \Phi_n$, and by $\mathrm{Con}_S\ \Phi_n$ we have $\mathrm{Con}_S\ \Theta_0$.

Now we can apply Zorn's Lemma (3.5) to $\mathfrak{U}$, thereby obtaining a maximal element Θ in $\mathfrak{U}$. From the definition of $\mathfrak{U}$ we know that $\Psi \subset \Theta \subset L^S$ and $\mathrm{Con}_S\ \Theta$. On the other hand Θ is also negation complete. For if $\varphi \in L^S$, then by IV.7.6(c), $\mathrm{Con}_S\ \Theta \cup \{\varphi\}$ or $\mathrm{Con}_S\ \Theta \cup \{\neg\varphi\}$; by maximality of Θ we have $\Theta = \Theta \cup \{\varphi\}$ or $\Theta = \Theta \cup \{\neg\varphi\}$. Therefore $\Theta \vdash \varphi$ or $\Theta \vdash \neg\varphi$. □

§4. The Completeness Theorem

As already mentioned in the introduction of this chapter, we can obtain the completeness of the sequent calculus from 2.4 (for at most countable S) and from 3.3 (for arbitrary S):

4.1 Completeness Theorem. *For $\Phi \subset L^S$ and $\varphi \in L^S$:*

$$\textit{If } \Phi \models \varphi \textit{ then } \Phi \vdash_S \varphi. \qquad \square$$

From 4.1, together with the Theorem on Correctness IV.6.2, we have:

$$\text{For } \Phi \subset L^S \text{ and } \varphi \in L^S, \quad \Phi \models \varphi \quad \text{iff} \quad \Phi \vdash_S \varphi,$$

and from 3.3 and IV.7.5 we obtain:

$$\text{For } \Phi \subset L^S, \quad \mathrm{Sat}\ \Phi \quad \text{iff} \quad \mathrm{Con}_S\ \Phi.$$

In III.4 we saw that the concepts of consequence and satisfiability are actually independent of the particular choice of S. It follows from the results above that the concepts of derivability and consistency are also independent of S (cf. the footnote on page 73). Thus we can simply write "$\vdash$" and "Con", omitting the subscript.

4.2 Theorem on the Adequacy of the Sequent Calculus.
(a) $\Phi \models \varphi$ *iff* $\Phi \vdash \varphi$.
(b) $\mathrm{Sat}\ \Phi$ *iff* $\mathrm{Con}\ \Phi$. □

Historical Note. The Completeness Theorem is due to Gödel. The program of setting up a calculus of reasoning was first formulated and pursued by Leibniz, although traces of it may be found in the works of earlier philosophers (e.g. Aristotle and Llull). At the beginning of this century, Russell and Whitehead developed a calculus, and within it, gave formal proofs for a large number of mathematical theorems. In 1928, Gödel [11] proved the Completeness Theorem. The method of proof used in this section is due to Henkin [13].

VI

The Löwenheim-Skolem Theorem and the Compactness Theorem

The equivalences of $\vdash$ and $\models$ and of Con and Sat, respectively, form a bridge between syntax and semantics which allows us to transfer properties of $\vdash$ to $\models$ and of Con to Sat. When proving the independence of $\vdash$ and Con from the underlying symbol set at the end of the previous chapter, we transferred properties of semantic notions to syntactic ones. In Section 2 we make use of this connection in the other direction and get several important results for $\models$ and Sat. Together with the theorems in Section 1 they will provide us with a deeper insight into the expressive power of first-order languages.

§1. The Löwenheim-Skolem Theorem

The domain of the model $\mathfrak{I}^{\Phi}$ defined in V.1 consists of equivalence classes of terms. We use this fact to obtain the following theorem:

1.1 Löwenheim-Skolem Theorem. *Every satisfiable and at most countable set of formulas is satisfiable over a domain which is at most countable (i.e. it has a model whose domain is at most countable).*

Proof. First let Φ be an at most countable set of S-*sentences* which is satisfiable and hence consistent. Since each S-formula contains only finitely many S-symbols, there are at most countably many S-symbols in Φ. Therefore we may assume, without loss of generality, that S itself is at most countable. Since Φ is satisfiable, Φ is consistent, and the proofs in V.1 and V.2 show that there is an interpretation which satisfies Φ and whose domain A consists of classes $\overline{t}$ of terms, where t ranges over T^S. Because T^S is countable (cf. II.3.3), A is at most countable.

This argument can easily be transferred from sets of sentences to sets of formulas; for, if Φ is a set of S-formulas and

$$\Phi' := \left\{ \psi \frac{c_0 \dots c_{n-1}}{v_0 \dots v_{n-1}} \middle| n \in \mathbf{N}, \psi \in L_n^S \cap \Phi \right\},$$

where $c_0, c_1, \dots$ are new constants, then Φ and Φ' are satisfiable over the same domains (cf. the proof of V.2.4). □

$\forall x \forall y\, x \equiv y$ is a sentence which has only finite models. For a unary function symbol f, the sentence $\forall x \forall y (fx \equiv fy \to x \equiv y) \wedge \neg \forall x \exists y\, fy \equiv x$ has only infinite models, since there is no function on a finite set which is injective but not surjective.

If one re-examines the proof of the Completeness Theorem for the case of uncountable symbol sets, one obtains the following generalization of 1.1, which we formulate for readers who are familiar with the concept of cardinality:

1.2 "Downward" Löwenheim-Skolem Theorem. *Every satisfiable set of formulas $\Phi \subset L^S$ is satisfiable over a domain of cardinality not greater than the cardinality of L^S.* □

In 1.1 (and 1.2) a certain weakness of first-order languages is already apparent. In the case of the symbol set $S_{\mathrm{ar}}^{<}$, for example, there cannot exist a set Φ of sentences which characterizes the ordered field $\mathfrak{R}^< = (\mathbf{R}, +, \cdot, 0, 1, <)$ of the real numbers up to isomorphism (in the sense that exactly $\mathfrak{R}^<$ and the structures isomorphic to $\mathfrak{R}^<$ are the models of Φ). Any such set Φ of $S_{\mathrm{ar}}^{<}$-sentences would be at most countable and satisfiable (since $\mathfrak{R}^< \models \Phi$ must hold); then by 1.1 there would be an at most countable structure $\mathfrak{A}$ such that $\mathfrak{A} \models \Phi$. But this could not be isomorphic to $\mathfrak{R}^<$ since $\mathbf{R}$ is uncountable.

In analysis, $\mathfrak{R}^<$ is characterized up to isomorphism, say, by the axioms for ordered fields and the so-called completeness axiom ("Every nonempty set which is bounded above has a supremum"). Since the former can be formulated as $S_{\mathrm{ar}}^{<}$-formulas, we see that the completeness axiom cannot be phrased in terms of $S_{\mathrm{ar}}^{<}$-formulas.

1.3 Exercise. Show that every at most countable set of formulas which is satisfiable over an infinite domain is satisfiable over a countable domain.

§2. The Compactness Theorem

From the definition of $\vdash$ and Con we obtained directly (cf. IV.6.1 and IV.7.4):

(a) $\Phi \vdash \varphi$ iff there is a finite $\Phi_0 \subset \Phi$ such that $\Phi_0 \vdash \varphi$.

(b) Con Φ iff for all finite $\Phi_0 \subset \Phi$, Con Φ_0.

Using the Adequacy Theorem V.4.2 we rephrase these results for the corresponding semantic concepts:

2.1 Compactness Theorem. (a) (*for the consequence relation*)

$\Phi \models \varphi$ *iff there is a finite* $\Phi_0 \subset \Phi$ *such that* $\Phi_0 \models \varphi$.

(b) (*for satisfiability*)

Sat Φ *iff for all finite* $\Phi_0 \subset \Phi$, Sat Φ_0.

The Compactness Theorem is so called because, in a suitable topological reformulation, it says that a certain topology is compact (cf. Exercise 2.5).

The Löwenheim-Skolem Theorem and the Compactness Theorem play a dominant role in the semantics of first-order languages and in applying them to mathematical structures. In Chapter XIII we shall show that, in a certain way, they even characterize the first-order languages.

We now use the Compactness Theorem to obtain variants of the Löwenheim-Skolem Theorem.

2.2 Theorem. *Let* Φ *be a set of formulas which is satisfiable over arbitrarily large finite domains* (*i.e. for every* $n \in \mathbf{N}$ *there is an interpretation satisfying* Φ *over a finite domain which contains at least* n *elements*). *Then* Φ *is also satisfiable over an infinite domain.*

Proof. Let

$$\Psi := \Phi \cup \{\varphi_{\geq n} \mid 2 \leq n\}$$

($\varphi_{\geq n}$ was introduced in III.6.3). Every interpretation which satisfies Ψ is a model of Φ and has an infinite domain. Therefore we need only prove that Ψ is satisfiable. By the Compactness Theorem it is sufficient to show that every finite subset Ψ_0 of Ψ is satisfiable. For each such Ψ_0 there is an $n_0 \in \mathbf{N}$ such that

$$(*) \qquad \Psi_0 \subset \Phi \cup \{\varphi_{\geq n} \mid 2 \leq n \leq n_0\}.$$

According to the hypothesis of the theorem there is an interpretation $\mathfrak{I}$ which satisfies Φ and whose domain contains at least n_0 elements. By $(*)$, $\mathfrak{I}$ is also a model of Ψ_0. □

2.3 "Upward" Löwenheim-Skolem Theorem. *Let* Φ *be a set of formulas which is satisfiable over an infinite domain. Then for every set* A *there is a model of* Φ *which contains at least as many elements as* A. (We say that M has at least as many elements as A if there exists an injective map of A into M.)

Proof. Let $\Phi \subset L^S$. For each $a \in A$ let c_a be a new constant (i.e. $c_a \notin S$) such that $c_a \neq c_b$ for distinct $a, b \in A$. First, we show that the set

$$\Psi := \Phi \cup \{\neg c_a \equiv c_b \mid a, b \in A,\ a \neq b\}$$

of $S \cup \{c_a \mid a \in A\}$-formulas is satisfiable.

Because of the Compactness Theorem we can restrict ourselves to showing, for every finite n-tuple of distinct elements $a_1, \ldots, a_n \in A$, that

$$(+) \qquad \Phi \cup \{\neg c_{a_i} \equiv c_{a_j} \mid 1 \leq i, j \leq n,\ i \neq j\}$$

is satisfiable (cf. the argument in the previous proof). By hypothesis, there is an S-interpretation $\mathfrak{I} = (\mathfrak{B}, \beta)$ which satisfies Φ and whose domain B is infinite. Therefore there are n distinct elements $b_1, \ldots, b_n \in B$ in B. We let $c_{a_i}^{\mathfrak{B}} := b_i$ for $1 \leq i \leq n$. Then the interpretation $\left(\left(\mathfrak{B}, c_{a_1}^{\mathfrak{B}}, \ldots, c_{a_n}^{\mathfrak{B}}\right), \beta\right)$ satisfies the set $(+)$. Since every finite subset of Ψ is satisfiable, we can find an interpretation $\mathfrak{I}'$ which satisfies Ψ and hence also Φ. Let D be the domain of $\mathfrak{I}'$. For $a, b \in A$ with $a \neq b$ we have $\mathfrak{I}' \models \neg c_a \equiv c_b$. Hence $\mathfrak{I}'(c_a)$ and $\mathfrak{I}'(c_b)$ are distinct elements of D. Therefore the map $\pi\colon A \to D$, where $\pi(a) = \mathfrak{I}'(c_a)$, is injective. Thus D has at least as many elements as A. □

For example, let $\Phi = \Phi_{\mathrm{gr}}$ be the set of group axioms. Since there are infinite groups, 2.3 proves the existence of arbitrarily large groups. Similarly, one can show that there are arbitrarily large orderings and arbitrarily large fields. For each of those theories this fact can easily be shown using algebraic methods specific to the theory. However, first-order logic provides us with the framework and with methods to state and prove such results in a *general* form. Investigations of this kind on (classes of) algebraic structures belong to the field of *model theory*. For further reading we refer to [6], [19].

The idea of the previous proof is used in the proof of the following theorem, which we state here for readers familiar with the notion of cardinal number.

2.4 Theorem of Löwenheim, Skolem and Tarski. *Let Φ be a set of formulas which is satisfiable over an infinite domain and let κ be an infinite cardinal greater than or equal to the cardinality of Φ. Then Φ has a model of cardinality κ.*

Proof. Let Φ and κ be given as in the statement of the theorem. Let A be a set of cardinality κ. We may assume that $\Phi \subset L^S$ for a symbol set S of cardinality $\leq \kappa$. Then the symbol set $S \cup \{c_a \mid a \in A\}$ given in the proof of 2.3 has cardinality κ, as does the set of $S \cup \{c_a \mid a \in A\}$-formulas. Again, let $\Psi := \Phi \cup \{\neg c_a \equiv c_b \mid a, b \in A,\ a \neq b\}$. By 1.2 there is a model $\mathfrak{I}'$ of Ψ (and hence of Φ) whose domain D has cardinality $\leq \kappa$. On the other hand, since $\neg c_a \equiv c_b \in \Psi$ for distinct $a, b \in A$, D has cardinality $\geq \kappa$; hence its cardinality is exactly κ. □

2.5 Exercise. Let S be a symbol set. For every satisfiable set Φ of S-sentences let $\mathfrak{A}_\Phi$ be an S-structure such that $\mathfrak{A}_\Phi \models \Phi$. Furthermore, write $\Sigma := \{\mathfrak{A}_\Phi \mid \Phi \subset L_0^S,\ \mathrm{Sat}\ \Phi\}$, and for every S-sentence φ define $X_\varphi := \{\mathfrak{A} \in \Sigma \mid \mathfrak{A} \models \varphi\}$.
(a) Show that the system $\{X_\varphi \mid \varphi \in L_0^S\}$ is a basis for a topology on Σ.
(b) Show that every set X_φ is closed.
(c) Use the Compactness Theorem to show that every open covering of Σ has a finite subcovering, so that Σ is (quasi-)compact.

§3. Elementary Classes

For a set Φ of S-sentences we call

$$\mathrm{Mod}^S\Phi := \{\mathfrak{A} \mid \mathfrak{A} \text{ is an } S\text{-structure and } \mathfrak{A} \models \Phi\}$$

the *class of models* of Φ. Instead of $\mathrm{Mod}^S\{\varphi\}$ we sometimes write $\mathrm{Mod}^S\varphi$.

3.1 Definition. Let $\mathfrak{K}$ be a class of S-structures.

(a) $\mathfrak{K}$ is called *elementary* if there is an S-sentence φ such that $\mathfrak{K} = \mathrm{Mod}^S\varphi$.

(b) $\mathfrak{K}$ is called Δ-*elementary* if there is a set Φ of S-sentences such that $\mathfrak{K} = \mathrm{Mod}^S\ \Phi$.

Every elementary class is Δ-elementary. Conversely, because

$$\mathrm{Mod}^S\Phi = \bigcap_{\varphi\in\Phi} \mathrm{Mod}^S\varphi,$$

every Δ-elementary class is the intersection of elementary classes.

From an algebraic point of view we can formulate the question of the expressive power of first-order languages as follows: Which classes of structures are elementary or Δ-elementary, i.e., which classes can be axiomatized by a first-order sentence φ or by a set Φ of first-order sentences?

Let us give some examples.

3.2. *The class of fields* (as S_{ar}-structures) *and the class of ordered fields* (as $S_{\mathrm{ar}}^{<}$-structures) *are elementary.* For example, the first class can be represented in the form $\mathrm{Mod}^{S_{\mathrm{ar}}}\varphi_F$, where φ_F is the conjunction of the field axioms in III.6.5. Similarly, the *class of groups*, the *class of equivalence structures* , the *class of partially defined orderings* (cf. III.6.4), and *the class of (directed) graphs* are elementary.

Let p be a prime. A field $\mathfrak{F}$ has characteristic p if $\underbrace{1^{\mathfrak{F}} + \ldots + 1^{\mathfrak{F}}}_{p-\text{times}} = 0^{\mathfrak{F}}$, that is, if $\mathfrak{F}$ satisfies the sentence $\chi_p := \underbrace{1 + \ldots + 1}_{p-\text{times}} \equiv 0$. If there is no prime p for which $\mathfrak{F}$ has characteristic p, $\mathfrak{F}$ is said to have characteristic 0. For every prime p the field $\mathbf{Z}/(p)$ of the integers modulo p has characteristic p. The field $\mathfrak{R}$ of real numbers has characteristic 0. $\mathrm{Mod}^{S_{\mathrm{ar}}}(\varphi_F \wedge \chi_p)$ is the *class of fields of characteristic p*. Hence, this class is *elementary. The class of fields of characteristic 0 is* Δ-*elementary*; it can be represented as $\mathrm{Mod}^{S_{\mathrm{ar}}}(\{\varphi_F\} \cup \{\neg\chi_p \mid p \text{ is prime}\})$. The following considerations will show that it is not elementary.

Let φ be an S_{ar}-sentence which is valid in all fields of characteristic 0, that is

$$\{\varphi_F\} \cup \{\neg\chi_p \mid p \text{ is prime}\} \models \varphi.$$

By the Compactness Theorem there is an n_0 (depending on φ) such that

$$\{\varphi_F\} \cup \{\neg\chi_p \mid p \text{ is prime}, p < n_0\} \models \varphi.$$

Hence, φ is valid in all fields of characteristic $\geq n_0$. Thus we have proved:

3.3 Theorem. *An S_{ar}-sentence which is valid in all fields of characteristic 0 is valid in all fields whose characteristic is sufficiently large.* □

We conclude from this that the *class of fields of characteristic 0 is not elementary*, for otherwise, there would have to be an S_{ar}-sentence φ which is valid precisely in the fields of characteristic 0.

As an instance of 3.3 one obtains the well-known algebraic result that two polynomials $\rho(x)$ and $\sigma(x)$, whose coefficients are integral multiples of the unit element, and which are relatively prime over all fields of characteristic 0, are also relatively prime over all fields of sufficiently large characteristic. In order to verify this, one rewrites the statement that $\rho(x)$ and $\sigma(x)$ are relatively prime as an S_{ar}-sentence φ. In the case $\rho(x) := 3x^2 + 1$ and $\sigma(x) := x^3 - 1$ one can take for φ the sentence

$$\begin{aligned}
&\neg\exists u_0\exists u_1\exists w_0\exists w_1\exists z_0\exists z_1\exists z_2\forall x\\
&\quad((u_0 + u_1 \cdot x) \cdot (w_0 + w_1 \cdot x) \equiv (1 + 1 + 1) \cdot x \cdot x + 1\\
&\quad\wedge (u_0 + u_1 \cdot x) \cdot (z_0 + z_1 \cdot x + z_2 \cdot x \cdot x) \equiv x \cdot x \cdot x - 1)\\
&\quad\wedge \neg\exists u_0\exists u_1\forall x(u_0 + u_1 \cdot x) \cdot ((1 + 1 + 1) \cdot x \cdot x + 1) \equiv x \cdot x \cdot x - 1.^1
\end{aligned}$$

Here "$\ldots \equiv x \cdot x \cdot x - 1$" stands for "$\ldots + 1 \equiv x \cdot x \cdot x$". (The symbol "$-$" does not belong to S_{ar}!)

3.4. *The class of finite S-structures (for a fixed S), the class of finite groups, and the class of finite fields are not Δ-elementary.* The proof is simple: If, for example, the class of finite fields were of the form $\mathrm{Mod}^{S_{\mathrm{ar}}}\Phi$, then Φ would be a set of sentences having arbitrarily large finite models (e.g. the fields of the form $\mathbf{Z}/(p)$) but no infinite model. That would contradict 2.2. □

On the other hand, Exercise 3.7 below shows that the corresponding classes of *infinite* S-structures (groups, fields) are Δ-elementary.

3.5. *The class of torsion groups is not Δ-elementary.* We give an indirect proof, assuming for a suitable set Φ of S_{gr}-sentences $\mathrm{Mod}^{S_{\mathrm{gr}}}\Phi$ to be the class of torsion groups. Let

$$\Psi := \Phi \cup \{\neg\underbrace{x \circ \ldots \circ x}_{n-\text{times}} \equiv e \mid n \geq 1\}.$$

Every finite subset Ψ_0 of Ψ has a model: Choose an n_0 such that $\Psi_0 \subset \Phi \cup \{\neg\underbrace{x \circ \ldots \circ x}_{n-\text{times}} \equiv e \mid 1 \leq n < n_0\}$. Then every cyclic group of order n_0 is

[1]Note that a polynomial of the kind in question is uniquely determined by its values as a function if the underlying field is large enough.

a model of Ψ_0 if x is interpreted by a generating element. Now let $(\mathfrak{G}, \beta)$ be a model of Ψ. Then $\beta(x)$ does not have finite order, showing that $\mathfrak{G}$ is a model of Φ but not a torsion group, a contradiction. □

3.6. *The class of connected graphs is not Δ-elementary.* A graph (G, R^G) is said to be *connected* if, for arbitrary $a, b \in G$ with $a \neq b$, there are $n \geq 2$ and $a_1, \ldots, a_n \in G$ with

$$a_1 = a,\ a_n = b \text{ and } R^G a_i a_{i+1} \text{ for } i = 1, \ldots, n-1$$

(i.e., if for any two distinct elements in G there is a path connecting them). For $n \in \mathbf{N}$, the regular $(n+1)$-gon $\mathfrak{G}_n$ with the vertices $0, \ldots, n$ is a connected graph. More precisely, $\mathfrak{G}_n$ is the structure (G_n, R^{G_n}) with $G_n := \{0, \ldots, n\}$ and

$$R^{G_n} := \{(i, i+1) \mid i < n\} \cup \{(i, i-1) \mid 1 \leq i \leq n\} \cup \{(0, n), (n, 0)\}.$$

To give an indirect proof of 3.6, we assume that, for a suitable set Φ of $\{R\}$-sentences, $\mathrm{Mod}^{\{R\}}\Phi$ is the class of connected graphs. For $n \geq 2$ we set

$$\psi_n := \neg x \equiv y \wedge \neg \exists x_1 \ldots \exists x_n (x_1 \equiv x \wedge x_n \equiv y \wedge Rx_1x_2 \wedge \ldots \wedge Rx_{n-1}x_n)$$

and

$$\Psi := \Phi \cup \{\psi_n \mid n \geq 2\}.$$

Then every finite subset Ψ_0 of Ψ has a model: For Ψ_0 choose an n_0 such that $\Psi_0 \subset \Phi \cup \{\psi_n \mid 2 \leq n \leq n_0\}$; then $\mathfrak{G}_{2n_0}$ is a model of Ψ_0, if x is interpreted by 0 and y by n_0. If $(\mathfrak{A}, \beta)$ is a model of Ψ, there is no path connecting $\beta(x)$ and $\beta(y)$. Therefore $\mathfrak{A}$ is a model of Φ, but not a connected graph. This contradicts the assumption on Φ. □

3.7 Exercise. Let $\mathfrak{K}$ be a Δ-elementary class of structures. Show that the class $\mathfrak{K}^\infty$ of structures in $\mathfrak{K}$ with infinite domain is also Δ-elementary.

3.8 Exercise. If $\mathfrak{K}$ is a class of S-structures, $\Phi \subset L_0^S$ and $\mathfrak{K} = \mathrm{Mod}^S\ \Phi$, then Φ is said to be a *system of axioms* for $\mathfrak{K}$. Show:
(a) $\mathfrak{K}$ is elementary if and only if there is a finite system of axioms for $\mathfrak{K}$.
(b) If $\mathfrak{K}$ is elementary and $\mathfrak{K} = \mathrm{Mod}^S\Phi$ then there is a finite subset Φ_0 of Φ such that $\mathfrak{K} = \mathrm{Mod}^S\Phi_0$.

3.9 Exercise. Let $\mathfrak{K}$ and $\mathfrak{K}_1$ be classes of S-structures such that $\mathfrak{K}_1 \subset \mathfrak{K}$. Let $\mathfrak{K}_2$ be the class of S-structures which are in $\mathfrak{K}$ but not in $\mathfrak{K}_1$, that is $\mathfrak{K}_2 = \mathfrak{K} \setminus \mathfrak{K}_1$. Furthermore, let $\mathfrak{K}$ be elementary and $\mathfrak{K}_1$ be Δ-elementary.

Show:
(a) $\mathfrak{K}_1$ is elementary iff $\mathfrak{K}_2$ is Δ-elementary
iff $\mathfrak{K}_2$ is elementary.
Conclude:
(b) The class of fields whose characteristic is a prime is not Δ-elementary.

3.10 Exercise. A set Φ of S-sentences is called *independent* if no $\varphi \in \Phi$ is a consequence of $\Phi \setminus \{\varphi\}$. Show:
(a) Every finite set Φ of S-sentences has an independent subset Φ_0 such that $\mathrm{Mod}^S\Phi = \mathrm{Mod}^S\Phi_0$.
(b) If S is at most countable then every Δ-elementary class of S-structures has an independent system of axioms. (Hint: Start by defining a system of axioms $\varphi_0, \varphi_1, \ldots$ such that $\models \varphi_{i+1} \to \varphi_i$ for $i \in \mathbf{N}$.)

3.11 Exercise. Let Φ be the finite system of axioms for vector spaces expressed in terms of the symbol set $S = \{\underline{F}, \underline{V}, +, \cdot, 0, 1, \circ, e, *\}$ (cf. III.7.2). Show:
(a) For every n the class of n-dimensional vector spaces is elementary.
(b) The class of infinite-dimensional vector spaces is Δ-elementary.
(c) The class of finite-dimensional vector spaces is not Δ-elementary.

§4. Elementarily Equivalent Structures

We begin by introducing two new concepts.

4.1 Definition. (a) Two S-structures $\mathfrak{A}$ and $\mathfrak{B}$ are called *elementarily equivalent* (written: $\mathfrak{A} \equiv \mathfrak{B}$) if for every S-sentence φ we have $\mathfrak{A} \models \varphi$ iff $\mathfrak{B} \models \varphi$.
(b) For an S-structure $\mathfrak{A}$ let $\mathrm{Th}(\mathfrak{A}) := \{\varphi \in L_0^S \mid \mathfrak{A} \models \varphi\}$. $\mathrm{Th}(\mathfrak{A})$ is called the (first-order) *theory* of $\mathfrak{A}$.

4.2 Lemma. *For two S-structures $\mathfrak{A}$ and $\mathfrak{B}$,*

$$\mathfrak{B} \equiv \mathfrak{A} \quad \textit{iff} \quad \mathfrak{B} \models \mathrm{Th}(\mathfrak{A}).$$

Proof. If $\mathfrak{B} \equiv \mathfrak{A}$ then, since $\mathfrak{A} \models \mathrm{Th}(\mathfrak{A})$, also $\mathfrak{B} \models \mathrm{Th}(\mathfrak{A})$. Conversely, if $\mathfrak{B} \models \mathrm{Th}(\mathfrak{A})$ then, given an S-sentence φ, we examine the two possibilities: (i) If $\mathfrak{A} \models \varphi$ then $\varphi \in \mathrm{Th}(\mathfrak{A})$ and hence $\mathfrak{B} \models \varphi$. (ii) If not $\mathfrak{A} \models \varphi$ then $\neg\varphi \in \mathrm{Th}(\mathfrak{A})$; thus $\mathfrak{B} \models \neg\varphi$ and therefore not $\mathfrak{B} \models \varphi$. □

In the following, let $\mathfrak{A}$ be a fixed S-structure. We consider

(1) the class $\{\mathfrak{B} \mid \mathfrak{B} \cong \mathfrak{A}\}$ of structures isomorphic to $\mathfrak{A}$,

(2) the class of structures which satisfy the same sentences as $\mathfrak{A}$, i.e. the class $\{\mathfrak{B} \mid \mathfrak{B} \equiv \mathfrak{A}\}$ of structures elementarily equivalent to $\mathfrak{A}$.

From the Isomorphism Lemma III.5.2 it follows directly that isomorphic structures are elementarily equivalent, that is

$$(+) \qquad \{\mathfrak{B} \mid \mathfrak{B} \cong \mathfrak{A}\} \subset \{\mathfrak{B} \mid \mathfrak{B} \equiv \mathfrak{A}\}.$$

4.3 Theorem. (a) *If $\mathfrak{A}$ is infinite then the class $\{\mathfrak{B} \mid \mathfrak{B} \cong \mathfrak{A}\}$ is not Δ-elementary; in other words, no infinite structure can be characterized up to isomorphism in a first-order language.*

(b) *For every structure $\mathfrak{A}$, the class $\{\mathfrak{B} \mid \mathfrak{B} \equiv \mathfrak{A}\}$ is Δ-elementary; in fact $\{\mathfrak{B} \mid \mathfrak{B} \equiv \mathfrak{A}\} = \mathrm{Mod}^S\ \mathrm{Th}(\mathfrak{A})$. Moreover, $\{\mathfrak{B} \mid \mathfrak{B} \equiv \mathfrak{A}\}$ is the smallest Δ-elementary class which contains $\mathfrak{A}$.*

From 4.3 together with (+) we obtain the result that for infinite $\mathfrak{A}$ the class $\{\mathfrak{B} \mid \mathfrak{B} \cong \mathfrak{A}\}$ must be a proper subclass of $\{\mathfrak{B} \mid \mathfrak{B} \equiv \mathfrak{A}\}$; in particular:

4.4 Corollary. *For each infinite structure there exists an elementarily equivalent, nonisomorphic structure.* □

Proof of 4.3. (a) We assume $\mathfrak{A}$ to be infinite and Φ to be a set of S-sentences such that

$$(*) \qquad \mathrm{Mod}^S\Phi = \{\mathfrak{B} \mid \mathfrak{B} \cong \mathfrak{A}\}.$$

Φ has an infinite model, and hence by 2.3, it has a model $\mathfrak{B}$ with at least as many elements as the power set of A. Hence $\mathfrak{B}$ is not isomorphic to $\mathfrak{A}$ (cf. II.1.5), in contradiction to $(*)$.

(b) From 4.2 it follows immediately that $\{\mathfrak{B} \mid \mathfrak{B} \equiv \mathfrak{A}\} = \mathrm{Mod}^S\mathrm{Th}(\mathfrak{A})$. Now, if $\mathrm{Mod}^S\Phi$ is another Δ-elementary class containing $\mathfrak{A}$, then $\mathfrak{A} \models \Phi$ and therefore $\mathfrak{B} \models \Phi$ for every $\mathfrak{B}$ with $\mathfrak{B} \equiv \mathfrak{A}$; hence $\{\mathfrak{B} \mid \mathfrak{B} \equiv \mathfrak{A}\} \subset \mathrm{Mod}^S\ \Phi$. □

4.3(b) shows that a Δ-elementary class contains, together with any given structure, all elementarily equivalent ones. In certain cases one can use this fact to show that a class $\mathfrak{K}$ is not Δ-elementary. To do this one simply specifies two elementarily equivalent structures, one of which belongs to $\mathfrak{K}$, and the other does not. We illustrate this method in the case of archimedean fields.

An ordered field $\mathfrak{F}$ is called *archimedean* if for every $a \in F$ there is a natural number n such that $a <^F \underbrace{1^F + \ldots + 1^F}_{n-\text{times}}$. For example, the ordered field of rational numbers and the ordered field $\mathfrak{R}^<$ of real numbers are archimedean. We show that there is an ordered field elementarily equivalent to $\mathfrak{R}^<$ which is not archimedean. This will prove:

4.5 Theorem. *The class of achimedean fields is not Δ-elementary.*

Proof. Let

$$\Psi := \mathrm{Th}(\mathfrak{R}^<) \cup \{\mathbf{0} < x,\ \mathbf{1} < x,\ \mathbf{2} < x, \ldots\},$$

where $\mathbf{0}, \mathbf{1}, \mathbf{2}, \ldots$ stand for the S_{ar}-terms $0, 1, 1{+}1, \ldots$. Every finite subset of Ψ is satisfiable, for example, by an interpretation of the form $(\mathfrak{R}^<, \beta)$, where $\beta(x)$ is a sufficiently large natural number. By the Compactness Theorem there is a model $(\mathfrak{B}, \beta)$ of Ψ. Since $\mathfrak{B} \models \mathrm{Th}(\mathfrak{R}^<)$, $\mathfrak{B}$ is an ordered field

elementarily equivalent to $\mathfrak{R}^<$, but (as shown by the element $\beta(x)$) it is not archimedean. □

The application of the Compactness Theorem in the preceding proof is typical and has already been used several times (cf. 2.2, 2.3, 3.5). In each case the problem consists in finding a structure with certain properties which can be expressed in first-order logic by means of a suitable set Ψ of formulas. To prove satisfiability of Ψ one employs the Compactness Theorem. In the preceding proof Ψ contains (in addition to $\mathrm{Th}(\mathfrak{R}^<)$) formulas which guarantee that there is an element which violates the archimedean ordering property. The Compactness Theorem says in this case that, from the existence of ordered fields with arbitrarily large "finite" elements, one can conclude the existence of an ordered field with an "infinitely large" element. We shall give some further applications of this method.

The system of axioms Π from III.7.5 characterizes the structure $\mathfrak{N}$ up to isomorphism. However, $\mathfrak{N}$ cannot be characterized up to isomorphism by means of first-order formulas (cf. 4.4). Hence the induction axiom, being the only second-order axiom of Π, cannot be formulated as a first-order formula or as a set of first-order formulas.

A structure which is elementarily equivalent, but not isomorphic to $\mathfrak{N}$ is called a *nonstandard model of arithmetic*. The proof of 2.3 shows that there exists an *uncountable* nonstandard model of arithmetic. We now prove:

4.6 Skolem's Theorem. *There is a countable nonstandard model of arithmetic.*

Proof. Let

$$\Psi := \mathrm{Th}(\mathfrak{N}) \cup \{\neg x \equiv \mathbf{0},\ \neg x \equiv \mathbf{1},\ \neg x \equiv \mathbf{2}, \ldots\}.$$

Every finite subset of Ψ has a model of the form $(\mathfrak{N}, \beta)$, where $\beta(x)$ is a sufficiently large natural number. By the Compactness Theorem there is a model $(\mathfrak{A}, \beta)$ of Ψ, which by the Löwenheim-Skolem Theorem and the countability of Ψ we may assume to be at most countable. $\mathfrak{A}$ is a structure elementarily equivalent to $\mathfrak{N}$. Since for $m \neq n$ the sentence $\neg \mathbf{m} \equiv \mathbf{n}$ belongs to $\mathrm{Th}(\mathfrak{N})$, $\mathfrak{A}$ is infinite and hence is countable. $\mathfrak{A}$ and $\mathfrak{N}$ are not isomorphic, since an isomorphism π from $\mathfrak{N}$ onto $\mathfrak{A}$ would have to map $n = \mathbf{n}^{\mathfrak{N}}$ to $\mathbf{n}^{\mathfrak{A}}$ (cf. (i) in the proof of III.5.2), and thus $\beta(x)$ would not belong to the range of π. □

Considering the set $\mathrm{Th}(\mathfrak{N}^<) \cup \{\neg x \equiv \mathbf{0},\ \neg x \equiv \mathbf{1},\ \neg x \equiv \mathbf{2}, \ldots\}$, we obtain analogously:

4.7 Theorem. *There is a countable structure elementarily equivalent to $\mathfrak{N}^<$ which is not isomorphic to $\mathfrak{N}^<$. (In other words, there is a countable* nonstandard model *of* $\mathrm{Th}(\mathfrak{N}^<)$). □

What do nonstandard models of $\mathrm{Th}(\mathfrak{N})$ or $\mathrm{Th}(\mathfrak{N}^<)$ look like? In the following we gain some insight into the order structure of a nonstandard model $\mathfrak{A}$ of $\mathrm{Th}(\mathfrak{N}^<)$ (and hence also into the structure of a nonstandard model of $\mathrm{Th}(\mathfrak{N})$; cf. Exercise 4.9).

In $\mathfrak{N}^<$ the sentences

$$\begin{aligned}&\forall x(\mathbf{0} \equiv x \vee \mathbf{0} < x),\\ &\mathbf{0} < \mathbf{1} \wedge \forall x(\mathbf{0} < x \rightarrow (\mathbf{1} \equiv x \vee \mathbf{1} < x)),\\ &\mathbf{1} < \mathbf{2} \wedge \forall x(\mathbf{1} < x \rightarrow (\mathbf{2} \equiv x \vee \mathbf{2} < x)), \ldots\end{aligned}$$

hold. They say that 0 is the smallest element, 1 the next smallest element after 0, 2 the next smallest element after 1, and so on. Since these sentences also hold in $\mathfrak{A}$, the "initial segment" of $\mathfrak{A}$ looks as follows:

In addition, A contains a further element, say a, since otherwise $\mathfrak{A}$ and $\mathfrak{N}^<$ would be isomorphic. Furthermore, $\mathfrak{N}^<$ satisfies a sentence φ which says that for every element there is an immediate successor and for every element other than 0 there is an immediate predecessor. From this it follows easily that A contains, in addition to a, infinitely many other elements which together with a are ordered by $<^A$ like the integers:

If we consider the element $a +^A a$ we are led to further elements of A:

The reader should give a proof of this and also verify that between every two copies of $(\mathbf{Z}, <)$ in $\mathfrak{A}$ there lies another copy.

The examples in this and the previous section show that there are important classes of structures which cannot be axiomatized in first-order logic. On the other hand, this weakness of expressive power also has pleasant consequences. For example, the argument establishing that the class of archimedean fields is not axiomatizable yields a proof of the existence of non-archimedean ordered fields; and the fact that the class of fields of characteristic 0 cannot be axiomatized by means of a single S_{ar}-sentence is complemented by the interesting result 3.3. Using similar methods, one can obtain structures elementarily equivalent to the ordered field $\mathfrak{R}^<$ of real numbers which contain, in addition to the real numbers, infinitely large elements and infinitely small positive elements (so-called *infinitesimals*). Such structures can be used in a development of analysis which avoids the ϵ-δ-technique (*nonstandard analysis*; cf. [14], [28]).

4.8 Exercise. Show: If an $S_{\mathrm{ar}}^{<}$-sentence φ is valid in all non-archimedean ordered fields, then φ is valid in all ordered fields.

4.9 Exercise. Let the S_{ar}-structure $\mathfrak{A}$ be a model of $\mathrm{Th}(\mathfrak{N})$. Let the binary relation $<^A$ be defined on A as follows:
For all $a, b \in A$: $a <^A b$ iff ($a \neq b$ and there is $c \in A$ such that $a +^A c = b$).
Show that $(\mathfrak{A}, <^A)$ is a model of $\mathrm{Th}(\mathfrak{N}^{<})$.

4.10 Exercise. If $\mathfrak{A}$ is a model of arithmetic (that is, $\mathfrak{A} \models \mathrm{Th}(\mathfrak{N})$) and if $a, b \in A$, then a is said to be a divisor of b (written: $a|b$) if $a \cdot^A c = b$ for a suitable $c \in A$. Let Q be a set of prime numbers. Show that there is a model $\mathfrak{A}$ of arithmetic which contains an element a whose prime divisors are just the members of Q, that is, for every prime p:

$$\underbrace{1^A + \ldots + 1^A}_{p-\text{times}} | a \quad \text{iff} \quad p \in Q.$$

Conclude that there are at least as many pairwise nonisomorphic countable models of arithmetic as there are subsets of $\mathbf{N}$.

4.11 Exercise. Let $\mathfrak{A} = (A, <^A)$ be a partially defined ordering (cf. III.6.4). We say that $<^A$ (or also $(A, <^A)$) has an *infinite descending chain*, if there are elements $a_0, a_1, a_2, \ldots$ in field $<^A$ such that

$$\ldots <^A a_2 <^A a_1 <^A a_0.$$

Show:
(a) $(\mathbf{N}, <^{\mathbf{N}})$ contains no infinite descending chain; on the other hand, if $\mathfrak{A}$ is a nonstandard model of $\mathrm{Th}(\mathfrak{N}^{<})$, then $(A, <^A)$ contains an infinite descending chain.
(b) Let $< \in S$ and $\Phi \subset L_0^S$. Assume that for every $m \in \mathbf{N}$ there is a model $\mathfrak{A}$ of Φ such that $(A, <^A)$ is a partially defined ordering and field $<^A$ contains at least m elements. Then there exists also a model $\mathfrak{B}$ of Φ such that $(B, <^B)$ is a partially defined ordering containing an infinite descending chain.

VII

The Scope of First-Order Logic

In Chapter I we realized that investigations into the logical reasoning used in mathematics require an analysis of the concepts of mathematical proposition and proof. In undertaking such an analysis, we were led to introduce the first-order languages. We also defined a notion of formal proof which corresponds to the intuitive concept of mathematical proof. The Completeness Theorem then shows that every proposition which is mathematically provable from a system of axioms (and thus follows from it) can also be obtained by means of a formal proof, provided the proposition and the system of axioms admit a first-order formulation.

In this chapter we discuss what has been achieved so far and what implications this has for the foundations of mathematics. To start our discussion let us consider the following questions:

(1) One goal of our investigations was a clarification of the notion of proof. However, we carried out mathematical proofs before the notion of proof was made precise. Are we not trapped in a vicious circle? Furthermore, even if there are no problems of this kind in our approach, how can we justify the rules of the sequent calculus $\mathfrak{S}$?

(2) We realized, particularly in Chapter VI, that the first-order languages have certain deficiencies in expressive power. Hence the question: What effect does the restriction to first-order languages have on the scope of our investigations?

We deal with the second question in Section 2. There we shall see that the first-order languages are in principle sufficient for present-day mathematics. Hence, the following discussion pertaining to the first question applies, in fact, to the whole of mathematics.

§1. The Notion of Formal Proof

In answering question (1), we want to show that no mathematical proofs are needed to introduce the notion of formal proof. In our discussion we also investigate the nature of the sequent rules and consider possible means of justifying them.

In Section 2 we shall argue that a finite set S of concretely chosen symbols suffices to represent the statements and arguments arising in mathematics. Therefore, in this discussion we can specify the symbols as concrete signs; thus terms, formulas, and sequents are concrete strings of symbols and not abstract mathematical entities such as are, for example, formulas in a language whose symbol set is $\{c_r \mid r \in \mathbf{R}\}$.

The notion of formal proof is based on the manipulation of symbol strings such as terms, formulas, and sequents. These manipulations are governed by a series of calculi, like the calculus of terms and the sequent calculus. The application of rules in these calculi consists of simple syntactic operations. We illustrate this in the case of the sequent calculus. To clarify the aspect we have in mind, let us start by a comparison with the rules of chess.

The rules of chess permit certain operations on concrete objects, the chess pieces. Applying a rule, that is, making a move, consists of proceeding from one configuration of the pieces to another. Each individual rule of chess is so simple that everyone who knows the rules – even if he is not a chess player – can carry out moves by himself, or can check moves to determine whether they were made according to the rules.

A similar situation pertains in the case of sequent rules. Clearly the rules are motivated by the intended meanings, but their application does not require any knowledge of these meanings: one merely performs concrete syntactic operations on strings of symbols. Anyone who knows the rules – even if he is not a logician or a mathematician – can apply them and check whether an application has been carried out correctly. Admittedly, when dealing with sequents, we have often relied on mathematical propositions (for example, we invoked the unique decomposition of a sequent into formulas when speaking of *the* succedent). But this can be avoided if, when applying a rule, we not only note the sequent, but also keep a record of how the symbol strings in it were obtained. We give some examples.

(a) Let Θ_1 and Θ_2 be sequents which occur in a derivation. One reads from the record accompanying the derivation that Θ_1 was obtained by forming a string from $\varphi_0, \ldots, \varphi_n$ and then Θ_2 was obtained similarly from $\psi_0, \ldots, \psi_m$. If one wants to apply the rule ($\vee$A), for example, one must first check whether $n = m \geq 1$, and whether the symbol strings φ_i and ψ_i agree for every $i \neq n-1$. If so, one can apply ($\vee$A) by forming the symbol string $\varphi_0 \cdots \varphi_{n-2}(\varphi_{n-1} \vee \psi_{n-1})\varphi_n$ from the components $\varphi_0, \ldots, \varphi_{n-2}, \varphi_{n-1}, \psi_{n-1}, \varphi_n, (, \vee,)$. Moreover, one notes in the record that this symbol string was obtained from the components $\varphi_0, \ldots, \varphi_{n-2}, (\varphi_{n-1} \vee \psi_{n-1})$, and φ_n.

(b) An application of the rule ($\equiv$) consists of writing down a sequent of the form $t \equiv t$, where the term t, for its part, has to be given by means of a derivation in the calculus of terms (cf. II.3.1).

(c) Similarly, when one uses the rule ($\exists$A) to proceed from the sequent

$\Gamma\ \varphi\frac{y}{x}\ \psi$ to the sequent $\Gamma\ \exists x\varphi\ \psi$, one must supply a derivation of $\varphi\ x\ y\ \varphi\frac{y}{x}$ in the substitution calculus (cf. III.8.11), and, for every χ in $\Gamma\ \exists x\varphi\ \psi$, one must supply a derivation of $y\ \chi$ in the calculus of nonfree occurrence for variables (cf. II.5.2) in order to show that the condition "y is not free in $\Gamma\ \exists x\varphi\ \psi$" is fulfilled. Then, starting from the sequent $\Gamma\ \varphi\frac{y}{x}\ \psi$, one needs only to write down the sequent $\Gamma\ \exists x\varphi\ \psi$.

From these examples it becomes clear that an application of the sequent rules consists of purely syntactic manipulations, which can be carried out without any reference to mathematical arguments. Since, by definition, a formal proof is just a sequence consisting of sequents, each of which is obtained by an application of a sequent rule (to preceding sequents), it is obvious from our previous remarks that no mathematical proofs are needed in order to introduce the notion of formal proof. Thus, our approach is not circular. The proofs we have given before defining the notion of formal proof, and the mathematical arguments we have used in building up the semantics, merely served the purpose of gaining insight into first-order languages and of motivating our development.

A word of warning is in order when considering this reduction of the notion of proof to a triviality by the calculus of sequents: We have seen that only patience, not mathematical talent is needed to verify a formal proof in accordance with the rules. However, it is a completely different matter to understand the idea of a proof, not to speak of developing such ideas oneself. Likewise, in chess there is also a great difference between knowing the rules and being able to checkmate a skillful opponent. Thus when determining the notion of formal proof we did not really touch upon the more creative part of mathematical activity (and this includes not only the development of proof ideas, but also the introduction of adequate concepts, setting up suitable systems of axioms, and finding new interesting conjectures). On the other hand, the formal character of the sequent rules leads to new interesting questions: It is possible to implement the syntactic manipulations on a computer and write a program which, for example, checks whether a proof is correct (in the sense of the sequent calculus), or which systematically produces all possible derivations. How far can we go using such computational methods, and what are their limitations? In Chapters X and XI we shall discuss this question in more detail.

Does our formal notion of proof provide a *justification* of common mathematical reasoning? Certainly not; for we have merely imitated methods of inference in the framework of a precisely defined language. However, we can at least claim that the sequent rules correspond to the normal usage of connectives, quantifiers, and equality in mathematics. For example, the $\vee$-rules reflect the use of the inclusive "or", according to which the disjunction of two propositions is true if and only if at least one of the propositions is true. Admittedly, such usage of "or" rests on certain assumptions; for

example, it must be meaningful to speak of the truth or falsehood of a mathematical proposition, and every such proposition must be either true or false (tertium non datur). In traditional mathematics (which in this regard is also called *classical* mathematics) these assumptions are accepted. Thus the rules of the sequent calculus are based upon the classical usage of the logical connectives.

Some mathematicians engaged in foundational questions, among them *intuitionists*, do not share the classical point of view. An intuitionist associates with the assertion of a mathematical proposition the requirement that it be proved in a "constructive" way. For instance, an existential statement must be proved by presenting an example, and a disjunction must be proved by establishing one of its members. To illustrate this, we consider the following two statements.

A: *Every even number* ≥ 4 *is the sum of two primes* (Goldbach's conjecture);

not A: *Not every even number* ≥ 4 *is the sum of two primes.*

From the classical point of view (A or not A) is true. However, an intuitionist cannot assert (A or not A) since neither the proposition A nor the proposition (not A) has hitherto been proved (even using classical methods).

This example already shows that mathematics as pursued by an intuitionist, so-called *intuitionistic mathematics* (cf. [17]), differs considerably from classical mathematics. Intuitionists investigate "mental mathematical constructions as such, without reference to questions regarding the nature of the constructed object, such as whether these objects exist independently of our knowledge of them " (cf. [17], p. 1). By contrast, some mathematicians adopt the classical point of view from the conviction that "the objects in mathematics, together with the mathematical domains, exist as such, like the platonic ideas" (cf. [29], p. 1), i.e. that propositions concerning these objects describe properties which either do or do not hold, and hence are either true or false.

We see from this discussion that the possibilities for justifying methods of mathematical reasoning (and specifically for justifying a proof calculus) depend essentially on epistemological assumptions. We shall continue to adopt the classical point of view.

The interested reader will find more information in [4].

§2. Mathematics Within the Framework of First-Order Logic

In this section we wish to discuss the second question raised at the beginning of this chapter: How serious is the restriction to first-order languages?

To treat this question we start with the example of arithmetic. In this case, the weakness of the expressive power of first-order languages manifests itself in the fact that the structure $\mathfrak{N}_\sigma = (\mathbf{N}, \sigma, 0)$ (cf. III.7.3) cannot be characterized up to isomorphism in $L^{\{\boldsymbol{\sigma},0\}}$. On the other hand, according to Dedekind's Theorem, $\mathfrak{N}_\sigma$ can be characterized in a second-order language by the Peano axioms (cf. III.7.4):

(P1) $\forall x \neg \boldsymbol{\sigma} x \equiv 0$
(P2) $\forall x \forall y (\boldsymbol{\sigma} x \equiv \boldsymbol{\sigma} y \to x \equiv y)$
(P3) $\forall X((X0 \wedge \forall x(Xx \to X\boldsymbol{\sigma} x)) \to \forall y Xy)$

Let us call a structure which satisfies (P1)–(P3) a *Peano structure*. Then we can formulate Dedekind's Theorem as follows:

2.1. *Any two Peano structures are isomorphic.*

Since Peano structures cannot be characterized in the first-order language, one might suspect that the result 2.1 cannot be formulated in the framework given by first-order logic, and in particular, that the proof of Theorem III.7.4, which involves (P1)–(P3), cannot be carried out within this framework. Nevertheless, this can be achieved, as we now show.

First, let us note that in 2.1 a statement is made about $\{\boldsymbol{\sigma}, 0\}$-structures. We want to interpret 2.1 as a statement about a domain which comprises as *elements* all Peano structures and also with any two such structures an isomorphism between them. Furthermore, this domain should contain the elements and subsets of Peano structures, since these play a role in the formulation of (P1)–(P3) and in the proof of 2.1.

To avoid drawing arbitrary boundaries and enable us to apply our discussion to other propositions besides 2.1, we shall consider as domain the totality of *all* objects which are treated in mathematics; this we shall call the (mathematical) *universe*. The universe contains not only "simple" objects, such as the natural numbers or the points of the euclidean plane, but also "more complicated" objects, such as sets, functions, structures or topological spaces. A mathematician assumes in his arguments that this universe has certain properties: for example, that for every two objects a_1 and a_2 the set $\{a_1, a_2\}$ is an object as well, likewise for any two sets M_1, M_2 the union $M_1 \cup M_2$, and for every injective function f the inverse f^{-1}. Mathematical statements can then be regarded as propositions about the universe. From this point of view, 2.1 says that for every two Peano

structures $\mathfrak{A}$ and $\mathfrak{B}$ in the universe there is another object in the universe which is an isomorphism between $\mathfrak{A}$ and $\mathfrak{B}$.

Now it is possible to present in a suitable first-order language a rather simple set of sentences expressing all the properties of the universe which mathematicians use. Proposition 2.1 can be formalized in this language. In other words, 2.1 can be formalized as a proposition about the universe in a first-order language L^S appropriate to the universe, just as the proposition "there is no largest real number" can be formalized as a proposition about the structure $(\mathbf{R}, <^{\mathbf{R}})$ in the language $L^{\{<\}}$ appropriate to $(\mathbf{R}, <^{\mathbf{R}})$.

In order to give a concrete impression, we carry out the essential steps of this idea more carefully: A preliminary analysis of the totality of mathematical objects leads us to a symbol set S which is suitable for the universe. In a second step we present parts of a system $\Phi_0 \subset L^S$ of axioms which comprise the properties of the universe used in mathematics. (A complete presentation of such a system Φ_0 follows in Section 3.) Finally, we indicate how to obtain a first-order formalization of 2.1 in L^S.

When introducing the universe, we spoke of "simple" objects (numbers, points, ...) and "complex" objects (sets, functions, ...). For the sake of simplicity we make use of the empirical fact that the whole spectrum of "complex" objects can be reduced to the concept of set. (We shall carry out this reduction for ordered pairs and functions.) We call the "simple" objects *urelements*. Thus, the universe contains only urelements and sets. The sets consist of elements which are either urelements or else sets themselves. Therefore Φ_0 essentially collects basic properties of sets and hence is called a *system of axioms for set theory*.

We use the unary relation symbols $\mathbf{U}$ ("... is an urelement") and $\mathbf{M}$ ("... is a set") to distinguish between urelements and sets, and we use the binary relation symbol $\in$ for the relation "... is an element of ...". Thus we are led to the symbol set $S := \{\mathbf{U}, \mathbf{M}, \in\}$.

Now we give four axioms from Φ_0 which formalize simple properties of the universe.

(A1) $\forall x(\mathbf{U}x \vee \mathbf{M}x)$
"Every object (of the universe) is an urelement or a set."

(A2) $\forall x \neg(\mathbf{U}x \wedge \mathbf{M}x)$
"No object is both an urelement and a set."

(A3) $\forall x \forall y((\mathbf{M}x \wedge \mathbf{M}y \wedge \forall z(z{\in}x \leftrightarrow z{\in}y)) \rightarrow x \equiv y)$
"Two sets which contain the same elements are equal."

(A4) $\forall x \forall y \exists z(\mathbf{M}z \wedge \forall u(u{\in}z \leftrightarrow (u \equiv x \vee u \equiv y)))$
"For every two objects x and y, the pair set $\{x, y\}$ exists."

The set z, whose existence is guaranteed by (A4), is uniquely determined by (A3). Repeated application of (A4) yields the existence of the set $\{\{x,x\}, \{x,y\}\}$. This set is normally written (x,y) and called the *ordered pair* of x and y. It is not difficult to show from (A1)–(A4) that

$$(x,y) = (x',y') \quad \text{iff} \quad x = x' \text{ and } y = y'.$$

Ordered triples can then be introduced by

$$(x,y,z) := ((x,y),z).$$

In order to obtain formalizations in L^S which are easier to read, we introduce a number of abbreviations.

($\subset$) $x \subset y$ for $\mathbf{M}x \wedge \mathbf{M}y \wedge \forall z(z \in x \rightarrow z \in y)$
"x is a subset of y."

(Instead of treating "$x \subset y$" as an abbreviation we could have added the binary relation symbol $\subset$ to S and expanded Φ_0 by adding the axiom

$$\forall x \forall y(x \subset y \leftrightarrow (\mathbf{M}x \wedge \mathbf{M}y \wedge \forall z(z \in x \rightarrow z \in y))).$$

Both approaches are equivalent, as we shall see in VIII.3.)

(**OP**) $\mathbf{OP}zxy$ for $\mathbf{M}z \wedge \forall u(u \in z \leftrightarrow (\mathbf{M}u \wedge (\forall v(v \in u \leftrightarrow v \equiv x) \vee \forall v(v \in u \leftrightarrow (v \equiv x \vee v \equiv y)))))$
"z is the ordered pair of x and y."

(**OT**) $\mathbf{OT}uxyz$ for $\mathbf{M}u \wedge \exists v(\mathbf{OP}uvz \wedge \mathbf{OP}vxy)$
"u is the ordered triple (x,y,z)."

(**E**) $\mathbf{E}uxy$ for $\mathbf{M}u \wedge \exists z(z \in u \wedge \mathbf{OP}zxy)$
"The ordered pair (x,y) is an element of u."

(**F**) $\mathbf{F}u$ for $\mathbf{M}u \wedge \forall z(z \in u \rightarrow \exists x \exists y \mathbf{OP}zxy) \wedge \forall x \forall y \forall y'((\mathbf{E}uxy \wedge \mathbf{E}uxy') \rightarrow y \equiv y')$
"u is a function, that is, a set of ordered pairs (x,y), where y is the value of u at x."

By means of (**F**) the concept of function is reduced in the usual manner to that of set: A function $f\colon A \rightarrow B$ is considered as the set $\{(x, f(x)) \mid x \in A\}$, which is also referred to as the *graph* of f.

(**D**) $\mathbf{D}uv$ for $\mathbf{F}u \wedge \mathbf{M}v \wedge \forall x(x \in v \leftrightarrow \exists y \mathbf{E}uxy)$
"v is the domain of the function u."

(**R**) $\mathbf{R}uv$ for $\mathbf{F}u \wedge \mathbf{M}v \wedge \forall y(y \in v \leftrightarrow \exists x \mathbf{E}uxy)$
"v is the range of the function u."

For simplicity, we regard a $\{\boldsymbol{\sigma}, 0\}$-structure (contrary to III.1.1) as an ordered triple (x,y,z) consisting of a set x, a function $y\colon x \rightarrow x$ and an element z of x. Then the following abbreviation "$\mathbf{PS}u$" expresses that u is a Peano structure, whereby parts (1), (2), and (3) are formulations of the Peano axioms (P1), (P2), and (P3), respectively.

$$
\begin{array}{llll}
(\mathbf{PS}) & \mathbf{PS}u & \text{for} & \exists x\exists y\exists z(\mathbf{OT}uxyz \wedge \mathbf{M}x \wedge z{\in}x \wedge \mathbf{F}y \wedge \mathbf{D}yx \wedge \\
 & & & \exists v(\mathbf{R}yv \wedge v{\subset}x) \wedge \\
 & (1) & & \forall w(w{\in}x \rightarrow \neg\mathbf{E}ywz) \wedge \\
 & (2) & & \forall w\forall w'\forall v((\mathbf{E}ywv \wedge \mathbf{E}yw'v) \rightarrow w \equiv w') \wedge \\
 & (3) & & \forall x'((x'{\subset}x \wedge z{\in}x' \wedge \forall w\forall v((w{\in}x' \wedge \mathbf{E}ywv) \rightarrow v{\in}x')) \rightarrow \\
 & & & \qquad\qquad x' \equiv x))
\end{array}
$$

The final abbreviation "$\mathbf{I}wuu'$" states the property that w is an isomorphism of the Peano structure u onto the Peano structure u':

$$
\begin{array}{lll}
(\mathbf{I}) & \mathbf{I}wuu' & \text{for} \quad \mathbf{PS}u \wedge \mathbf{PS}u' \wedge \mathbf{F}w \wedge \\
 & & \exists x\exists y\exists z\exists x'\exists y'\exists z'(\mathbf{OT}uxyz \wedge \mathbf{OT}u'x'y'z' \wedge \mathbf{D}wx \wedge \mathbf{R}wx' \wedge \\
 & & \forall r\forall s\forall v'((\mathbf{E}wrv' \wedge \mathbf{E}wsv') \rightarrow r \equiv s) \wedge \mathbf{E}wzz' \wedge \\
 & & \forall v\forall v'\forall r((\mathbf{E}yvr \wedge \mathbf{E}wvv') \rightarrow \exists r'(\mathbf{E}wrr' \wedge \mathbf{E}y'v'r')))
\end{array}
$$

Thus the following is a formalization of 2.1, Dedekind's Theorem:

$$(+) \qquad \forall u\forall v(\mathbf{PS}u \wedge \mathbf{PS}v \rightarrow \exists w\mathbf{I}wuv).$$

Clearly, (+) is a $\{\mathbf{U}, \mathbf{M}, \in\}$-sentence. So we have attained our goal of formulating 2.1 within a first-order language. This was possible because we did not distinguish between different types of mathematical objects, such as natural numbers and sets of natural numbers, but simply treated all objects in the universe as first-order ones (compare (P3) and (3) in (**PS**)).

We can achieve even more: Recall that the system Φ_0 (which we have given only in part) captures all properties of the universe needed for mathematical reasoning. By rewriting in L^S the proof of Dedekind's Theorem from III.7, one can obtain a derivation of the assertion (+) from axioms of Φ_0. Hence we have:

2.2. $\Phi_0 \vdash \forall u\forall v(\mathbf{PS}u \wedge \mathbf{PS}v \rightarrow \exists w\mathbf{I}wuv)$.

More generally: *Experience shows that all mathematical propositions can be formalized in L^S* (or in variants of it), *and that mathematically provable propositions have formalizations which are derivable from Φ_0. Thus it is in principle possible to imitate all mathematical reasoning in L^S using the rules of the sequent calculus. In this sense, first-order logic is sufficient for mathematics.*

At the same time this experience shows that the properties of the universe which are expressed in Φ_0 are a sufficient basis for a set-theoretic development of mathematics. Thus Φ_0 is a formalization of the set-theoretic assumptions about the universe upon which the mathematician ultimately relies. Since these set-theoretic assumptions can be viewed as the background for all mathematical considerations, we call Φ_0, in this context, a system of axioms for *background set theory*.

On the other hand, Φ_0 itself, like any other system of axioms, can also be the object of mathematical investigations. For example, one can ask

whether Φ_0 is consistent or study the models of Φ_0. In this context Φ_0 is called a system of axioms for *object set theory*.

A model of Φ_0 is of the form $\mathfrak{A} = (A, \mathbf{U}^A, \mathbf{M}^A, \in^A)$ and is, like every structure, an object of the universe, that is, an object in the sense of background set theory. The same is true of the domain A. Thus, as an object of the universe, A is distinct from the universe. Nevertheless, in a model $\mathfrak{A} = (A, \mathbf{U}^A, \mathbf{M}^A, \in^A)$ of Φ_0, all set-theoretical statements hold which are derivable from Φ_0; but note that, for example, $a \in^A b$ (for $a, b \in A$) does not mean that a is an element of b, i.e. that $a \in b$ holds.

Let us emphasize once again that Φ_0 plays two roles: It is both an object of mathematical investigations and a formalized description of basic properties of the universe. In other words, it is both a mathematical object and a framework for mathematics.

Thus we have two levels, object set theory and background set theory, which must be carefully distinguished. Many paradoxes arise from a confusion of these two levels. In Section 4 we shall discuss this in more detail. For the present we merely mention *Skolem's paradox*. It is well known that there are uncountably many sets (for example, there are uncountably many subsets of $\mathbf{N}$). This fact can be formalized by a sentence φ, which is derivable from Φ_0. By the Löwenheim-Skolem Theorem there is a countable model $\mathfrak{A}$ of Φ_0 and hence of φ. The *countable* model $\mathfrak{A}$ thus satisfies a sentence which says that there are *uncountably* many sets in $\mathfrak{A}$!

§3. The Zermelo-Fraenkel Axioms for Set Theory

We now present, in full, a system of axioms for set theory. Our exposition will be rather sketchy; for a more detailed treatment we refer the reader to [23], [24].

In Section 2 we assumed that the universe consists only of sets and urelements, and we saw by means of set-theoretic definitions for concepts such as "ordered pair" and "function" that this assumption is really no restriction. Furthermore, experience has shown that one can even replace the urelements arising in mathematics by suitable sets. Later, as an example, we shall give a set-theoretic substitute for the natural numbers.

Since we are abandoning the use of urelements, the symbols $\mathbf{U}$ and $\mathbf{M}$ become superfluous. Therefore, we formulate the axioms in $L^{\{\in\}}$, where the variables are intended to range over the sets of the universe. The resulting system of axioms, called ZFC, is originally due to Zermelo, Fraenkel and Skolem, and includes the axiom of choice.

ZFC contains the axioms EXT (*the axiom of extensionality*), PAIR (*the pair set axiom*), SUM (*the sum set axiom*), POW (*the power set axiom*), INF (*the axiom of infinity*), AC (*the axiom of choice*) and the axiom schemes SEP (*separation axioms*) and REP (*replacement axioms*):

EXT: $\forall x \forall y (\forall z (z \in x \leftrightarrow z \in y) \rightarrow x \equiv y)$
"Two sets which contain the same elements are equal."

SEP: For each $\varphi(z, x_1, \ldots, x_n)$[1] and arbitrary distinct variables x, y which are also distinct from z and the x_i, the axiom

$$\forall x_1 \ldots \forall x_n \forall x \exists y \forall z (z \in y \leftrightarrow (z \in x \wedge \varphi(z, x_1, \ldots, x_n)))$$

"Given a set x and a property P which can be formulated by an $\{\in\}$-formula φ, the set $\{z \in x \mid z \text{ has the property } P\}$ exists."

PAIR: $\forall x \forall y \exists z \forall w (w \in z \leftrightarrow (w \equiv x \vee w \equiv y))$
"Given two sets x, y, the pair set $\{x, y\}$ exists."

SUM: $\forall x \exists y \forall z (z \in y \leftrightarrow \exists w (w \in x \wedge z \in w))$
"Given a set x, the union of all sets in x exists."

POW: $\forall x \exists y \forall z (z \in y \leftrightarrow \forall w (w \in z \rightarrow w \in x))$
"Given a set x, the power set of x exists."

To formulate the remaining axioms more conveniently, we introduce more symbols and define their meaning. The considerations in VIII.3 show that formulas which contain these symbols can be regarded as abbreviations of $\{\in\}$-formulas. The symbols and their definitions are:

$\boldsymbol{\emptyset}$ (constant for the empty set):

$$\forall y (\boldsymbol{\emptyset} \equiv y \leftrightarrow \forall z \neg z \in y).$$

$\subset$ (binary relation symbol for the subset relation):

$$\forall x \forall y (x \subset y \leftrightarrow \forall z (z \in x \rightarrow z \in y)).$$

$\boldsymbol{\{,\}}$ (binary function symbol for pairing):

$$\forall x \forall y \forall z (\boldsymbol{\{}x, y\boldsymbol{\}} \equiv z \leftrightarrow \forall w (w \in z \leftrightarrow (w \equiv x \vee w \equiv y))).$$

(For the term $\boldsymbol{\{}y, y\boldsymbol{\}}$ we often write the shorter form $\boldsymbol{\{}y\boldsymbol{\}}$.)

$\cup$ (binary function symbol for the union):

$$\forall x \forall y \forall z (x \cup y \equiv z \leftrightarrow \forall w (w \in z \leftrightarrow (w \in x \vee w \in y))).$$

$\cap$ (binary function symbol for the intersection):

$$\forall x \forall y \forall z (x \cap y \equiv z \leftrightarrow \forall w (w \in z \leftrightarrow (w \in x \wedge w \in y))).$$

---

[1] Here and in the following we write $\psi(y_1, \ldots, y_n)$ to indicate that the variables occurring free in ψ are among the distinct variables $y_1, \ldots, y_n$.

P (unary function symbol for the power set operation):

$$\forall x \forall y(\mathbf{P}x \equiv y \leftrightarrow \forall z(z{\in}y \leftrightarrow \forall w(w{\in}z \rightarrow w{\in}x))).$$

The remaining axioms of ZFC are as follows:

INF: $\exists x(\emptyset{\in}x \wedge \forall y(y{\in}x \rightarrow y{\cup}\{y\}{\in}x))$
"There exists an infinite set, namely a set containing $\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \ldots$."

REP: For each $\varphi(x, y, x_1, \ldots, x_n)$ in $L^{\{\in\}}$ and all distinct variables u, v which are also distinct from x, y and the x_i, the axiom

$$\forall x_1 \ldots \forall x_n(\forall x \exists^{=1} y \varphi(x, y, x_1, \ldots, x_n) \rightarrow \\ \forall u \exists v \forall y(y{\in}v \leftrightarrow \exists x(x{\in}u \wedge \varphi(x, y, x_1, \ldots, x_n))))$$

"If for parameters $x_1, \ldots, x_n$ the formula $\varphi(x, y, x_1, \ldots, x_n)$ defines a map $x \mapsto y$, then the range of a set is again a set."

AC: $\forall x((\neg\emptyset{\in}x \wedge \forall u \forall v((u{\in}x \wedge v{\in}x \wedge \neg u \equiv v) \rightarrow u{\cap}v \equiv \emptyset)) \rightarrow \\ \exists y \forall w(w{\in}x \rightarrow \exists^{=1} z\, z{\in}w{\cap}y))$

"Given a set x of nonempty pairwise disjoint sets, there exists a set which contains exactly one element of each set in x."

Within the framework of ZFC one can now introduce the notions of ordered pair, ordered triple, function, etc. as we did in the preceding section, and, by examples, give evidence that all mathematical propositions can be formalized in $L^{\{\in\}}$, and that provable propositions correspond to sentences derivable from ZFC.

As announced earlier, we now show in the case of the natural numbers that one can replace the urelements by suitable sets: In our present framework we exhibit a Peano structure which can play the role of $\mathfrak{N}_\sigma$.

The sets $\tilde{0} := \emptyset$, $\tilde{1} := \{\emptyset\}$, $\tilde{2} := \{\emptyset, \{\emptyset\}\}, \ldots$ will play the role of the natural numbers $0, 1, 2, \ldots$. Thus $\tilde{0} = \emptyset$, $\tilde{1} = \{\tilde{0}\}$, $\tilde{2} = \{\tilde{0}, \tilde{1}\}$, and in general $\tilde{n} = \{\tilde{0}, \tilde{1}, \ldots, \widetilde{n-1}\}$. Let us call a set *inductive* if it contains $\emptyset$, and if whenever it contains x it also contains $x \cup \{x\}$; then the smallest inductive set assumes the role of **N**. It remains to show that the statement "there is a smallest inductive set" is derivable in ZFC. We give a guideline as to how to proceed. By INF there exists an inductive set, say x. Using SEP we obtain the set

$$\omega := \{z \mid z \in x \text{ and for all inductive } y,\ z \in y\},$$

which can be shown to be the smallest inductive set, i.e. ω is inductive and for every inductive y, $\omega \subset y$. The function $\nu{:}\,\omega \rightarrow \omega$ with $\nu(x) := x \cup \{x\}$ for $x \in \omega$ (i.e. the function $\nu = \{(x, x \cup \{x\}) \mid x \in \omega\}$) plays the role of the successor function. One can see that $(\omega, \nu, \tilde{0})$ is a Peano structure.

We close our presentation of ZFC with an important methodological aspect by briefly discussing the so-called *continuum hypothesis*. This hypothesis

was stated at the end of the nineteenth century by G. Cantor and has had a crucial influence on the development of set theory. We first give an intuitive formulation.

Two sets x, y are said to be *of the same cardinality* (written: $x \sim y$) if there is a bijection from x to y. A set is *finite* if and only if it is of the same cardinality as an element of ω; it is *countable* if it is of the same cardinality as ω. The set $\mathbf{R}$ of real numbers (the "continuum") is uncountable (cf. Exercise II.1.3).

Now the continuum hypothesis states: Every infinite subset of $\mathbf{R}$ is either countable or of the same cardinality as $\mathbf{R}$. Using canonically defined symbols $\mathbf{R}, \mathbf{Fin}, \ldots$, this statement can be formulated in $L^{\{\in\}}$ in the following form:

$$\forall x((x \subset \mathbf{R} \wedge \neg \mathbf{Fin}\ x) \rightarrow (\mathbf{Count}\ x \vee x \sim \mathbf{R})).$$

This formula is often denoted by "CH" (Continuum Hypothesis). The question whether the continuum hypothesis holds corresponds to the question whether CH is derivable from ZFC.

K. Gödel showed in 1938:

3.1. *If* ZFC *is consistent then not* ZFC $\vdash \neg$CH,

and P. Cohen showed in 1963:

3.2. *If* ZFC *is consistent then not* ZFC $\vdash$ CH.

Thus if we assume that ZFC is consistent (cf. Section 4), then neither CH nor $\neg$CH is derivable from it. For an exposition of these results we refer the reader to [23].

The axiom system ZFC embodies our knowledge of the intuitive concept of set which mathematicians, in fact, use. In the light of the results of Gödel and Cohen, we see that our concept is so vague that it does not definitely decide the truth or falsehood of the continuum hypothesis. One can even show (cf. X.7) that it is not possible to give "explicitly" an axiom system Ψ for set theory, which decides *every* set-theoretic statement (in the sense that for *every* $\{\in\}$-sentence ψ either $\Psi \vdash \psi$ or $\Psi \vdash \neg\psi$).

§4. Set Theory as a Basis for Mathematics

We now supplement our previous discussion by treating three aspects: In 4.1, taking ZFC as an example, we show how the question of the consistency of mathematics may be made precise by the use of suitable first-order axioms sufficient for mathematics. In 4.2 we discuss misunderstandings

which may arise from a confusion of object set theory with background set theory. Finally, in 4.3 we show how first-order logic, like every other mathematical theory, can be based on set theory.

4.1. In the preceding sections we have emphasized the experience that mathematical statements can be formalized in $L^{\{\in\}}$ and that provable statements lead to formalizations which are derivable from ZFC. Taking this for granted, suppose it were possible in mathematics to prove both a statement and its negation. Let φ be a formalization of this statement. Then both ZFC $\vdash \varphi$ and ZFC $\vdash \neg\varphi$ would hold, and thus ZFC would be inconsistent. Therefore, a proof that ZFC is consistent could be regarded as strong evidence for the consistency of mathematics. In fact, the question of the consistency of ZFC is one of the key problems of foundational investigations. In an explicit formulation it asks: Is there a derivation in the sequent calculus of a sequent of the form $\varphi_1 \ldots \varphi_n(\varphi \wedge \neg\varphi)$, where $\varphi_1, \ldots, \varphi_n$ are ZFC axioms? In this form, the problem is obviously of a purely syntactic character. Therefore, one might hope to solve it by elementary arguments concerning the manipulation of symbol strings by sequent rules. (Hilbert also demanded a proof of such an elementary nature to recognize "that the generally accepted methods of mathematics taken as a whole do not lead to a contradiction.") However, by *Gödel's Second Incompleteness Theorem*, such a consistency proof for ZFC is not possible (cf. X.7). A proof is not even possible if one admits all the auxiliary means of the background set theory described by ZFC (and if ZFC is indeed consistent). In particular, one cannot prove the existence of a model of ZFC (since Sat ZFC would imply Con ZFC). Nevertheless, the fact that ZFC has been investigated and used in mathematics for decades and no inconsistency has been discovered, attests to the consistency of ZFC.

In the following considerations we assume ZFC to be consistent.

4.2. We investigate the relationship between background set theory and object set theory by first discussing *Skolem's Paradox* (cf. Section 2). In terms of ZFC the paradox can be formulated as follows: ZFC, being a countable, consistent set of sentences, has a *countable* model $\mathfrak{A} = (A, \in^A)$ according to the Löwenheim-Skolem Theorem. On the other hand, $\mathfrak{A}$ satisfies an $\{\in\}$-sentence φ (derivable from ZFC) which says that there are *uncountably* many sets in A. If, for simplicity, we again use defined symbols, we can write

$$\varphi := \exists x \neg \exists y(\mathbf{Function}\ y \wedge \mathbf{injective}\ y \wedge \mathbf{Domain}\,(y) \equiv x \wedge \mathbf{Range}\,(y) \subset \omega).$$

φ symbolizes the property of the universe that there exists an uncountable set (and hence, also that uncountably many sets exist). Since $\mathfrak{A}$ is a model of ZFC, we have $\mathfrak{A} \models \varphi$, i.e. there is an $a \in A$ (for x) such that

$$(*) \qquad \mathfrak{A} \models \neg\exists y(\mathbf{Function}\ y \wedge \ldots \wedge \mathbf{Range}\,(y) \subset \omega)[a].$$

The set $\{b \in A \mid b \in^A a\}$ is at most countable because it is a subset of A. Therefore *in the universe* there exists an injective function whose domain is $\{b \in A \mid b \in^A a\}$ and whose range is a subset of ω. However, this does not contradict $(*)$. For $(*)$ merely says that *in* $\mathfrak{A}$ there is no injective function defined on a with values in $\boldsymbol{\omega}^A$, or more exactly, that there is no $b \in A$ such that $\mathbf{Function}^A b$, $\mathbf{Domain}^A(b) = a$, $\mathbf{injective}^A b$, and $\mathbf{Range}^A(b) \subset^A \boldsymbol{\omega}^A$: a is uncountable *in the sense of* $\mathfrak{A}$.

From this example we see that it is necessary to distinguish carefully between the set-theoretical concepts (which refer to the universe) and their meaning in a model.

Let us consider another example. The set of sentences

$$\Psi := \mathrm{ZFC} \cup \{c_r \in \boldsymbol{\omega} \mid r \in \mathbf{R}\} \cup \{\neg c_r \equiv c_s \mid r, s \in \mathbf{R}, r \neq s\}$$

is satisfiable, as one can easily show using the Compactness Theorem. Let $\mathfrak{B} = (B, \in^B)$ be a model of Ψ (more exactly, the $\{\in\}$-reduct of a model of Ψ). Then $\{b \in B \mid b \in^B \boldsymbol{\omega}^B\}$ is an uncountable set. On the other hand, $\boldsymbol{\omega}^B$ (being the set of natural numbers in $\mathfrak{B}$) is $\mathbf{Countable}^B$ (that is, we have $\mathbf{Countable}^B \boldsymbol{\omega}^B$).

As before, let $\mathfrak{A} = (A, \in^A)$ be a countable model of ZFC. Then $\{a \in A \mid a \in^A \boldsymbol{\omega}^A\}$ is countable because it is a subset of A, and we obtain:

(1) There is no bijection of $\{b \in B \mid b \in^B \boldsymbol{\omega}^B\}$ onto $\{a \in A \mid a \in^A \boldsymbol{\omega}^A\}$,

since one set is uncountable, whereas the other one is countable. At first glance (1) seems to contradict Dedekind's Theorem, according to which every two Peano structures are isomorphic. To analyze the situation, we take a formalization ψ of this theorem as an $\{\in\}$-sentence

$$\psi := \forall x \forall y((\mathbf{Peanostructure}\ x \wedge \mathbf{Peanostructure}\ y) \rightarrow x\ \mathbf{isomorphic}\ y).$$

Then we have

$$\mathrm{ZFC} \vdash \psi. \tag{2}$$

However, (1) and (2) do not contradict each other. (2) merely says that in each *individual* model $\mathfrak{C}$ of ZFC every two Peano structures are isomorphic (in the sense of $\mathfrak{C}$), whereas (1) speaks of Peano structures in *different* models of ZFC.

4.3. We provide a set-theoretic development of first-order logic, i.e. we show that its concepts can be based on the concept of set, as we have done already for functions and Peano structures. To be specific, we restrict ourselves to the symbol set $S = \{P^1, P^2, \ldots\}$ with n-ary P^n. Our first goal is to give a set-theoretic substitute for S-formulas.

As a substitute for the variables we use the elements $\tilde{0}, \tilde{1}, \ldots$ of ω. The roles of the symbols $\neg, \vee, \exists, \equiv$ are assumed by the ordered pairs $\tilde{\neg} := (\tilde{0}, \tilde{0})$,

$\tilde{\vee} := (\tilde{0}, \tilde{1})$, $\tilde{\exists} := (\tilde{0}, \tilde{2})$ and $\tilde{\equiv} := (\tilde{0}, \tilde{3})$. For the P^n (for $n \geq 1$) we take the ordered pairs $\tilde{P}^x := (\tilde{1}, x)$ where $x \in \omega \setminus \{0\}$. (Similarly, one could, for example, let ordered pairs $(\tilde{2}, x)$ with $x \in \omega$ stand for function symbols. In order to represent symbol sets with uncountably many elements, one could use an appropriate set of larger cardinality instead of ω.)

Now formulas of the form $v_n \equiv v_m$ correspond to triples $(x, \tilde{\equiv}, y)$ with $x, y \in \omega$. These triples are the elements of the set

$$At^{\equiv} := \omega \times \{\tilde{\equiv}\} \times \omega.$$

Ordered pairs of the form $(\tilde{P}^x, z)$, where $x \in \omega$ and z is a function from x into ω, play the role of formulas of the form $P^n v_{m_0} \ldots v_{m_{n-1}}$. (For instance, the formula $P^3 v_1 v_4 v_5$ corresponds to the ordered pair $(\tilde{P}^{\tilde{3}}, z)$ with $z = \{(\tilde{0}, \tilde{1}), (\tilde{1}, \tilde{4}), (\tilde{2}, \tilde{5})\}$.) Thus, we are led to the set At^R of atomic "relational" formulas

$$At^R := \{(\tilde{P}^x, z) \mid x \in \omega \text{ and } z\colon x \to \omega\}.$$

Likewise, one can define the set of all S-formulas set-theoretically to be the smallest set A which satisfies the conditions:

(1) $At^{\equiv} \cup At^R \subset A$;
(2) if $y \in A$ then $(\tilde{\neg}, y) \in A$;
(3) if $y, z \in A$ then $(y, \tilde{\vee}, z) \in A$;
(2) if $x \in \omega$ and $y \in A$ then $(\tilde{\exists}, x, y) \in A$.

We can now give a natural set-theoretic description of the notions of sequent and derivation, developing in this way the whole syntax set-theoretically. Semantic concepts such as the notions of structure or consequence can also be introduced set-theoretically. By doing so, we obtain a set-theoretic formulation of the Completeness Theorem. All considerations can be carried out in $L^{\{\in\}}$ on the basis of ZFC. In particular, the Completeness Theorem can be formalized as an $\{\in\}$-sentence and can be derived from ZFC.

What benefits do we obtain from such a set-theoretical treatment? We mention three points.

(1) The mathematical development of first-order logic (as given in the first six chapters) can be founded upon the axiomatic basis of ZFC.

(2) The set-theoretic treatment enables us to deal with uncountable symbol sets in a precise manner. Appropriate variations of this approach make it possible to define other languages, e.g. languages with infinitely long "formulas" of the form $\varphi_0 \vee \varphi_1 \vee \varphi_2 \vee \ldots$ (Chapter IX).

(3) In our discussion concerning the formal notion of proof and the scope of first-order logic, we did not appeal to the Completeness Theorem. This was done to avoid becoming trapped in a vicious circle, since the Completeness Theorem itself requires a proof. In a set-theoretical framework, one can investigate more closely the assumptions which are needed for a proof of

the Completeness Theorem. Doing this one finds that a considerably weaker axiom system than ZFC is sufficient for the proof (cf. [2]).

4.4 Exercise. A reader who has been confused by the discussion of this chapter says, "Now I'm completely mixed up. How can ZFC be used as a basis for first-order logic, while first-order logic was actually needed in order to build up ZFC?" Help such a reader out of his dilemma. (Hint: Again, be careful in distinguishing between the object and the background level.)

VIII

Syntactic Interpretations and Normal Forms

In this chapter we collect some results which show to which extent the choice of a symbol set for a mathematical theory is arbitrary. We show, for instance, that the expressive power of first-order languages for group theory does not depend on the choice of S_{grp} or S_{gr} as symbol set. The notion of syntactic interpretation will turn out to be a central concept for our considerations. In the section about normal forms we show that, for different syntactic properties, one can find for each formula a logically equivalent one which has this property, e.g., one which has syntactically an especially simple form.

We start with a preliminary investigation which will allow for some technical simplifications.

§1. Term-Reduced Formulas and Relational Symbol Sets

Terms in a formula usually contain "nested" occurrences of function symbols. For instance, the $\{f, g\}$-formula

$$\varphi := \forall x\ fgx \equiv y$$

(with unary f, g) contains the nested term fgx. But φ is logically equivalent to the formula

$$\forall x \exists u (gx \equiv u \wedge fu \equiv y),$$

which contains no more nested terms, and which, in this sense, is "term-reduced". We show this fact in general.

1.1 Definition. An S-formula is called *term-reduced* iff its atomic subformulas have the form $Rx_1 \ldots x_n$, $x \equiv y$, $fx_1 \ldots x_n \equiv x$ or $c \equiv x$.

The result just mentioned can now be formulated as follows:

1.2 Theorem. *For every S-formula ψ there is a logically equivalent, term-reduced S-formula ψ^* with* $\mathrm{free}(\psi) = \mathrm{free}(\psi^*)$.

Proof. For $\psi \in L^S$ let $x_1, x_2, x_3, \ldots$ be the enumeration of the variables not occurring in ψ in the order induced by $v_0, v_1, v_2, \ldots$. First we define ψ^* for formulas ψ of the form $t \equiv x$; this is done by induction on the term t:

$$[y \equiv x]^* := y \equiv x;$$

for $c \in S$:

$$[c \equiv x]^* := c \equiv x;$$

for n-ary $f \in S$:

$$[ft_1 \ldots t_n \equiv x]^* := \exists x_1 \ldots \exists x_n([t_1 \equiv x_1]^* \wedge \ldots \wedge [t_n \equiv x_n]^* \wedge fx_1 \ldots x_n \equiv x).$$

For the remaining atomic formulas ψ we define ψ^* as follows:

If t_2 is not a variable:

$$[t_1 \equiv t_2]^* := \exists x_1([t_2 \equiv x_1]^* \wedge [t_1 \equiv x_1]^*);$$

for n-ary $R \in S$:

$$[Rt_1 \ldots t_n]^* := \exists x_1 \ldots \exists x_n([t_1 \equiv x_1]^* \wedge \ldots \wedge [t_n \equiv x_n]^* \wedge Rx_1 \ldots x_n).$$

Finally we set

$$\begin{array}{lcl} [\neg\psi]^* & := & \neg\psi^*; \\ (\psi_1 \vee \psi_2)^* & := & (\psi_1^* \vee \psi_2^*); \\ [\exists x\psi]^* & := & \exists x\psi^*. \end{array}$$

Using this definition, it is quite easy to prove the claim. □

The following consideration gives a first example showing us how useful term-reduced formulas can be.

A symbol set is called *relational* if it contains only relation symbols. Sometimes it is convenient, as for example in Chapter XII, to be able to restrict oneself to relational symbol sets. We show how function symbols and constants can be replaced by relation symbols in order to obtain a relational symbol set. The idea is to consider the graph of a function, rather than the function itself.

Let S be an arbitrary symbol set. For every n-ary $f \in S$ let F be a new $(n+1)$-ary relation symbol, and for $c \in S$ let C be a new unary relation symbol. Let S^r consist of the relation symbols from S together with the new relation symbols. Thus S^r is relational.

We associate with every S-structure $\mathfrak{A}$ an S^r-structure $\mathfrak{A}^r$ by replacing the functions and constants by their graphs. We define:

(1) $A^r := A$;

(2) for $P \in S$:

$$P^{\mathfrak{A}^r} := P^{\mathfrak{A}};$$

(3) for n-ary $f \in S$:

$$F^{\mathfrak{A}^r} := \text{ the graph of } f^{\mathfrak{A}},$$

that is,

$$F^{\mathfrak{A}^r} a_1 \dots a_n a \quad \text{:iff} \quad f^{\mathfrak{A}}(a_1, \dots, a_n) = a;$$

(4) for $c \in S$:

$$C^{\mathfrak{A}^r} := \text{ the graph of } c^{\mathfrak{A}},$$

that is,

$$C^{\mathfrak{A}^r} a \ \text{:iff} \ c^{\mathfrak{A}} = a.$$

Then the following holds:

1.3 Theorem. (a) *For every $\psi \in L^S$ there is $\psi^r \in L^{S^r}$ such that for all S-interpretations $\mathfrak{I} = (\mathfrak{A}, \beta)$,*

$$(\mathfrak{A}, \beta) \models \psi \quad \textit{iff} \quad (\mathfrak{A}^r, \beta) \models \psi^r.$$

(b) *For every $\psi \in L^{S^r}$ there is $\psi^{-r} \in L^S$ such that for all S-interpretations $\mathfrak{I} = (\mathfrak{A}, \beta)$,*

$$(\mathfrak{A}, \beta) \models \psi^{-r} \quad \textit{iff} \quad (\mathfrak{A}^r, \beta) \models \psi.$$

Proof. (a) By Theorem 1.2 it suffices to define ψ^r for term-reduced ψ. This is done inductively:

$$\begin{array}{lcl}
[Ry_1 \dots y_n]^r & := & Ry_1 \dots y_n; \\
[x \equiv y]^r & := & x \equiv y; \\
[fy_1 \dots y_n \equiv x]^r & := & Fy_1 \dots y_n x; \\
[\neg\psi]^r & := & \neg\psi^r; \\
(\psi_1 \vee \psi_2)^r & := & (\psi_1^r \vee \psi_2^r); \\
[\exists x\psi]^r & := & \exists x\psi^r.
\end{array}$$

The proof of the equivalence is easy.
(b) We argue similarly; in particular we set

$$\begin{array}{lcl}
[Ft_1 \dots t_n t]^{-r} & := & ft_1 \dots t_n \equiv t, \\
[Ct]^{-r} & := & c \equiv t.
\end{array}$$

□

From Theorem 1.3 we obtain immediately:

1.4 Corollary. *For two S-structures $\mathfrak{A}$ and $\mathfrak{B}$,*

$$\mathfrak{A} \equiv \mathfrak{B} \quad \textit{iff} \quad \mathfrak{A}^r \equiv \mathfrak{B}^r.$$

□

§2. Syntactic Interpretations

We now aim towards the notion of syntactic interpretation. In the following paragraphs A to D we present some motivating examples.

A. Axiom Systems for Groups

We introduced two axiom systems for the class of groups: the system Φ_{gr} in $L_0^{S_{\mathrm{gr}}}$ with $S_{\mathrm{gr}} = \{\circ, e\}$ and the system Φ_{grp} in $L_0^{S_{\mathrm{grp}}}$ with $S_{\mathrm{grp}} = \{\circ, {}^{-1}, e\}$. For $S_{\mathrm{g}} := \{\circ\}$ we have the following axiom system $\Phi_{\mathrm{g}} \subset L_0^{S_{\mathrm{g}}}$:

$$\begin{aligned}\Phi_{\mathrm{g}} \quad := \quad & \{\forall x \forall y \forall z (x \circ y) \circ z \equiv x \circ (y \circ z), \\ & \exists z (\forall x\ x \circ z \equiv x \wedge \forall x \exists y\ x \circ y \equiv z)\}.\end{aligned}$$

All three axiom systems are equivalent, since the same statements are expressible in each of these languages and the same statements provable in the corresponding axiom system. For instance, the S_{grp}-sentence

$$\forall x\ x \circ x^{-1} \equiv e$$

corresponds to the S_{g}-sentence

$$\exists z (\forall x\ x \circ z \equiv x \wedge \forall x \exists y\ x \circ y \equiv z),$$

and, in this case, the first sentence is provable from Φ_{grp} and the second from Φ_{g}.

B. Axiom Systems for Orderings

Let $S := \{<\}$. In III.6.4 we introduced the axiom system Φ_{ord} for the class of orderings. Often one extends the symbol set by a symbol $\leq$ whose interpretation is given by

$$\forall x \forall y (x \leq y \leftrightarrow (x < y \vee x \equiv y)).$$

In the new symbol set $S' := \{<, \leq\}$ we have the axiom system

$$\Phi'_{\mathrm{ord}} := \Phi_{\mathrm{ord}} \cup \{\forall x \forall y (x \leq y \leftrightarrow (x < y \vee x \equiv y))\}.$$

Since $\leq$ can always be replaced by its definition, we can associate with each S'-formula φ an S-formula $\varphi^<$ such that

$$\Phi'_{\mathrm{ord}} \models \varphi \quad \text{iff} \quad \Phi_{\mathrm{ord}} \models \varphi^<.$$

(For $\varphi \in L^S$ we have $\varphi^< = \varphi$ and therefore $\Phi'_{\mathrm{ord}} \models \varphi$ iff $\Phi_{\mathrm{ord}} \models \varphi$.) It is in this sense that L^S and $L^{S'}$ have the same expressive power for the class of orderings.

C. Rings

If, in the axiom system Φ_{fd} for fields (cf. III.6.5), we leave out the axiom $\forall x(\neg x \equiv 0 \to \exists y\, x \cdot y \equiv 1)$ about the existence of the multiplicative inverse and the commutative law $\forall x \forall y\; x \cdot y \equiv y \cdot x$ for the multiplication, we obtain the axiom system Φ_{rg} for *rings*, more precisely: for rings with 1. Every field (as an S_{ar}-structure) is a ring. The set of integers, under the natural interpretation of the S_{ar}-symbols, forms a ring, the *ring of integers*. For $n \geq 1$, the $n \times n$-matrices over $\mathbf{R}$ under the usual interpretation of the symbols from S_{ar} also form a ring $\mathfrak{M}(n)$.

Let $\mathfrak{A}$ be a ring. An element $a \in A$ is a *unit* in $\mathfrak{A}$ iff there is some $b \in A$ such that $a \cdot^{\mathfrak{A}} b = b \cdot^{\mathfrak{A}} a = 1$. In the ring of integers only 1 and -1 are units, in the rings $\mathfrak{M}(n)$ the units are the invertible matrices.

We set, with x for v_0,

$$\varepsilon := \exists y(x \cdot y \equiv 1 \wedge y \cdot x \equiv 1).$$

Then

$$E(\mathfrak{A}) := \{a \in A \mid \mathfrak{A} \models \varepsilon[a]\}$$

is the set of units in $\mathfrak{A}$. It is easy to show that $1^{\mathfrak{A}} \in E(\mathfrak{A})$, that $E(\mathfrak{A})$ is closed under multiplication, and that $E(\mathfrak{A})$ with $1^{\mathfrak{A}}$ and the multiplication even forms a group (as an S_{gr}-structure), the *group* $\mathfrak{E}(\mathfrak{A})$ *of units in* $\mathfrak{A}$. It turns out that in $\mathfrak{A}$ one can talk about $\mathfrak{E}(\mathfrak{A})$ in the sense that for every $\varphi \in L_0^{S_{\text{gr}}}$ there exists a $\varphi' \in L_0^{S_{\text{ar}}}$ such that

$$(\circ) \qquad \mathfrak{E}(\mathfrak{A}) \models \varphi \quad \text{iff} \quad \mathfrak{A} \models \varphi'.$$

For example, if φ is the commutative law $\forall x \forall y\; x \circ y \equiv y \circ x$, φ' can be chosen to be the S_{ar}-sentence

$$\forall x \forall y((\varepsilon \wedge \varepsilon \tfrac{y}{x}) \to x \cdot y \equiv y \cdot x).$$

D. Relativizations

An important aspect of the translation of the commutative law into the language of rings which we just discussed, is the restriction or, as we shall say, the *relativization* of the quantifiers to the set of units. Relativizations have already come up in III.7.2: If one regards a vector space as a one-sorted structure, then the domain consists of scalars and vectors. When formulating the vector space axioms in the corresponding language, one must *relativize* the field axioms to the set of scalars and the group axioms (for the vectors) to the set of vectors. For the field axiom $\forall x(\neg x \equiv 0 \to \exists y\, x \cdot y \equiv 1)$ this can be done by using the relation symbol $\underline{F}$ for the set of scalars and reformulating the axiom as $\forall x(\underline{F}x \to (\neg x \equiv 0 \to \exists y(\underline{F}y \wedge x \cdot y \equiv 1)))$. Similarly, the formula in the field language

$$\varphi := \forall x(x \equiv 0 \vee x \equiv 1),$$

when relativized to $\underline{F}$, becomes

$$\varphi^{\underline{F}} := \forall x(\underline{F}x \to (x \equiv 0 \vee x \equiv 1)).$$

In a vector space, $\varphi^{\underline{F}}$ just says that the field of scalars satisfies φ. It turns out that, in this sense, one can transform every formula of the language $L^{S_{\mathrm{ar}}}$ into the language of vector spaces. In this section we shall discuss the connection between a formula and its relativization in more detail.

E. Syntactic Interpretations

A common feature in all previous examples is the fact that in one structure one talks about another structure: in groups as S_{g}-structures about groups as S_{gr}-structures, in orderings with underlying symbol set $\{<\}$ about orderings with underlying symbol set $\{<, \leq\}$, in rings about the group of units, and in structures (e.g., vector spaces) about substructures whose domains are given by unary relation symbols (e.g., scalar fields). The concept of syntactic interpretation comprises the aspect common to all of these examples: A syntactic interpretation of a symbol set S' in a symbol set S will allow us to talk in S-structures about induced S'-structures.

For this purpose, an S-formula $\varphi_{S'}(v_0)$ will be specified in order to define the domain of the S'-structure under consideration, and for each relation symbol (function symbol, constant) in S' an S-formula describing a relation (function, element) will be given. Again, we write $\varphi(v_0, \ldots, v_{n-1})$ for a formula $\varphi \in L_n^S$, and $\varphi(t_0, \ldots, t_{n-1})$ for $\varphi \frac{t_0 \ldots t_{n-1}}{v_0 \ldots v_{n-1}}$.

2.1 Definition. Let S and S' be symbol sets. A *syntactic interpretation of S' in S* is a map $I: S' \cup \{S'\} \to L^S$ where

$I(S')$	is a formula	$\varphi_{S'}(v_0) \in L_1^S$,	
$I(R)$	is a formula	$\varphi_R(v_0, \ldots, v_{n-1}) \in L_n^S$	for n-ary $R \in S'$,
$I(f)$	is a formula	$\varphi_f(v_0, \ldots, v_{n-1}, v_n) \in L_{n+1}^S$	for n-ary $f \in S'$,
$I(c)$	is a formula	$\varphi_c(v_0)$	for $c \in S'$.

In many applications one has $\varphi_{S'}(v_0) = v_0 \equiv v_0$.

The following set Φ_I of S-sentences says that $\varphi_{S'}(v_0)$ defines the domain of an S'-structure.

$$\Phi_I = \begin{cases} \exists v_0 \varphi_{S'}(v_0), & \\ \forall v_0 \ldots \forall v_{n-1}((\varphi_{S'}(v_0) \wedge \ldots \wedge \varphi_{S'}(v_{n-1})) \to & \\ \quad \exists^{=1} v_n(\varphi_{S'}(v_n) \wedge \varphi_f(v_0, \ldots, v_{n-1}, v_n))) & \text{for } f \in S' \ n\text{-ary}, \\ \exists^{=1} v_0(\varphi_{S'}(v_0) \wedge \varphi_c(v_0)) & \text{for } c \in S'. \end{cases}$$

If $\varphi_{S'}(v_0) = v_0 \equiv v_0$ then Φ_I is equivalent[1] to

$$\{\forall v_0 \ldots \forall v_{n-1} \exists^{=1} v_n \varphi_f(v_0, \ldots, v_{n-1}, v_n) \mid f \in S' \text{ } n\text{-ary}\} \cup \{\exists^{=1} v_0 \varphi_c(v_0) \mid c \in S'\}.$$

For an S-structure $\mathfrak{A}$ with $\mathfrak{A} \models \Phi_I$ we define an S'-structure $\mathfrak{A}^{-I}$ as follows:

$$A^{-I} := \{a \in A \mid \mathfrak{A} \models \varphi_{S'}[a]\};$$

for n-ary $R \in S'$ and $a_0, \ldots, a_{n-1} \in A^{-I}$:

$$R^{A^{-I}} a_0 \ldots a_{n-1} \quad \text{:iff} \quad \mathfrak{A} \models \varphi_R[a_0, \ldots, a_{n-1}];$$

for n-ary $f \in S'$ and $a_0, \ldots, a_{n-1}, a \in A^{-I}$:

$$f^{A^{-I}}(a_0, \ldots, a_{n-1}) = a \quad \text{:iff} \quad \mathfrak{A} \models \varphi_f[a_0, \ldots, a_{n-1}, a];$$

for $c \in S'$ and $a \in A^{-I}$:

$$c^{A^{-I}} = a \quad \text{:iff} \quad \mathfrak{A} \models \varphi_c[a].$$

If $R \in S \cap S'$ is n-ary and $\varphi_R = Rv_0 \ldots v_{n-1}$, we say that I is the *identity on* R. Similarly, I is the identity on $f \in S'$ (for n-ary f) and $c \in S'$, if $\varphi_f = fv_0 \ldots v_{n-1} \equiv v_n$ and $\varphi_c = c \equiv v_0$, respectively. If $S \subset S'$, $\varphi_{S'} = v_0 \equiv v_0$, and if I is the identity on all symbols from S, then

$$\mathfrak{A}^{-I}|_S = \mathfrak{A}$$

for all S-structures $\mathfrak{A}$ with $\mathfrak{A} \models \Phi_I$.

Using a syntactic interpretation of S' in S we can talk in S-structures about induced S'-structures:

2.2 Theorem on Syntactic Interpretations. *Let I be a syntactic interpretation of S' in S. Then, to every $\psi \in L^{S'}$ one can associate a $\psi^I \in L^S$ with* $\text{free}(\psi) = \text{free}(\psi^I)$ *such that for all S-structures $\mathfrak{A}$ with $\mathfrak{A} \models \Phi_I$ and all assignments β in $\mathfrak{A}^{-I}$,*

$$(*) \quad (\mathfrak{A}, \beta) \models \psi^I \quad \textit{iff} \quad (\mathfrak{A}^{-I}, \beta) \models \psi.$$

In particular, for $\psi \in L_0^{S'}$,

$$\mathfrak{A} \models \psi^I \quad \textit{iff} \quad \mathfrak{A}^{-I} \models \psi.$$

Before proving the theorem we want to apply it to clear up the claims in the paragraphs A to C. The relativization from paragraph D will be discussed at the end. In the sequel we use $x, y, \ldots$ for $v_0, v_1, \ldots$.

In the ring-theoretic example from paragraph C, concerning the group of units, we choose the syntactic interpretation I of $S_{\text{gr}} = \{\circ, e\}$ in $S_{\text{ar}} = \{+, \cdot, 0, 1\}$ given by

[1] We call two sets Φ and Ψ of S-sentences *equivalent* if $\text{Mod}^S\Phi = \text{Mod}^S\Psi$. Then, in particular, $\Phi \models \chi$ iff $\Psi \models \chi$ for all $\chi \in L^S$.

$$\begin{aligned} I(S_{\mathrm{gr}}) &:= \varepsilon(x), \\ I(\circ) &:= x \cdot y = z. \end{aligned}$$

Then Φ_I is equivalent to

$$\{\exists x \varepsilon(x), \forall x \forall y (\varepsilon(x) \wedge \varepsilon(y) \to \varepsilon(x \cdot y))\},$$

and for a ring $\mathfrak{A}$ we have $\mathfrak{A} \models \Phi_I$ and $\mathfrak{A}^{-I} = \mathfrak{E}(\mathfrak{A})$. For $\varphi \in L_0^{S_{\mathrm{gr}}}$ the equivalence $(*)$ in 2.2 says

$$\mathfrak{E}(\mathfrak{A}) \models \varphi \quad \text{iff} \quad \mathfrak{A} \models \varphi^I,$$

which is the claim $(\circ)$ in paragraph C (if we set $\varphi' := \varphi^I$).

In the example about orderings in paragraph B we define the syntactic interpretation I of $S' = \{<, \leq\}$ in $S = \{<\}$ as follows:

$$\begin{aligned} \varphi_{S'} &:= v_0 \equiv v_0; \\ \varphi_{<} &:= v_0 < v_1; \\ \varphi_{\leq} &:= (v_0 < v_1 \vee v_0 \equiv v_1). \end{aligned}$$

Then Φ_I is equivalent to the empty set, and φ^I is a $\{<\}$-sentence for $\varphi \in L_0^{S'} = L^{\{<,\leq\}}$. Theorem 2.2 yields for every $\varphi \in L_0^{S'}$ and every S-structure $\mathfrak{A}$

$$\mathfrak{A}^{-I} \models \varphi \quad \text{iff} \quad \mathfrak{A} \models \varphi^I.$$

Since $\mathfrak{A} \models \Phi_{\mathrm{ord}}$ implies $\mathfrak{A}^{-I} \models \Phi'_{\mathrm{ord}}$ (Φ'_{ord} was defined in paragraph B), and since for every S'-structure $\mathfrak{B}$ with $\mathfrak{B} \models \Phi'_{\mathrm{ord}}$ we have $\mathfrak{B}|_S \models \Phi_{\mathrm{ord}}$ and $(\mathfrak{B}|_S)^{-I} = \mathfrak{B}$, we obtain

$$\Phi'_{\mathrm{ord}} \models \varphi \quad \text{iff} \quad \Phi_{\mathrm{ord}} \models \varphi^I.$$

Finally we discuss the group theoretic example from paragraph A. We use the following syntactic interpretation I of S_{grp} in S_{g} (writing $x, y, \ldots$ for $v_0, v_1, \ldots$):

$$\begin{aligned} \varphi_{S_{\mathrm{grp}}}(x) &:= x \equiv x, \\ \varphi_{\circ}(x, y, z) &:= x \circ y \equiv z, \\ \varphi_{-1}(x, y) &:= \exists z (\forall u\, u \circ z \equiv u \wedge x \circ y \equiv z), \\ \varphi_e(x) &:= \forall y\, y \circ x \equiv y. \end{aligned}$$

Then we can argue as in the previous example and obtain: If $\mathfrak{A} = (A, \circ^A)$ is a group (as an S_{g}-structure) with identity element e^A and inverse function ${}^{-1^A}$, then $\mathfrak{A}^{-I} = (A, \circ^A, {}^{-1^A}, e^A)$, and for all $\varphi \in L_0^{S_{\mathrm{grp}}}$ we have

$$\mathfrak{A}^{-I} \models \varphi \quad \text{iff} \quad \mathfrak{A} \models \varphi^I$$

and

$$\Phi_{\mathrm{grp}} \models \varphi \quad \text{iff} \quad \Phi_{\mathrm{g}} \models \varphi^I.$$

We now turn to the *proof* of Theorem 2.2. It suffices to define ψ^I for term-reduced $\psi \in L^{S'}$. (Then, for arbitrary $\psi \in L^{S'}$, we can set $\psi^I :=$

$[\psi^*]^I$, where (cf. Theorem 1.2) ψ^* is a term-reduced S'-formula logically equivalent to ψ with $\text{free}(\psi) = \text{free}(\psi^*)$.) We set

for n-ary $R \in S'$,

$$\begin{array}{lll} [Rx_0 \ldots x_{n-1}]^I & := & \varphi_R(x_0, \ldots, x_{n-1}); \\ [x \equiv y]^I & := & x \equiv y; \end{array}$$

for n-ary $f \in S'$,

$$[fx_0 \ldots x_{n-1} \equiv x]^I := \varphi_f(x_0, \ldots, x_{n-1}, x);$$

for $c \in S'$,

$$[c \equiv x]^I := \varphi_c(x);$$

and

$$\begin{array}{lll} [\neg\varphi]^I & := & \neg\varphi^I; \\ (\varphi_1 \vee \varphi_2)^I & := & (\varphi_1^I \vee \varphi_2^I); \\ [\exists x\varphi]^I & := & \exists x(\varphi_{S'}(x) \wedge \varphi^I). \end{array}$$

Using this definition it is not difficult to prove $(*)$. We demonstrate the step involving a quantifier. So, let $\mathfrak{A}$ be an S-structure with $\mathfrak{A} \models \Phi_I$, and let β be an assignment in $\mathfrak{A}^{-I}$. Then:

$(\mathfrak{A}, \beta) \models [\exists x\varphi]^I$ iff $(\mathfrak{A}, \beta) \models \exists x(\varphi_{S'}(x) \wedge \varphi^I)$
iff for some $a \in A$, $(\mathfrak{A}, \beta\frac{a}{x}) \models \varphi_{S'}(x)$ and $(\mathfrak{A}, \beta\frac{a}{x}) \models \varphi^I$
iff for some $a \in A^{-I}$, $(\mathfrak{A}, \beta\frac{a}{x}) \models \varphi^I$
iff for some $a \in A^{-I}$, $(\mathfrak{A}^{-I}, \beta\frac{a}{x}) \models \varphi$ (induction hypothesis)
iff $(\mathfrak{A}^{-I}, \beta) \models \exists x\varphi$. □

Finally, we come back to the relativizations as introduced in paragraph D and present as a further application of Theorem 2.2 a precise statement of the connection between a formula and its relativization.

Let $S = S' \cup \{P\}$, where P is a unary relation symbol not contained in S'. Let the syntactic interpretation I of S' in S be the identity on the symbols from S', and let

$$\varphi_{S'}(v_0) := Pv_0.$$

Then Φ_I is equivalent to

$$\{\exists v_0 Pv_0\} \cup \{Pc \mid c \in S'\} \cup \\ \{\forall v_0 \ldots \forall v_{n-1}(Pv_0 \wedge \ldots \wedge Pv_{n-1} \to Pfv_0 \ldots v_{n-1}) \mid f \in S',\ f \text{ is } n\text{-ary}\},$$

and for an S-structure $(\mathfrak{A}, P^A)$ we have:

(+) $(\mathfrak{A}, P^A) \models \Phi_I$ iff P^A is S'-closed in $\mathfrak{A}$.

(++) If P^A is S'-closed in $\mathfrak{A}$ then $(\mathfrak{A}, P^A)^{-I} = [P^A]^{\mathfrak{A}}$.

(Remember that for an S-closed subset X of an S-structure $\mathfrak{A}$, $[X]^{\mathfrak{A}}$ is the substructure of $\mathfrak{A}$ with domain X; cf. III.5.)

If $\psi \in L^{S'}$, we also write ψ^P for ψ^I, and we call ψ^P the *relativization of* ψ *to* P. Hence (+) and (++) yield (note that P^A being S-closed implies that it is $S \cup \{P\}$-closed):

2.3 Relativization Lemma. *Let* $\mathfrak{A}$ *be an* $S \cup \{P\}$*-structure such that* $P \notin S$ *and* P *is unary. Suppose the set* $P^A \subset A$ *is* S*-closed in* $\mathfrak{A}$*. Then for* $\psi \in L_0^S$,

$$[P^A]^{\mathfrak{A}} \models \psi \quad \textit{iff} \quad \mathfrak{A} \models \psi^P.$$

(This means: *The relativization* ψ^P *says in* $\mathfrak{A}$ *the same as* ψ *does in* $[P^A]^{\mathfrak{A}}$.) □

It is easy to give a direct proof of the Relativization Lemma. For this purpose one defines for $\psi \in L^S$ the formula $\psi^P \in L^{S \cup \{P\}}$ inductively by

$$\begin{array}{lcl} \psi^P & := & \psi, \text{ if } \psi \text{ is atomic} \\ [\neg\psi]^P & := & \neg\psi^P \\ (\psi_1 \vee \psi_2)^P & := & (\psi_1^P \vee \psi_2^P) \\ [\exists x\psi]^P & := & \exists x(Px \wedge \psi^P). \end{array}$$

Then one shows by induction on the formula ψ that for all assignments $\beta : \{v_n \mid n \in \mathbf{N}\} \to P^A$,

$$([P^A]^{\mathfrak{A}}, \beta) \models \psi \quad \text{iff} \quad (\mathfrak{A}, \beta) \models \psi^P.$$

2.4 Exercise. Let U and V be distinct unary relation symbols, $U, V \notin S$. Assume $(\mathfrak{A}, U^A, V^A)$ to be an $S \cup \{U, V\}$-structure such that U^A and V^A are S-closed in $\mathfrak{A}$ and $U^A \subset V^A$. Show that for $\varphi \in L_0^S$,

$$(\mathfrak{A}, U^A, V^A) \models ([\varphi^V]^U \leftrightarrow \varphi^U).$$

2.5 Exercise. Let $<$ and $\leq$ be two binary relation symbols. Show that for every $\varphi \in L_0^{\{<\}}$ there is a $\psi \in L_0^{\{\leq\}}$, and that for every $\psi \in L_0^{\{\leq\}}$ there is a $\varphi \in L_0^{\{<\}}$ such that (a) and (b), respectively, hold:
(a) An ordering $(A, <^A)$ satisfies φ iff the corresponding ordering $(A, \leq^A)$ satisfies the sentence ψ.
(b) An ordering $(A, \leq^A)$ in the sense of "$\leq$" satisfies ψ iff the corresponding ordering $(A, <)$ satisfies the sentence φ.

2.6 Exercise. In the discussion of groups (Example A) following the statement of Theorem 2.2, interchange the roles of Φ_{grp} and Φ_{g}.

2.7 Exercise. (a) Give a syntactic interpretation I of S_{ar} in S_{ar} such that for all $\varphi \in L_0^{S_{\text{ar}}}$:

$$(\mathbf{N}, +, \cdot, 0, 1) \models \varphi \quad \text{iff} \quad (\mathbf{Z}, +, \cdot, 0, 1) \models \varphi^I.$$

Hint: Every natural number can be written as the sum of four squares of integers.
(b) Prove the analogue of (a) obtained by interchanging the roles of $\mathbf{N}$ and $\mathbf{Z}$.

2.8 Exercise. Prove Theorem 1.3 using Theorem 2.2 by applying suitable syntactic interpretations.

§3. Extensions by Definitions

In some of the previous examples we dealt with two axiom systems: the axiom systems Φ_{g} and Φ_{grp} for group theory (Example A), and the axiom systems Φ_{ord} and Φ'_{ord} for orderings (Example B).

Usually a mathematician does not work with two or more symbol sets for one and the same theory, but considers a single underlying symbol set which possibly is extended by "defined" symbols. Thus, in group theory one can start with the symbol set $S_{\mathrm{g}} = \{\circ\}$ and extend it to $S_{\mathrm{grp}} = \{\circ, {}^{-1}, e\}$ by the defined symbols for the inverse function and the unit element. For orderings one can start with $S = \{<\}$ and extend S to $S' = \{<, \leq\}$ by the defined symbol $\leq$. We proceeded in the same way when discussing set theory in VII.3; there we extended the symbol set $S = \{\in\}$ successively by the defined symbols $\emptyset$, $\cap$, $\cup$,.... Our goal in this section is to analyze these extensions by definitions. To clarify our intuitive expectation and to explain the idea, we take the transition from $S_{\mathrm{g}} = \{\circ\}$ to $S_{\mathrm{gr}} = \{\circ, e\}$ in the example from group theory. We use x, y, z for v_0, v_1, v_2.

The starting point is the axiom system $\Phi_{\mathrm{g}} \subset L_0^{S_{\mathrm{g}}}$. We notice that the unit element is uniquely determined, namely

$$\Phi_{\mathrm{g}} \models \exists^{=1} x \forall y\, y \circ x \equiv y.$$

Hence, we can introduce a new constant e to denote the unit element and fix its interpretation by the following definition:

$$\delta_e := \forall x (e \equiv x \leftrightarrow \forall y\, y \circ x \equiv y),$$

thus arriving at the new symbol set $S_{\mathrm{gr}} = \{\circ, e\}$ and the "extension by definitions"

$$\Phi_{\mathrm{g}} \cup \{\delta_e\}$$

as the new axiom system. (It is easy to show that the sets $\Phi_{\mathrm{g}} \cup \{\delta_e\}$ and Φ_{gr} of S_{gr}-sentences are equivalent.) Introducing e simplifies the notation, but we do not expect any major changes by this transition from $L^{S_{\mathrm{g}}}$ to $L^{S_{\mathrm{gr}}}$. This can be made precise as follows:

(E1) "*Extensions by definitions are conservative*":
For all $\varphi \in L_0^{S_{\mathrm{g}}}$: $\Phi_{\mathrm{g}} \cup \{\delta_e\} \models \varphi$ iff $\Phi_{\mathrm{g}} \models \varphi$
(thus, adding definitions does not increase the set of provable sentences).

(E2) "*Defined symbols can be eliminated*":
For the syntactic interpretation I of S_{gr} in S_{g} with

$$\begin{array}{lcl} \varphi_{S_{\mathrm{gr}}}(x) & := & x \equiv x \\ \varphi_{\circ}(x,y,z) & := & x \circ y \equiv z \\ \varphi_e(x) & := & \forall y\, y \circ x \equiv y \end{array}$$

the following holds for all $\chi \in L^{S_{\mathrm{gr}}}$:

$$\Phi_{\mathrm{g}} \cup \{\delta_e\} \models \chi \leftrightarrow \chi^I.$$

(E3) "*The elimination of defined symbols respects the theory*":
For I as in (E2) and $\varphi \in L_0^{S_{\mathrm{gr}}}$:

$$\Phi_{\mathrm{g}} \cup \{\delta_e\} \models \varphi \quad \text{iff} \quad \Phi_{\mathrm{g}} \models \varphi^I.$$

Note that (E3) follows immediately from (E1) and (E2), since for $\varphi \in L_0^{S_{\mathrm{gr}}}$ we have

$$\begin{array}{lll} \Phi_{\mathrm{g}} \cup \{\delta_e\} \models \varphi & \text{iff} \quad \Phi_{\mathrm{g}} \cup \{\delta_e\} \models \varphi^I & \text{(by (E2))} \\ & \text{iff} \quad \Phi_{\mathrm{g}} \models \varphi^I & \text{(by (E1))}. \end{array}$$

From the Theorem on Definitions which we now present, it will follow immediately that (E1) to (E3) are fulfilled.

3.1 Definition. Let Φ be a set of S-sentences.

(a) Suppose $P \notin S$ is an n-ary relation symbol and $\varphi_P(v_0, \ldots, v_{n-1})$ an S-formula. Then we say that

$$\forall v_0 \ldots \forall v_{n-1}(Pv_0 \ldots v_{n-1} \leftrightarrow \varphi_P(v_0, \ldots, v_{n-1}))$$

is an *S-definition of P in Φ*.

(b) Suppose $f \notin S$ is an n-ary function symbol and $\varphi_f(v_0, \ldots, v_{n-1}, v_n)$ an S-formula. We say that

$$\forall v_0 \ldots \forall v_n(fv_0 \ldots v_{n-1} \equiv v_n \leftrightarrow \varphi_f(v_0, \ldots, v_{n-1}, v_n))$$

is an *S-definition of f in Φ* provided

$$\Phi \models \forall v_0 \ldots \forall v_{n-1} \exists^{=1} v_n \varphi_f(v_0, \ldots, v_n).$$

(c) Suppose $c \notin S$ is a constant and $\varphi_c(v_0)$ an S-formula. We say that

$$\forall v_0(c \equiv v_0 \leftrightarrow \varphi_c(v_0))$$

is an *S-definition of c in Φ* provided

$$\Phi \models \exists^{=1} v_0 \varphi_c(v_0).$$

Thus

$$\forall x \forall y(x \leq y \leftrightarrow (x < y \vee x \equiv y))$$

is an $\{<\}$-definition of $\leq$ in Φ_{ord},

$$\forall x(e \equiv x \leftrightarrow \forall y\, y \circ x \equiv y)$$

is an S_g-definition of e in Φ_g, and

$$\forall x \forall y \forall z (x \cap y \equiv z \leftrightarrow \forall w (w \in z \leftrightarrow (w \in x \wedge w \in y)))$$

is an $\{\in\}$-definition of $\cap$ in ZFC.

Let S be given, and let s be a relation symbol, a function symbol or a constant with $s \notin S$. Furthermore, let $\Phi \subset L_0^S$ and let δ_s be an S-definition of s in Φ. We define, in the obvious way, the *associated* syntactic interpretation I of $S' := S \cup \{s\}$ in S to be the identity on the symbols from S and

$$(I(S') =)\varphi_{S'}(v_0) := v_0 \equiv v_0,\ I(s) := \varphi_s$$

(where φ_s is as in Definition 3.1). So Φ_I is logically equivalent to

- the empty set of sentences, if s is a relation symbol,
- $\{\forall v_0 \ldots \forall v_{n-1} \exists^{=1} v_n \varphi_f(v_0, \ldots, v_{n-1}, v_n)\}$, if s is an n-ary function symbol f,
- $\{\exists^{=1} v_0 \varphi_c(v_0)\}$, if s is a constant c.

Therefore, we have for every S-structure $\mathfrak{A}$ with $\mathfrak{A} \models \Phi$:

$(*)$ $\quad \mathfrak{A} \models \Phi_I$

$(**)$ $\quad (\mathfrak{A}, s^A) \models \delta_s \quad$ iff $\quad \mathfrak{A}^{-I} = (\mathfrak{A}, s^A)$.

Now we easily reach our goal:

3.2 Theorem on Definitions. *Let Φ be a set of S-sentences, s a new symbol, δ_s an S-definition of s in Φ and I the associated syntactic interpretation of $S \cup \{s\}$ in S. Then:*

(a) *For all $\varphi \in L_0^S$,*

$$\Phi \cup \{\delta_s\} \models \varphi \quad \textit{iff} \quad \Phi \models \varphi.$$

(b) *For all $\chi \in L_0^{S \cup \{s\}}$,*

$$\Phi \cup \{\delta_s\} \models \chi \leftrightarrow \chi^I.$$

(c) *For all $\varphi \in L_0^{S \cup \{s\}}$,*

$$\Phi \cup \{\delta_s\} \models \varphi \quad \textit{iff} \quad \Phi \models \varphi^I.$$

Proof. (a) For the proof of the non-trivial direction, assume that $\Phi \cup \{\delta_s\} \models \varphi$, and let $\mathfrak{A}$ be an S-structure with $\mathfrak{A} \models \Phi$. By $(*)$, $\mathfrak{A}^{-I}$ is defined, say $\mathfrak{A}^{-I} = (\mathfrak{A}, s^A)$. By $(**)$ it follows that $(\mathfrak{A}, s^A) \models \Phi \cup \{\delta_s\}$, therefore by assumption $(\mathfrak{A}, s^A) \models \varphi$, and hence $\mathfrak{A} \models \varphi$ by the Coincidence Lemma.
(b) Let $\chi \in L_0^{S \cup \{s\}}$ and let $(\mathfrak{A}, s^A)$ be an $(S \cup \{s\})$-structure such that

$$(\mathfrak{A}, s^A) \models \Phi \cup \{\delta_s\}.$$

By the Theorem on Syntactic Interpretations, the following holds for $\mathfrak{A}^{-I}$ $(= (\mathfrak{A}, s^A)$; cf. $(**))$:

$$\begin{aligned} (\mathfrak{A}, s^A) \models \chi \quad &\text{iff} \quad \mathfrak{A} \models \chi^I \\ &\text{iff} \quad (\mathfrak{A}, s^A) \models \chi^I. \end{aligned}$$

□

3.3 Exercise. Generalize Theorem 3.2 to the case of more (possibly infinitely many) definitions of new symbols.

3.4 Exercise. Formulate precisely and show: For a set Φ of S-sentences the following holds: An extension by definitions of an extension by definitions of Φ is an extension by definitions of Φ.

3.5 Exercise. Let P be a k-ary relation symbol, $P \notin S$, and Φ' a set of $(S \cup \{P\})$-sentences which *implicitly defines* P, in the sense that for every S-structure $\mathfrak{A}$ and all $P^1, P^2 \subset A^k$ the following holds:

If $(\mathfrak{A}, P^1) \models \Phi'$ and $(\mathfrak{A}, P^2) \models \Phi'$ then $P^1 = P^2$.

Then, by Beth's Definability Theorem (see Exercise XIII.3.7), there is an *explicit definition* of P with respect to Φ', i.e. there is an S-formula $\varphi_P(v_0, \ldots, v_{k-1})$ such that

$$\Phi' \models \forall v_0 \ldots \forall v_{k-1}(Pv_0 \ldots v_{k-1} \leftrightarrow \varphi_P(v_0, \ldots, v_{k-1})).$$

Using this, show that there is a set Φ of S-sentences and a definition δ_P of P in Φ such that for all $\varphi \in L_0^{S \cup \{P\}}$,

$$\Phi \cup \{\delta_P\} \models \varphi \quad \text{iff} \quad \Phi' \models \varphi;$$

thus Φ' is, up to equivalence, an extension of Φ by definitions.

§4. Normal Forms

In this section we show that one can associate with every formula a logically equivalent formula which has a special syntactic form.

Let S be a fixed symbol set. For an arbitrary set Φ of S-formulas let $\langle \Phi \rangle$ be the smallest subset of L^S containing Φ, and containing with any ψ and χ the formulas $\neg\psi$ and $(\psi \vee \chi)$. Note that $\Phi \subset L_r^S$ implies $\langle \Phi \rangle \subset L_r^S$.

4.1 Lemma. *Let $\Phi \subset L_r^S$. Suppose $\mathfrak{A}$ and $\mathfrak{B}$ are S-structures, and $a_0, \ldots, a_{r-1} \in A$, $b_0, \ldots, b_{r-1} \in B$. If*

$$(*) \qquad \mathfrak{A} \models \varphi[a_0, \ldots, a_{r-1}] \text{ iff } \mathfrak{B} \models \varphi[b_0, \ldots, b_{r-1}]$$

holds for all $\varphi \in \Phi$, then $(*)$ holds for all $\varphi \in \langle \Phi \rangle$.

Proof. The set of φ for which $(*)$ holds includes Φ and with any ψ and χ also contains $\neg\psi$ and $(\psi \vee \chi)$. □

4.2 Lemma. *Let $\Phi = \{\varphi_0, \ldots, \varphi_n\}$ be a finite set of formulas. Then every satisfiable formula in $\langle \Phi \rangle$ is logically equivalent to a formula of the form*

$$(+) \qquad (\psi_{0,0} \wedge \cdots \wedge \psi_{0,n}) \vee \cdots \vee (\psi_{k,0} \wedge \cdots \psi_{k,n})$$

where $k < 2^{n+1}$ and for $i \leq k$ and $j \leq n$, $\psi_{i,j} = \varphi_j$ or $\psi_{i,j} = \neg\varphi_j$. In particular, there are only finitely many pairwise logically nonequivalent formulas in $\langle\Phi\rangle$.

Thus, we see that every formula in $\langle\Phi\rangle$ is logically equivalent to a disjunction of conjunctions of formulas from $\{\varphi_0, \ldots, \varphi_n, \neg\varphi_0, \ldots, \neg\varphi_n\}$.

Proof. We choose an r such that $\Phi = \{\varphi_0, \ldots, \varphi_n\} \subset L_r^S$. For a structure $\mathfrak{A}$ and an r-tuple $\vec{a} := (a_0, \ldots, a_{r-1}) \in A^r$ let

$$\psi_{(\mathfrak{A},\vec{a})} := \psi_0 \wedge \cdots \wedge \psi_n, \tag{1}$$

where

$$\psi_i := \begin{cases} \varphi_i, & \text{if } \mathfrak{A} \models \varphi_i[a_0, \ldots, a_{r-1}], \\ \neg\varphi_i, & \text{if } \mathfrak{A} \models \neg\varphi_i[a_0, \ldots, a_{r-1}]. \end{cases}$$

Then

$$\mathfrak{A} \models \psi_{(\mathfrak{A},\vec{a})}[a_0, \ldots, a_{r-1}], \tag{2}$$

and $\psi_{(\mathfrak{A},\vec{a})}$ is a conjunction of the form in (+). Moreover, for any $\mathfrak{B}$ and $b_0, \ldots, b_{r-1} \in B$,

$$\mathfrak{B} \models \psi_{(\mathfrak{A},\vec{a})}[b_0, \ldots, b_{r-1}] \tag{3}$$

iff for $i = 0, \ldots, n$:

$$\mathfrak{A} \models \varphi_i[a_0, \ldots, a_{r-1}] \quad \text{iff} \quad \mathfrak{B} \models \varphi_i[b_0, \ldots, b_{r-1}]$$

iff (cf. 4.1) for all $\varphi \in \langle\Phi\rangle$:

$$\mathfrak{A} \models \varphi[a_0, \ldots, a_{r-1}] \quad \text{iff} \quad \mathfrak{B} \models \varphi[b_0, \ldots, b_{r-1}].$$

From (1) it follows that the set

$$\{\psi_{(\mathfrak{A},\vec{a})} \mid \mathfrak{A} \text{ is an } S\text{-structure and } \vec{a} \in A^r\}$$

has at most 2^{n+1} elements.

The proof is complete if we can show that every satisfiable $\varphi \in \langle\Phi\rangle$ is logically equivalent to the disjunction χ of the finitely many formulas from the set

$$\{\psi_{(\mathfrak{A},\vec{a})} \mid \mathfrak{A} \text{ is an } S\text{-structure}, \vec{a} \in A^r, \ \mathfrak{A} \models \varphi[a_0, \ldots, a_{r-1}]\}.$$

In a suggestive notation, we write

$$\chi = \bigvee\{\psi_{(\mathfrak{A},\vec{a})} \mid \mathfrak{A} \text{ is an } S\text{-structure}, \vec{a} \in A^r, \ \mathfrak{A} \models \varphi[a_0, \ldots, a_{r-1}]\}.$$

In order to verify the equivalence between φ and χ, assume first that $\mathfrak{B} \models \varphi[b_0, \ldots, b_{r-1}]$. Then $\psi_{(\mathfrak{B},\vec{b})}$ is a member of the disjunction χ. Since $\mathfrak{B} \models$

$\psi_{(\mathfrak{B},\vec{b})}[b_0,\ldots,b_{r-1}]$ (cf. (2)) it follows that $\mathfrak{B} \models \chi[b_0,\ldots,b_{r-1}]$. Conversely, if $\mathfrak{B} \models \chi[b_0,\ldots,b_{r-1}]$, then by definition of χ there is a structure $\mathfrak{A}$ and there are $a_0,\ldots,a_{r-1} \in A$ such that

$$\mathfrak{A} \models \varphi[a_0,\ldots,a_{r-1}] \qquad \text{and} \qquad \mathfrak{B} \models \psi_{(\mathfrak{A},\vec{a})}[b_0,\ldots,b_{r-1}].$$

Then, by (3), $b_0,\ldots,b_{r-1}$ satisfy the same formulas of $\langle\Phi\rangle$ in $\mathfrak{B}$ as $a_0,\ldots,a_{r-1}$ do in $\mathfrak{A}$. In particular, $\mathfrak{B} \models \varphi[b_0,\ldots,b_{r-1}]$. □

A formula which is a disjunction of conjunctions of atomic or negated atomic formulas is called a *formula in disjunctive normal form.* A formula which contains no quantifiers is said to be *quantifier-free.* As a corollary to 4.2 we obtain

4.3 Theorem on the Disjunctive Normal Form. *If φ is quantifier-free, then φ is logically equivalent to a formula in disjunctive normal form.*

Proof. Let φ be a quantifier-free formula. If φ is not satisfiable then φ is logically equivalent to $\neg v_0 \equiv v_0$. If φ is satisfiable and $\psi_0,\ldots,\psi_n$ are the atomic subformulas in φ, then $\varphi \in \langle\{\psi_0,\ldots,\psi_n\}\rangle$. The theorem now follows from 4.2. □

We turn to formulas which also contain quantifiers. A formula ψ is said to be in *prenex normal form* if it has the form $Q_0x_0\ldots Q_{m-1}x_{m-1}\psi_0$, where $Q_i = \exists$ or $Q_i = \forall$ for $i < m$ and ψ_0 is quantifier-free. $Q_0x_0\ldots Q_{m-1}x_{m-1}$ is called the *prefix* and ψ_0 the *matrix* of ψ.

4.4 Theorem on the Prenex Normal Form. *Every formula φ is logically equivalent to a formula ψ in prenex normal form with* $\mathrm{free}(\varphi) = \mathrm{free}(\psi)$.

Proof. First, we note some simple properties of logical equivalence. By "$\varphi \sim \psi$" we mean that φ and ψ are logically equivalent.

(1) If $\varphi \sim \psi$, then $\neg\varphi \sim \neg\psi$.
(2) If $\varphi_0 \sim \psi_0$ and $\varphi_1 \sim \psi_1$, then $(\varphi_0 \vee \varphi_1) \sim (\psi_0 \vee \psi_1)$.
(3) If $\varphi \sim \psi$ and $Q = \exists$ or $Q = \forall$, then $Qx\varphi \sim Qx\psi$.
(4) $\neg\exists x\varphi \sim \forall x\neg\varphi$, $\neg\forall x\varphi \sim \exists x\neg\varphi$.
(5) If $x \notin \mathrm{free}(\psi)$, then $(\exists x\varphi \vee \psi) \sim \exists x(\varphi \vee \psi)$, $(\forall x\varphi \vee \psi) \sim \forall x(\varphi \vee \psi)$, $(\psi \vee \exists x\varphi) \sim \exists x(\psi \vee \varphi)$, and $(\psi \vee \forall x\varphi) \sim \forall x(\psi \vee \varphi)$.

We shall see how one can transform a given formula into prenex normal form by repeated applications of (1)–(5). For instance, if $\varphi = \neg\exists xPx \vee \forall xRx$ we can proceed as follows:

$$\begin{aligned}
\neg\exists xPx \vee \forall xRx &\sim \forall x\neg Px \vee \forall xRx && \text{(by (2) and (4))}\\
&\sim \forall x\neg Px \vee \forall yRy && \text{(since } \forall xRx \sim \forall yRy \text{ and by (2))}\\
&\sim \forall x(\neg Px \vee \forall yRy) && \text{(by (5))}\\
&\sim \forall x\forall y(\neg Px \vee Ry) && \text{(by (3) and (5)).}
\end{aligned}$$

In general, we argue as follows: For $\varphi \in L^S$ let $\mathrm{qn}(\varphi)$ be the *quantifier number* of φ, i.e. the number of quantifiers occurring in φ. Using induction on n, we prove:

$(*)_n$ For φ with $\mathrm{qn}(\varphi) \leq n$ there is a $\psi \in L^S$ in prenex normal form such that $\varphi \sim \psi$, $\mathrm{free}(\varphi) = \mathrm{free}(\psi)$, and $\mathrm{qn}(\varphi) = \mathrm{qn}(\psi)$.

We leave the arguments for "$\mathrm{free}(\varphi) = \mathrm{free}(\psi)$" to the reader.

$n = 0$: If $\mathrm{qn}(\varphi) = 0$, φ is quantifier-free and we can set $\psi := \varphi$.

$n > 0$: We show $(*)_n$ by induction on φ. Suppose $\mathrm{qn}(\varphi) \leq n$. The quantifier-free case is clear. If $\varphi = \neg\varphi'$ and $\mathrm{qn}(\varphi) > 0$, then $\mathrm{qn}(\varphi') = \mathrm{qn}(\varphi) > 0$, and by induction hypothesis there is a formula of the form $Qx\chi$ which is a prenex normal form for φ' (where $\mathrm{qn}(Qx\chi) = \mathrm{qn}(\varphi)$ and where χ may contain quantifiers). By (1) and (4), $\varphi \sim Q^{-1}x\neg\chi$ (where $\forall^{-1} := \exists$ and $\exists^{-1} := \forall$). Since $\mathrm{qn}(\neg\chi) = \mathrm{qn}(Qx\chi) - 1 = \mathrm{qn}(\varphi) - 1 \leq n - 1$, there exists a formula ψ logically equivalent to $\neg\chi$ which is in prenex normal form such that $\mathrm{qn}(\psi) = \mathrm{qn}(\neg\chi)$. By (3), $Q^{-1}x\psi$ is a formula logically equivalent to φ with the desired properties.

Let $\varphi = (\varphi' \vee \varphi'')$ and let $\mathrm{qn}(\varphi) > 0$, e.g., $\mathrm{qn}(\varphi') > 0$. By induction hypothesis there is a formula of the form $Qx\chi$ which is a prenex normal form for φ'. Let y be a variable which does not occur in $Qx\chi$ or in φ''. It is then easy to show that

$$Qx\chi \sim Qy\chi\tfrac{y}{x}$$

and thus, by (2) and (5), to obtain

$$\begin{aligned} \varphi = (\varphi' \vee \varphi'') &\sim (Qy\chi\tfrac{y}{x} \vee \varphi'') \\ &\sim Qy(\chi\tfrac{y}{x} \vee \varphi''). \end{aligned}$$

Since $\mathrm{qn}(\chi\frac{y}{x} \vee \varphi'') = \mathrm{qn}(\varphi) - 1 \leq n - 1$, we can find a formula ψ in prenex normal form which is logically equivalent to $(\chi\frac{y}{x} \vee \varphi'')$. $Qy\psi$ has the desired properties.

Let $\varphi = \exists x\varphi'$. Since $\mathrm{qn}(\varphi') \leq n - 1$ there is a formula ψ' in prenex normal form which is logically equivalent to φ'. $\exists x\psi'$ is a formula in prenex normal form which, by (3), is logically equivalent to φ and has the same quantifier number as φ. □

If φ and ψ are formulas such that

$$\mathrm{Sat}\ \varphi \quad \text{iff} \quad \mathrm{Sat}\ \psi,$$

we call φ and ψ *equivalent for satisfaction.* If, in the Theorem on the Prenex Normal Form, the condition of logical equivalence is weakened to $\psi \models \varphi$ and equivalence for satisfaction, the formula ψ can, in addition, be chosen universal, i.e. in such a way that its prefix contains only universal

quantifiers. The following example serves as an illustration. Let $S = \{R\}$ and let φ be the S-formula $\forall x \exists y Rxy$. We set $S' := \{R, f\}$ with unary f and $\psi := \forall x Rxfx$. Then ψ is universal and $\psi \models \varphi$. Hence each model of ψ is a model of φ. On the other hand, let (A, R^A) be a model of $\forall x \exists y Rxy$. As we have, for every $a \in A$, an element $b \in A$ with $R^A ab$, we can choose an interpretation f^A of f in such a way that $R^A a f^A(a)$ for all $a \in A$, i.e. $(A, R^A, f^A) \models \forall x Rxfx$. Hence $\forall x Rxfx$ has a model, too.

4.5 Theorem on the Skolem Normal Form. *To each formula φ one can associate a universal formula ψ in prenex normal form with $\psi \models \varphi$ and* $\text{free}(\varphi) = \text{free}(\psi)$ *such that φ and ψ are equivalent for satisfaction. Besides the symbols from φ, the formula ψ may contain additional function symbols or constants.*

Proof. We describe how we can "construct" ψ from φ. Often such a ψ is called a *Skolem normal form* of φ.

Let φ be an S-formula. According to 4.4, we can assume that φ is in prenex normal form, say,

$$\varphi = Q_1 x_1 \ldots Q_m x_m \varphi_0,$$

where φ_0 is quantifier-free. We proceed by induction on the number of existential quantifiers in the prefix $Q_1 x_1 \ldots Q_m x_m$.

If the number equals zero, we set $\psi := \varphi$. In the induction step, let φ be of the shape

$$\varphi = \forall x_1 \ldots \forall x_k \exists x_{k+1} Q_{k+2} x_{k+2} \ldots Q_m x_m \varphi_0.$$

We may assume that $x_1, \ldots, x_k$ are pairwise distinct. Let

$$\varphi_1 := Q_{k+2} x_{k+2} \ldots Q_m x_m \varphi_0$$

and let f be a new k-ary function symbol if $k \neq 0$ and a constant if $k = 0$. We show for

$$\psi' := \forall x_1 \ldots \forall x_k \varphi_1 \frac{f x_1 \ldots x_k}{x_{k+1}} :$$

(1) If Sat φ, then Sat ψ'.

(2) $\psi' \models \varphi$.

Then we are done: As the prefix of ψ' contains less existential quantifiers than the prefix of φ, the induction hypothesis yields a formula ψ in Skolem normal form such that

(3) ψ' and ψ are equivalent for satisfaction and $\text{free}(\psi') = \text{free}(\psi)$.

(4) $\psi \models \psi'$.

As $\text{free}(\psi') = \text{free}(\varphi)$, (1) to (4) yield that ψ is a formula with the desired properties.

To prove (1), let $\mathfrak{A}$ be an S-structure and $\mathfrak{I} = (\mathfrak{A}, \beta)$ a model of φ. Then, for all $a_1, \ldots, a_k \in A$, we have

$$\mathfrak{I}\frac{a_1 \ldots a_k}{x_1 \ldots x_k} \models \exists x_{k+1}\varphi_1,$$

hence, we can choose a function f^A on A such that

$$\text{for all } a_1, \ldots, a_k \in A : \mathfrak{I}\frac{a_1 \ldots a_k f^A(a_1, \ldots, a_k)}{x_1 \ldots x_k \qquad x_{k+1}} \models \varphi_1.$$

By the Substitution Lemma we get

$$\text{for all } a_1, \ldots, a_k \in A : ((\mathfrak{A}, f^A), \beta)\frac{a_1 \ldots a_k}{x_1 \ldots x_k} \models \varphi_1 \frac{fx_1 \ldots x_k}{x_{k+1}}.$$

Therefore, $((\mathfrak{A}, f^A), \beta)$ is a model of $\forall x_1 \ldots \forall x_k \varphi_1 \frac{fx_1 \ldots x_k}{x_{k+1}}$. Hence ψ' is satisfiable.

To prove (2), let $\mathfrak{I} = ((\mathfrak{A}, f^A), \beta)$ be a model of ψ'. Then we have for all $a_1, \ldots, a_k \in A$

$$\mathfrak{I}\frac{a_1 \ldots a_k}{x_1 \ldots x_k} \models \varphi_1 \frac{fx_1 \ldots x_k}{x_{k+1}},$$

hence,

$$\mathfrak{I}\frac{a_1 \ldots a_k}{x_1 \ldots x_k} \models \exists x_{k+1}\varphi_1$$

and $\mathfrak{I}$ is a model of φ. □

4.6 Exercise. Let φ be an S-sentence and ψ the universal sentence that we get from φ by the preceding proof. Furthermore, let $S \subset S'$ and $\psi \in L_0^{S'}$. Show for every S-structure $\mathfrak{A}$ that the following are equivalent:

(1) $\mathfrak{A} \models \varphi$.
(2) There is an S'-expansion $\mathfrak{A}'$ of $\mathfrak{A}$ that is a model of ψ.

4.7 Exercise (Conjunctive Normal Form)**.** Show: If φ is quantifier-free, then φ is logically equivalent to a formula which is a conjunction of disjunctions of atomic or negated atomic formulas.

4.8 Exercise. Let S be a relational symbol set and $\varphi \in L_0^S$ of the shape $\exists x_0 \ldots \exists x_n \forall y_0 \ldots y_m \psi$ with ψ quantifier-free. Show that every model of φ contains a substructure with at most $n+1$ elements which also is a model of φ. Conclude that the sentence $\forall x \exists y Rxy$ cannot be logically equivalent to a sentence of the same shape as φ.

PART B

IX

Extensions of First-Order Logic

We have seen that the structure $\mathfrak{N}$ of natural numbers cannot be characterized in the first-order language corresponding to $\mathfrak{N}$. The same situation holds for the field of real numbers and the class of torsion groups. As we showed in Chapter VII, one can, at least in principle, overcome this weakness by a set-theoretical formulation: One introduces a system of axioms for set theory in a first-order language, e.g. ZFC, which is sufficient for mathematics, and then, in this system, carries out the arguments which are required, say, for a definition and characterization of $\mathfrak{N}$. However, this approach necessitates an explicit use of set theory to an extent not usual in ordinary mathematical practice.

The situation may encourage to consider languages with more expressive power, which permit us to avoid this detour through set theory. For example, in so-called second-order language we can directly characterize the natural numbers by means of Peano's axioms. However, already at this stage we wish to remark that in order to set up the semantics of such a language and to prove the correctness of inference rules, one has to make more extensive use of set-theoretic assumptions (for example, of the ZFC axioms) than for first-order logic.

There is another reason for introducing and investigating more powerful languages. We saw that results such as the Compactness Theorem are useful in algebraic investigations (cf. VI.4). Therefore, it seems worthwhile to seek other, more expressive languages in the hope of obtaining tools for more far-reaching applications in mathematics.

In this chapter we introduce the reader to some of the languages which have been considered with these aims in mind.

§1. Second-Order Logic

The difference between second-order and first-order languages lies in the fact that in the former one can quantify over second-order objects (for example, subsets of the domain of a structure) whereas in the latter this is not possible.

1.1 The Second-Order Languages L_{II}^S. Let S be a symbol set, that is, a set of relation symbols, function symbols and constants. The alphabet of L_{II}^S contains, in addition to the symbols of L^S, for each $n \geq 1$ countably many n-ary relation variables $V_0^n, V_1^n, V_2^n, \ldots$. To denote relation variables we use letters $X, Y, \ldots$, where we indicate the arity by superscripts, if necessary. We define the set L_{II}^S of second-order S-formulas to be the set generated by the rules of the calculus for first-order formulas (cf. II.3.2), extended by the following two rules:

(a) If X is an n-ary relation variable and $t_1, \ldots, t_n$ are S-terms, then $Xt_1 \ldots t_n$ is an S-formula.

(b) If φ is an S-formula and X is a relation variable, then $\exists X\varphi$ is an S-formula.

1.2 The Satisfaction Relation for L_{II}^S. A *second-order assignment* γ in a structure $\mathfrak{A}$ is a map which assigns to each variable v_i an element of A and to each relation variable V_i^n an n-ary relation on A. We extend the notion of satisfaction from L^S to L_{II}^S by taking (a) and (b) into account as follows:

If $\mathfrak{A}$ is an S-structure, γ a second-order assignment in $\mathfrak{A}$ and $\mathfrak{I} = (\mathfrak{A}, \gamma)$, then we set:

(a′) $\mathfrak{I} \models Xt_1 \ldots t_n$:iff $\gamma(X)$ holds for $\mathfrak{I}(t_1), \ldots, \mathfrak{I}(t_n)$.

(b′) For n-ary X:
$\mathfrak{I} \models \exists X\varphi$:iff there is a $C \subset A^n$ such that $\mathfrak{I}\frac{C}{X} \models \varphi$

(where $\mathfrak{I}\frac{C}{X} = \left(\mathfrak{A}, \gamma\frac{C}{X}\right)$ and $\gamma\frac{C}{X}$ is the assignment which maps X to C but which otherwise agrees with γ).

We let $\mathcal{L}_{\mathrm{II}}$ denote *second-order logic*, that is, the logical system given by the languages L_{II}^S together with the satisfaction relation for these languages. Similarly, $\mathcal{L}_{\mathrm{I}}$ denotes first-order logic. For the present, we still use the term "logical system" in an informal sense. A precise definition will be given in XIII.1.

1.3 Remarks and Examples. (1) One defines the free occurrence of variables and relation variables in second-order formulas in the obvious way and can then prove the analogue of the Coincidence Lemma. In particular, when φ is an L_{II}^S-sentence, i.e. a formula without free variables or free

relation variables, it is meaningful to say that $\mathfrak{A}$ is a model of φ, written $\mathfrak{A} \models \varphi$.

(2) Let $\forall X\varphi$ be an abbreviation for $\neg\exists X\neg\varphi$. Then

$$\mathfrak{I} \models \forall X^n\varphi \quad \text{iff} \quad \text{for all } C \subset A^n\text{: } \mathfrak{I}\frac{C}{X} \models \varphi.$$

(3) If X is a unary relation variable, then the following formalizations of Peano's axioms, which we already encountered in III.7.3, are $L_{\mathrm{II}}^{\{\boldsymbol{\sigma},0\}}$-sentences:

(P1) $\forall x \neg \boldsymbol{\sigma} x \equiv 0$;
(P2) $\forall x\forall y(\boldsymbol{\sigma} x \equiv \boldsymbol{\sigma} y \to x \equiv y)$;
(P3) $\forall X((X0 \wedge \forall x(Xx \to X\boldsymbol{\sigma} x)) \to \forall y Xy)$.

Hence, by passing from first-order to second-order logic we have gained expressive power, since no first-order axioms can characterize the structure $(\mathbf{N}, \boldsymbol{\sigma}, 0)$ up to isomorphism.

(4) The ordered field $\mathfrak{R}^<$ of the real numbers is, up to isomorphism, the only completely ordered field. Therefore, if $\psi_{\mathfrak{R}^<}$ is the conjunction of the axioms for ordered fields (cf. III.6.5) and the second-order S_{ar}-sentence "Every nonempty set which is bounded above has a supremum," i.e.,

$$\begin{aligned}&\forall X((\exists x Xx \wedge \exists y\forall z(Xz \to z < y))\\ &\quad \to \exists y(\forall z(Xz \to (z < y \vee z \equiv y)) \wedge \forall x(x < y \to \exists z(x < z \wedge Xz)))),\end{aligned}$$

then the following holds for all $S_{\mathrm{ar}}^<$-structures $\mathfrak{A}$:

$$\mathfrak{A} \models \psi_{\mathfrak{R}^<} \quad \text{iff} \quad \mathfrak{A} \cong \mathfrak{R}^<.$$

(5) Let S be arbitrary. Then the L_{II}^S-sentence

$$(+) \qquad \forall x\forall y(x \equiv y \leftrightarrow \forall X(Xx \leftrightarrow Xy))$$

is valid: two things are equal precisely when there is no property which distinguishes them (the *identitas indiscernibilium* of Leibniz). Thus, in the development of L_{II}^S we could have done without the equality symbol, using (+) to express equality.

(6) When setting up the second-order languages we could have introduced, in addition to relation variables, *function variables* which can also be quantified. This procedure would increase convenience, but not the expressive power of the languages. We illustrate this by means of an example (cf. the elimination of function symbols in VIII.1).

Let g be a unary function variable and let φ be the "second-order formula"

$$\forall g(\forall x\forall y(gx \equiv gy \to x \equiv y) \to \forall x\exists y\, x \equiv gy).$$

Then (for the natural extension of the notion of satisfaction) the following holds for every structure $\mathfrak{A}$:

$$\begin{aligned}\mathfrak{A} \models \varphi \quad &\text{iff} \quad \text{every injective function from } A \text{ to } A \text{ is surjective}\\ &\text{iff} \quad A \text{ is finite.}\end{aligned}$$

Considering the graph of a unary function instead of the function itself, we can use a binary relation variable and replace φ by the following formula:

$$\varphi_{\text{fin}} := \forall X((\forall x \exists^{=1} y Xxy \wedge \forall x \forall y \forall z((Xxz \wedge Xyz) \to x \equiv y)) \\ \to \forall y \exists x Xxy).$$

The formulas φ and φ_{fin} have the same models. Therefore,

$$\mathfrak{A} \models \varphi_{\text{fin}} \quad \text{iff} \quad A \text{ is finite.}$$

In later examples we shall often use function variables to obtain formulas which are easier to read.

(7) In $\mathcal{L}_{\text{II}}$ one can introduce operations such as substitution and relativization by definitions analogous to those for $\mathcal{L}_{\text{I}}$. One can also verify basic semantic properties such as the analogue of the Isomorphism Lemma.

The situation is different when we consider deeper semantic properties such as the Completeness Theorem, the Compactness Theorem, and the Löwenheim-Skolem Theorem: the price we have to pay for being able to quantify over second-order objects is the loss of all these central properties.

1.4 Theorem. *The Compactness Theorem does not hold for $\mathcal{L}_{\text{II}}$.*

Proof. The following set of sentences is a counterexample:

$$\{\varphi_{\text{fin}}\} \cup \{\varphi_{\geq n} \mid n \geq 2\}.$$

This set is not satisfiable, but, of course, every finite subset is satisfiable. □

1.5 Theorem. *The Löwenheim-Skolem Theorem does not hold for $\mathcal{L}_{\text{II}}$.*

Proof. We give a sentence $\varphi_{\text{unc}} \in L_{\text{II}}^{\emptyset}$ such that for all structures $\mathfrak{A}$,

$$\mathfrak{A} \models \varphi_{\text{unc}} \quad \text{iff} \quad A \text{ is uncountable.}$$

Then φ_{unc} is satisfiable but it has no model that is at most countable.

To define φ_{unc} we use an $L_{\text{II}}^{\emptyset}$-formula $\psi_{\text{fin}}(X)$, similar to φ_{fin}, with just one free unary relation variable X, for which

$$(\mathfrak{A}, \gamma) \models \psi_{\text{fin}}(X) \quad \text{iff} \quad \gamma(X) \text{ is finite.}$$

(We leave it to the reader to write down such a formula.) Clearly, a set A is at most countable if and only if there is an ordering relation on A such that every element has only finitely many predecessors. So let us define, using a binary relation variable Y,

$$\begin{aligned} \varphi_{\leq \text{ctbl}} \quad := \quad & \exists Y(\forall x \neg Yxx \wedge \forall x \forall y \forall z((Yxy \wedge Yyz) \to Yxz) \\ & \wedge \forall x \forall y(Yxy \vee x \equiv y \vee Yyx) \\ & \wedge \forall x \exists X(\psi_{\text{fin}}(X) \wedge \forall y(Xy \leftrightarrow Yyx))). \end{aligned}$$

Then we have

$$\mathfrak{A} \models \varphi_{\leq \text{ctbl}} \quad \text{iff} \quad A \text{ is at most countable}$$

and hence we can set $\varphi_{\text{unc}} := \neg\varphi_{\leq \text{ctbl}}$. □

1.6. For first-order logic we obtained the Compactness Theorem from the existence of an adequate system of derivation rules (cf. VI.2). *For $\mathcal{L}_{\mathrm{II}}$ there is no correct and complete system of derivation rules.* Otherwise we could use the same argument as we did for $\mathcal{L}_{\mathrm{I}}$ to prove the Compactness Theorem for $\mathcal{L}_{\mathrm{II}}$.

This negative result does not, of course, hinder us from setting up correct rules for second-order logic. For example, one can add to the first-order rules the following correct rules for quantification over relation variables:

$$\frac{\Gamma \quad \varphi}{\Gamma \quad \exists X\varphi}\ ; \quad \frac{\Gamma \quad \varphi \qquad \psi}{\Gamma \quad \exists X\varphi \quad \psi}\ , \text{ if } X \text{ is not free in } \Gamma\psi.$$

In the introduction to this chapter we provided two motivations for investigating more expressive languages, namely: (a) to facilitate the formalization of mathematical statements and arguments, and (b) to supply us with more powerful tools for mathematical investigations. In regard to (a) and (b), what have we accomplished by second-order logic?

To begin with, we note that by supplementing the second-order rules presented above, one can obtain a system largely sufficient for the purposes of mathematics. (However, by 1.6, one never gets a complete system, so that the choice of rules can only be made from a pragmatic point of view, and not with the aim of attaining completeness.) In addition, bearing in mind that mathematics can be formulated more conveniently in a second-order language, one can tend to the opinion that progress in the sense of (a) has indeed been made. However, as far as (b) is concerned, $\mathcal{L}_{\mathrm{II}}$ is hardly an appropriate system. The results 1.4 and 1.5 already hint at this. The expressive power of second-order languages is so great that results such as the Compactness Theorem or the Löwenheim-Skolem Theorem, which are of value for mathematical applications, no longer hold. In view of these remarks it is natural to investigate other extensions of first-order logic (cf. Sections 2 and 3).

By considering a further aspect, we explain how, in a certain sense, second-order logic has overshot the mark: We show that set theory, as based, e.g. on ZFC, is not sufficient to decide basic semantic questions for $\mathcal{L}_{\mathrm{II}}$. This follows because we can write down a sentence $\varphi_{\mathrm{CH}} \in L_{\mathrm{II}}^{\emptyset}$ which is valid if and only if Cantor's continuum hypothesis CH holds. Since neither CH nor its negation can be proved in ZFC (cf. VII.3), the validity of φ_{CH} can neither be established nor refuted within the framework of ZFC. CH says:

(1) For every subset A of $\mathbf{R}$, either A is at most countable, or there is a bijection of $\mathbf{R}$ onto A.

φ_{CH} will be essentially a formalization of (1).

First, similar to $\varphi_{\leq \mathrm{ctbl}}$, we can easily give a formula $\chi_{\leq \mathrm{ctbl}}(X)$ with the property

$$(\mathfrak{A}, \gamma) \models \chi_{\leq \mathrm{ctbl}}(X) \quad \text{iff} \quad \gamma(X) \text{ is at most countable.}$$

Further, there is a formula $\varphi_{\mathbf{R}}$ such that

(2) $\qquad \mathfrak{A} \models \varphi_{\mathbf{R}} \quad$ iff $\quad A$ and $\mathbf{R}$ have the same cardinality.

Note that for the $S_{\mathrm{ar}}^{<}$-sentence $\psi_{\mathfrak{R}^<}$ introduced in 1.3(4) and for all $S_{\mathrm{ar}}^{<}$-structures $\mathfrak{A}$,

$$\mathfrak{A} \models \psi_{\mathfrak{R}^<} \quad \text{iff} \quad \mathfrak{A} \cong \mathfrak{R}^<.$$

So, to satisfy (2), we can choose $\varphi_{\mathbf{R}}$ to be an $L_{\mathrm{II}}^{\emptyset}$-sentence which says:

"There are functions $+, \cdot$, elements $0, 1$, and a relation $<$ such that $\varphi_{\mathfrak{R}^<}$."

(We leave it to the reader to write down $\varphi_{\mathbf{R}}$ as a second-order sentence.) Now we can take as φ_{CH} a sentence which says that "if the domain is of the same cardinality as $\mathbf{R}$, then every subset of the domain is either at most countable or else of the same cardinality as the whole domain",

$$\varphi_{\mathrm{CH}} := \varphi_{\mathbf{R}} \to \forall X(\chi_{\leq \mathrm{ctbl}}(X) \vee \exists g(\forall x \forall y(gx \equiv gy \to x \equiv y) \wedge \forall y(Xy \leftrightarrow \exists x\, gx \equiv y))).$$

It is easy to prove (cf. (1)) that

$$\models \varphi_{\mathrm{CH}} \quad \text{iff} \quad \mathrm{CH} \text{ holds.}$$

1.7 Exercise (The System $\mathcal{L}_{\mathrm{II}}^{w}$ of Weak Second-Order Logic). For every S, let $L_{\mathrm{II}}^{w,S} = L_{\mathrm{II}}^{S}$. Change the notion of satisfaction for $\mathcal{L}_{\mathrm{II}}$ by specifying, for $\mathfrak{I} = (\mathfrak{A}, \gamma)$:

$$\mathfrak{I} \models_w \exists X^n \varphi \quad \text{:iff} \quad \text{there is a } \textit{finite } C \subset A^n \text{ such that } \mathfrak{I}\frac{C}{X^n} \models \varphi.$$

Thus, only quantifications over finite sets (and relations) are allowed. Show the following:
(a) There is a sentence φ and a structure $\mathfrak{A}$ such that $\mathfrak{A} \models_w \varphi$ but not $\mathfrak{A} \models \varphi$.
(b) For each sentence $\varphi \in L_{\mathrm{II}}^{w,S}$, there is a sentence $\psi \in L_{\mathrm{II}}^{S}$ such that for all S-structures $\mathfrak{A}$, $\mathfrak{A} \models_w \varphi \quad$ iff $\quad \mathfrak{A} \models \psi$.
(c) The Compactness Theorem does not hold for L_{II}^{w}. (However, the Löwenheim-Skolem Theorem does hold for L_{II}^{w}. This follows from the result 2.4 in the following section; cf. Exercise 2.7.)

§2. The System $\mathcal{L}_{\omega_1\omega}$

In VI.3.5 we showed that the class of torsion groups cannot be characterized in first-order logic. But we can axiomatize this class if we add to the group axioms the "formula"

$$(*) \qquad \forall x(x \equiv e \vee x \circ x \equiv e \vee x \circ x \circ x \equiv e \vee \ldots).$$

Thus we gain expressive power when allowing infinite disjunctions and conjunctions. Such formations are characteristic of the so-called *infinitary languages*. In the simplest case one restricts to conjunctions and disjunctions of countable length. This leads to the system $\mathcal{L}_{\omega_1\omega}$. (The notation $\mathcal{L}_{\omega_1\omega}$ follows the systematic terminology usual in the study of infinitary languages, cf. [3]). To define the formulas of $\mathcal{L}_{\omega_1\omega}$ we use the jargon of calculi. Nevertheless it should be noted that the rule in 2.1(b) below is not a rule of a calculus in the strict sense, since it has infinitely many premises. (For example, in order to obtain the "formula" $(*)$ one must already have obtained the formulas $x \equiv e$, $x \circ x \equiv e$, ...) A precise version of such "calculi" and their usage can be given within the framework of set theory (cf. VII.4.3). For example, the definition of formulas and proofs by induction on formulas can be based on the principle of transfinite induction.

2.1 Definition of $\mathcal{L}_{\omega_1\omega}$. Compared with the first-order language L^S, we add the following to constitute the language $L^S_{\omega_1\omega}$:

(a) the symbol $\bigvee$ (for infinite disjunctions);

(b) to the calculus of formulas the following "rule":
If Φ is an at most countable set of S-formulas, then $\bigvee \Phi$ is an S-formula (the *disjunction* of the formulas in Φ);

(c) to the definition of the notion of satisfaction the following clause:
If Φ is an at most countable set of $L^S_{\omega_1\omega}$-formulas, $\mathfrak{A}$ an S-structure, β an assignment in $\mathfrak{A}$, and $\mathfrak{I} = (\mathfrak{A}, \beta)$, then

$$\mathfrak{I} \models \bigvee \Phi \quad \text{:iff} \quad \mathfrak{I} \models \varphi \text{ for some } \varphi \in \Phi.$$

There are many classes of structures which can be characterized in $\mathcal{L}_{\omega_1\omega}$, but not in first-order logic. Examples are:

the class of torsion groups, characterized by the conjunction of the group axioms and

$$\forall x \bigvee \{\underbrace{x \circ \ldots \circ x}_{n-\text{times}} \equiv e \mid n \geq 1\},$$

the class of fields with characteristic a prime, by the conjunction of the field axioms and

$$\bigvee \{\underbrace{1 + \ldots + 1}_{n-\text{times}} \equiv 0 \mid n \geq 2\},$$

the class of archimedean ordered fields, by the conjunction of the axioms for ordered fields and

$$\forall x \bigvee \{x < \underbrace{1 + \ldots + 1}_{n-\text{times}} \mid n \geq 1\},$$

the class of structures isomorphic to $(\mathbf{N}, \sigma, 0)$, by the conjunction of the first two Peano axioms and

$$\forall x \bigvee\{x \equiv \underbrace{\sigma \ldots \sigma}_{n-\text{times}} 0 \mid n \geq 0\},$$

the class of connected graphs, by the conjunction of the axioms for graphs and

$$\forall x \forall y(\neg x \equiv y \rightarrow \\ \bigvee\{\exists z_0 \ldots \exists z_n(x \equiv z_0 \wedge y \equiv z_n \wedge Rz_0z_1 \wedge \ldots \wedge Rz_{n-1}z_n \mid n \geq 1\}.$$

2.2 Remarks. (a) For a set Φ which is at most countable let $\bigwedge \Phi$ be an abbreviation for the $\mathcal{L}_{\omega_1\omega}$-formula $\neg \bigvee\{\neg\varphi \mid \varphi \in \Phi\}$. Then

$$\mathfrak{I} \models \bigwedge \Phi \quad \text{iff} \quad \text{for all } \varphi \in \Phi, \mathfrak{I} \models \varphi.$$

$\bigwedge \Phi$ is called the *conjunction* of the formulas in Φ.

(b) The definition of the set $\mathrm{SF}(\varphi)$ of subformulas of a formula φ in $\mathcal{L}_{\omega_1\omega}$ is obtained from the corresponding definition for first-order formulas in II.4.5 by adding the clause

$$\mathrm{SF}(\bigvee \Phi) := \{\bigvee \Phi\} \cup \bigcup_{\psi \in \Phi} \mathrm{SF}(\psi).$$

It can be proved for arbitrary φ that $\mathrm{SF}(\varphi)$ is at most countable. The proof is by induction on formulas; we give the $\bigvee$-step: Let $\varphi = \bigvee \Phi$, where by induction hypothesis $\mathrm{SF}(\psi)$ is at most countable for every $\psi \in \Phi$. Since $\mathrm{SF}(\bigvee \Phi) = \{\bigvee \Phi\} \cup \bigcup_{\psi \in \Phi} \mathrm{SF}(\psi)$ is an at most countable union of at most countable sets, $\mathrm{SF}(\varphi)$ is at most countable. In particular, for every $\varphi \in L^S_{\omega_1\omega}$ there exists an at most countable $S' \subset S$ such that $\varphi \in L^{S'}_{\omega_1\omega}$.

(c) Define the set $\mathrm{free}(\bigvee \Phi)$ of variables occurring free in the formula $\bigvee \Phi$ to be $\bigcup_{\psi \in \Phi} \mathrm{free}(\psi)$. The formula $\bigvee\{v_n \equiv v_n \mid n \in \mathbf{N}\}$ has infinitely many free variables. However, one can easily prove by induction that in case $\mathrm{free}(\varphi)$ is finite, $\mathrm{free}(\psi)$ is also finite for any subformula ψ of φ. In particular, subformulas of $\mathcal{L}_{\omega_1\omega}$-sentences have only finitely many free variables.

Consider the $L^{\emptyset}_{\omega_1\omega}$-sentence

$$\psi_{\text{fin}} := \bigvee\{\neg\varphi_{\geq n} \mid n \geq 2\}.$$

Then for every structure $\mathfrak{A}$ we have

$$\mathfrak{A} \models \psi_{\text{fin}} \quad \text{iff} \quad A \text{ is finite.}$$

Hence the set of sentences $\{\psi_{\text{fin}}\} \cup \{\varphi_{\geq n} \mid n \geq 2\}$ is an example showing

2.3 Theorem. *The Compactness Theorem does not hold for $\mathcal{L}_{\omega_1\omega}$.* □

Nevertheless, many results for $\mathcal{L}_{\mathrm{I}}$ have their counterparts in $\mathcal{L}_{\omega_1\omega}$. We mention some examples and refer the reader to [22] for more information.

(1) The analogue of the Löwenheim-Skolem Theorem holds (see 2.4).

(2) Extend the sequent calculus $\mathfrak{S}$ for first-order logic by the following "rules" for $\bigvee$:

$$(\bigvee\mathrm{A})\ \frac{\Gamma\quad\varphi\qquad\psi\quad\text{for every }\varphi\in\Phi}{\Gamma\quad\bigvee\Phi\quad\psi}\ ;$$

$$(\bigvee\mathrm{S})\ \frac{\Gamma\quad\varphi}{\Gamma\quad\bigvee\Phi}\ ,\quad\text{if }\varphi\in\Phi.$$

Here Γ stands for a finite sequence of $\mathcal{L}_{\omega_1\omega}$-formulas.

In this way one obtains a correct and complete "calculus": for $\mathcal{L}_{\omega_1\omega}$-sentences $\varphi_1,\ldots,\varphi_n,\varphi$, the sequent $\varphi_1\ldots\varphi_n\varphi$ is derivable if and only if it is correct. However, one must allow infinitely long derivations as is obvious from ($\bigvee$A).

(3) An analysis of (2) shows that by suitably generalizing the concept of finiteness one can transfer other results from $\mathcal{L}_{\mathrm{I}}$ to $\mathcal{L}_{\omega_1\omega}$. Among these is the *Barwise Compactness Theorem* for $\mathcal{L}_{\omega_1\omega}$, cf. [2].

2.4 Löwenheim-Skolem Theorem for $\mathcal{L}_{\omega_1\omega}$. *Every satisfiable $\mathcal{L}_{\omega_1\omega}$-sentence has a model over an at most countable domain.*

Since for every $\mathcal{L}_{\omega_1\omega}$-sentence φ there is an at most countable S such that $\varphi\in L^S_{\omega_1\omega}$, 2.4 follows directly from

2.5 Lemma. *Let S be at most countable, φ an $L^S_{\omega_1\omega}$-sentence and $\mathfrak{B}$ an S-structure such that $\mathfrak{B}\models\varphi$. Then there is an at most countable substructure $\mathfrak{A}\subset\mathfrak{B}$ such that $\mathfrak{A}\models\varphi$.*

Proof. We first present the idea of the proof. Let B_0 be a nonempty at most countable subset of B which is S-closed, that is, which contains all $c^{\mathfrak{B}}$ for $c\in S$ and which is closed under application of $f^{\mathfrak{B}}$ for $f\in S$. Then B_0 is the domain of an at most countable substructure $\mathfrak{B}_0$ of $\mathfrak{B}$. If one tries to prove by induction that $\mathfrak{B}_0\models\varphi$, the proof breaks down at the point where $\exists$-quantifiers are considered. For example, in a simple case where φ is of the form $\exists xPx$, one must ensure that there is a $b\in B_0$ such that $P^{\mathfrak{B}}b$. Therefore we shall close B_0 with respect to all possible existential requirements arising from subformulas of φ.

We now begin the proof. For pairwise distinct variables $x_1,\ldots,x_n$ we write $\psi(x_1,\ldots,x_n)$ to denote a formula ψ with $\mathrm{free}(\psi)\subset\{x_1,\ldots,x_n\}$. Furthermore, $\mathfrak{D}\models\psi[a_1,\ldots,a_n]$ says that ψ holds in $\mathfrak{D}$ if the variables x_i get the assignment a_i for $1\le i\le n$.

We define a sequence $A_0,A_1,A_2,\ldots$ of at most countable subsets of B so that for $m\in\mathbf{N}$

(a) $A_m\subset A_{m+1}$;

(b) for $\psi(x_1,\ldots,x_n,x)\in\mathrm{SF}(\varphi)$ or $\psi=fx_1\ldots x_n\equiv x$ (with n-ary $f\in S$) and $a_1,\ldots,a_n\in A_m$,
if $\mathfrak{B}\models\exists x\psi[a_1,\ldots,a_n]$, then there is an $a\in A_{m+1}$ such that $\mathfrak{B}\models\psi[a_1,\ldots,a_n,a]$.

Let A_0 be a nonempty at most countable subset of B which contains $\{c^{\mathfrak{B}} \mid c \in S\}$. Suppose A_m is already defined and is at most countable. In order to define A_{m+1}, for every formula $\psi(x_1, \ldots, x_n, x)$ which belongs to $\mathrm{SF}(\varphi)$ or which has the form $fx_1 \ldots x_n \equiv x$ (with n-ary $f \in S$), and for $a_1, \ldots, a_n \in A_m$ with $\mathfrak{B} \models \exists x \psi[a_1, \ldots, a_n]$ we choose a $b \in B$ such that $\mathfrak{B} \models \psi[a_1, \ldots, a_n, b]$. (In case $\varphi = fx_1 \ldots x_n \equiv x$, b will be $f^{\mathfrak{B}}(a_1, \ldots, a_n)$.) Let A'_m be the set of b's chosen in this way. Since $\mathrm{SF}(\varphi)$ and A_m are at most countable, so is A'_m. We set $A_{m+1} := A_m \cup A'_m$. Then A_{m+1} is at most countable, and (a) and (b) are satisfied.

For

$$A := \bigcup_{m \in \mathbf{N}} A_m$$

we have:

(1) A is at most countable.

(2) A is S-closed. By choice of A_0, we need only show that A is closed under $f^{\mathfrak{B}}$ for n-ary $f \in S$. Let $a_1, \ldots, a_n \in A$. Since the sets A_m form an ascending chain, $a_1, \ldots, a_n$ lie in some A_k. As $\mathfrak{B} \models \exists x\ fx_1 \ldots x_n \equiv x$ $[a_1, \ldots, a_n]$, by (b) the element $f^{\mathfrak{A}}(a_1, \ldots, a_n)$ lies in A_{k+1}, hence in A.

By (1) and (2), A is the domain of an at most countable substructure $\mathfrak{A}$ of $\mathfrak{B}$. Therefore we are done if we can show:

$$(*) \qquad \mathfrak{A} \models \varphi.$$

This follows immediately from the following claim:

$$(**) \quad \begin{array}{l} \text{For all } \psi(x_1, \ldots, x_n) \in \mathrm{SF}(\varphi) \text{ and all } a_1, \ldots, a_n \in A, \\ \qquad \mathfrak{A} \models \psi[a_1, \ldots, a_n] \quad \text{iff} \quad \mathfrak{B} \models \psi[a_1, \ldots, a_n]. \end{array}$$

We prove $(**)$ by induction on ψ, but limit ourselves to the $\exists$-case.

Let $\psi(x_1, \ldots, x_n) = \exists x \chi(x_1, \ldots, x_n, x)$, and suppose $a_1, \ldots, a_n \in A$. If $\mathfrak{A} \models \exists x \chi[a_1, \ldots, a_n]$ then we obtain successively:

There is an $a \in A$ such that $\mathfrak{A} \models \chi[a_1, \ldots, a_n, a]$.
There is an $a \in A$ such that $\mathfrak{B} \models \chi[a_1, \ldots, a_n, a]$
(induction hypothesis).
$\mathfrak{B} \models \exists x \chi[a_1, \ldots, a_n]$.

Conversely, if $\mathfrak{B} \models \exists x \chi[a_1, \ldots, a_n]$, we choose k such that $a_1, \ldots, a_n \in A_k$, and we obtain successively:

There is an $a \in A_{k+1}$ such that $\mathfrak{B} \models \chi[a_1, \ldots, a_n, a]$ (by (b)).
There is an $a \in A_{k+1}$ such that $\mathfrak{A} \models \chi[a_1, \ldots, a_n, a]$
(induction hypothesis).
$\mathfrak{A} \models \exists x \chi[a_1, \ldots, a_n]$. □

Consider an at most countable set Φ of first-order sentences and let $\varphi := \bigwedge \Phi$. Then it follows from 2.5 that every model of Φ has an at most countable substructure which is also a model of Φ. In particular, this yields a

proof of the Löwenheim-Skolem Theorem for first-order logic which does not rely on the proof of the Completeness Theorem. Note that an $\mathcal{L}_{\omega_1\omega}$-sentence characterizing $(\mathbf{N}, \sigma, 0)$ has no uncountable model; hence in $\mathcal{L}_{\omega_1\omega}$ we do not have an analogue of the upward Löwenheim-Skolem Theorem VI.2.3.

To conclude this section, we give a mathematical application of 2.5 by choosing φ appropriately.

We consider groups as S_{grp}-structures with $S_{\text{grp}} := \{\circ, e, ^{-1}\}$. A group $\mathfrak{G}$ is said to be *simple* if $\{e^G\}$ and G are the only normal subgroups of $\mathfrak{G}$. If for $a \in G$ we denote by $\langle a \rangle_{\mathfrak{G}}$ the normal subgroup of $\mathfrak{G}$ generated by a, then clearly

$$\mathfrak{G} \text{ is simple} \quad \text{iff} \quad \langle a \rangle_{\mathfrak{G}} = \mathfrak{G} \text{ for all } a \in G \text{ with } a \neq e^G.$$

Since

$$\langle a \rangle_{\mathfrak{G}} = \{g_0 a^{z_0} g_0^{-1} \dots g_n a^{z_n} g_n^{-1} \mid n \in \mathbf{N}, z_0, \dots, z_n \in \mathbf{Z}, g_0, \dots, g_n \in G\},$$

the class of simple groups can be axiomatized in $L_{\omega_1\omega}^{S_{\text{grp}}}$ by the conjunction φ_s of the group axioms and the following sentence:

$$\forall x(\neg x \equiv e \to \forall y \bigvee\{\exists u_0 \dots \exists u_n \bigvee\{y \equiv u_0 x^{z_0} u_0^{-1} \dots u_n x^{z_n} u_n^{-1} \mid \\ z_0, \dots, z_n \in \mathbf{Z}\} \mid n \in \mathbf{N}\}.$$

Using 2.5 we now show

2.6. *If $\mathfrak{G}$ is a simple group and M a countable subset of G then there is a countable simple subgroup of $\mathfrak{G}$ which contains M.*

Proof. Let $S' := S_{\text{grp}} \cup \{c_a \mid a \in M\}$, where c_a are new constants for $a \in M$. We expand $\mathfrak{G}$ to an S'-structure $\mathfrak{G}'$, interpreting each c_a by the corresponding a, and apply 2.5 to $\mathfrak{G}'$ and φ_s. □

2.7 Exercise. Show that for every $L_{\text{II}}^{w,S}$-sentence φ (cf. Exercise 1.7) there is an $L_{\omega_1\omega}^S$-sentence ψ with the same models (that is, for all S-structures $\mathfrak{A}$, $\mathfrak{A} \models_w \varphi$ iff $\mathfrak{A} \models \psi$). Conclude from this that the Löwenheim-Skolem Theorem holds for $\mathcal{L}_{\text{II}}^w$.

2.8 Exercise. Show that the following classes can be axiomatized by an $\mathcal{L}_{\omega_1\omega}$-sentence:
(a) the class of finitely generated groups;
(b) the class of structures isomorphic to $(\mathbf{Z}, <)$.

2.9 Exercise. (a) For arbitrary S, show that $L_{\omega_1\omega}^S$ is uncountable.
(b) Give an uncountable structure $\mathfrak{B}$ (for a suitable countable symbol set S) such that there is no countable structure $\mathfrak{A}$ satisfying the same $L_{\omega_1\omega}^S$-sentences as $\mathfrak{B}$.

§3. The System $\mathcal{L}_Q$

The system $\mathcal{L}_Q$ is obtained from first-order logic by adding the quantifier Q, where a formula $Qx\varphi$ says "there are uncountably many x satisfying φ".

3.1 Definition of $\mathcal{L}_Q$. Compared with the first-order language L^S, we add the following to constitute the language L^S_Q:

(a) to the alphabet: the symbol Q;

(b) to the calculus of formulas the rule:
If φ is an S-formula then so is $Qx\varphi$;

(c) to the definition of the notion of satisfaction the clause:
If φ is an S-formula and $\mathfrak{I} = (\mathfrak{A}, \beta)$ an S-interpretation then

$$\mathfrak{I} \models Qx\varphi \quad \text{:iff} \quad \{a \in A \mid \mathfrak{I}\tfrac{a}{x} \models \varphi\} \text{ is uncountable.}$$

$\mathcal{L}_Q$ has more expressive power than $\mathcal{L}_{\mathrm{I}}$. For example, the class of at most countable structures can be axiomatized in $\mathcal{L}_Q$ by the sentence $\neg Qx\, x \equiv x$. For $S = \{<\}$ let φ_0 be the conjunction of the axioms for orderings and

$$(Qx\, x \equiv x \wedge \forall x \neg Qy\, y < x).$$

Then φ_0 is an L^S_Q-sentence characterizing the class of uncountable orderings in which every element has at most countably many predecessors. These so-called *ω_1-like orderings* play an important role in investigations of $\mathcal{L}_Q$.

Note that the sentence φ_0, or even the sentence $Qx\, x \equiv x$, has an uncountable, but no at most countable model. Hence, the strict analogue of the Löwenheim-Skolem Theorem (VI.1.1) does not hold for $\mathcal{L}_Q$. However, each satisfiable $\mathcal{L}_Q$-sentence has a model of cardinality $\leq \aleph_1$; cf. Exercise 3.3.

One can set up an adequate sequent calculus for $\mathcal{L}_Q$ by adding the following rules to the sequent calculus $\mathfrak{S}$ for first-order logic. (After each rule an explanatory comment is given, containing the essence of a correctness proof.)

$$\frac{\Gamma \quad Qx\varphi}{\Gamma \quad Qy\varphi\frac{y}{x}}, \quad \text{if } y \text{ is not free in } \varphi$$

(Renaming of bound variables);

$$\frac{}{\neg Qx(x \equiv y \vee x \equiv z)}, \quad \text{if } y \text{ and } z \text{ are distinct from } x$$

("Singletons and pair sets are not uncountable");

$$\frac{\Gamma \quad \forall x(\varphi \to \psi)}{\Gamma \quad Qx\varphi \to Qx\psi}$$

("Sets having uncountable subsets are uncountable");

$$\begin{array}{ll} \Gamma & \neg Qx\exists y\varphi \\ \Gamma & Qy\exists x\varphi \\ \hline \Gamma & \exists x Qy\varphi \end{array}$$

("If the union of at most countably many sets is uncountable then at least one of these sets is uncountable").

One can show (cf. [21]) that this calculus allows us to derive exactly the correct sequents. Furthermore, the Completeness Theorem holds for countable sets Φ of L_Q^S-formulas: $\Phi \models \varphi$ iff $\Phi \vdash \varphi$. As for first-order logic we conclude (cf. VI.2):

3.2 $\mathcal{L}_Q$-Compactness Theorem. *For every countable set Φ of L_Q^S-formulas, Φ is satisfiable if and only if every finite subset of Φ is satisfiable.* □

The following example shows that the Compactness Theorem does not hold for uncountable sets of formulas. Let S be an uncountable set of constants and let

$$\Phi := \{\neg c \equiv d \mid c, d \in S, c \neq d\} \cup \{\neg Qx\, x \equiv x\}.$$

Then every finite subset of Φ is satisfiable, but Φ itself is not.

In Chapter VI we saw that the Compactness Theorem and the Löwenheim-Skolem Theorem are useful for mathematical applications. None of the extensions of $\mathcal{L}_{\mathrm{I}}$ which we have discussed in this chapter satisfies both theorems. The Compactness Theorem fails for $\mathcal{L}_{\omega_1\omega}$, the Löwenheim-Skolem Theorem for $\mathcal{L}_Q$, and both for $\mathcal{L}_{\mathrm{II}}$. Does there exist any logical system at all which has more expressive power than first-order logic and for which both the Compactness Theorem and the Löwenheim-Skolem Theorem hold? We shall give a negative answer to this question in Chapter XIII.

3.3 Exercise. Show that every satisfiable $\mathcal{L}_Q$-sentence has a model over a domain of cardinality at most $\aleph_1$ (where $\aleph_1$ is the smallest uncountable cardinal).
(Hint: Use a method similar to that in the proof of 2.5: for formulas $Qx\varphi$ which hold in $\mathfrak{B}$, add $\aleph_1$ elements satisfying φ.)

3.4 Exercise. Let $\mathcal{L}_Q^{\circ}$ be obtained from $\mathcal{L}_Q$ by changing the notion of satisfaction 3.1.(c) as follows:

$$\mathfrak{I} \models Qx\varphi \quad \text{:iff} \quad \{a \in A \mid \mathfrak{I}\tfrac{a}{x} \models \varphi\} \text{ is infinite.}$$

Show that the Compactness Theorem does not hold for $\mathcal{L}_Q^{\circ}$, but that the Löwenheim-Skolem Theorem does.

X

Limitations of the Formal Method

Only in methodological questions have we thus far referred to the fact that applications of sequent rules consist ultimately of mechanical operations on symbol strings (cf. VII.1). In this chapter we make stronger use of this formal-syntactic aspect also in mathematical considerations. Let us give a first idea, taking as an example the system of axioms $\Phi_{\mathrm{gr}} = \{\varphi_0, \varphi_1, \varphi_2\}$ for group theory . It follows from the Completeness Theorem that for all S_{gr}-sentences φ,

$$\Phi_{\mathrm{gr}} \models \varphi \quad \text{iff} \quad \Phi_{\mathrm{gr}} \vdash \varphi.$$

Thus φ is a theorem of group theory

$$\mathrm{Th}_{\mathrm{gr}} := \{\psi \in L_0^{S_{\mathrm{gr}}} \mid \Phi_{\mathrm{gr}} \models \psi\},$$

if and only if the sequent $\varphi_0\varphi_1\varphi_2\varphi$ is derivable. By systematically applying all the sequent rules one can generate all possible derivations and thus compile a list of the theorems of $\mathrm{Th}_{\mathrm{gr}}$: One adds a sentence $\varphi \in L_0^{S_{\mathrm{gr}}}$ to the list if one arrives at a derivation whose last sequent is $\varphi_0\varphi_1\varphi_2\varphi$.

Hence there is a procedure by which one can, in a "mechanical" way, list all theorems of $\mathrm{Th}_{\mathrm{gr}}$. It should be plausible that one could use a suitably programmed computer to carry out such a procedure. Of course, one would have to be able to increase the capacity of the computer if necessary since the derivations and the sequents and formulas therein can be arbitrarily long. A set such as $\mathrm{Th}_{\mathrm{gr}}$ which can be listed by means of such a procedure is said to be *enumerable*.

Of course, the above enumeration procedure yields many trivialities such as $\forall x(x \equiv x \to x \equiv x)$. Moreover, one does not know how soon it will yield any useful theorem. On the other hand, a group theorist is only interested in specific statements φ which are relevant for his investigations. His aim is to determine for such a φ whether $\varphi \in \mathrm{Th}_{\mathrm{gr}}$ or not. Usually this is accomplished either by a proof of φ or by a counterexample to φ (i.e. a group $\mathfrak{G}$ with $\mathfrak{G} \models \neg\varphi$). Often it is difficult to accomplish either of the above. So it is natural to ask whether there is a procedure which can be applied to arbitrary S_{gr}-sentences and which decides for each of these sentences, in finitely many steps, whether it belongs to $\mathrm{Th}_{\mathrm{gr}}$ or not. In

other words, can one program a computer so that whenever it is given an S_{gr}-sentence φ it "computes" whether φ belongs to $\mathrm{Th}_{\mathrm{gr}}$ or not? If such a procedure exists for a given theory, we call that theory *decidable*.

The present chapter is devoted to questions of this kind. First we discuss the concepts of enumerability and decidability in more detail, in Section 1 from a naive point of view, and in Section 2 on the basis of the precise notion of *register machine*. These topics form part of the so-called *recursion theory* (*theory of computability*). The remaining sections of the chapter then contain applications to first-order and second-order logic.

For further information about recursion theory we refer the reader to [8], [15], [20].

§1. Decidability and Enumerability

A. Procedures, Decidability

It is well known how to decide whether an arbitrary natural number n is prime: If $n = 0$ or $n = 1$, n is not prime. If $n \geq 2$, one tests the numbers $2, \ldots, n-1$ to see whether they divide n. If none of these numbers divides n then n is prime; otherwise it is not.

This procedure operates with strings of symbols. For example, in the case of decimal representation of natural numbers it operates with strings over the alphabet $\{0, \ldots, 9\}$. Our description has not specified it in complete detail (for instance, we have not described how division is to be carried out), but it should be clear that it is possible to fill these gaps in order to ensure that all steps are completely determined. In view of its purpose we call the procedure a *decision procedure for the set of primes*.

Other procedures which are well known include those for

(a) multiplication of two natural numbers,

(b) computing the square root of a natural number,

(c) listing the primes in increasing order.

Common to all of these procedures is the fact that they proceed *step by step*, they *operate on symbol strings* and they *can be carried out by a suitably programmed computer*. A procedure can operate on one or more *inputs* (as in (a) or (b)) or it can be started without any particular input (as in (c)). It can *stop* after finitely many steps and yield an *output* (as in (a) for any input and in (b) for inputs which are squares), or it can run without ever stopping, possibly giving an output from time to time (as in (c)).

Procedures in our sense (effective procedures, processes, algorithms) operate with concrete objects such as symbol strings. Sometimes mathematicians use these notions in a wider sense, speaking, for instance, of the Gram-Schmidt orthogonalization *process* even when referring to "abstract" vector spaces.

Concerning the following definition and the subsequent discussion the reader should bear in mind that the notion of procedure has so far been introduced only in an intuitive way and by examples.

1.1 Definition. Let $\mathcal{A}$ be an alphabet, W a set of words over $\mathcal{A}$, i.e. $W \subset \mathcal{A}^*$, and $\mathfrak{P}$ a procedure.

(a) $\mathfrak{P}$ is a *decision procedure* for W if, for every input $\zeta \in \mathcal{A}^*$, $\mathfrak{P}$ eventually stops, having previously given exactly one output $\eta \in \mathcal{A}^*$, where

$$\eta = \square, \text{ if } \zeta \in W \quad \text{and} \quad \eta \neq \square, \text{ if } \zeta \notin W.$$

(b) W is *decidable* if there is a decision procedure for W.

Thus, when a decision procedure for W is applied to an arbitrary word ζ over $\mathcal{A}$, it yields an answer to the question "$\zeta \in W$?" in finitely many steps. The answer is "yes" if the output is the empty word; it is "no" if the output is a nonempty word.

To formulate the above decision procedure for the set of primes according to Definition 1.1 we set $\mathcal{A} := \{0, \ldots, 9\}$, $W :=$ set of primes (in decimal representation), and we agree that the empty word shall be the output for primes and, say, 1 the output for nonprimes.

Further examples of decidable sets are the set of terms and the set of formulas for a concretely given symbol set. In the case of S_∞ (cf. II.2) for instance, terms and formulas are strings over the alphabet

$$\mathcal{A}_\infty := \{v_0, v_1, \ldots, \neg, \vee, \exists, \equiv,), (\} \cup S_\infty.$$

(The same can be done for other explicitly given symbol sets.) We sketch a decision procedure for the terms.

Let $\zeta \in \mathcal{A}_\infty^*$ be given. First, determine the length $l(\zeta)$ of ζ. If $l(\zeta) = 0$, ζ is not a term. If $l(\zeta) = 1$, ζ is a term if and only if ζ is a variable or a constant. If $l(\zeta) > 1$, ζ is not a term unless it begins with a function symbol. If ζ begins with a function symbol, say $\zeta = f_1^3\zeta'$, then check whether there is a decomposition $\zeta_1\zeta_2\zeta_3$ of ζ', where the ζ_i are terms. ζ is a term if and only if such a decomposition exists. To check whether each ζ_i is a term, use the same procedure as for ζ. Clearly, since the ζ_i are shorter than ζ, an answer will be obtained after finitely many steps.

If one analyzes this procedure or tries to write a computer program for it, a difficulty arises: programs (or descriptions of procedures) are finite and therefore can only refer to finitely many symbols in $\mathcal{A}_\infty$, whereas $\mathcal{A}_\infty$ con-

tains, among other things, the infinite list of symbols $v_0, v_1, v_2, \ldots$. Therefore we introduce the new finite alphabet

$$\mathcal{A}_0 := \{v, \underline{0}, \underline{1}, \ldots, \underline{9}, \overline{0}, \overline{1}, \ldots, \overline{9}, \neg, \vee, \exists, \equiv,), (, R, f, c\}$$

and then represent the symbols in $\mathcal{A}_\infty$ in terms of the symbols of $\mathcal{A}_0$ in the natural way. For example, we represent v_{71} by $v\underline{7}\,\underline{1}$, c_{11} by $c\underline{1}\,\underline{1}$, R^3_{18} by $R\overline{3}\underline{1}\,\underline{8}$ and the S_∞-formula $\exists v_3(R^1_1 v_3 \vee c_{11} \equiv f^1_0 v_1)$ by $\exists v\underline{3}(R\overline{1}\underline{1}v\underline{3} \vee c\underline{1}\,\underline{1} \equiv f\overline{1}\underline{0}v\underline{1})$. With this in mind, we only consider *finite* alphabets in the sequel.

1.2 Exercise. Let $\mathcal{A}$ be an alphabet, and let W, W' be decidable subsets of $\mathcal{A}^*$. Show that $W \cup W'$, $W \cap W'$ and $\mathcal{A}^* \setminus W$ are also decidable.

1.3 Exercise. Describe decision procedures for the following subsets of $\mathcal{A}_0^*$:
(a) the set of strings $x\varphi$ over $\mathcal{A}_0$ such that $x \in \text{free}(\varphi)$,
(b) the set of S_∞-sentences.

B. Enumerability

Consider a computing machine which operates as follows: it successively generates the numbers $n = 0, 1, 2, \ldots$, tests in each case whether n is a prime, and yields n as output if the answer is positive. The machine runs without ever stopping, and it generates a list of all primes, i.e. a list in which every prime eventually appears.

Sets, such as the primes, which can be listed by means of a procedure are said to be *enumerable*:

1.4 Definition. Let $\mathcal{A}$ be an alphabet, $W \subset \mathcal{A}^*$, and $\mathfrak{P}$ a procedure.
(a) $\mathfrak{P}$ is an *enumeration procedure* for W if $\mathfrak{P}$, once having been started, eventually yields as outputs exactly the words in W (in some order, possibly with repetitions).
(b) W is *enumerable* if there is an enumeration procedure for W.

We give some further examples for enumerable sets.

1.5. *If $\mathcal{A}$ is a (finite) alphabet, then $\mathcal{A}^*$ is enumerable.*

Proof. Suppose $\mathcal{A} = \{a_0, \ldots, a_n\}$. We first define the *lexicographic order* on $\mathcal{A}^*$ (with respect to the indexing $a_0, \ldots, a_n$). In this ordering ζ precedes ζ' if either

$l(\zeta) < l(\zeta')$ or
$l(\zeta) = l(\zeta')$ and "ζ precedes ζ' in a dictionary", that is, there are $a_i, a_j \in \mathcal{A}$ with $i < j$, such that for suitable $\xi, \eta, \eta' \in \mathcal{A}^*$, $\zeta = \xi a_i \eta$ and $\zeta' = \xi a_j \eta'$.

For example, if $\mathcal{A} = \{a, b, c, \ldots, x, y, z\}$, then "papa" comes before "papi", but after "zuu". In general, the ordering begins as follows:

$$\square, a_0, \ldots, a_n, a_0a_0, a_0a_1, \ldots, a_0a_n, a_1a_0, \ldots, a_na_n, a_0a_0a_0, \ldots .$$

It is easy to set up a procedure which lists the elements of $\mathcal{A}^*$ in lexicographic order. □

1.6. $\{\varphi \in L_0^{S_\infty} \mid \models \varphi\}$ *is enumerable.*

Proof. By the Completeness Theorem we have to describe a procedure which lists the S_∞-sentences φ with $\vdash \varphi$. We use the same idea as in the procedure for listing $\mathrm{Th}_{\mathrm{gr}}$ at the beginning of this chapter: We systematically generate all possible derivations for the symbol set S_∞. If the last sequent in such a derivation consists of a single sentence φ, we include φ in the list. Note that the derivations can be generated as follows: For $n = 1, 2, 3, \ldots$ one constructs the first n terms and formulas in the lexicographic ordering, and one forms the finitely many derivations of length $\leq n$ which use only these formulas and terms, and which consist of sequents containing at most n members. □

C. The Relationship Between Enumerability and Decidability

We have just seen that the set of "logically true" sentences can be listed by means of an enumeration procedure. Is it possible to go farther than this and *decide* whether an arbitrary given sentence is "logically true"? The enumeration procedure given above does not help to solve this problem. For example, if we want to test a sentence φ for validity we might start the enumeration procedure in 1.6 and wait to see whether φ appears; we obtain a positive decision as soon as φ is added to the list. But as long as φ has not appeared, we cannot say anything about φ, since we do not know whether φ will never appear (because it is not valid) or whether it will appear at a later time. In fact, we shall show (cf. 4.1) that the set of valid S_∞-sentences is not decidable.

On the other hand, if a set is decidable we can conclude that it is enumerable:

1.7 Theorem. *Every decidable set is enumerable.*

Proof. Suppose $W \subset \mathcal{A}^*$ is decidable and $\mathfrak{P}$ is a decision procedure for W. To list W, generate the strings of $\mathcal{A}^*$ in lexicographic order, use $\mathfrak{P}$ to check for each string ζ thus obtained whether it belongs to W or not, and, if the answer is positive, add ζ to the list. □

As an extension of 1.7. we have:

1.8 Theorem. *A subset W of $\mathcal{A}^*$ is decidable if and only if W and the complement $\mathcal{A}^* \setminus W$ are enumerable.*

Proof. Suppose W is decidable. Then $\mathcal{A}^* \setminus W$ is also decidable (one can use a decision procedure for W, merely interchanging the outputs "yes" and

"no"). Thus by 1.7, W and $\mathcal{A}^* \setminus W$ are enumerable. Conversely, suppose W and $\mathcal{A}^* \setminus W$ are enumerable by means of procedures $\mathfrak{P}$ and $\mathfrak{P}'$. We combine $\mathfrak{P}$ and $\mathfrak{P}'$ into a decision procedure for W which operates as follows: Given ζ, $\mathfrak{P}$ and $\mathfrak{P}'$ run simultaneously until ζ is yielded by either $\mathfrak{P}$ or $\mathfrak{P}'$. This will eventually be the case since every symbol string in $\mathcal{A}^*$ is either in W or in $\mathcal{A}^* \setminus W$. If ζ is listed by $\mathfrak{P}$, it belongs to W, otherwise to $\mathcal{A}^* \setminus W$. □

1.9 Exercise. Suppose $U \subset \mathcal{A}^*$ is decidable and $W \subset U$. Show that if W and $U \setminus W$ are enumerable, then W is decidable.

Our definitions of decidability and enumerability were given with respect to a fixed alphabet. However, this reference is not essential:

1.10 Exercise. Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be alphabets such that $\mathcal{A}_1 \subset \mathcal{A}_2$, and suppose $W \subset \mathcal{A}_1^*$. Show that W is decidable (enumerable) with respect to $\mathcal{A}_1$ if and only if it is decidable (enumerable) with respect to $\mathcal{A}_2$.

1.11 Exercise. Show:
(a) The set PIR of polynomials in several unknows with integer coefficients which have an integer root, is enumerable. (Choose, for example, the alphabet $\{x, +, -, 0, \ldots, 9, \underline{0}, \ldots, \underline{9}, \overline{0}, \ldots, \overline{9}\}$ and represent the polynomial $-3x_1 + x_2^3 x_5 + 2$ by $-3x\underline{1} + x\underline{2}\overline{3}x\underline{5} + 2$.)
(b) The set PIR_1 of polynomials in *one* unknown which belong to PIR, is decidable.
See also the remarks before 6.13 for the question of the decidability of PIR (*Hilbert's 10. Problem*).

D. Computable Functions

Let $\mathcal{A}$ and $\mathcal{B}$ be alphabets. A procedure which for each input from $\mathcal{A}^*$ yields a word in $\mathcal{B}^*$ determines a function from $\mathcal{A}^*$ to $\mathcal{B}^*$. A function whose values can be computed in this way by a procedure is said to be *computable*. An example of a computable function is the length function l, which assigns to every $\zeta \in \mathcal{A}^*$ the length of ζ (in decimal notation as a word over $\{0, \ldots, 9\}$).

Whereas our discussion of effective procedures deals mainly with the notions of enumerability and decidability, many presentations of recursion theory start with computability of functions as the key concept. Both approaches are equivalent in the sense that the above notions are definable from each other. The following exercise shows that the notion of computable function can be reduced to both the notion of enumerability and the notion of decidability.

1.12 Exercise. Let $\mathcal{A}$ and $\mathcal{B}$ be alphabets, $\# \notin \mathcal{A} \cup \mathcal{B}$ and $f\colon \mathcal{A}^* \to \mathcal{B}^*$. Show that the following are equivalent:
(i) f is computable. (ii) $\{\zeta \# f(\zeta) \mid \zeta \in \mathcal{A}^*\}$ is enumerable.
(iii) $\{\zeta \# f(\zeta) \mid \zeta \in \mathcal{A}^*\}$ is decidable.

The set $\{\zeta\#f(\zeta) \mid \zeta \in \mathcal{A}^*\}$ can be considered as the graph of f, and hence the equivalences in 1.12 can be formulated as follows: A function is computable iff its graph is enumerable (decidable).

§2. Register Machines

In the foregoing discussion we have used an intuitive notion of procedure which we illustrated by use of examples. The conception we have thus acquired is perhaps sufficient for recognizing in a given case whether a proposed procedure can be accepted as such. But in general, our informal concept does not enable us to prove that a particular set is not decidable. Namely, in this case one must show that *every* possible procedure is not a decision procedure for the set in question. However, such a proof is usually not possible without a precise notion of procedure.

We now introduce such a precise concept, starting from the idea that a procedure should be programmable on a computer. For this purpose we set up a programming language and define procedures in the formal sense to be exactly those procedures which can be programmed in this language. A. Turing was the first to introduce a similar and equivalent concept (cf. [34]).

For the following discussion we fix an alphabet $\mathcal{A} = \{a_0, \ldots, a_r\}$.

The programs are executed by computers with a memory consisting of units $\mathrm{R}_0, \ldots, \mathrm{R}_m$, called *registers*. (In the literature such machines are frequently called *register machines*.) At each stage in a computation every register contains a word from $\mathcal{A}^*$. We assume that we have machines with arbitrarily many registers at our disposal, and that the individual registers can store words of arbitrary length. This idealization agrees with our objective of encompassing all procedures which can be carried out "in principle" by a computer, i.e. disregarding problems of capacity.

A program (over $\mathcal{A} = \{a_0, \ldots, a_r\}$) consists of instructions, where each instruction begins with a natural number L, its *label*. Only instructions of the form (1) through (5) below are permitted.

(1) L LET $\mathrm{R}_i = \mathrm{R}_i + a_j$

for $L, i, j \in \mathbf{N}$ with $j \leq r$ (Add-instruction: "Add the letter a_j at the end of the word in register R_i");

(2) L LET $\mathrm{R}_i = \mathrm{R}_i - a_j$

for $L, i, j \in \mathbf{N}$ with $j \leq r$ (Subtract-instruction: "If the word in register R_i ends with the letter a_j, delete this a_j; otherwise leave the word unchanged");

(3) L IF $\mathrm{R}_i = \square$ THEN L' ELSE L_0 OR ... OR L_r

for $L, i, L', L_0, \ldots, L_r \in \mathbf{N}$ (Jump-instruction: "If register R_i contains the empty word go to instruction labelled L'; if the word in register R_i ends with a_0 (resp. $a_1, \ldots, a_r$) go to instruction labelled L_0 (resp. $L_1, \ldots, L_r$)");

(4) L PRINT

for $L \in \mathbf{N}$ (Print-instruction: "Print as output the word stored in register R_0");

(5) L HALT

for $L \in \mathbf{N}$ (Halt-instruction: "Halt").

2.1 Definition. A *register program* (or simply *program*) is a finite sequence $\alpha_0, \ldots, \alpha_k$ of instructions of the form (1) through (5) with the following properties:

(i) α_i has label i ($i = 0, \ldots, k$).
(ii) Every jump-instruction refers to labels $\leq k$.
(iii) Only the last line α_k is a halt-instruction.

Each program P gives rise to a procedure: Imagine we have a computer which contains all registers occurring in P and which has been programmed with P. At the beginning of a computation all registers with the possible exception of R_0 are empty, i.e. they contain the empty word, whereas R_0 contains a possible input. The computation proceeds stepwise, each step corresponding to the execution of one instruction of the program. Beginning with the first instruction one proceeds line-by-line through the program, jumping only as required by a jump-instruction. Whenever a print-instruction is encountered, the respective content of R_0 is given as an output ("printed out"). The machine stops when the halt-instruction is reached.

Examples of Programs

2.2. Let $\mathcal{A} = \{|\}$. We identify the strings $\square, |, ||, \ldots, |^n, \ldots$ over $\mathcal{A}$ with the natural numbers $0, 1, 2, \ldots, n, \ldots$. The following program P_0 decides whether an input in the register R_0 is an even number or not: P_0 successively deletes strokes $|$ from the string n given as an input in R_0 until the empty string $\square$ is obtained. It ascertains whether n is even or odd and prints out $\square$ or $|$ accordingly and then stops.

0 IF $\mathrm{R}_0 = \square$ THEN 6 ELSE 1
1 LET $\mathrm{R}_0 = \mathrm{R}_0 - |$
2 IF $\mathrm{R}_0 = \square$ THEN 5 ELSE 3
3 LET $\mathrm{R}_0 = \mathrm{R}_0 - |$
4 IF $\mathrm{R}_0 = \square$ THEN 6 ELSE 1
5 LET $\mathrm{R}_0 = \mathrm{R}_0 + |$
6 PRINT
7 HALT

We say that the program P is *started* with a word $\zeta \in \mathcal{A}^*$, if P begins the computation with ζ in R_0 and $\square$ in the remaining registers. If P, started with ζ, eventually reaches the halt-instruction, we write

$$P: \zeta \to \text{halt};$$

otherwise we write

$$P: \zeta \to \infty.$$

For $\zeta, \eta \in \mathcal{A}^*$,

$$P: \zeta \to \eta$$

means that P started with ζ eventually stops, having – in the course of the computation – given exactly one output, namely η. In the above example,

$$P_0: n \to \square, \quad \text{if } n \text{ is even},$$
$$P_0: n \to |, \quad \text{if } n \text{ is odd}.$$

2.3. Let $\mathcal{A} = \{a_0, \ldots, a_r\}$. For the program P:

```
0  PRINT
1  LET R0 = R0 + a0
2  IF R0 = □ THEN 0 ELSE 0 OR ... OR 0
3  HALT
```

we have $P: \zeta \to \infty$ for all ζ. If P is started with a word ζ, P prints out successively the words $\zeta, \zeta a_0, \zeta a_0 a_0, \ldots$.

Line 2 of P has the form

L IF $R_0 = \square$ THEN L' ELSE L' OR ... OR L'.

In every case such an instruction results in a jump to line L'. For the sake of simplicity we shall in the sequel abbreviate it by

L GOTO L'.

2.4. We present a program P for the alphabet $\mathcal{A} = \{a_0, a_1\}$, such that $P: \zeta \to \zeta\zeta$ for $\zeta \in \mathcal{A}^*$. Started with ζ in R_0, instructions 0 - 8 serve to build up ζ in reverse order in the registers R_1 and R_2, after which $\zeta\zeta$ is built up in R_0, taking the first copy from R_1 (instructions 9 - 15) and the second copy from R_2 (instructions 16 - 22).

```
0 IF R0 = □ THEN 9 ELSE 1 OR 5
1 LET R0 = R0 − a0
2 LET R1 = R1 + a0
3 LET R2 = R2 + a0
4 GOTO 0
5 LET R0 = R0 − a1
6 LET R1 = R1 + a1
7 LET R2 = R2 + a1
```

```
 8 GOTO 0
 9 IF R1 = □ THEN 16 ELSE 10 OR 13
10 LET R1 = R1 − a0
11 LET R0 = R0 + a0
12 GOTO 9
13 LET R1 = R1 − a1
14 LET R0 = R0 + a1
15 GOTO 9
16 IF R2 = □ THEN 23 ELSE 17 OR 20
17 LET R2 = R2 − a0
18 LET R0 = R0 + a0
19 GOTO 16
20 LET R2 = R2 − a1
21 LET R0 = R0 + a1
22 GOTO 16
23 PRINT
24 HALT
```

As an exercise the reader should write a program P over the alphabet $\mathcal{A} = \{a_0, a_1, a_2\}$ which accomplishes the following:

$$\mathrm{P}\colon \zeta \to \text{halt}, \quad \text{if } \zeta = a_0 a_0 a_2,$$

$$\mathrm{P}\colon \zeta \to \infty, \quad \text{if } \zeta \neq a_0 a_0 a_2.$$

By analogy with the naive definitions in Section 1, we can introduce the exact notions of register-decidability and register-enumerability.

2.5 Definition. Let $W \subset \mathcal{A}^*$.

(a) A program P *decides* W if for all $\zeta \in \mathcal{A}^*$,

$$\mathrm{P}\colon \zeta \to \square, \quad \text{if } \zeta \in W;$$

$$\mathrm{P}\colon \zeta \to \eta \text{ with } \eta \neq \square, \quad \text{if } \zeta \notin W.$$

(b) W is said to be *register-decidable* (abbreviated: *R-decidable*) if there is a program which decides W.

Example 2.2 shows that the set of even natural numbers is R-decidable.

2.6 Definition. Let $W \subset \mathcal{A}^*$.

(a) A program P *enumerates* W, if P, started with $\square$, prints out exactly the words in W (in any order, possibly with repetitions).

(b) W is said to be *register-enumerable* (abbreviated: *R-enumerable*), if there is a program which enumerates W.

If P enumerates an infinite set, then $\mathrm{P}\colon \square \to \infty$. By 2.3, the set $W = \{\square, a_0, a_0 a_0, \ldots\}$ is R-enumerable. The program 0 HALT enumerates the empty set, as does the program

```
0 LET R1 = R1 + a0
1 GOTO 0
2 HALT
```

For the sake of completeness, we add the following definition of register-computable functions.

2.7 Definition. Let $\mathcal{A}$ and $\mathcal{B}$ be alphabets and $F: \mathcal{A}^* \to \mathcal{B}^*$.

(a) A program P over $\mathcal{A} \cup \mathcal{B}$ *computes* F if for all $\zeta \in \mathcal{A}^*$,

$$\mathrm{P}: \zeta \to F(\zeta).$$

(b) F is said to be *register-computable* (abbreviated: *R-computable*) if there is a program over $\mathcal{A} \cup \mathcal{B}$ which computes F.

In this terminology, program P from example 2.4 computes the function

$$F: \{a_0, a_1\}^* \to \{a_0, a_1\}^* \text{ with } F(\zeta) = \zeta\zeta.$$

Definitions 2.5 through 2.7 can easily be extended to n-ary relations and functions. For example, in order to use a program to compute a binary function, one enters the two arguments in the first two registers.

Since any program describes a procedure it is clear that every R-decidable set is decidable, every R-enumerable set is enumerable, and every R-computable function is computable. Does the converse also hold? In other words, can every procedure in the intuitive sense be simulated by means of a program? A mathematical treatment of this problem is not possible because the concept of procedure is an intuitive one, without an exact definition. Nevertheless, in spite of the simple form of the instructions allowed in register programs, it is widely accepted today that all procedures can indeed be simulated by register programs, and, consequently, that the intuitive concepts of decidability and enumerability coincide with their mathematically precise R-analogues. This view was first expressed by A. Church in 1935 (referring to a different but equivalent precise notion of decidability and enumerability). Therefore, the claim that every procedure can be simulated by a program and, hence, that the concepts of enumerability and decidability coincide with their precise counterparts, is called *Church's Thesis* (sometimes also *Church-Turing-Thesis*, as Turing independently stated a similar claim in [34]). We mention two arguments which support this thesis.

Argument 1: *Experience.* Hitherto it has always been possible to simulate any given procedure by a register program. In particular, programs in programming languages such as FORTRAN or PASCAL etc. can be rewritten as register programs.

Argument 2: Since 1930 numerous mathematical concepts have been proposed as precise counterparts to the notion of procedure. Although developed from different starting points, all these definitions have turned out to be equivalent.

In the literature R-decidable sets and R-computable functions are often called *recursive*, and R-enumerable sets are called *recursively enumerable.*

Proofs of R-enumerability or R-decidability often require a considerable amount of programming work. To avoid getting lost in details, rather than actually writing down register programs, we shall usually content ourselves with describing procedures intuitively. The following example should help to illustrate this.

2.8. *The set of valid S_∞-sentences is R-enumerable.*

We accept the procedure described in 1.6 as a proof. □

In the following exercises the critical reader is invited to practice writing programs for given procedures. The more trusting reader may instead draw upon the experience of others and rely on Church's Thesis.

2.9 Exercise. Suppose $W, W' \subset \mathcal{A}^*$. Show that if W and W' are R-decidable, then so are $\mathcal{A}^* \setminus W$, $W \cap W'$, and $W \cup W'$.

2.10 Exercise. Show:
(a) $\mathcal{A}^*$ is R-enumerable.
(b) If $W \subset \mathcal{A}^*$, then W is R-decidable if and only if W and $\mathcal{A}^* \setminus W$ are R-enumerable.

2.11 Exercise. Suppose $W \subset \mathcal{A}^*$. Show that (a) and (b) are equivalent.
(a) W is R-enumerable.
(b) There is a program P such that $\mathrm{P}\colon \zeta \to \square$ if $\zeta \in W$, and $\mathrm{P}\colon \zeta \to \infty$ if $\zeta \notin W$.

2.12 Exercise. A set $W \subset \mathcal{A}^*$ is called lexicographically R-enumerable if there is a program which enumerates W in lexicographic order. Show that W is R-decidable if and only if W is lexicographically R-enumerable.

2.13 Exercise. Restrict the jump-instruction for register programs to the form

(3′) L IF $\mathrm{R}_i = \square$ THEN L' ELSE L''

(If R_i contains the empty word go to instruction labelled L'; otherwise go to instruction labelled L'').

Show that there is no register program P over $\{a_0, a_1\}$ of this new kind such that $\mathrm{P}\colon \zeta \to \zeta\zeta$ for all $\zeta \in \{a_0, a_1\}^*$.

§3. The Halting Problem for Register Machines

Again we fix an alphabet $\mathcal{A} = \{a_0, \ldots, a_r\}$. Our aim is to present a subset of $\mathcal{A}^*$ which is not R-decidable. The set will consist of register programs (over $\mathcal{A}$) which are suitably coded as words over $\mathcal{A}$.

For this purpose we associate with every program P (over $\mathcal{A}$) a word $\xi_{\mathrm{P}} \in \mathcal{A}^*$. First we extend $\mathcal{A}$ by new symbols to an alphabet $\mathcal{B}$

(+) $\mathcal{B} := \mathcal{A} \cup \{A, B, C, \ldots, X, Y, Z\} \cup \{0, 1, \ldots, 8, 9\} \cup \{=, +, -, \Box, \S\}$,

and we order $\mathcal{B}^*$ lexicographically according to the order of letters given in (+). We represent a program P as a word over $\mathcal{B}$, e.g. the program

0 LET $\mathrm{R}_1 = \mathrm{R}_1 - a_0$
1 PRINT
2 HALT

is represented by the word

0LETR1=R1$-a_0$§1PRINT§2HALT.

If this word is the nth word in the lexicographic ordering on $\mathcal{B}^*$, let $\xi_{\mathrm{P}} := \underbrace{a_0 \ldots a_0}_{n-\text{times}}$. Set $\Pi := \{\xi_{\mathrm{P}} \mid \mathrm{P} \text{ is a program over } \mathcal{A}\}$.

The transition from P to ξ_{P} (i.e. the "numbering" of programs over $\mathcal{A}$ with words in $\{a_0\}^*$) is an example of a *Gödel numbering* (Gödel was the first to apply this method); and ξ_{P} is called the *Gödel number* of P.

Clearly, for each P we can effectively determine the corresponding $\xi_{\mathrm{P}} \in \mathcal{A}^*$; conversely, given $\zeta \in \mathcal{A}^*$, we can decide whether it belongs to Π or not, and if it does we can effectively determine the program P with $\xi_{\mathrm{P}} = \zeta$. The corresponding procedures can be programmed for register machines (cf. the discussion at the end of Section 2). In particular, we have

3.1 Lemma. Π *is R-decidable.* $\Box$

The following theorem presents first examples of R-undecidable sets.

3.2 Theorem (Undecidability of the Halting Problem).

(a) *The set*

$$\Pi'_{\mathrm{halt}} = \{\xi_{\mathrm{P}} \mid \mathrm{P} \textit{ is a program over } \mathcal{A} \textit{ and } \mathrm{P}\colon \xi_{\mathrm{P}} \to \textit{halt}\}$$

is not R-decidable.

(b) *The set*

$$\Pi_{\mathrm{halt}} = \{\xi_{\mathrm{P}} \mid \mathrm{P} \textit{ is a program over } \mathcal{A} \textit{ and } \mathrm{P}\colon \Box \to \textit{halt}\}$$

is not R-decidable.

Part (b) says that there is no register program which decides the set Π_{halt}. Hence, by Church's Thesis there is no procedure whatsoever which decides Π_{halt}. From this we obtain the following formulation of (b):

There is no procedure which decides for an arbitrarily given program P *whether* $\mathrm{P}\colon \Box \to \text{halt}$.

For, if such a procedure $\mathfrak{P}$ did exist, one could use it to decide Π_{halt} as follows. First, for a given ζ, check whether $\zeta \in \Pi$ (cf. 3.1). If $\zeta \notin \Pi$ then $\zeta \notin \Pi_{\text{halt}}$. If $\zeta \in \Pi$, construct the program P for which $\xi_{\mathrm{P}} = \zeta$ and then apply $\mathfrak{P}$ to P.

Proof of 3.2. (a) For a contradiction, suppose that there is a program P_0 deciding Π'_{halt}. Then for all P:

$$(1)\quad \begin{array}{ll} \mathrm{P}_0 : \xi_{\mathrm{P}} \to \Box, & \text{if } \mathrm{P} : \xi_{\mathrm{P}} \to \text{halt}, \\ \mathrm{P}_0 : \xi_{\mathrm{P}} \to \eta \text{ for some } \eta \neq \Box, & \text{if } \mathrm{P} : \xi_{\mathrm{P}} \to \infty. \end{array}$$

From this we easily obtain a program P_1 (see below) such that

$$(2)\quad \begin{array}{ll} \mathrm{P}_1\colon \xi_{\mathrm{P}} \to \infty, & \text{if } \mathrm{P} : \xi_{\mathrm{P}} \to \text{halt}, \\ \mathrm{P}_1\colon \xi_{\mathrm{P}} \to \text{halt}, & \text{if } \mathrm{P}\colon \xi_{\mathrm{P}} \to \infty. \end{array}$$

Then the following holds for all programs P:

$$(3)\quad \mathrm{P}_1\colon \xi_{\mathrm{P}} \to \infty \quad \text{iff} \quad \mathrm{P}\colon \xi_{\mathrm{P}} \to \text{halt}.$$

In particular, if we set $\mathrm{P} = \mathrm{P}_1$, we have

$$(4)\quad \mathrm{P}_1\colon \xi_{\mathrm{P}_1} \to \infty \quad \text{iff} \quad \mathrm{P}_1\colon \xi_{\mathrm{P}_1} \to \text{halt},$$

a contradiction.

To complete the proof we show how to construct P_1 from P_0: We change P_0 in such a way that if P_0 prints the empty word, the new program P_1 will not reach the halt instruction. This is achieved by replacing the last instruction k HALT in P_0 by

$$\begin{array}{rl} k & \text{IF } \mathrm{R}_0 = \Box \text{ THEN } k \text{ ELSE } k+1 \text{ OR} \ldots \text{OR } k+1 \\ k+1 & \text{HALT} \end{array}$$

and all instructions of the form L PRINT by L GOTO k.

(b) To each program P we assign in an effective way a program P^+ such that

$$(*)\quad \begin{array}{lll} \mathrm{P}\colon \xi_{\mathrm{P}} \to \text{halt} & \text{iff} & \mathrm{P}^+\colon \Box \to \text{halt}, \\ \text{i.e. } \xi_{\mathrm{P}} \in \Pi'_{\text{halt}} & \text{iff} & \xi_{\mathrm{P}^+} \in \Pi_{\text{halt}}. \end{array}$$

Using $(*)$ we can prove (b) indirectly as follows: Suppose that Π_{halt} is R-decidable, say by means of the program P_0. Then, in contradiction to (a), we obtain the following decision procedure for Π'_{halt}: For an arbitrarily given $\zeta \in \mathcal{A}^*$ first check whether $\zeta \in \Pi$ (cf. 3.1). If $\zeta \notin \Pi$ then $\zeta \notin \Pi'_{\text{halt}}$. If $\zeta \in \Pi$ take the program P with Gödel number ζ, i.e. with $\xi_{\mathrm{P}} = \zeta$, and construct P^+. Using P_0, decide whether $\xi_{\mathrm{P}^+} \in \Pi_{\text{halt}}$. On account of $(*)$ one

thus obtains an answer to the question whether $\xi_{\mathrm{P}} \in \Pi'_{\mathrm{halt}}$, i.e. whether $\zeta \in \Pi'_{\mathrm{halt}}$.

It remains to define a program P^+ satisfying $(*)$. If $\xi_{\mathrm{P}} = \underbrace{a_0 \ldots a_0}_{n-\text{times}}$, let P^+ be the program which begins with the lines

$$\begin{array}{rl} 0 & \text{LET } \mathrm{R}_0 = \mathrm{R}_0 + a_0 \\ \vdots & \\ n-1 & \text{LET } \mathrm{R}_0 = \mathrm{R}_0 + a_0 \end{array}$$

followed by the lines of P with all labels increased by n. When P^+ is started with $\square$ as input, it first builds up the word ξ_{P} in R_0 and then operates in the same way as the program P applied to ξ_{P}. Hence $(*)$ holds. Since we can build the word ξ_{P} from P in an effective way, we can also build P^+ from P in an effective way. $\square$

The reader should note that the only properties of the map $\mathrm{P} \mapsto \xi_{\mathrm{P}}$ used in the proof were its injectivity and properties of effectiveness, as mentioned before 3.1. Therefore the undecidability of the halting problem does not depend on our particular choice of Gödel numbering.

Of course, for particular programs P it may be easy to determine whether $\mathrm{P}: \square \to$ halt or not. But Theorem 3.2 tells us that there cannot exist a procedure which decides this question "uniformly" for each P. (Strictly speaking, 3.2 only refers to procedures which can be simulated by register programs. However, we obtain our preceding formulation if we accept Church's thesis. Henceforth we shall tacitly do this in explanatory remarks.)

The following lemma together with 3.2 shows that Π_{halt} is an example of an enumerable set which is not decidable.

3.3 Lemma. Π_{halt} *is R-enumerable.*

Proof. We sketch an enumeration procedure: for $n = 1, 2, 3, \ldots$ generate the finitely many programs whose Gödel numbers are of length $\leq n$. Start each such program with $\square$ as input, and let each one perform n steps of its computation. To compile the desired list, note each program which stops. $\square$

Applying 1.8 (cf. Exercise 2.10), we obtain

3.4 Corollary. $\mathcal{A}^* \setminus \Pi_{\mathrm{halt}}$ *is not R-enumerable.* $\square$

Before discussing questions about decidability in first-order and second-order logic in the next section, we briefly consider the aspect of *costs* of computations, which we will not get into otherwise.

Propositional formulas are built up from propositional variables $p_0, p_1, \ldots$ using $\neg$ and $\vee$ (in the same way as quantifier-free first-order formulas are built up from atomic formulas). A propositional formula is said to be satisfiable if one can assign the truth values T, F to the occurring propositional

variables in such a way that the truth value T is assigned to the whole formula if we interpret $\neg$ and $\vee$ in the usual way. (We will give a precise definition of propositional logic in XI.4.)

The set SAT of satisfiable propositional formulas α is decidable: Suppose the propositional variables occurring in α are among $p_0, \ldots, p_n$. Then check systematically for all $b_0, \ldots, b_n \in \{T, F\}$ whether α is assigned the truth value T if we assign b_i to p_i (for $i \leq n$). If α contains, say, 1000 propositional variables then, in the worst case, we have to check this for 2^{1000} tuples. Not even with the fastest existent computers is this possible within a human life time. Therefore, even for "relatively short" inputs it may be impossible to carry out a decision procedure. Thus it is conceivable that a set is "theoretically", but not "practically" decidable, since all decision procedures are too costly, e.g., they need too many computation steps or too much memory (in the registers). Questions of this kind are the subject of *complexity theory* (cf. [10], [20]). We give a first impression by considering the number of computation steps, the so-called *time complexity*.

Let $t\colon \mathbf{N} \to \mathbf{N}$. Then a register program P over $\mathcal{A}$ is said to be *t-bounded in time* if for all $n \in \mathbf{N}$ the following holds: If $\zeta \in \mathcal{A}^*$ is a word of length n, then, started with ζ, P stops after at most $t(n)$ many steps. We say that a program is *polynomially bounded in time* if it is t-bounded in time for a suitable polynomial t (with coefficients in the natural numbers). Let $\mathcal{P}$ be the class of R-decidable sets which can be decided by a program which is polynomially bounded in time.

Experience shows that, as far as problems in practical applications are concerned (or problems that arise naturally in mathematics), the existence of a procedure executable in practice corresponds to the existence of a procedure which is polynomially bounded in time. Therefore, one often identifies the "practically decidable" sets with the sets in $\mathcal{P}$. This "Church's Thesis of practical computability" (also called Thesis of Cobham and Edmonds) can only be justified to a certain extent: note, for example, that no restriction is imposed on the degree of the polynomials.

The set Π of register programs lies in $\mathcal{P}$; on the other hand it is not known whether the set PRIM of (decimal representations of) primes or the set SAT lie in $\mathcal{P}$; it is conjectured that PRIM $\in \mathcal{P}$ and SAT $\notin \mathcal{P}$.

SAT lies in $\mathcal{NP}$, the class of sets accepted by the so-called *non-deterministic* polynomially bounded in time register programs. In non-deterministic programs instructions of the form

$$L \text{ GOTO } \mathfrak{Z}$$

are allowed in addition to the usual instructions. Here $\mathfrak{Z}$ is a nonempty finite set of labels. To follow instructions of this kind the machine chooses "non-deterministically" a label from $\mathfrak{Z}$ and jumps to the corresponding instruction. So, by successively choosing appropriate labels, non-deterministic

machines are able to "guess" words, e.g. a satisfying assignment for a propositional formula. In this way one shows that SAT $\in \mathcal{NP}$. From the exact definitions it follows immediately that $\mathcal{P} \subset \mathcal{NP}$. Furthermore, it can be shown that SAT $\notin \mathcal{P}$ if and only if $\mathcal{P} \neq \mathcal{NP}$. If one could show that SAT $\notin \mathcal{P}$, then $\mathcal{P} \neq \mathcal{NP}$ would be proved, and the so-called "$\mathcal{P} = \mathcal{NP}$"-problem, the probably most well-known unsolved problem in theoretical computer science, would be settled in the expected way.

The proof of 3.2(a) is based on a so-called "diagonal argument". The following exercise contains an abstract version of this method of proof.

3.5 Exercise. (a) Suppose M is a nonempty set and $R \subset M \times M$ is a binary relation over M. For $a \in M$ let $M_a := \{b \in M \mid Rab\}$. Show that the set $D := \{b \in M \mid \text{not } Rbb\}$ is different from each M_a.
(b) Let $M = \mathcal{A}^*$ for some given alphabet $\mathcal{A} = \{a_0, \ldots, a_r\}$ and define $R \subset M \times M$ by

$R\xi\eta$:iff ξ is the Gödel number of a program enumerating a set in which η occurs.

Show that $D := \{\eta \mid \text{not } R\eta\eta\}$ is not R-enumerable. Thus the set of programs which do not print their own Gödel number is not enumerable.
(c) Again, let $M = \mathcal{A}^*$ for $\mathcal{A} = \{a_0, \ldots, a_r\}$ and $R \subset M \times M$ be defined by

$R\xi\eta$:iff ξ is not the Gödel number of a program P with P: $\eta \to$ halt.

Show that all R-decidable subsets of $\mathcal{A}^*$ $(= M)$ occur among the sets M_ξ and that $D = \Pi'_{\text{halt}}$. (Here M_ξ and D are defined as in (a).)

§4. The Undecidability of First-Order Logic

The set of valid first-order S_∞-sentences is enumerable (cf. 1.6). On the other hand we now show:

4.1 Theorem on the Undecidability of First-Order Logic. *The set* $\{\varphi \in L_0^{S_\infty} \mid \models \varphi\}$ *of valid* S_∞*-sentences is not R-decidable.*

Thus there is no procedure which decides, for an arbitrary S_∞-sentence, whether it is valid or not.

Proof. We adopt the notation of Section 3 with $\mathcal{A} = \{|\}$. Again we identify words over $\mathcal{A}$ with natural numbers. By 3.2 we know that the set

$$\Pi_{\text{halt}} = \{\xi_{\text{P}} \mid \text{P is a program over } \mathcal{A} \text{ and P}: \square \to \text{ halt}\}$$

is not R-decidable. We shall assign to every program P, in an effective way, an S_∞-sentence φ_{P} such that

$$(*) \qquad \models \varphi_{\text{P}} \quad \text{iff} \quad \text{P}: \square \to \text{ halt}.$$

Then we are done: If the set $\{\varphi \in L_0^{S_\infty} \mid \models \varphi\}$ were decidable, we would have the following decision procedure for Π_{halt} (a contradiction): Given $\zeta \in \mathcal{A}^*$, first check whether ζ is of the form ξ_{P}. If so, take P, construct φ_{P} and decide whether φ_{P} is valid. By $(*)$ we obtain an answer to the question whether $\text{P}\colon \Box \to$ halt, i.e. whether $\xi_{\text{P}} \in \Pi_{\text{halt}}$.

The following considerations are preparatory to the definition of the sentences φ_{P}. Let P be a program with instructions $\alpha_0, \ldots, \alpha_k$. Denote by n the smallest number such that the registers occurring in P are among $\text{R}_0, \ldots, \text{R}_n$. An $(n+2)$-tuple $(L, m_0, \ldots, m_n)$ of natural numbers with $L \leq k$ is called a *configuration* of P. We say that $(L, m_0, \ldots, m_n)$ is the *configuration of* P *after* s *steps* if P, started with $\Box$, runs for at least s steps, and if after s steps instruction L is to be executed next, while the numbers $m_0, \ldots, m_n$ are in $R_0, \ldots, R_n$, respectively. In particular, $(0, 0, \ldots, 0)$ is the configuration of P after 0 steps (the *initial configuration of* P). Since only α_k is a halt-instruction we have

(1) $\text{P}\colon \Box \to \text{halt}$ iff for suitable $s, m_0, \ldots, m_n$, $(k, m_0, \ldots, m_n)$ is the configuration of P after s steps.

If $\text{P}\colon \Box \to$ halt, we let s_{P} be the number of steps carried out by P until it arrives at the halt-instruction.

Finally we choose symbols R ($(n+3)$-ary), $<$ (binary), f (unary), and $c \in S_\infty$ (e.g. R_0^{n+3}, R_0^2, f_0^1 and c_0), and set $S := \{R, <, f, c\}$. With the program P we associate an S-structure $\mathfrak{A}_{\text{P}}$ within which we shall describe how P operates. We distinguish two cases:

Case 1: $\text{P}\colon \Box \to \infty$. We set $A_{\text{P}} := \mathbf{N}$ and interpret $<$ by the usual ordering on $\mathbf{N}$, c by 0, f by the successor function, and R by $\{(s, L, m_0, \ldots, m_n) \mid (L, m_0, \ldots, m_n)$ is the configuration of P after s steps$\}$.

Case 2: $\text{P}\colon \Box \to$ halt. We set $e := \max\{k, s_{\text{P}}\}$ and $A_{\text{P}} := \{0, \ldots, e\}$ and interpret $<$ by the usual ordering on A_{P} and c by 0; furthermore we define $f^{A_{\text{P}}}$ by $f^{A_{\text{P}}}(m) = m + 1$ for $m < e$ and $f^{A_{\text{P}}}(e) = e$, and set $R^{A_{\text{P}}} := \{(s, L, m_0, \ldots, m_n) \mid (L, m_0, \ldots, m_n)$ is the configuration of P after s steps$\}$. Note that $R^{A_{\text{P}}}$ is indeed a relation on A_{P}, since at each step P increases the contents of each register by at most 1, and hence we have $m_0, \ldots, m_n \leq s_{\text{P}} \leq e$ as well as $L \leq k \leq e$ for all $(s, L, m_0, \ldots, m_n) \in R^{A_{\text{P}}}$.

Now we provide an S-sentence ψ_{P} which, in a suitable way, describes the operations of P on $\Box$. We abbreviate $c, fc, ffc, \ldots$ by $\overline{0}, \overline{1}, \overline{2} \ldots$, respectively. In reading ψ_{P} one should check that the following holds:

(2) (a) $\mathfrak{A}_{\text{P}} \models \psi_{\text{P}}$.
(b) If $\mathfrak{A}$ is an S-structure with $\mathfrak{A} \models \psi_{\text{P}}$ and $(L, m_0, \ldots, m_n)$ is the configuration of P after s steps, then the elements $\overline{0}^{\mathfrak{A}}, \overline{1}^{\mathfrak{A}}, \ldots, \overline{s}^{\mathfrak{A}}$ are pairwise distinct and $\mathfrak{A} \models R\overline{s}\overline{L}\overline{m_0}, \ldots, \overline{m_n}$ [1].

[1] We shall need the fact that $\overline{0}^{\mathfrak{A}}, \ldots, \overline{s}^{\mathfrak{A}}$ are distinct only in the next section.

We set

$$\psi_{\mathrm{P}} := \psi_0 \wedge R\overline{0}\ldots\overline{0} \wedge \psi_{\alpha_0} \wedge \ldots \wedge \psi_{\alpha_{k-1}}.$$

Here the sentence ψ_0 says that $<$ is an ordering whose first element is c, that $x \leq fx$ holds for every x and that fx is the immediate successor of x in case x is not the last element:

$$\begin{aligned}\psi_0 := \text{“}<\text{ is an ordering”} &\wedge \forall x(c < x \vee c \equiv x) \wedge \forall x(x < fx \vee x \equiv fx)\\ &\wedge \forall x(\exists y\, x < y \rightarrow (x < fx \wedge \forall z(x < z \rightarrow (fx < z \vee fx \equiv z)))).\end{aligned}$$

For $\alpha = \alpha_0, \ldots, \alpha_{k-1}$, the sentence ψ_α describes the operation corresponding to instruction α. ψ_α is defined as follows:

If α is an add-instruction, say L LET $\mathrm{R}_i = \mathrm{R}_i+\,|$, then let

$$\begin{aligned}\psi_\alpha := \forall x \forall y_0 \ldots \forall y_n(Rx\overline{L}y_0 \ldots y_n \rightarrow (x < fx \wedge\\ Rfx\overline{L+1}y_0 \ldots y_{i-1} f y_i y_{i+1} \ldots y_n)).\end{aligned}$$

If α is the instruction L LET $\mathrm{R}_i = \mathrm{R}_i-\,|$, then let

$$\begin{aligned}\psi_\alpha := \forall x \forall y_0 \ldots \forall y_n(Rx\overline{L}y_0 \ldots y_n \rightarrow (x < fx \wedge\\ ((y_i \equiv \overline{0} \wedge Rfx\overline{L+1}y_0 \ldots y_n) \vee (\neg y_i \equiv \overline{0} \wedge\\ \exists u(fu \equiv y_i \wedge Rfx\overline{L+1}y_0 \ldots y_{i-1} u y_{i+1} \ldots y_n))))).\end{aligned}$$

If α is the instruction L IF $\mathrm{R}_i = \square$ THEN L' ELSE L_0, then let

$$\begin{aligned}\psi_\alpha := \forall x \forall y_0 \ldots \forall y_n(Rx\overline{L}y_0 \ldots y_n \rightarrow (x < fx \wedge\\ ((y_i \equiv \overline{0} \wedge Rfx\overline{L'}y_0 \ldots y_n) \vee (\neg y_i \equiv \overline{0} \wedge Rfx\overline{L_0}y_0 \ldots y_n)))).\end{aligned}$$

Finally for $\alpha = L$ PRINT let

$$\psi_\alpha := \forall x \forall y_0 \ldots \forall y_n(Rx\overline{L}y_0 \ldots y_n \rightarrow (x < fx \wedge Rfx\overline{L+1}y_0 \ldots y_n)).$$

Now we set

$$\varphi_{\mathrm{P}} := \psi_{\mathrm{P}} \rightarrow \exists x \exists y_0 \ldots \exists y_n Rx\overline{k}y_0 \ldots y_n. \tag{3}$$

Then φ_{P} is an S-sentence which satisfies $(*)$, i.e.

$$\models \varphi_{\mathrm{P}} \quad \text{iff} \quad \mathrm{P}\colon \square \rightarrow \text{ halt}.$$

Indeed, suppose first that φ_{P} is valid. Then in particular $\mathfrak{A}_{\mathrm{P}} \models \varphi_{\mathrm{P}}$. Since $\mathfrak{A}_{\mathrm{P}} \models \psi_{\mathrm{P}}$ (cf. (2)(a)), we have $\mathfrak{A}_{\mathrm{P}} \models \exists x \exists y_0 \ldots \exists y_n Rx\overline{k}y_0 \ldots y_n$ (cf. (3)), i.e. for suitable $s, m_0, \ldots, m_n$, the tuple $(k, m_0, \ldots, m_n)$ is the configuration of P after s steps. Now, (1) yields $\mathrm{P}\colon \square \rightarrow$ halt.

Conversely, if $\mathrm{P}\colon \square \rightarrow$ halt, then for suitable $s, m_0, \ldots, m_n$, the tuple $(k, m_0, \ldots, m_n)$ is the configuration of P after s steps. Hence φ_{P} is valid, because if $\mathfrak{A}$ is an S-structure such that $\mathfrak{A} \models \psi_{\mathrm{P}}$, then $\mathfrak{A} \models R\overline{s}\overline{k}\overline{m_0} \ldots \overline{m_n}$ by (2)(b) and hence $\mathfrak{A} \models \varphi_{\mathrm{P}}$. $\square$

The undecidability of first-order logic was first proved by A. Church in 1936. Thus the so-called *Entscheidungsproblem*, the question on the decidability of valid first-order sentences, was solved negatively. In traditional logic the

problem of finding a decision procedure for "logically true propositions" had already been considered (Llull, Leibniz). Theorem 4.1 shows that such a search was bound to fail.

4.2 Exercise. Prove (2)(b) by induction on s.

4.3 Exercise. Show that the set of satisfiable S_∞-sentences is not R-enumerable.

4.4 Exercise. Show that the set

$\{(\psi, \chi) \mid \psi, \chi \in L_0^{S_\infty}$ do not contain the equality symbol, $\models \psi \to \chi$, ψ is a universal Horn sentence, and χ is of the form $\exists x_1 \ldots \exists x_n \chi_0$ with atomic $\chi_0\}$

is not R-decidable. (Hint: In the proof of 4.1 leave out the ordering $<$ and formalize in such a way that the ψ_{P} become universal Horn sentences.)

§5. Trahtenbrot's Theorem and the Incompleteness of Second-Order Logic

The object of this section is to prove that the set of valid second-order S_∞-sentences is not enumerable, and to briefly discuss the methodological consequences. A useful tool in this context will be Trahtenbrot's Theorem, which says that the set of first-order sentences valid in all finite structures is not enumerable.

5.1 Definition. (a) An S-sentence φ is said to be *fin-satisfiable* if there is a finite S-structure which satisfies φ.

(b) An S-sentence φ is said to be *fin-valid* if every finite S-structure satisfies φ.

For $S = S_\infty$ we set

(a) $\Phi_{\mathrm{fs}} := \{\varphi \in L_0^{S_\infty} \mid \varphi$ is fin-satisfiable$\}$,

(b) $\Phi_{\mathrm{fv}} := \{\varphi \in L_0^{S_\infty} \mid \varphi$ is fin-valid$\}$.

As an example, we note that over a finite domain any injective function is also surjective; therefore the sentence $\varphi := (\forall x \forall y (fx \equiv fy \to x \equiv y) \to \forall x \exists y\, x \equiv fy)$ is fin-valid; however, φ is not valid. The sentence $\neg\varphi$ is satisfiable but not fin-satisfiable.

5.2 Lemma. Φ_{fs} *is R-enumerable.*

Proof. First we describe a procedure which decides, for every S_∞-sentence φ and every n, whether or not φ is satisfiable over a domain with $n+1$

elements. Suppose φ and n are given. Since for every structure with $n+1$ elements there is an isomorphic structure with domain $\{0,\dots,n\}$, we only need to check (by the Isomorphism Lemma) whether φ is satisfiable over $\{0,\dots,n\}$. Let S be the (finite!) set of symbols occurring in φ and $\mathfrak{A}_0,\dots,\mathfrak{A}_k$ the finitely many S-structures with domain $\{0,\dots,n\}$ (cf. III.1.5). We can describe the $\mathfrak{A}_i$ explicitly by means of finite tables for the relations, functions and constants. φ is satisfiable over $\{0,\dots,n\}$ if and only if $\mathfrak{A}_i \models \varphi$ for some $i \leq k$. Thus we only need to test whether $\mathfrak{A}_i \models \varphi$ for $i = 0,\dots,k$. These tests can be reduced to questions which can be answered from the respective tables as follows: If $\varphi = \neg\psi$ then the problem "$\mathfrak{A}_i \models \varphi$?" can be reduced to the question of whether $\mathfrak{A}_i \models \psi$. If $\varphi = (\psi \vee \chi)$, then similarly the problem can be reduced to the questions of whether $\mathfrak{A}_i \models \psi$ and whether $\mathfrak{A}_i \models \chi$. If $\varphi = \exists v_0\psi$, we reduce to the questions "$\mathfrak{A}_i \models \psi[0]$",..., "$\mathfrak{A}_i \models \psi[n]$". Continuing in this way we eventually arrive at questions of the form $\mathfrak{A}_i \models \psi[n_0,\dots,n_{m-1}]$ for atomic formulas $\psi(v_0,\dots,v_{m-1})$ and $n_0,\dots,n_{m-1} \leq n$. Clearly these can be answered effectively by inspecting the tables for $\mathfrak{A}_i$.

Now Φ_{fs} can be enumerated as follows: For $m = 0,1,2,\dots$ generate the (finitely many) words over $\mathcal{A}_0$ which are S_∞-sentences and are of length $\leq m$, and use the procedure just described to decide, for $n = 0,\dots,m$, whether they are satisfiable over a domain with $n+1$ elements. List the sentences where this is the case. □

5.3 Theorem. Φ_{fs} *is not R-decidable.*

Proof. For a program P over $\mathcal{A} = \{|\}$, let $\mathfrak{A}_{\mathrm{P}}$ and ψ_{P} be defined as in the proof of 4.1. We show

$$(*) \qquad \mathrm{P}\colon \square \to \text{ halt} \quad \text{iff} \quad \psi_{\mathrm{P}} \in \Phi_{\mathrm{fs}}.$$

This proves the theorem; for otherwise, using $(*)$, one could obtain from a decision procedure for Φ_{fs} a procedure to decide whether $\mathrm{P}\colon \square \to$ halt (cf. the corresponding argument in the proof of 4.1).

Proof of $(*)$: If $\mathrm{P}\colon \square \to$ halt, then $\mathfrak{A}_{\mathrm{P}}$ is finite and is a model of ψ_{P}. Hence $\psi_{\mathrm{P}} \in \Phi_{\mathrm{fs}}$. Conversely, if $\mathrm{P}\colon \square \to \infty$, then by (2)(b) in the proof of 4.1, the elements $\overline{0}^{\mathfrak{A}}, \overline{1}^{\mathfrak{A}}, \dots$ are pairwise distinct in every model $\mathfrak{A}$ of ψ_{P}. Thus every model of ψ_{P} is infinite, and hence $\psi_{\mathrm{P}} \notin \Phi_{\mathrm{fs}}$. □

From 5.2 and 5.3 we now obtain

5.4 Trahtenbrot's Theorem. *The set* Φ_{fv} *of first-order* S_∞*-sentences valid in all finite structures is not R-enumerable.*

Proof. Clearly, for $\varphi \in L_0^{S_\infty}$,

$$(*) \qquad \varphi \in L_0^{S_\infty} \setminus \Phi_{\mathrm{fs}} \quad \text{iff} \quad \neg\varphi \in \Phi_{\mathrm{fv}}.$$

For a contradiction assume that Φ_{fv} is R-enumerable. Then, using $(*)$, one can enumerate $L_0^{S_\infty} \setminus \Phi_{\mathrm{fs}}$: one simply starts an enumeration procedure for

Φ_{fv}, and whenever it lists a sentence $\neg\varphi$, one writes down φ. This would lead to a decision procedure for Φ_{fs} (in contradiction to 5.3) as follows: For a string ζ over $\mathcal{A}_0$, decide first whether ζ is an S_∞-sentence. If so, start enumeration procedures for Φ_{fs} (cf. 5.2) and for $L_0^{S_\infty} \setminus \Phi_{\text{fs}}$, and let both procedures continue until one of them yields ζ as output. Thus one obtains a decision whether $\zeta \in \Phi_{\text{fs}}$. □

5.5 Theorem (Incompleteness of Second-Order Logic). *The set of valid second-order S_∞-sentences is not R-enumerable.*

Proof. Let φ_{fin} be a second-order S_∞-sentence with the property that for all $\mathfrak{A}$,

$$\mathfrak{A} \models \varphi_{\text{fin}} \quad \text{iff} \quad A \text{ is finite}$$

(cf. IX.1.3(6)). Then for all first-order(!) S_∞-sentences φ,

$$(*) \qquad \varphi \in \Phi_{\text{fv}} \quad \text{iff} \quad \models \varphi_{\text{fin}} \to \varphi.$$

Now, if the set of valid second-order S_∞-sentences is R-enumerable, then one can start an enumeration procedure for this set, and each time it yields a sentence of the form $\varphi_{\text{fin}} \to \varphi$, where $\varphi \in L_0^{S_\infty}$, one adds φ to the list. In this way, by $(*)$, we obtain an enumeration of Φ_{fv}, in contradiction to Trahtenbrot's Theorem. □

Theorem 5.5 is due to Gödel. It is a stronger version of a result obtained in IX.1. There we concluded from the failure of the Compactness Theorem for second-order logic $\mathcal{L}_{\text{II}}$ that there cannot be any correct and complete proof calculus for $\mathcal{L}_{\text{II}}$. In other words, there is no calculus whose derivability relation $\vdash$ satisfies

$$(+) \quad \begin{array}{l} \text{For all } \mathcal{L}_{\text{II}}\text{-sentences } \varphi \text{ and all sets } \Phi \text{ of } \mathcal{L}_{\text{II}}\text{-sentences,} \\ \Phi \models \varphi \quad \text{iff} \quad \Phi \vdash \varphi. \end{array}$$

However, $(+)$ leaves open the question of whether there is a calculus which satisfies $(+)$ for $\Phi = \emptyset$, that is, whether there is a correct calculus in which all valid second-order sentences are derivable. Now 5.5 shows that in this sense second-order logic is also incomplete: If such a calculus existed, one could apply its rules systematically to generate all possible derivations and hence all valid second-order sentences (cf. the proof of 1.6).

At this point we see how useful it has been to introduce the notion of enumerability: By employing this notion, we were relieved of the task of giving precise definitions for the notions of derivation rule and calculus, but were nevertheless able to conclude that there is no adequate proof calculus for the second-order sentences.

The above argument for 5.5 is based on the fact that the finite sets are characterizable in second-order logic. Thus, it can also be applied to weak second-order logic (cf. IX.1.7).

For the sake of simplicity, we have, in the last two sections, referred to the symbol set S_∞ although we have actually needed only a few symbols from S_∞. It should be clear that the results are also valid for other symbol sets S which are effectively given, as is S_∞, and contain symbols which allow to describe the execution of programs. One can even show that it is sufficient for S to contain only one binary relation symbol. Moreover, the incompleteness of second-order logic already holds for $S = \emptyset$ (cf. 5.6). On the other hand, the set of valid first-order S-sentences is decidable provided S contains only unary relation symbols (cf. XII.3.18(b)).

5.6 Exercise. The set of valid second-order $\emptyset$-sentences is not R-enumerable.

§6. Theories and Decidability

In this section we investigate several theories, especially with regard to enumerability and decidability. Among the results obtained is the undecidability of arithmetic. We shall always assume that the symbol sets considered are effectively given.

A. First-Order Theories

6.1 Definition. $T \subset L_0^S$ is said to be a *theory* if T is satisfiable and if it is closed under consequence (i.e. every S-sentence which follows from T already belongs to T).

For every S-structure $\mathfrak{A}$ the set $\mathrm{Th}(\mathfrak{A}) = \{\varphi \in L_0^S \mid \mathfrak{A} \models \varphi\}$ is a theory, the *theory of* $\mathfrak{A}$ (cf. VI.4.1). $\mathrm{Th}(\mathfrak{N})$ is called (elementary) *arithmetic* where $\mathfrak{N}$ is the S_{ar}-structure $\mathfrak{N} = (\mathbf{N}, +, \cdot, 0, 1)$.

For $\Phi \in L_0^S$ let $\Phi^{\models} := \{\varphi \in L_0^S \mid \Phi \models \varphi\}$. If T is a theory, then $T = T^{\models}$, and if Φ is a satisfiable set of S-sentences, then $\Phi^{\models}$ is a theory. We give a few examples.

(1) $\emptyset^{\models} = \{\varphi \in L_0^S \mid \models \varphi\}$.

(2) For $S = S_{\mathrm{gr}}$: (elementary) *group theory* $\quad \mathrm{Th}_{\mathrm{gr}} := \Phi_{\mathrm{gr}}^{\models}$.

(3) For $S = \{\in\}$: ZFC *set theory* $\quad \mathrm{Th}_{\mathrm{ZFC}} := \mathrm{ZFC}^{\models}$.

(4) For $S = S_{\mathrm{ar}}$: (first-order) *Peano arithmetic* $\quad \mathrm{Th}_{\mathrm{PA}} := \Phi_{\mathrm{PA}}^{\models}$.

The axiom system Φ_{PA} consists of the Peano axioms given in III.7.5, where the usual induction axiom (a second-order sentence) is replaced by the first-order "induction axioms" $(*)$ below. The axioms of Φ_{PA} are:

$$\begin{array}{ll} \forall x \neg x+1 \equiv 0 & \forall x \forall y (x+1 \equiv y+1 \to x \equiv y) \\ \forall x\, x+0 \equiv x & \forall x \forall y\, x+(y+1) \equiv (x+y)+1 \\ \forall x\, x \cdot 0 \equiv 0 & \forall x \forall y\, x \cdot (y+1) \equiv x \cdot y + x \end{array}$$

$$(*) \quad \begin{cases} \text{for all } x_1, \ldots, x_n, y \text{ and all } \varphi \in L^{S_{\mathrm{ar}}} \text{ such that} \\ \text{free}(\varphi) \subset \{x_1, \ldots, x_n, y\} \text{ the sentence} \\ \forall x_1 \ldots \forall x_n \left((\varphi \frac{0}{y} \wedge \forall y (\varphi \to \varphi \frac{y+1}{y})) \to \forall y \varphi \right). \end{cases}$$

$\mathfrak{N}$ is a model of Φ_{PA} and therefore $\Phi_{\mathrm{PA}}^{\models} \subset \mathrm{Th}(\mathfrak{N})$. The induction schema $(*)$ is a natural substitute for the induction axiom, because it expresses the induction axiom for properties which are definable in first-order logic. Many theorems of elementary arithmetic (i.e. sentences in $\mathrm{Th}(\mathfrak{N})$) can be derived from Φ_{PA}. Nevertheless, it turns out that not *all* sentences of $\mathrm{Th}(\mathfrak{N})$ are derivable from Φ_{PA}: in 6.10 we shall show that $\Phi_{\mathrm{PA}}^{\models} \subsetneq \mathrm{Th}(\mathfrak{N})$.

6.2 Definition. (a) A theory T is said to be *R-axiomatizable* if there is an R-decidable set Φ of sentences such that $T = \Phi^{\models}$.
(b) A theory T is said to be *finitely axiomatizable* if there is a finite set Φ of sentences such that $T = \Phi^{\models}$.

Every finitely axiomatizable theory can be axiomatized by means of a single sentence. (Take the conjunction of the axioms.) Every finitely axiomatizable theory is also R-axiomatizable. The theories $\mathrm{Th}_{\mathrm{PA}}$ and $\mathrm{Th}_{\mathrm{ZFC}}$ are R-axiomatizable, but not finitely axiomatizable (which we will not show here).

6.3 Theorem. *An R-axiomatizable theory is R-enumerable.*

Proof. Let T be a theory and let Φ be an R-decidable set of S-sentences such that $T = \Phi^{\models}$. The sentences of T may be listed as follows: Generate systematically all derivable sequents and (with a decision procedure for Φ) check in each case whether the members of the antecedent belong to Φ. If so, list the succedent provided it is a sentence. □

An R-axiomatizable theory T need not necessarily be R-decidable. Examples are $T = \emptyset^{\models}$ (for $S = S_\infty$; cf. 4.1) and $T = T_{\mathrm{gr}}$ (cf. [33]). The situation is different, however, if T is complete in the following sense.

6.4 Definition. A theory $T \subset L_0^S$ is *complete* if for every S-sentence φ we have $\varphi \in T$ or $\neg\varphi \in T$.

$\mathrm{Th}(\mathfrak{A})$ is complete for every structure $\mathfrak{A}$.

6.5 Theorem. (a) *Every R-axiomatizable, complete theory is R-decidable.*
(b) *Every R-enumerable, complete theory is R-decidable.*

Proof. By 6.3 it is sufficient to prove (b). Let T be an R-enumerable complete theory. In order to decide whether a given sentence φ belongs to T, we use a procedure to enumerate T, continuing until either φ or $\neg\varphi$ has

been listed. Since T is complete, one of these two sentences will eventually be listed. If φ is listed, φ belongs to T; if $\neg\varphi$ is listed, φ does not belong to T. □

From 6.5 we obtain the decidability of an axiomatizable theory once we have proved its completeness. A method for proving completeness will be introduced in Chapter XII. In certain cases one can use the assertion in 6.7 for this purpose.

6.6 Exercise. Let $T = \Phi^{\models}$ be a theory, where Φ is R-enumerable. Show that T is R-axiomatizable. (Hint: Starting with an enumeration $\varphi_0, \varphi_1, \ldots$ of Φ, consider the set $\{\varphi_0, \varphi_0 \wedge \varphi_1, \ldots\}$.)

6.7 Exercise. (a) For an at most countable S, let $T \subset L_0^S$ be a theory having only infinite models. Further, suppose there is an infinite cardinal κ such that any two models of T of cardinality κ are isomorphic. Show that T is complete.
(b) Set up a decidable system of axioms for the theory of algebraically closed fields of fixed characteristic and use (a) to show its completeness (and hence, by 6.5, its decidability).

B. The Undecidability of Arithmetic

In this section we prove the undecidability of arithmetic, i.e. we show that there is no procedure which decides for every S_{ar}-sentence whether it holds in $\mathfrak{N}$. We shall use the same method of proof as in showing the undecidability of first-order logic: we effectively assign to every register program P over $\mathcal{A} = \{|\}$ an S_{ar}-sentence φ_{P} such that

$$\mathfrak{N} \models \varphi_{\text{P}} \quad \text{iff} \quad \text{P}: \square \to \text{ halt.}$$

The undecidability of $\text{Th}(\mathfrak{N})$ then follows immediately from the undecidability of Π_{halt} (cf. 3.2).

In defining φ_{P} we shall make use of a formula χ_{P} which, in $\mathfrak{N}$, describes how the program P operates. The following lemma provides such a formula.

Assume the register program P consists of the instructions $\alpha_0, \ldots, \alpha_k$, and let n be the smallest number such that all registers mentioned in P are among $\text{R}_0, \ldots, \text{R}_n$. Recall (cf. Section 4) that a configuration of P is an $(n+2)$-tuple $(L, m_0, \ldots, m_n)$ of natural numbers such that $L \leq k$. $(L, m_0, \ldots, m_n)$ stands for a situation where α_L is the next instruction to be executed and the contents of the registers $\text{R}_0, \ldots, \text{R}_n$ are $m_0, \ldots, m_n$, respectively.

6.8 Lemma. *With any given program* P *one can effectively associate an* S_{ar}*-formula* $\chi_{\text{P}}(v_0, \ldots, v_{2n+2})$ *such that for all* $l_0, \ldots, l_n, L, m_0, \ldots, m_n \in \mathbf{N}$ *the following holds:*

$\mathfrak{N} \models \chi_{\mathrm{P}}[l_0, \ldots, l_n, L, m_0, \ldots, m_n]$ *iff*
P, *beginning with the configuration* $(0, l_0, \ldots, l_n)$, *after finitely many steps reaches the configuration* $(L, m_0, \ldots, m_n)$.

Using χ_{P}, we can write down the desired formula φ_{P}: We set

$$\varphi_{\mathrm{P}} := \exists v_{n+2} \ldots \exists v_{2n+2} \chi_{\mathrm{P}}(\mathbf{0}, \ldots, \mathbf{0}, \boldsymbol{k}, v_{n+2}, \ldots, v_{2n+2}).^2$$

Then we have (note that α_k is the halt-instruction of P):
$\mathfrak{N} \models \varphi_{\mathrm{P}}$
iff P, beginning with the configuration $(0, \ldots, 0)$, after finitely many steps reaches the configuration $(k, m_0, \ldots, m_n)$ for some $m_0, \ldots, m_n$
iff $\mathrm{P}: \square \to$ halt.

Thus we have

6.9 Theorem on the Undecidability of Arithmetic. *Arithmetic, i.e. the theory* $\mathrm{Th}(\mathfrak{N})$, *is not R-decidable.* □

Since $\mathrm{Th}(\mathfrak{N})$ is complete, using 6.5, we obtain

6.10 Corollary. *Arithmetic, i.e. the theory* $\mathrm{Th}(\mathfrak{N})$, *is neither R-axiomatizable nor R-enumerable. In particular,* $\Phi_{\mathrm{PA}}^{\models} \subsetneq \mathrm{Th}(\mathfrak{N})$. □

According to 6.9 and 6.10, arithmetic is not amenable to a purely "mechanical" treatment in the following sense: There is no procedure for deciding whether any given arithmetical sentence is true, nor is there even a procedure which lists all true arithmetical sentences. In other words, every procedure which lists only true arithmetical sentences must necessarily omit some true arithmetical sentences. Thus, mathematicians will never possess a method for systematically proving all true arithmetical sentences. In particular, one cannot effectively give a system of axioms $\Phi \subset \mathrm{Th}(\mathfrak{N})$ from which all sentences in $\mathrm{Th}(\mathfrak{N})$ are derivable.

Proof of Lemma 6.8. Let P be given as above. We must find an S_{ar}-formula $\chi_{\mathrm{P}}(x_0, \ldots, x_n, z, y_0, \ldots, y_n)$ which says (in $\mathfrak{N}$) that P, beginning with the configuration $(0, x_0, \ldots, x_n)$, proceeds through a series of configurations, ending finally with the configuration $(z, y_0, \ldots, y_n)$. That is, $\chi_{\mathrm{P}}(x_0, \ldots, x_n, z, y_0, \ldots, y_n)$ should be a formalization of the following statement (1):

(1) "There is an $s \in \mathbf{N}$ and a sequence $C_0, \ldots, C_s$ of configurations such that
$C_0 = (0, x_0, \ldots, x_n), \quad C_s = (z, y_0, \ldots, y_n),$
for all $i < s$: $C_i \underset{\mathrm{P}}{\to} C_{i+1}$."

---

[2] In case $\varphi \in L_2^{S_{\mathrm{ar}}}$, for example, we write $\varphi(n, v_1)$ for $\varphi\frac{n}{v_0}$ and $\varphi(n, m)$ for $\varphi\frac{n\ m}{v_0\ v_1}$. Here, as before, $0, 1, 2, \ldots$ stand for the S_{ar}-terms $0, 1, 1+1, \ldots$.

"$C_i \underset{\mathrm{P}}{\rightarrow} C_{i+1}$" means that P passes from configuration C_i to C_{i+1} when executing the instruction addressed in C_i.

We form a single sequence from $C_0, \ldots, C_s$ and thus obtain the following formulation (2) of (1):

(2) "There is an $s \in \mathbf{N}$ and a sequence

$$(\underbrace{a_0, \ldots, a_{n+1}}_{C_0}, \underbrace{a_{n+2}, \ldots, a_{(n+2)+(n+1)}}_{C_1}, \ldots, \underbrace{a_{s\cdot(n+2)}, \ldots, a_{s\cdot(n+2)+(n+1)}}_{C_s}),$$

such that

$a_0 = 0,\ a_1 = x_0, \ldots,\ a_{n+1} = x_n,$

$a_{s\cdot(n+2)} = z,\ a_{s\cdot(n+2)+1} = y_0, \ldots,\ a_{s\cdot(n+2)+(n+1)} = y_n,$

and for all $i < s$:

$(a_{i\cdot(n+2)}, \ldots, a_{i\cdot(n+2)+(n+1)}) \underset{\mathrm{P}}{\rightarrow} (a_{(i+1)\cdot(n+2)}, \ldots, a_{(i+1)\cdot(n+2)+(n+1)})$."

The principal difficulty in formalizing (2) as a first-order $L^{S_{\mathrm{ar}}}$-sentence arises with the quantifier "there exists a sequence." We overcome this problem by using natural numbers as codes for finite sequences. Often one codes a sequence $(a_0, \ldots, a_r)$ by the number $p_0^{a_0+1} \cdots p_r^{a_r+1}$, where p_i denotes the ith prime. However, when using this code, we would be forced to give an $L^{S_{\mathrm{ar}}}$-definition of exponentiation x^y. Since such a definition is rather involved, we provide another coding where a sequence $(a_0, \ldots, a_r)$ is coded by two suitably chosen numbers t and p.

6.11 β-Function Lemma[3]. *There is a function $\beta\colon \mathbf{N}^3 \to \mathbf{N}$ with the following properties:*

(a) *For every sequence $(a_0, \ldots, a_r)$ over $\mathbf{N}$ there exist $t, p \in \mathbf{N}$ such that for all $i \leq r$*

$$\beta(t, p, i) = a_i.$$

(b) *β is definable in $L^{S_{\mathrm{ar}}}$, i.e. there exists an S_{ar}-formula $\varphi_\beta(v_0, v_1, v_2, v_3)$ such that for all $t, p, i, a \in \mathbf{N}$,*

$$\mathfrak{N} \models \varphi_\beta[t, p, i, a] \quad \textit{iff} \quad \beta(t, p, i) = a.$$

Proof. Given $(a_0, \ldots, a_r)$, we choose a prime p which is larger than $a_0, \ldots, a_r, r+1$ and set

$$(*)\quad t := 1 \cdot p^0 + a_0 p^1 + 2p^2 + a_1 p^3 + \ldots + (i+1)p^{2i} + a_i p^{2i+1} + \ldots + (r+1)p^{2r} + a_r p^{2r+1}.$$

By choice of p the right-hand side is the p-adic representation of t.

First, we show that for all i, $0 \leq i \leq r$,

[3]This nomenclature stems from Gödel's use of β for a function with the properties (a) and (b) of the lemma.

$$
\begin{array}{lll}
a = a_i & \text{iff} & \text{there are } b_0, b_1, b_2 \text{ such that} \\
 & & \text{(i) } t = b_0 + b_1((i+1) + ap + b_2p^2), \\
(**) & & \text{(ii) } a < p, \\
 & & \text{(iii) } b_0 < b_1, \\
 & & \text{(iv) } b_1 = p^{2m} \text{ for a suitable } m.
\end{array}
$$

The implication from left to right follows immediately from $(*)$ with

$$
\begin{aligned}
b_0 &:= 1 \cdot p^0 + \ldots + a_{i-1}p^{2i-1}, \\
b_1 &:= p^{2i}, \\
b_2 &:= (i+2) + a_{i+1}p + \ldots + a_r p^{2(r-i)-1}.
\end{aligned}
$$

Conversely, suppose (i) – (iv) hold for b_0, b_1, b_2 and let $b_1 = p^{2m}$. From (i) we obtain

$$t = b_0 + (i+1)p^{2m} + ap^{2m+1} + b_2p^{2m+2}.$$

Since $b_0 < p^{2m}$, $a < p$, and $i + 1 < p$, and since the p-adic representation of t is unique, a comparison with $(*)$ yields $m = i$ and $a = a_i$.

Obviously, (iv) from $(**)$ is equivalent to

(iv)$'$ b_1 is a square and for all $d \neq 1$ with $d|b_1$ we have $p|d$.

We define $\beta(t, p, i)$ to be the uniquely determined (and hence the smallest) a for which the right-hand side of $(**)$ (with (iv)$'$ instead of (iv)) holds. We extend this definition to arbitrary triples of natural numbers by specifying:

Let $\beta(u, q, j)$ be the smallest a such that there are b_0, b_1, b_2 with

(i) $u = b_0 + b_1((j+1) + aq + b_2q^2)$,

(ii) $a < q$,

(iii) $b_0 < b_1$,

(iv)$'$ b_1 is a square, and for all $d \neq 1$ with $d|b_1$ we have $q|d$.

If no such a exists let $\beta(u, q, j) = 0$.

Then β has the properties required in (a).

The definition of β just given leads immediately to an S_{ar}-formula $\varphi_\beta(v_0, v_1, v_2, v_3)$ defining β. So (b) holds as well. □

We now return to the proof of 6.8, that is, to the problem of giving an S_{ar}-formula χ_{P} which says that the program P passes in finitely many steps from the configuration $(0, x_0, \ldots, x_n)$ to the configuration $(z, y_0, \ldots, y_n)$. As we have seen, this statement about P is equivalent to statement (2) at the beginning of the proof. We can formalize (2) with the aid of the formula φ_β from the β-Function Lemma 6.11 in the following way (where we now use $s, t, \ldots$ to denote variables):

$$
\begin{aligned}
&\chi_{\mathrm{P}}(x_0,\dots,x_n,z,y_0,\dots,y_n):=\\
&\exists s\exists p\exists t(\varphi_\beta(t,p,0,0)\wedge\varphi_\beta(t,p,1,x_0)\wedge\dots\wedge\varphi_\beta(t,p,\boldsymbol{n}+\mathbf{1},x_n)\\
&\wedge\varphi_\beta(t,p,s\cdot(\boldsymbol{n}+\mathbf{2}),z)\wedge\varphi_\beta(t,p,s\cdot((\boldsymbol{n}+\mathbf{2})+1),y_0)\wedge\dots\\
&\wedge\varphi_\beta(t,p,s\cdot(\boldsymbol{n}+\mathbf{2})+(\boldsymbol{n}+\mathbf{1}),y_n)\\
&\wedge\forall i(i<s\to\forall u\forall u_0\dots\forall u_n\forall u'\forall u'_0\dots\forall u'_n\\
&\quad[\varphi_\beta(t,p,i\cdot(\boldsymbol{n}+\mathbf{2}),u)\wedge\dots\wedge\varphi_\beta(t,p,i\cdot(\boldsymbol{n}+\mathbf{2})+(\boldsymbol{n}+\mathbf{1}),u_n)\\
&\quad\wedge\varphi_\beta(t,p,(i+1)\cdot(\boldsymbol{n}+\mathbf{2}),u')\wedge\dots\\
&\quad\wedge\varphi_\beta(t,p,(i+1)\cdot(\boldsymbol{n}+\mathbf{2})+(\boldsymbol{n}+\mathbf{1}),u'_n)\\
&\quad\to\text{“}(u,u_0,\dots,u_n)\underset{\mathrm{P}}{\to}(u',u'_0,\dots,u'_n)\text{”}]).
\end{aligned}
$$

Here

$$\text{“}(u,u_0,\dots,u_n)\underset{\mathrm{P}}{\to}(u',u'_0,\dots,u'_n)\text{”}$$

stands for a formula which describes the direct transition from configuration $(u,u_0,\dots,u_n)$ to configuration $(u',u'_0,\dots,u'_n)$; such a formula can be obtained as a conjunction $\psi_0\wedge\dots\wedge\psi_{k-1}$, where ψ_j describes transitions induced by instruction α_j of P. For example, if α_j is of the form

j LET $\mathrm{R}_1 = \mathrm{R}_1 + |$,

then we take

$$
\begin{aligned}
\psi_j := u\equiv\boldsymbol{j}\to(u'\equiv u+1\\
\wedge u'_0\equiv u_0\wedge u'_1\equiv u_1+1\wedge u'_2\equiv u_2\wedge\dots\wedge u'_n\equiv u_n).
\end{aligned}
$$

Thus a formula χ_{P} with the desired properties is obtained, and the proof of 6.8 is completed. □

Finally, we note another consequence of the fact that computations of register machines can be described in $\mathfrak{N}$.

6.12 Theorem. *Let $r\geq 1$.*

(a) *Given an r-ary R-decidable relation $\mathfrak{Q}$ over $\mathbf{N}$, there is an S_{ar}-formula $\varphi(v_0,\dots,v_{r-1})$ such that for all $l_0,\dots,l_{r-1}\in\mathbf{N}$*

$$\mathfrak{Q}l_0\dots l_{r-1}\quad\textit{iff}\quad\mathfrak{N}\models\varphi(\boldsymbol{l}_0,\dots,\boldsymbol{l}_{r-1}).$$

(b) *Given an R-computable function $f\colon\mathbf{N}^r\to\mathbf{N}$, there is an S_{ar}-formula $\varphi(v_0,\dots,v_{r-1},v_r)$ such that for all $l_0,\dots,l_{r-1},l_r\in\mathbf{N}$*

$$f(l_0,\dots,l_{r-1})=l_r\quad\textit{iff}\quad\mathfrak{N}\models\varphi(\boldsymbol{l}_0,\dots,\boldsymbol{l}_{r-1},\boldsymbol{l}_r),$$

and in particular,

$$\mathfrak{N}\models\exists^{=1}v_r\varphi(\boldsymbol{l}_0,\dots,\boldsymbol{l}_{r-1},v_r).$$

Proof. (a) Suppose $r\geq 1$ and let $\mathfrak{Q}$ be an r-ary R-decidable relation over $\mathbf{N}$. Let P be a register program which decides $\mathfrak{Q}$ and let R_n be the largest register mentioned in P. Without loss of generality, let $n\geq r-1$. Suppose $\alpha_{L_0},\dots,\alpha_{L_m}$ are the print-instructions in P. Then, choosing χ_{P} as in 6.8, we have for arbitrary $l_0,\dots,l_{r-1}\in\mathbf{N}$:

$\mathfrak{Q}l_0, \ldots, l_{r-1}$

iff P, beginning with configuration $(0, l_0, \ldots, l_{r-1}, \underbrace{0, \ldots, 0}_{n+1-r})$, after finitely many steps reaches a configuration of the form $(L_i, 0, m_1, \ldots, m_n)$ with $0 \leq i \leq m$ (i.e. a print-instruction with the empty word in R_0)

iff $\mathfrak{N} \models \exists v_{n+3} \ldots \exists v_{2n+2}$
$$(\chi_{\mathrm{P}}(\boldsymbol{l}_0, \ldots, \boldsymbol{l}_{r-1}, \mathbf{0}, \ldots, \mathbf{0}, \boldsymbol{L}_0, \mathbf{0}, v_{n+3}, \ldots, v_{2n+2}) \vee \ldots \vee \chi_{\mathrm{P}}(\boldsymbol{l}_0, \ldots, \boldsymbol{l}_{r-1}, \mathbf{0}, \ldots, \mathbf{0}, \boldsymbol{L}_m, \mathbf{0}, v_{n+3}, \ldots, v_{2n+2})).$$

Thus, for $\varphi(v_0, \ldots, v_{r-1})$ one can take the formula

$$\exists v_{n+3} \ldots \exists v_{2n+2} \bigvee_{i=0}^{m} \chi_{\mathrm{P}}(v_0, \ldots, v_{r-1}, \mathbf{0}, \ldots, \mathbf{0}, \boldsymbol{L}_i, \mathbf{0}, v_{n+3}, \ldots, v_{2n+2}).$$

(b) We proceed as in (a), noting that

$f(l_0, \ldots, l_{r-1}) = l_r$

iff P, beginning with configuration $(0, l_0, \ldots, l_{r-1}, 0, \ldots, 0)$, after finitely many steps reaches a configuration of the form $(L_i, l_r, m_1, \ldots, m_n)$ with $0 \leq i \leq m$.

Hence, the required formula $\varphi(v_0, \ldots, v_{r-1}, v_r)$ can be chosen as

$$\exists v_{n+3} \ldots \exists v_{2n+2} \bigvee_{i=0}^{m} \chi_{\mathrm{P}}(v_0, \ldots, v_{r-1}, \mathbf{0}, \ldots, \mathbf{0}, \boldsymbol{L}_i, v_r, v_{n+3}, \ldots, v_{2n+2}). \qquad \square$$

Relations and functions over $\mathbf{N}$ which can be described by an S_{ar}-formula as in 6.12 are said to be *arithmetical*. Thus 6.12 says that all R-decidable relations and all R-computable functions over $\mathbf{N}$ are arithmetical.

The Theorem on the Undecidability of Arithmetic has been strengthened: In 1970 Matijasevič proved that the set PIR of polynomials in integer coefficients with integer roots (cf. 1.11) is not R-decidable. Since to every polynomial p with integer coefficients one can assign effectively an existential S_{ar}-sentence φ_p such that

$$p \in \mathrm{PIR} \quad \text{iff} \quad \varphi_p \in \mathrm{Th}(\mathfrak{N}),^4$$

we obtain that already the set $\{\varphi \in \mathrm{Th}(\mathfrak{N}) \mid \varphi \text{ is existential}\}$ is undecidable. The considerations by Matijasevič show that every R-enumerable subset of $\mathbf{N}$ can be written in the form

$$\{n \in \mathbf{N} \mid \text{ there are integers } z_1, \ldots, z_r \text{ with } p(z_1, \ldots, z_r) = n\},$$

where p is a polynomial with integer coefficients. Hence, the R-enumerable subsets of $\mathbf{N}$ coincide with the "$\mathbf{N}$-parts" of the ranges of such polynomials.

---

[4]E.g. to the polynomial $p = x + 2y^2 - 5$ one assigns the existential sentence $\varphi_p = \exists x \exists y\, (x + \mathbf{2}(y \cdot y) \equiv \mathbf{5} \ \vee\ \mathbf{2}(y \cdot y) \equiv x + \mathbf{5})$ (φ_p also takes negative roots into account).

6.13 Exercise. Let $\mathfrak{Z} = (\mathbf{Z}, +, \cdot, 0, 1)$ be the ring of integers (as S_{ar}-structure). Show that $\text{Th}(\mathfrak{Z})$ is not R-decidable. (Hint: Use the fact that an integer is a natural number if and only if it is the sum of four squares of integers.)

§7. Self-Referential Statements and Gödel's Incompleteness Theorems

In the preceding section we have shown that arithmetic is not R-axiomatizable. Originally Gödel [12] used another method to prove this result. He showed that within sufficiently strong axiom systems there are self-referential formulas, i.e. formulas which make statements about themselves. Such self-referential formulas are the main theme of this section. We close by taking up our original objective of this chapter and obtain some important results concerning the limitations of the formal method. With this aim in mind, we shall often conduct the arguments on the syntactic level.

In the following, we take Φ to be a set of S_{ar}-sentences.

7.1 Definition. (a) A relation $\mathfrak{Q} \subset \mathbf{N}^r$ is said to be *representable in* Φ if there is an S_{ar}-formula $\varphi(v_0, \ldots, v_{r-1})$ such that for all $n_0, \ldots, n_{r-1} \in \mathbf{N}$:

$$\text{If } \mathfrak{Q}n_0 \ldots n_{r-1} \text{ then } \Phi \vdash \varphi(\boldsymbol{n}_0, \ldots, \boldsymbol{n}_{r-1});$$
$$\text{if not } \mathfrak{Q}n_0 \ldots n_{r-1} \text{ then } \Phi \vdash \neg\varphi(\boldsymbol{n}_0, \ldots, \boldsymbol{n}_{r-1}).$$

In this case we say that $\varphi(v_0, \ldots, v_{r-1})$ *represents* $\mathfrak{Q}$ *in* Φ.

(b) A function $F\colon \mathbf{N}^r \to \mathbf{N}$ is said to be *representable in* Φ if there is an S_{ar}-formula $\varphi(v_0, \ldots, v_{r-1}, v_r)$ such that for all $n_0, \ldots, n_{r-1}, n_r \in \mathbf{N}$:

$$\text{If } F(n_0, \ldots, n_{r-1}) = n_r \quad \text{then } \Phi \vdash \varphi(\boldsymbol{n}_0, \ldots, \boldsymbol{n}_{r-1}, \boldsymbol{n}_r);$$
$$\text{if } F(n_0, \ldots, n_{r-1}) \neq n_r \quad \text{then } \Phi \vdash \neg\varphi(\boldsymbol{n}_0, \ldots, \boldsymbol{n}_{r-1}, \boldsymbol{n}_r);$$
$$\Phi \vdash \exists^{=1} v_r \varphi(\boldsymbol{n}_0, \ldots, \boldsymbol{n}_{r-1}, v_r).$$

In this case we say that $\varphi(v_0, \ldots, v_{r-1}, v_r)$ *represents* F *in* Φ.

7.2 Lemma. (a) *If Φ is inconsistent then every relation over $\mathbf{N}$ and every function over $\mathbf{N}$ is representable in Φ.*

(b) *If $\Phi \subset \Phi' \subset L_0^{S_{\text{ar}}}$ then the relations and functions representable in Φ are also representable in Φ'.*

(c) *Let Φ be consistent. If Φ is R-decidable then every relation representable in Φ is R-decidable and every function representable in Φ is R-computable.*

Proof. The assertions (a) and (b) follow immediately from Definition 7.1. We show (c) for a function $F\colon \mathbf{N} \to \mathbf{N}$. Let F be represented in Φ by

$\varphi(v_0, v_1)$. We obtain a procedure to compute F in the following way: Let $n \in \mathbf{N}$ be given. Then $\Phi \vdash \varphi(\boldsymbol{n}, \boldsymbol{F(n)})$ and $\Phi \vdash \neg\varphi(\boldsymbol{n}, \boldsymbol{m})$ for $m \neq F(n)$; hence by the consistency of Φ, we have not $\Phi \vdash \varphi(\boldsymbol{n}, \boldsymbol{m})$ for $m \neq F(n)$. To determine $F(n)$, i.e. to find k with $\Phi \vdash \varphi(\boldsymbol{n}, \boldsymbol{k})$, we start an enumeration procedure for $\{\psi \in L_0^{S_{\mathrm{ar}}} \mid \Phi \vdash \psi\}$ and at the same time produce the sentences $\varphi(\boldsymbol{n}, \mathbf{0}), \varphi(\boldsymbol{n}, \mathbf{1}), \varphi(\boldsymbol{n}, \mathbf{2}), \ldots$. As soon as the enumeration procedure yields a sentence $\varphi(\boldsymbol{n}, \boldsymbol{k})$, we have found k to be the value for $F(n)$. □

We say that Φ *allows representations* if all R-decidable relations and all R-computable functions over $\mathbf{N}$ are representable in Φ.

Φ allowing representations says, in a certain sense, that Φ is rich enough to describe how procedures operate. In the preceding section we have described the execution of register programs in $\Phi = \mathrm{Th}(\mathfrak{N})$. Indeed, we have

7.3 Theorem. $\mathrm{Th}(\mathfrak{N})$ *allows representations.*

The proof is immediate from 6.12 if one notes that for every S_{ar}-sentence φ we have ($\mathfrak{N} \models \varphi$ iff $\mathrm{Th}(\mathfrak{N}) \vdash \varphi$) and (not $\mathfrak{N} \models \varphi$ iff $\mathrm{Th}(\mathfrak{N}) \vdash \neg\varphi$). □

A closer analysis (not pursued here) of the considerations leading to the proof of 6.12 shows that one can describe the execution of register programs already on the basis of Peano arithmetic, i.e. in Φ_{PA}.Thus, one can obtain

7.4 Theorem. Φ_{PA} *allows representations.* □

As an important technical means we assume in the following that an effective coding of the S_{ar}-formulas by natural numbers (a "Gödel numbering") is given, and moreover, that the Gödel numbering is surjective, i.e. that every number is the Gödel number of some formula. We write n^{φ} for the Gödel number of φ.

In this way it is possible to translate statements about formulas into arithmetical statements. For example, a statement about the derivability of a formula φ becomes an arithmetical statement about the Gödel number of φ, and this in turn can be formalized as an S_{ar}-sentence. This idea gives us the key to construct self-referential formulas.

The way we shall proceed originates from the liar paradox. This paradox amounts to the fact that the statement

(∗) "*I am not telling the truth now*"

can neither be true nor false; for if it were true it would have to be false, and if it were false it would have to be true.

Note that (∗) makes a statement about itself, and hence is an example of a *self-referential* statement. In a first step we consider statements of this kind in general. We show that within a sufficiently rich system (i.e. in a system which allows representations), *every* property which is expressible in the system gives rise to a self-referential sentence; more precisely:

7.5 Fixed Point Theorem. *Suppose Φ allows representations. Then, for every $\psi \in L_1^{S_{\mathrm{ar}}}$, there is an S_{ar}-sentence φ such that*

$$\Phi \vdash \varphi \leftrightarrow \psi(\boldsymbol{n}^{\varphi}).$$

Intuitively, φ says: "I have the property ψ."

Proof. Let $F\colon \mathbf{N} \times \mathbf{N} \to \mathbf{N}$ be given by

$$F(n,m) = \begin{cases} n^{\chi(\boldsymbol{m})}, & \text{if } n = n^{\chi} \text{ for some } \chi \in L_1^{S_{\mathrm{ar}}}; \\ 0, & \text{otherwise.} \end{cases}$$

Clearly, F is computable, and for $\chi \in L_1^{S_{\mathrm{ar}}}$ we have

$$F(n^{\chi}, m) = n^{\chi(\boldsymbol{m})}.$$

Since Φ allows representations, F can be represented in Φ by a suitable S_{ar}-formula $\alpha(v_0, v_1, v_2)$. We write x, y, z for v_0, v_1, v_2. For given $\psi \in L_1^{S_{\mathrm{ar}}}$ we set

$$\beta := \forall z(\alpha(x,x,z) \to \psi(z)),$$
$$\varphi := \forall z(\alpha(\boldsymbol{n}^{\beta}, \boldsymbol{n}^{\beta}, z) \to \psi(z)).$$

Since $\beta \in L_1^{S_{\mathrm{ar}}}$ and $\varphi = \beta\frac{\boldsymbol{n}^{\beta}}{x}$, we have $F(n^{\beta}, n^{\beta}) = n^{\varphi}$ and hence

(1) $$\Phi \vdash \alpha(\boldsymbol{n}^{\beta}, \boldsymbol{n}^{\beta}, \boldsymbol{n}^{\varphi}).$$

Now we show the claim for φ and ψ, i.e.

$$\Phi \vdash \varphi \leftrightarrow \psi(\boldsymbol{n}^{\varphi}).$$

For the direction from left to right, we have by definition of φ that

$$\Phi \cup \{\varphi\} \vdash \alpha(\boldsymbol{n}^{\beta}, \boldsymbol{n}^{\beta}, \boldsymbol{n}^{\varphi}) \to \psi(\boldsymbol{n}^{\varphi}),$$

by (1) therefore $\Phi \vdash \varphi \to \psi(\boldsymbol{n}^{\varphi})$.

On the other hand, α represents the function F in Φ, in particular

$$\Phi \vdash \exists^{=1} z\alpha(\boldsymbol{n}^{\beta}, \boldsymbol{n}^{\beta}, z);$$

thus by (1)

$$\Phi \vdash \forall z(\alpha(\boldsymbol{n}^{\beta}, \boldsymbol{n}^{\beta}, z) \to z \equiv \boldsymbol{n}^{\varphi})$$

and therefore

$$\Phi \vdash \psi(\boldsymbol{n}^{\varphi}) \to \forall z(\alpha(\boldsymbol{n}^{\beta}, \boldsymbol{n}^{\beta}, z) \to \psi(z)),$$

that is

$$\Phi \vdash \psi(\boldsymbol{n}^{\varphi}) \to \varphi. \qquad \square$$

The following theorem shows that in a system which is rich enough one cannot speak about the truth of all its statements. Formally, we consider a consistent system of axioms Φ which allows representations. The "true" statements correspond to the sentences in $\Phi^{\vdash} = \{\varphi \in L_0^{S_{\mathrm{ar}}} \mid \Phi \vdash \varphi\}$. To say that one can speak of "truth" or "falsity" in Φ is to say that $\Phi^{\vdash}$ (more precisely: $\{n^{\varphi} \mid \varphi \in \Phi^{\vdash}\}$) is representable in Φ.

7.6 Lemma. *Let Φ be consistent and suppose Φ allows representations. Then $\Phi^{\vdash}$ is not representable in Φ.*

Proof. Suppose the assumptions of the lemma hold and let $\chi(v_0)$ represent $\Phi^{\vdash}$ in Φ. By the consistency of Φ we get for an arbitrary $\alpha \in L_0^{S_{\mathrm{ar}}}$,

$$\Phi \vdash \neg\chi(\boldsymbol{n}^{\alpha}) \quad \text{iff} \quad \text{not } \Phi \vdash \alpha. \tag{1}$$

For $\psi := \neg\chi$ we choose, by 7.5, a "fixed point" $\varphi \in L_0^{S_{\mathrm{ar}}}$ such that

$$\Phi \vdash \varphi \leftrightarrow \neg\chi(\boldsymbol{n}^{\varphi}). \tag{2}$$

(φ says intuitively "I am not true.") But then

$$\begin{aligned} \Phi \vdash \varphi \quad &\text{iff} \quad \Phi \vdash \neg\chi(\boldsymbol{n}^{\varphi}) \quad \text{(by (2))} \\ &\text{iff} \quad \text{not } \Phi \vdash \varphi \quad \text{(by (1))}, \end{aligned}$$

a contradiction. □

Lemma 7.6 has interesting consequences, both on the syntactical and semantical levels. In semantical formulations one usually refers to $\Phi^{\models}$ instead of $\Phi^{\vdash}$. From 7.6 we obtain Tarski's Theorem (7.7) and Gödel's First Incompleteness Theorem (7.8).

7.7 Tarski's Theorem [32]. (a) *Suppose Φ is consistent and allows representations. Then $\Phi^{\models}$ is not representable in Φ.*
(b) $\mathrm{Th}(\mathfrak{N})$ *is not representable in* $\mathrm{Th}(\mathfrak{N})$.

Proof. Since $\Phi^{\vdash} = \Phi^{\models}$, (a) follows immediately from 7.6. As $\mathrm{Th}(\mathfrak{N})$ is consistent and allows representations (cf. 7.3), (b) is a special case of (a). □

Tarski's Theorem is of great significance in the study of semantics. Part (b) can be formulated succinctly as "there is no truth definition for arithmetic within arithmetic".

7.8 Gödel's First Incompleteness Theorem. *Let Φ be consistent and R-decidable and suppose Φ allows representations. Then there is an S_{ar}-sentence φ such that neither $\Phi \vdash \varphi$ nor $\Phi \vdash \neg\varphi$.*

Proof. Suppose that for every S_{ar}-sentence φ, either $\Phi \vdash \varphi$ or $\Phi \vdash \neg\varphi$. Then $\Phi^{\vdash}$ is complete and hence R-decidable (cf. 6.5(a)). Thus, since Φ allows representations, $\Phi^{\vdash}$ is representable in Φ, a contradiction to 7.6. □

A refinement of the above argumentations leads to results concerning the consistency of mathematics. In particular, Gödel's Second Incompleteness Theorem, which we shall now derive, shows that the consistency of a sufficiently rich system cannot be proved using only the means available within the system.

In the following let $\Phi \subset L_0^{S_{\mathrm{ar}}}$ be decidable and allow representations.

We choose an effective enumeration of all derivations in the sequent calculus associated with S_{ar} and define a relation H by

Hnm iff the mth derivation ends with a sequent of the form $\psi_0 \ldots \psi_{k-1}\ \varphi$, where $\psi_0, \ldots, \psi_{k-1} \in \Phi$ and $n = n^\varphi$.

Since Φ is decidable, so is H, and clearly

$$\Phi \vdash \varphi \quad \text{iff} \quad \text{there is } m \in \mathbf{N} \text{ such that } Hn^\varphi m.$$

Since Φ allows representations, H can be represented in Φ by a suitable formula $\varphi_H(v_0, v_1) \in L_2^{S_{\mathrm{ar}}}$. Again we write x, y for v_0, v_1 and set

$$\mathrm{Der}_\Phi(x) := \exists y \varphi_H(x, y).$$

For $\psi = \neg\mathrm{Der}_\Phi(x)$ we choose with 7.5 a fixed point $\varphi \in L_0^{S_{\mathrm{ar}}}$, i.e. an S_{ar}-sentence φ with

$$(*) \qquad \Phi \vdash \varphi \leftrightarrow \neg\mathrm{Der}_\Phi(\boldsymbol{n}^\varphi).$$

φ says intuitively "I am not provable from Φ".

7.9 Lemma. *If* Con Φ *then not* $\Phi \vdash \varphi$.

Proof. Suppose $\Phi \vdash \varphi$ holds. Choose m such that $Hn^\varphi m$. Then $\Phi \vdash \varphi_H(\boldsymbol{n}^\varphi, \boldsymbol{m})$, and so $\Phi \vdash \mathrm{Der}_\Phi(\boldsymbol{n}^\varphi)$. From $(*)$ we have $\Phi \vdash \neg\varphi$, and hence Inc Φ. □

Since $\Phi \vdash 0 \equiv 0$, we have

$$\mathrm{Con}\ \Phi \quad \text{iff} \quad \text{not } \Phi \vdash \neg 0 \equiv 0.$$

The S_{ar}-sentence

$$\mathrm{Consis}_\Phi := \neg\mathrm{Der}_\Phi(\boldsymbol{n}^{\neg 0 \equiv 0})$$

thus expresses the consistency of Φ. Lemma 7.9 may then be formalized as

$$(**) \qquad \mathrm{Consis}_\Phi \to \neg\mathrm{Der}_\Phi(\boldsymbol{n}^\varphi).$$

An argument which is in principle simple, though technically rather tedious, could now be used to show that for $(**)$ the proof of 7.9 can be carried out on the basis of Φ, i.e., one can show that

$$(***) \qquad \Phi \vdash \mathrm{Consis}_\Phi \to \neg\mathrm{Der}_\Phi(\boldsymbol{n}^\varphi),$$

in case $\Phi \supset \Phi_{\mathrm{PA}}$ (and if a sufficiently simple formula $\varphi_H(x, y)$ to be used in Der_Φ has been chosen; cf. Exercise 7.12). Thus we obtain:

7.10 Gödel's Second Incompleteness Theorem. *Let* Φ *be consistent and R-decidable with* $\Phi \supset \Phi_{\mathrm{PA}}$. *Then*

$$\textit{not}\ \ \Phi \vdash \mathrm{Consis}_\Phi.$$

Proof. If $\Phi \vdash \mathrm{Consis}_\Phi$ then by $(***)$ $\Phi \vdash \neg\mathrm{Der}_\Phi(\boldsymbol{n}^\varphi)$. Since $\Phi \vdash \varphi \leftrightarrow \neg\mathrm{Der}_\Phi(\boldsymbol{n}^\varphi)$ (cf. $(*)$) it would follow that $\Phi \vdash \varphi$, in contradiction to 7.9. □

For $\Phi = \Phi_{\mathrm{PA}}$, Gödel's Second Incompleteness Theorem says intuitively that the consistency of Φ_{PA} cannot be proved on the basis of Φ_{PA}. This result shows that Hilbert's program cannot be carried out in its original

form. In particular, this program aimed at a consistency proof for Φ_{PA} with elementary, so-called *finitistic* means. The concept "finitistic", though not defined precisely (cf. [18], I, p. 32), was taken in a very narrow sense; in particular it was meant that finitistic proof methods be carried out on the basis of Φ_{PA}.

The above argument can be transferred to other systems where there is a substitute for the natural numbers and where R-decidable relations and R-computable functions are representable. In particular, it applies to systems of axioms for set theory such as ZFC. One uses the natural numbers as defined in ZFC (cf. VIII.3). Then one can define an $\{\in\}$-sentence $\mathrm{Consis}_{\mathrm{ZFC}}$, which expresses the consistency of ZFC, to obtain:

7.11 Theorem. *If* Con ZFC *then not* ZFC $\vdash \mathrm{Consis}_{\mathrm{ZFC}}$. □

Since contemporary mathematics can be based on the ZFC axioms, and since "not ZFC $\vdash \mathrm{Consis}_{\mathrm{ZFC}}$" says that the consistency of ZFC cannot be proved using only means available within ZFC, we can formulate 7.11 as follows: If mathematics is consistent, we cannot prove its consistency by mathematical means.

In a similar way also Tarski's Theorem and Gödel's First Incompleteness Theorem can be transferred to axiom systems for set theory. For example, 7.8 would then assert that for every decidable and consistent system Φ of axioms for set theory which contains ZFC, there is an $\{\in\}$-sentence φ such that neither $\Phi \vdash \varphi$ nor $\Phi \vdash \neg\varphi$. Intuitively this means that there is no decidable consistent system of axioms for mathematics which, for every mathematical statement, allows us to either prove or disprove it. In this fact an inherent limitation of the axiomatic method is manifested.

With the results of Matijasevič mentioned at the end of Section 6 we can formulate 7.11 in the following form which is easy to remember: One can write down a polynomial p in finitely many indeterminates with integer coefficients for which the following holds: Mathematics is consistent if and only if p has no (integer) root. By 7.11 we have therefore: If p has no root, then mathematics cannot prove it.

7.12 Exercise. For the (effectively given) symbol set S, fix a Gödel numbering of the S-formulas; let n^{φ} be the Gödel number of φ. Furthermore, for $n \in \mathbf{N}$, let $\underline{n}$ be a variable free S-term.

For $\Phi \subset L_0^S$ let the S-formula $\mathrm{der}(v_0)$ ("v_0 is derivable from Φ") satisfy the so-called *Löb axioms*, i.e. for arbitrary $\varphi, \psi \in L^S$,

(L1) If $\Phi \vdash \varphi$ then $\Phi \vdash \mathrm{der}(\underline{n^{\varphi}})$;

(L2) $\Phi \vdash (\mathrm{der}(\underline{n^{\varphi}}) \wedge \mathrm{der}(\underline{n^{(\varphi\to\psi)}}) \to \mathrm{der}(\underline{n^{\psi}}))$;

(L3) $\Phi \vdash (\mathrm{der}(\underline{n^{\varphi}}) \to \mathrm{der}(\underline{n^{\mathrm{der}(\underline{n^{\varphi}})}}))$.

Show: If Φ is consistent and if there is an S-sentence φ_0 such that $\Phi \vdash (\varphi_0 \leftrightarrow \neg\mathrm{der}(\underline{n^{\varphi_0}}))$, then not $\Phi \vdash \neg\mathrm{der}(\underline{n^{\neg\underline{0}\equiv\underline{0}}})$. (Hint: Show that, if (L1), (L2), (L3) hold for all $\varphi, \psi \in L^S$, then also $\Phi \vdash (\mathrm{der}(\underline{n^{\varphi}}) \wedge \mathrm{der}(\underline{n^{\psi}}) \rightarrow \mathrm{der}(\underline{n^{(\varphi\wedge\psi)}}))$ and $\Phi \vdash (\mathrm{der}(\underline{n^{\varphi_0}}) \rightarrow \mathrm{der}(\underline{n^{\neg\varphi_0}}))$.)

XI

Free Models and Logic Programming

In general, the following statement is false:

(*) If $\Phi \vdash \exists x\varphi$ then there is a term t with $\Phi \vdash \varphi\frac{t}{x}$.

We get a counterexample for $S = \{R\}$ with unary R, $\Phi = \{\exists xRx\}$, and $\varphi = Rx$.

The main subject of this chapter are results showing that (*) – or variants of (*) – hold under certain conditions on Φ and φ. The corresponding proofs start from the term structures introduced in V.1. These structures turn out to be *free* or *minimal* and therefore have algebraically important properties.

Statement (*) says that an existential proposition $\exists x\varphi$ which holds (under the assumptions of Φ) has a "concrete" solution t. Are there efficient procedures for finding such solutions? This question leads to the fundamentals of *logic programming*, a subject which plays an important role in certain areas of computer science (data structures, knowledge-based systems). So this chapter establishes a bridge between central problems in logic and questions oriented to applications.

The techniques mentioned above are mainly based on an analysis of quantifier-free formulas. This motivates the study of so-called *propositional logic*, the logic of connectives to be treated in Section 4 below.

To emphasize the aspect of effectiveness we formulate many results and proofs using the derivation relation $\vdash$, but we recommend following the arguments on the semantic level, i.e. using the equivalent consequence relation $\models$.

§1. Herbrand's Theorem

In V.1 we have assigned to each set Φ of formulas its *term interpretation* $\mathfrak{I}^\Phi = (\mathfrak{T}^\Phi, \beta^\Phi)$. For this purpose we have introduced an equivalence relation

$\sim$ on the set T^S of S-terms as follows:

$$t \sim t' \quad \text{:iff} \quad \Phi \vdash t \equiv t'.$$

For $t \in T^S$ we have denoted the equivalence class of t modulo $\sim$ by $\overline{t}$ and set:

$$T^\Phi := \{\overline{t} \mid t \in T^S\};$$

for n-ary $R \in S$: $\quad R^{\mathfrak{T}^\Phi}\overline{t_1}\ldots\overline{t_n} \quad$:iff $\quad \Phi \vdash Rt_1\ldots t_n$;

for n-ary $f \in S$: $\quad f^{\mathfrak{T}^\Phi}(\overline{t_1},\ldots,\overline{t_n}) := \overline{ft_1\ldots t_n}$;

for $c \in S$: $\quad c^{\mathfrak{T}^\Phi} := \overline{c}$;

and finally, $\beta^\Phi(x) := \overline{x}$.

Writing $\varphi(\overset{n}{x} \mid \overset{n}{t})$ instead of $\varphi\frac{t_1\ldots t_n}{x_1\ldots x_n}$ we obtained (cf. Lemma V.1.7):

1.1 Reminder. (a) *For all t: $\mathfrak{I}^\Phi(t) = \overline{t}$.*
(b) *For all atomic formulas φ:*

$$\mathfrak{I}^\Phi \models \varphi \quad \textit{iff} \quad \Phi \vdash \varphi.$$

(c) *For all formulas φ and pairwise distinct variables $x_1,\ldots,x_n$:*

(i) $\mathfrak{I}^\Phi \models \exists x_1\ldots\exists x_n\varphi$ *iff there are S-terms $t_1,\ldots,t_n$ with* $\mathfrak{I}^\Phi \models \varphi(\overset{n}{x} \mid \overset{n}{t})$;

(ii) $\mathfrak{I}^\Phi \models \forall x_1\ldots\forall x_n\varphi$ *iff for all S-terms $t_1,\ldots,t_n$,* $\mathfrak{I}^\Phi \models \varphi(\overset{n}{x} \mid \overset{n}{t})$.

For formulas of the form $\exists x_1\ldots\exists x_n\varphi$ or of the form $\forall x_1\ldots\forall x_n\varphi$ we throughout assume $x_1,\ldots,x_n$ to be pairwise distinct.

In analogy to L_k^S, for $k \in \mathbf{N}$, we define the set

$$T_k^S := \{t \in T^S \mid \mathrm{var}(t) \subset \{v_0,\ldots,v_{k-1}\}\}.$$

We consider the substructure of $\mathfrak{T}^\Phi$ whose universe

$$T_k^\Phi := \{\overline{t} \mid t \in T_k^S\}$$

consists of the term classes $\overline{t}$ with $t \in T_k^S$. To ensure in case $k = 0$ the existence of such a term, i.e. that T_k^S is nonempty, we assume from now on:

If $k = 0$ then S contains at least one constant.

T_k^Φ is the universe of a substructure of $\mathfrak{T}^\Phi$ since it is S-closed in $\mathfrak{T}^\Phi$. In fact, if $c \in S$ then $c \in T_k^S$ and therefore $\overline{c} \in T_k^\Phi$; and if $f \in S$ is n-ary and $a_1,\ldots,a_n \in T_k^\Phi$, say $a_1 = \overline{t_1},\ldots,a_n = \overline{t_n}$ for suitable $t_1,\ldots,t_n \in T_k^S$, then $f^{\mathfrak{T}^\Phi}(a_1,\ldots,a_n) = f^{\mathfrak{T}^\Phi}(\overline{t_1},\ldots,\overline{t_n}) = \overline{ft_1\ldots t_n} \in T_k^\Phi$.

Let $\mathfrak{T}_k^\Phi$ be the substructure of $\mathfrak{T}^\Phi$ with universe T_k^Φ and let β_k^Φ be an assignment in $\mathfrak{T}_k^\Phi$ with

$$(+) \qquad \beta_k^\Phi(v_i) = \beta^\Phi(v_i)\ (= \overline{v_i}) \text{ for } i < k$$

and for $i \geq k$ say

$$\beta_k^\Phi(v_i) = \begin{cases} \overline{v_0} & \text{if } k \neq 0, \\ \overline{c} & \text{if } k = 0, \end{cases}$$

where, in case $k = 0$, c is a constant from S. Finally, let

$$\mathfrak{I}_k^\Phi := (\mathfrak{T}_k^\Phi, \beta_k^\Phi).$$

By (+) and the Coincidence Lemma the following holds for $t \in T_k^S$ and $\varphi \in L_k^S$:

$$(\mathfrak{T}^\Phi, \beta_k^\Phi)(t) = (\mathfrak{T}^\Phi, \beta^\Phi)(t) = \overline{t} \quad \text{(cf. 1.1)},$$

$$(\mathfrak{T}^\Phi, \beta_k^\Phi) \models \varphi \quad \text{iff} \quad (\mathfrak{T}^\Phi, \beta^\Phi) \models \varphi,$$

respectively. Since $\mathfrak{T}_k^\Phi \subset \mathfrak{T}^\Phi$ we conclude, using the Substructure Lemma:

1.2 Lemma. (a) *$\mathfrak{I}_k^\Phi(t) = \overline{t}$ for $t \in T_k^S$, and therefore $t^{\mathfrak{T}_0^\Phi} = \overline{t}$ for $t \in T_0^S$.*
(b) *For quantifier-free $\psi \in L_k^S$: $\mathfrak{I}^\Phi \models \psi$ iff $\mathfrak{I}_k^\Phi \models \psi$.*
(c) *For universal $\psi \in L_k^S$: If $\mathfrak{I}^\Phi \models \psi$ then $\mathfrak{I}_k^\Phi \models \psi$, hence in case $k = 0$: If $\mathfrak{T}^\Phi \models \psi$ then $\mathfrak{T}_0^\Phi \models \psi$.* □

The next lemma is the main step towards Herbrand's Theorem, the first result which is of the form (∗) mentioned at the beginning of this chapter.

1.3 Lemma. *For a set $\Phi \subset L_k^S$ of universal formulas in prenex normal form the following are equivalent:*
(a) *Φ is satisfiable.*
(b) *The set $\Phi_0 :=$*

$$\{\varphi(\overset{m}{x} \mid \overset{m}{t}) \mid \forall x_1 \ldots \forall x_m \varphi \in \Phi,\ \varphi \text{ quantifier-free and } t_1, \ldots, t_m \in T_k^S\}$$

is satisfiable.

Proof. From (a) we obtain (b), since always $\forall x_1 \ldots \forall x_m \varphi \models \varphi(\overset{m}{x} \mid \overset{m}{t})$ for $t_1, \ldots, t_m \in T_k^S$. For the direction from (b) to (a), an easy argument using the Compactness Theorem shows that it suffices to consider finite S. So let S be finite and let Φ_0 be satisfiable and therefore consistent. Since $\Phi_0 \subset L_k^S$, free(Φ_0) is finite. Therefore (cf. V.2.1 and V.2.2) there is Θ with $\Phi_0 \subset \Theta \subset L^S$ which is negation complete and contains witnesses. By Henkin's Theorem, $\mathfrak{I}^\Theta$ is a model of Θ, in particular $\mathfrak{I}^\Theta \models \Phi_0$. Since Φ_0 contains only quantifier-free formulas from L_k^S, $\mathfrak{I}_k^\Theta$ is a model of Φ_0 (by 1.2(b)). Hence for all formulas $\forall x_1 \ldots \forall x_m \varphi \in \Phi$ with quantifier-free φ we have:

$$\text{for all } t_1, \ldots, t_m \in T_k^S\colon \quad \mathfrak{I}_k^\Theta \models \varphi(\overset{m}{x} \mid \overset{m}{t}),$$

thus with $\mathfrak{I}_k^\Theta(t_i) = \overline{t_i}$ (cf. 1.2(a)) and the Substitution Lemma we get:

$$\text{for all } t_1, \ldots, t_m \in T_k^S\colon \quad \mathfrak{I}_k^\Theta \frac{\overline{t_1} \ldots \overline{t_m}}{x_1 \ldots x_m} \models \varphi.$$

Since $T_k^\Theta = \{\bar{t} \mid t \in T_k^S\}$, we obtain $\mathfrak{I}_k^\Theta \models \forall x_1 \ldots \forall x_m \varphi$. Thus $\mathfrak{I}_k^\Theta$ is a model of Φ. □

1.4 Herbrand's Theorem. *Let $k \in \mathbf{N}$, and let the symbol set S contain a constant in case $k = 0$. For formulas $\forall x_1 \ldots \forall x_m \varphi$ and $\exists y_1 \ldots \exists y_n \psi$ from L_k^S with quantifier-free φ, ψ and pairwise distinct variables $x_1, \ldots, x_m$ and $y_1, \ldots, y_n$, respectively, the following are equivalent:*

(a) $\forall x_1 \ldots \forall x_m \varphi \vdash \exists y_1 \ldots \exists y_n \psi$.

(b) *There are $j \geq 1$ and terms $t_{11}, \ldots, t_{1n}, \ldots, t_{j1}, \ldots, t_{jn} \in T_k^S$ with*

$$\forall x_1 \ldots \forall x_m \varphi \vdash \psi(\overset{n}{y} \mid \overset{n}{t_1}) \vee \ldots \vee \psi(\overset{n}{y} \mid \overset{n}{t_j})^1.$$

(c) *There are $i, j \geq 1$ and terms $s_{11}, \ldots, s_{1m}, \ldots, s_{i1}, \ldots, s_{im}$ and $t_{11}, \ldots, t_{1n}, \ldots, t_{j1}, \ldots, t_{jn} \in T_k^S$ with*

$$\varphi(\overset{m}{x} \mid \overset{m}{s_1}) \wedge \ldots \wedge \varphi(\overset{m}{x} \mid \overset{m}{s_i}) \vdash \psi(\overset{n}{y} \mid \overset{n}{t_1}) \vee \ldots \vee \psi(\overset{n}{y} \mid \overset{n}{t_j}).$$

Proof. Since $\forall x_1 \ldots \forall x_m \varphi \vdash \varphi(\overset{m}{x} \mid \overset{m}{s})$ and $\psi(\overset{n}{y} \mid \overset{n}{t}) \vdash \exists y_1 \ldots \exists y_n \psi$, we easily get (b) from (c) and (a) from (b). Therefore we only have to show that (a) implies (c). So let $\forall x_1 \ldots \forall x_m \varphi \vdash \exists y_1 \ldots \exists y_n \psi$. Thus the set $\{\forall x_1 \ldots \forall x_m \varphi,\ \neg \exists y_1 \ldots \exists y_n \psi\}$ is not satisfiable, and neither is the set $\{\forall x_1 \ldots \forall x_m \varphi,\ \forall y_1 \ldots \forall y_n \neg \psi\}$. With the previous lemma we obtain that

$$\{\varphi(\overset{m}{x} \mid \overset{m}{s}) \mid s_1, \ldots, s_m \in T_k^S\} \cup \{\neg \psi(\overset{n}{y} \mid \overset{n}{t}) \mid t_1, \ldots, t_n \in T_k^S\}$$

is not satisfiable either. By the Compactness Theorem this holds for a finite subset; hence there are $i, j \geq 1$ and terms $s_{11}, \ldots, s_{1m}, \ldots, s_{i1}, \ldots, s_{im}$ and $t_{11}, \ldots, t_{1n}, \ldots, t_{j1}, \ldots, t_{jn} \in T_k^S$ so that

$$\{\varphi(\overset{m}{x} \mid \overset{m}{s_1}), \ldots, \varphi(\overset{m}{x} \mid \overset{m}{s_i})\} \cup \{\neg \psi(\overset{n}{y} \mid \overset{n}{t_1}), \ldots, \neg \psi(\overset{n}{y} \mid \overset{n}{t_j})\}$$

is not satisfiable. Thus we have

$$\varphi(\overset{m}{x} \mid \overset{m}{s_1}) \wedge \ldots \wedge \varphi(\overset{m}{x} \mid \overset{m}{s_i}) \models \psi(\overset{n}{y} \mid \overset{n}{t_1}) \vee \ldots \vee \psi(\overset{n}{y} \mid \overset{n}{t_j}),$$

and therefore (c) holds. □

As special cases of 1.3 and 1.4 we get:

1.5 Corollary. *Let $\forall x_1 \ldots \forall x_n \varphi \in L_k^S$ with φ quantifier-free.*

(a) *The following are equivalent:*
  (i) Sat $\forall x_1 \ldots \forall x_n \varphi$.
  (ii) Sat $\{\varphi(\overset{n}{x} \mid \overset{n}{t}) \mid t_1, \ldots, t_n \in T_k^S\}$.

(b) *The following are equivalent:*
  (i) $\vdash \exists x_1 \ldots \exists x_n \varphi$.
  (ii) *There are $j \geq 1$ and terms $t_{11}, \ldots, t_{1n}, \ldots, t_{j1}, \ldots, t_{jn} \in T_k^S$ with*

$$\vdash \varphi(\overset{n}{x} \mid \overset{n}{t_1}) \vee \ldots \vee \varphi(\overset{n}{x} \mid \overset{n}{t_j}).$$ □

---

[1] Here, e.g., $\overset{n}{t_1}$ stands for $t_{11}, \ldots, t_{1n}$.

In general, the disjunctions in 1.5(b)(ii) and in 1.4 consist of several members (cf. Exercise 1.7). In the next section we present a special but important case in which we may ensure $j = 1$. – The following exercise shows that 1.5(b) does not hold for arbitrary formulas.

1.6 Exercise. Let $S = \{R, c\}$ with unary R and $\varphi = \forall x(Ry \vee \neg Rx)$. Show:
(a) $\vdash \exists y\varphi$.
(b) For $j \geq 1$ and arbitrary $t_1, \ldots, t_j \in T^S$, not $\vdash \varphi(y \mid t_1) \vee \ldots \vee \varphi(y \mid t_j)$.

1.7 Exercise. Show that 1.4 und 1.5 cannot be strengthened by claiming $j = 1$ at the appropriate places.

§2. Free Models and Universal Horn Formulas

Let Φ be a set of formulas. In general, the term interpretation $\mathfrak{I}^\Phi$ is not a model of Φ. (This is why in Chapter V we have enlarged Φ to a negation complete set of formulas containing witnesses.) If, however, $\mathfrak{I}^\Phi$ is a model of Φ then, as we shall see, $\mathfrak{I}^\Phi$ is a distinguished model of Φ, a so-called *free* model. For instance, $\mathfrak{I}^\Phi$ is a model of Φ if Φ consists of atomic formulas (cf. 1.1(b)). The same holds for other sufficiently "simple" sets of formulas which are important in algebra and of central interest in logic programming: for sets of universal Horn formulas. They allow (Theorem 2.7) a positive answer to the question raised at the beginning of this chapter about the existence of satisfying terms.

Throughout, let S be a fixed symbol set.

For a set Φ of S-formulas we have defined the term interpretation $\mathfrak{I}^\Phi = (\mathfrak{T}^\Phi, \beta^\Phi)$ in such a way that an atomic formula φ holds in $\mathfrak{I}^\Phi$ if and only if $\Phi \vdash \varphi$ (cf. 1.1(b)). So, if $R \in S$ is n-ary and if $t_1, \ldots, t_n \in T^S$, we have:

If $\Phi \vdash Rt_1 \ldots t_n$ then $R^{\mathfrak{T}^\Phi}\overline{t_1} \ldots \overline{t_n}$;
if not $\Phi \vdash Rt_1 \ldots t_n$ then not $R^{\mathfrak{T}^\Phi}\overline{t_1} \ldots \overline{t_n}$.

And similarly:

If $\Phi \vdash t_1 \equiv t_2$ then $\overline{t_1} = \overline{t_2}$;
if not $\Phi \vdash t_1 \equiv t_2$ then $\overline{t_1} \neq \overline{t_2}$.

So, if φ is atomic and neither $\Phi \vdash \varphi$ nor $\Phi \vdash \neg\varphi$ then $\mathfrak{I}^\Phi$ is a model of $\neg\varphi$. Therefore, we see that in the definition of $\mathfrak{I}^\Phi$ we have chosen the "positive atomic information" only if it was required by Φ. In this sense $\mathfrak{I}^\Phi$ is a minimal model. From an algebraic point of view the minimality is reflected in the fact that $\mathfrak{I}^\Phi$ is free:

2.1 Theorem. *Let $\mathfrak{I}^\Phi \models \Phi$. Then $\mathfrak{I}^\Phi$ $(= (\mathfrak{T}^\Phi, \beta^\Phi))$ is a* free model *of Φ, i.e. $\mathfrak{I}^\Phi$ is a model of Φ, and if $\mathfrak{I} = (\mathfrak{A}, \beta)$ is another model of Φ then*

$$\pi(\overline{t}) := \mathfrak{I}(t) \quad \textit{for } t \in T^S$$

defines a map from T^Φ *to* A *which is a* homomorphism *from* $\mathfrak{T}^\Phi$ *to* $\mathfrak{A}$, *i.e.*

(i) *for n-ary* $R \in S$ *and* $a_1, \ldots, a_n \in T^\Phi$:

$$\textit{If } R^{\mathfrak{T}^\Phi} a_1 \ldots a_n \textit{ then } R^{\mathfrak{A}} \pi(a_1) \ldots \pi(a_n);$$

(ii) *for n-ary* $f \in S$ *and* $a_1, \ldots, a_n \in T^\Phi$:

$$\pi(f^{\mathfrak{T}^\Phi}(a_1, \ldots, a_n)) = f^{\mathfrak{A}}(\pi(a_1), \ldots, \pi(a_n));$$

(iii) *for* $c \in S$: $\pi(c^{\mathfrak{T}^\Phi}) = c^{\mathfrak{A}}$.

Proof. Assume the hypotheses of the theorem. First we show that π is well defined: If $t, t' \in T^S$ with $\overline{t} = \overline{t'}$ then $\Phi \vdash t \equiv t'$, by $\mathfrak{I} \models \Phi$ therefore $\mathfrak{I}(t) = \mathfrak{I}(t')$. For the proof that π is a homomorphism we only show (i). So let $a_1, \ldots, a_n \in T^\Phi$, say $a_i = \overline{t_i}$ with suitable $t_i \in T^S$ for $1 \leq i \leq n$. Now, if $R^{\mathfrak{T}^\Phi} a_1 \ldots a_n$, i.e. $R^{\mathfrak{T}^\Phi} \overline{t_1} \ldots \overline{t_n}$, then $\Phi \vdash Rt_1 \ldots t_n$. Since $\mathfrak{I} \models \Phi$ we get $\mathfrak{I} \models Rt_1 \ldots t_n$, i.e. $R^{\mathfrak{A}} \mathfrak{I}(t_1) \ldots \mathfrak{I}(t_n)$, and by definition of π finally $R^{\mathfrak{A}} \pi(a_1) \ldots \pi(a_n)$. □

If Φ is a set of S-sentences with $\mathfrak{I}^\Phi \models \Phi$, i.e. $\mathfrak{T}^\Phi \models \Phi$, algebraists call the structure $\mathfrak{T}^\Phi$ a *free model of* Φ *over* $\{\overline{v_n} \mid n \in \mathbf{N}\}$. Similarly, one can show that $\mathfrak{I}_k^\Phi$ is *free over* $\{\overline{v_n} \mid n < k\}$. We do not present the details of the definitions here (however, see Exercise 2.9).

Next, we show that for a set Φ of universal Horn formulas the interpretation $\mathfrak{I}^\Phi$ is a model of Φ. This will lead us to concrete applications of 2.1. We define universal Horn formulas to be formulas which are both universal and Horn formulas (cf. III.4.16). They can be introduced by the following calculus:

2.2 Definition. Formulas which are obtained using the following calculus are called *universal Horn formulas*:

(1) $\dfrac{}{(\neg\varphi_1 \vee \ldots \vee \neg\varphi_n \vee \varphi)}$ if $n \in \mathbf{N}$ and $\varphi_1, \ldots, \varphi_n, \varphi$ are atomic

(2) $\dfrac{}{(\neg\varphi_0 \vee \ldots \vee \neg\varphi_n)}$ if $n \in \mathbf{N}$ and $\varphi_0, \ldots, \varphi_n$ are atomic

(3) $\dfrac{\varphi, \psi}{(\varphi \wedge \psi)}$ (4) $\dfrac{\varphi}{\forall x \varphi}$.

The decisive restriction which distinguishes universal Horn formulas from universal formulas is expressed in (1), allowing only a single unnegated atom as member of the disjunction. Thus $(Pc \vee Pd)$ and $(\neg Px \vee Py \vee x \equiv y)$ are not universal Horn formulas and – as we shall see in 2.8 – not even logically equivalent to universal Horn formulas.

2.3 Lemma. *For* $k \in \mathbf{N}$ *the following holds:*

(a) *Every universal Horn formula in* L_k^S *is logically equivalent to a conjunction of formulas in* L_k^S *of the form*

(H1) $\forall x_1 \ldots \forall x_m \varphi$

(H2) $\forall x_1 \ldots \forall x_m (\varphi_0 \wedge \ldots \wedge \varphi_n \rightarrow \varphi)$

(H3) $\forall x_1 \ldots \forall x_m (\neg\varphi_0 \vee \ldots \vee \neg\varphi_n)$

with atomic φ *and* φ_i.

(b) *Every universal Horn formula in* L_k^S *is logically equivalent to a universal Horn formula from* L_k^S *in prenex normal form.*

(c) *If* φ *is a universal Horn formula and if* $x_1, \ldots, x_n$ *are pairwise distinct then, for* $t_1, \ldots, t_n \in T^S$, $\varphi(\overset{n}{x} \mid \overset{n}{t})$ *is also a universal Horn formula.*

Proof. (a) follows from the fact that for $n \geq 1$ the formula $(\neg\varphi_1 \vee \ldots \vee \neg\varphi_n \vee \varphi)$ is logically equivalent to $(\varphi_1 \wedge \ldots \wedge \varphi_n \rightarrow \varphi)$ and the formula $\forall x(\varphi \wedge \psi)$ logically equivalent to $(\forall x\varphi \wedge \forall x\psi)$. (b) follows similarly, (c) can easily be proved by induction on universal Horn formulas. □

Now we show:

2.4 Theorem. *Let* Φ *be a consistent set of formulas and* ψ *a universal Horn formula with* $\Phi \vdash \psi$. *Then* $\mathfrak{I}^\Phi \models \psi$.

With 2.1 we get:

2.5 Corollary. *Let* Φ *be a consistent set of universal Horn formulas. Then* $\mathfrak{I}^\Phi$ *is a free model of* Φ. □

And with 1.2(c) we conclude:

2.6 Corollary. *Let* S *contain a constant and let* Φ *be a set of universal Horn sentences. Then* $\mathfrak{T}_0^\Phi$ *is a model of* Φ. □

Proof of 2.4. If ψ is atomic, 1.1(b) gives:

$$(*) \qquad \mathfrak{I}^\Phi \models \psi \quad \text{iff} \quad \Phi \vdash \psi.$$

Now we prove the theorem by induction on $\mathrm{rk}(\psi)$ using Definition 2.2.

(1): Let $\psi = (\neg\varphi_1 \vee \ldots \vee \neg\varphi_n \vee \varphi)$ and let $\Phi \vdash \psi$. The case $n = 0$ is covered by $(*)$. Let $n > 0$. We have to show that $\mathfrak{I}^\Phi \models (\varphi_1 \wedge \ldots \wedge \varphi_n \rightarrow \varphi)$. So let $\mathfrak{I}^\Phi \models (\varphi_1 \wedge \ldots \wedge \varphi_n)$. Then $\Phi \vdash \varphi_1, \ldots, \Phi \vdash \varphi_n$ by $(*)$. Since $\Phi \vdash (\varphi_1 \wedge \ldots \wedge \varphi_n \rightarrow \varphi)$, we also have $\Phi \vdash \varphi$ and, again by $(*)$, we get $\mathfrak{I}^\Phi \models \varphi$.

(2): Let $\psi = (\neg\varphi_0 \vee \ldots \vee \neg\varphi_n)$ and let $\Phi \vdash \psi$. Then $\Phi \vdash \neg(\varphi_0 \wedge \ldots \wedge \varphi_n)$. Suppose $\mathfrak{I}^\Phi$ is not a model of $(\neg\varphi_0 \vee \ldots \vee \neg\varphi_n)$. Then $\mathfrak{I}^\Phi \models \varphi_i$ for $i = 0, \ldots, n$, hence $\Phi \vdash \varphi_i$ for $i = 0, \ldots, n$ by $(*)$, i.e. $\Phi \vdash (\varphi_0 \wedge \ldots \wedge \varphi_n)$. Thus Φ is not consistent which contradicts the hypothesis.

(3): For $\psi = (\varphi_1 \wedge \varphi_2)$, where φ_1 and φ_2 are universal Horn formulas, the claim is immediate from the induction hypothesis for φ_1 and φ_2.

(4): Let $\psi = \forall x\varphi$ and $\Phi \vdash \forall x\varphi$. Then $\Phi \vdash \varphi\frac{t}{x}$ for all $t \in T^S$. Since $\varphi\frac{t}{x}$ is a universal Horn formula (cf. 2.3(c)) and since $\mathrm{rk}(\varphi\frac{t}{x}) = \mathrm{rk}(\varphi) < \mathrm{rk}(\psi)$, the

induction hypothesis gives $\mathfrak{I}^\Phi \models \varphi\frac{t}{x}$ for all $t \in T^S$, and 1.1(c)(ii) yields $\mathfrak{I}^\Phi \models \forall x\varphi$. □

As an example we consider the axiom system Φ_{grp} for the class of all groups as $\{\circ, ^{-1}, e\}$-structures (cf. III.5). It consists of universal Horn sentences. Hence, by 2.5, $\mathfrak{T}^{\Phi_{\text{grp}}}$ is a free model, the *free group* over $\{\overline{v_n} \mid n \in \mathbf{N}\}$. If we set $\Phi_{\text{ab}} := \Phi_{\text{grp}} \cup \{\forall x \forall y\, x \circ y \equiv y \circ x\}$, $\mathfrak{T}^{\Phi_{\text{ab}}}$ is the *free abelian group* over $\{\overline{v_n} \mid n \in \mathbf{N}\}$.

Sentences of the form $\forall x_1 \ldots \forall x_r\, t_1 \equiv t_2$ are also called *equations*. So equations are universal Horn sentences. The axioms of Φ_{grp} and Φ_{ab} are equations. Many classes of structures studied in algebra can be axiomatized by equations and therefore have free models (see also Exercise 2.10).

For the axiom system Φ_{grp} we have $\Phi_{\text{grp}} \vdash \exists z\, z \circ x \equiv y$. A "solution" is provided by $y \circ x^{-1}$ (a term in the free variables of $\exists z\, z \circ x \equiv y$). An analogous fact holds in general; it is contained in the following strengthening of 1.4:

2.7 Theorem. *Let $k \in \mathbf{N}$ and S contain a constant in case $k = 0$. Furthermore, let $\Phi \subset L_k^S$ be a consistent set of universal Horn formulas. Then the following are equivalent for every formula in L_k^S of the form $\exists x_1 \ldots \exists x_n(\psi_0 \wedge \ldots \wedge \psi_l)$ with atomic $\psi_0, \ldots, \psi_l$:*

(i) $\Phi \vdash \exists x_1 \ldots \exists x_n(\psi_0 \wedge \ldots \wedge \psi_l)$.

(ii) $\mathfrak{I}_k^\Phi \models \exists x_1 \ldots \exists x_n(\psi_0 \wedge \ldots \wedge \psi_l)$.

(iii) *There are $t_1, \ldots, t_n \in T_k^S$ with $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{n}{x} \mid \overset{n}{t})$.*

Proof. Obviously, (iii) implies (i) and (i) implies (ii). We show how to obtain (iii) from (ii). Let $\mathfrak{I}_k^\Phi \models \exists x_1 \ldots \exists x_n(\psi_0 \wedge \ldots \wedge \psi_l)$, i.e. for suitable terms $t_1, \ldots, t_n \in T_k^S$ we have $\mathfrak{I}_k^\Phi \models (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{n}{x} \mid \overset{n}{t})$. Since $(\psi_0 \wedge \ldots \wedge \psi_l)(\overset{n}{x} \mid \overset{n}{t})$ is a quantifier-free formula from L_k^S, 1.2(b) yields $\mathfrak{I}^\Phi \models (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{n}{x} \mid \overset{n}{t})$. Therefore $\mathfrak{I}^\Phi \models \psi_i(\overset{n}{x} \mid \overset{n}{t})$ for $i = 0, \ldots, l$, and as the ψ_i are atomic we get $\Phi \vdash \psi_i(\overset{n}{x} \mid \overset{n}{t})$, and so altogether $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{n}{x} \mid \overset{n}{t})$. □

If in part (i) we replace the derivation relation $\vdash$ by the consequence relation $\models$, we see that the validity of $\Phi \models \exists x_1 \ldots \exists x_n(\psi_0 \wedge \ldots \wedge \psi_l)$ can be checked by a *single* interpretation, namely $\mathfrak{I}_k^\Phi$.

In mathematics and its applications one is usually interested not only in the derivation of an existential formula but also in the presentation of concrete terms satisfying it. In view of the formal character of the sequent calculus we see that in the cases covered by Theorem 2.7 it is possible to find concrete solutions in a systematic way. Thus one can think of a programming language where, for a given problem, a programmer only has to formalize in first-order language the hypotheses (as universal Horn formulas) and the

"query" (as an existential formula); then, by systematically applying the sequent calculus, the computer searches for terms satisfying the existential formula, i.e. solving the given problem. The area in which this approach is pursued is called *logic programming,* the most popular programming language in this context being PROLOG (Programming in Logic).

The central idea in this subject is often expressed by the following equation:

$$\textit{algorithm} = \textit{logic} + \textit{control}$$

"Logic" here refers to the static (the *declarative*) aspects of the problem, e.g. its adequate formalization. "Control" stands for the part concerned with the strategies for applying rules of derivation which therefore characterizes the dynamic (the *procedural*) aspect.

We shall deal with the fundamentals of logic programming in Sections 6 and 7. In Sections 4 and 5 we consider rules of derivation which are more suitable for logic programming than the rules of the sequent calculus that primarily follow the proof patterns used by mathematicians. In many concrete applications the equality symbol does not appear in the formalizations. This will simplify the exposition. The next section contains some preliminary results for equality-free formulas.

2.8 Exercise. Let $S := \{P, c, d\}$ with unary P and $\Phi := \{(Pc \vee Pd)\}$. Show that not $\mathfrak{I}^{\Phi} \models \Phi$ and conclude that $(Pc \vee Pd)$ is not logically equivalent to a universal Horn sentence. Using III.4.16, show that it is not even logically equivalent to a Horn sentence. Prove this last statement also for $(\neg Px \vee Py \vee x \equiv y)$.

2.9 Exercise. Show: Every at most countable group $\mathfrak{G}$ (as $\{\circ, ^{-1}, e\}$-structure) is a homomorphic image of $\mathfrak{T}^{\Phi_{\mathrm{grp}}}$ (i.e. there is a homomorphism from $\mathfrak{T}^{\Phi_{\mathrm{grp}}}$ onto $\mathfrak{G}$). Similarly, show that for $k \in \mathbf{N}$ every group $\mathfrak{G}$ generated by at most k elements is a homomorphic image of $\mathfrak{T}_k^{\Phi_{\mathrm{grp}}}$.

2.10 Exercise. Let $\Phi := \{\forall x_1 \ldots \forall x_{n_i}\, t_i \equiv t_i' \mid i \in \mathbf{N}\}$ be a set of equations in the language $L^{S_{\mathrm{grp}}}$ of group theory. Show:
(a) $\Phi_{\mathrm{grp}} \cup \Phi$ is satisfiable.
(b) $\mathfrak{T}^{\Phi_{\mathrm{grp}} \cup \Phi}$ is a model of $\Phi_{\mathrm{grp}} \cup \Phi$, the so-called *free group over* $\{\overline{v_n} \mid n \in \mathbf{N}\}$ *with defining relations* $t_i \equiv t_i'$ $(i \in \mathbf{N})$.
(c) $\{\overline{t} \mid t \in T^S$ and $\Phi_{\mathrm{grp}} \cup \Phi \vdash t \equiv e\}$ is the universe of a normal subgroup $\mathfrak{U}$ of $\mathfrak{T}^{\Phi_{\mathrm{grp}}}$ (the equivalence classes are taken with respect to Φ_{grp}). We have $\mathfrak{T}^{\Phi_{\mathrm{grp}} \cup \Phi} \cong \mathfrak{T}^{\Phi_{\mathrm{grp}}}/_{\mathfrak{U}}$.

§3. Herbrand Structures

A formula is called *equality-free* if the equality symbol does not occur in it. Our first goal is to show that no non-trivial equations are derivable from equality-free formulas. This allows us to present the term interpretations $\mathfrak{I}^{\Phi}$ in an especially simple form in case Φ consists of equality-free formulas.

3.1 Theorem. *If Φ is a consistent set of equality-free S-formulas then the following holds for all terms $t_1, t_2 \in T^S$:*

$$(*) \qquad \textit{If} \quad \Phi \vdash t_1 \equiv t_2 \quad \textit{then } t_1 = t_2.$$

The crucial part in the proof is the following lemma:

3.2 Lemma. *For an S-interpretation $\mathfrak{I} = (\mathfrak{A}, \beta)$ let $\mathfrak{I}' = (\mathfrak{A}', \beta')$ be the S-interpretation given by*

(1) $A' := T^S$;
(2) *for n-ary $f \in S$ and $t_1, \ldots, t_n \in T^S$:*

$$f^{\mathfrak{A}'}(t_1, \ldots, t_n) := f t_1 \ldots t_n;$$

(3) *for $c \in S$:* $c^{\mathfrak{A}'} := c$;
(4) *for n-ary $R \in S$ and $t_1, \ldots, t_n \in T^S$:*

$$R^{\mathfrak{A}'} t_1 \ldots t_n \quad \textit{:iff} \quad R^{\mathfrak{A}} \mathfrak{I}(t_1) \ldots \mathfrak{I}(t_n);$$

(5) $\beta'(x) := x$ *for all variables x.*

Then the following holds:

(i) *for all $t \in T^S$:* $\mathfrak{I}'(t) = t$;
(ii) *for all universal and equality-free formulas $\psi \in L^S$:*

$$\textit{If } \mathfrak{I} \models \psi \textit{ then } \mathfrak{I}' \models \psi.$$

Proof of 3.2. Part (i) follows immediately from the definitions. Every equality-free atomic formula φ is of the form $Rt_1 \ldots t_n$; so by (4) we have

$$\mathfrak{I}' \models \varphi \quad \text{iff} \quad \mathfrak{I} \models \varphi.$$

Now we can show the implication in (ii) by induction on $\mathrm{rk}(\psi)$. For $\psi = \forall x \varphi$ for example, we argue as follows: If $\mathfrak{I} \models \forall x \varphi$ then for all $t \in T^S$ we have $\mathfrak{I}\frac{\mathfrak{I}(t)}{x} \models \varphi$, hence $\mathfrak{I} \models \varphi \frac{t}{x}$, so by induction hypothesis $\mathfrak{I}' \models \varphi \frac{t}{x}$ (note that $\mathrm{rk}(\varphi \frac{t}{x}) < \mathrm{rk}(\psi)$). Since $\mathfrak{I}'(t) = t$ we have $\mathfrak{I}' \frac{t}{x} \models \varphi$. Therefore $\mathfrak{I}' \frac{t}{x} \models \varphi$ holds for all $t \in T^S$ $(= A')$, and so $\mathfrak{I}' \models \forall x \varphi$. □

Proof of 3.1. Suppose Φ satisfies the hypotheses of the theorem. Furthermore let $\Phi \vdash t_1 \equiv t_2$.

First, we consider the case where Φ consists of universal formulas and choose a model $\mathfrak{I}$ of Φ. Then, by 3.2(ii), we have $\mathfrak{I}' \models \Phi$. Since $\Phi \vdash t_1 \equiv t_2$

it follows that $\mathfrak{I}' \models t_1 \equiv t_2$, and therefore $t_1 = \mathfrak{I}'(t_1) = \mathfrak{I}'(t_2) = t_2$ (cf. 3.2(i)).

In the general case, applying the Compactness Theorem, we first replace Φ by a finite subset Φ_0 with $\Phi_0 \vdash t_1 \equiv t_2$. Let φ_0 be the conjunction of the formulas from Φ_0. Then φ_0 is satisfiable and equality-free, and we have $\varphi_0 \vdash t_1 \equiv t_2$. By the Theorem on the Skolem Normal Form (cf. VIII.4.5 and the proof given there) there is a satisfiable, universal, equality-free ψ with $\psi \vdash \varphi_0$. By $\varphi_0 \vdash t_1 \equiv t_2$ we therefore have $\psi \vdash t_1 \equiv t_2$. So, by the case of universal formulas already considered, $t_1 = t_2$ holds. □

Now let Φ be consistent and equality-free. For the equivalence relation

$$t_1 \sim t_2 \quad \text{iff} \quad \Phi \vdash t_1 \equiv t_2$$

on T^S, given by Φ, the previous theorem yields

$$t_1 \sim t_2 \quad \text{iff} \quad t_1 = t_2.$$

So $\overline{t} = \{t\}$. For simplicity we identify $\overline{t}$ and t and get:

3.3 Remark. *Let Φ be a consistent set of equality-free S-formulas. Then the following holds for the term interpretation $\mathfrak{I}^\Phi = (\mathfrak{T}^\Phi, \beta^\Phi)$:*

(a) $T^\Phi = T^S$.

(b) *For n-ary $f \in S$ and $t_1, \ldots, t_n \in T^S$:*

$$f^{\mathfrak{T}^\Phi}(t_1, \ldots, t_n) = f t_1 \ldots t_n.$$

(c) *For $c \in S$, $c^{\mathfrak{T}^\Phi} = c$.*

(d) *For n-ary $R \in S$ and $t_1, \ldots, t_n \in T^S$:*

$$R^{\mathfrak{T}^\Phi} t_1 \ldots t_n \quad \textit{iff} \quad \Phi \vdash R t_1 \ldots t_n.$$

(e) *For every variable x: $\beta^\Phi(x) = x$.* □

We now consider the case where Φ is a set of *sentences*, assuming throughout that S contains a constant. The substructure $\mathfrak{T}_0^\Phi$ of $\mathfrak{T}^\Phi$ from 3.3, consisting of variable-free terms, is a Herbrand structure in the following sense.

3.4 Definition. An S-structure $\mathfrak{A}$ is called *Herbrand structure* :iff

(i) $A = T_0^S$.
(ii) For n-ary $f \in S$ and $t_1, \ldots, t_n \in T^S$, $f^{\mathfrak{A}}(t_1, \ldots, t_n) = f t_1 \ldots t_n$.
(iii) For $c \in S$, $c^{\mathfrak{A}} = c$.

We note:

3.5 Remark. *For a consistent set Φ of equality-free sentences, $\mathfrak{T}_0^\Phi$ is a Herbrand structure.* □

3.6 Remark. *For a Herbrand structure $\mathfrak{A}$ and $t \in T_0^S$ we have $t^{\mathfrak{A}} = t$.* □

For a Herbrand structure the interpretation of the function symbols and constants is fixed. However, Definition 3.4 says nothing about the interpretation of the relation symbols; it can be chosen "freely."

3.7 Theorem. *Let Φ be a satisfiable set of universal and equality-free sentences. Then Φ has a Herbrand model, i.e. a model which is a Herbrand structure.*

Proof. Let $\mathfrak{I} = (\mathfrak{A}, \beta)$ be an interpretation with $\mathfrak{I} \models \Phi$. For the corresponding interpretation $\mathfrak{I}' = (\mathfrak{A}', \beta')$ (see 3.2) we have that $\mathfrak{I}' \models \Phi$ and therefore $\mathfrak{A}' \models \Phi$. By definition of $\mathfrak{A}'$, T_0^S is the universe of a substructure $\mathfrak{B}'$ of $\mathfrak{A}'$. $\mathfrak{B}'$ is a Herbrand structure and also a model of Φ as Φ consists of universal sentences. □

The minimality of the term structure mentioned in the previous section (before 2.1) is reflected in the following characterization of $\mathfrak{T}_0^\Phi$.

3.8 Theorem. *Let Φ be a consistent set of universal and equality-free Horn sentences. Then the following holds:*
(a) *$\mathfrak{T}_0^\Phi$ is a Herbrand model of Φ.*
(b) *For every Herbrand model $\mathfrak{A}$ of Φ and every n-ary $R \in S$, $R^{\mathfrak{T}_0^\Phi} \subset R^{\mathfrak{A}}$.*

Therefore $\mathfrak{T}_0^\Phi$ is called the *minimal Herbrand model* of Φ.

Proof. (a): $\mathfrak{T}_0^\Phi$ is a Herbrand structure (cf. 3.5) and a model of Φ (cf. 2.6).

(b): Let $\mathfrak{A}$ be a Herbrand model of Φ and let $R \in S$ be n-ary. For $t_1, \ldots, t_n \in T_0^S$ $(= A)$ we have by definition (cf. 3.3(d)):

$$R^{\mathfrak{T}_0^\Phi} t_1 \ldots t_n \quad \text{iff} \quad \Phi \vdash R t_1 \ldots t_n.$$

Since $\mathfrak{A} \models \Phi$ it follows with $R^{\mathfrak{T}_0^\Phi} t_1 \ldots t_n$ that $\mathfrak{A} \models R t_1 \ldots t_n$, i.e. $R^{\mathfrak{A}} t_1 \ldots t_n$. □

We finish this section by restating Theorem 2.7 in terms of the Herbrand structure $\mathfrak{T}_0^\Phi$:

3.9 Theorem. *Let Φ be a consistent set of equality-free universal Horn sentences. Then the following are equivalent for every Horn sentence of the form $\exists x_1 \ldots \exists x_n (\psi_0 \wedge \ldots \wedge \psi_l)$ with atomic $\psi_0, \ldots, \psi_l$:*
(i) $\Phi \vdash \exists x_1 \ldots \exists x_n (\psi_0 \wedge \ldots \wedge \psi_l)$.
(ii) $\mathfrak{T}_0^\Phi \models \exists x_1 \ldots \exists x_n (\psi_0 \wedge \ldots \wedge \psi_l)$.
(iii) *There are $t_1, \ldots, t_n \in T_0^S$ with $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{n}{x} \mid \overset{n}{t})$.* □

§4. Propositional Logic

In propositional logic we consider formulas which are built up from atoms, the so-called *propositional variables* , only using connectives. The proposi-

tional variables are interpreted by the truth-values T (for "true") and F (for "false") (cf. III.2).

4.1 Definition. Let $\mathcal{A}_a$ be the alphabet $\{\neg, \vee,), (\} \cup \{p_0, p_1, p_2, \ldots\}$. We define the *formulas of the language of propositional logic* (the *propositional formulas*) to be the strings over $\mathcal{A}_a$ which are obtained by means of the following rules:

$$\frac{}{p_i} \quad (i \in \mathbf{N}), \quad \frac{\alpha}{\neg\alpha}, \quad \frac{\alpha, \beta}{(\alpha \vee \beta)}.$$

For propositional variables we often use the letters $p, q, r, \ldots$, for propositional formulas the letters $\alpha, \beta, \ldots$. PF stands for the set of propositional formulas. For $\alpha \in PF$ let $\mathrm{pvar}(\alpha)$ be the set of propositional variables occurring in α,

$$\mathrm{pvar}(\alpha) := \{p \mid p \text{ occurs in } \alpha\}.$$

Furthermore for $n \geq 1$ we set

$$PF_n := \{\alpha \in PF \mid \mathrm{pvar}(\alpha) \subset \{p_0, \ldots, p_{n-1}\}\}.$$

We call a map $b\colon \{p_i \mid i \in \mathbf{N}\} \to \{T, F\}$ a (propositional) *assignment*. The other semantic notions are defined as in the first-order case:

The truth-value $\alpha[b]$ of a propositional formula α under the assignment b is defined inductively by[2]

$$\begin{aligned} p_i[b] &:= b(p_i) \\ \neg\alpha[b] &:= \dot{\neg}(\alpha[b]) \\ (\alpha \vee \beta)[b] &:= \dot{\vee}(\alpha[b], \beta[b]) \end{aligned}$$

(cf. III.2 for the definition of $\dot{\neg}$ and $\dot{\vee}$). If $\alpha[b] = T$ we say that b is a *model* of α or *satisfies* α. The assignment b is a *model* of the set of formulas $\Delta \subset PF$ if b is a model of each formula in Δ.

Similar to the Coincidence Lemma of first-order logic, the truth-value $\alpha[b]$ depends only on the assignment of the propositional variables occurring in the formula α:

4.2 Coincidence Lemma of Propositional Logic. *Let α be a propositional formula and let b and b' be assignments with $b(p) = b'(p)$ for all $p \in \mathrm{pvar}(\alpha)$. Then $\alpha[b] = \alpha[b']$.*

The easy proof is left to the reader. □

By the Coincidence Lemma, for $\alpha \in PF_{n+1}$ and $b_0, \ldots, b_n \in \{T, F\}$ it makes sense to write

$$\alpha[b_0, \ldots, b_n]$$

[2]Inductive proofs and definitions on propositional formulas can be justified as those for first-order logic in II.4.

for the truth-value $\alpha[b]$ where b is any assignment for which $b(p_i) = b_i$ for $i \leq n$.

We say:

- α *is a consequence of* Δ (written: $\Delta \models \alpha$) :iff every model of Δ is a model of α;
- α is *valid* (written: $\models \alpha$) :iff α holds under all assignments;
- Δ is *satisfiable* (written: Sat Δ) :iff there is an assignment which is a model of Δ;
- α is *satisfiable* (written: Sat α) :iff Sat $\{\alpha\}$;
- α and β are *logically equivalent* :iff $\models (\alpha \leftrightarrow \beta)$.

Again, $(\alpha \wedge \beta), (\alpha \rightarrow \beta)$, and $(\alpha \leftrightarrow \beta)$ are abbreviations for $\neg(\neg\alpha \vee \neg\beta)$, $(\neg\alpha \vee \beta)$, and $(\neg(\alpha \vee \beta) \vee \neg(\neg\alpha \vee \neg\beta))$, respectively.

Some essential aspects of logic programming can better be explained on the level of propositional logic; we will do so in the next section. The results obtained there have to be transferred to first-order language. Let us consider a technique for such a transfer. It is based on the intuitively evident fact that an equality-free formula such as $((Rxy \wedge Ryfx) \vee (\neg Rzz \wedge Rxy))$ has the "same models" as the propositional formula $((p_0 \wedge p_1) \vee (\neg p_2 \wedge p_0))$.

Let S be an at most countable symbol set containing at least one relation symbol. Then the set

$$A^S := \{Rt_1 \ldots t_n \mid R \in S \text{ } n\text{-ary}, t_1, \ldots, t_n \in T^S\}$$

of equality-free atomic S-formulas is countable. Furthermore let

$$\pi_0 \colon A^S \rightarrow \{p_i \mid i \in \mathbf{N}\}$$

be a bijection. We extend π_0 to a map π which is defined on the set of S-formulas which are both equality-free and quantifier-free, by setting:

$$\begin{aligned} \pi(\varphi) &:= \pi_0(\varphi) \text{ for } \varphi \in A^S \\ \pi(\neg\varphi) &:= \neg\pi(\varphi) \\ \pi(\varphi \vee \psi) &:= (\pi(\varphi) \vee \pi(\psi)). \end{aligned}$$

Then the following holds:

4.3. *The map $\varphi \mapsto \pi(\varphi)$ is a bijection from the set of equality-free and quantifier-free S-formulas onto PF.*

Proof. We define a map $\rho \colon PF \rightarrow L^S$ by

$$\begin{aligned} \rho(p) &:= \pi_0^{-1}(p) \\ \rho(\neg\alpha) &:= \neg\rho(\alpha) \\ \rho(\alpha \vee \beta) &:= (\rho(\alpha) \vee \rho(\beta)). \end{aligned}$$

By induction on φ and α, respectively, one can easily show:

$\rho(\pi(\varphi)) = \varphi$ for equality-free and quantifier-free φ,
$\pi(\rho(\alpha)) = \alpha$ for $\alpha \in PF$.

Hence π is a bijection and $\rho = \pi^{-1}$. □

4.4. *If $\Phi \cup \{\varphi, \psi\}$ is a set of equality-free and quantifier-free S-formulas then the following holds:*
(a) Sat Φ *iff* Sat $\pi(\Phi)$.
(b) $\Phi \models \varphi$ *iff* $\pi(\Phi) \models \pi(\varphi)$.
(c) *φ and ψ are logically equivalent iff $\pi(\varphi)$ and $\pi(\psi)$ are logically equivalent.*

Proof. Since (b) follows immediately from (a) and (c) from (b) we only have to show (a). For the implication from left to right let $\mathfrak{I}$ be an S-interpretation with $\mathfrak{I} \models \Phi$. We define a propositional assignment b by

$$b(p_i) = \begin{cases} T & \text{if } \mathfrak{I} \models \rho(p_i) \\ F & \text{otherwise} \end{cases}$$

for $i \in \mathbf{N}$. Using induction on propositional formulas one can easily show that for all $\alpha \in PF$

$$\alpha[b] = T \quad \text{iff} \quad \mathfrak{I} \models \rho(\alpha).$$

Since $\mathfrak{I} \models \Phi$ the assignment b is a model of $\pi(\Phi)$.

For the other direction, let $\pi(\Phi)$ be satisfiable and b be a model of $\pi(\Phi)$. It suffices to find an S-interpretation $\mathfrak{I}$ with

$$(*) \qquad \mathfrak{I} \models \varphi \quad \text{iff} \quad \pi(\varphi)[b] = T$$

for all $\varphi \in A^S$. Then a proof by induction shows that $\mathfrak{I} \models \Phi$. We define $\mathfrak{I} = (\mathfrak{A}, \beta)$ by (cf. 3.2):

$A := T^S$;

$f^{\mathfrak{A}}(t_1, \ldots, t_n) := ft_1 \ldots t_n$ for n-ary $f \in S$ and $t_1, \ldots, t_n \in T^S$;

$c^{\mathfrak{A}} := c$ for $c \in S$;

$\beta(x) := x$;

and for n-ary $P \in S$ and $t_1, \ldots, t_n \in T^S$:

$P^{\mathfrak{A}} t_1 \ldots t_n \quad :\text{iff} \quad \pi(Pt_1 \ldots t_n)[b] = T.$

Then obviously $(*)$ holds. □

Proposition 4.4 depends essentially on the fact that the equality symbol does not occur in Φ. If we drop this hypothesis we get a counterexample to 4.4(a) taking a unary relation symbol P and setting $\pi(Pv_0) = p_0$, $\pi(Pv_1) = p_1$, $\pi(v_0 \equiv v_1) = p_2$ and $\Phi = \{Pv_0, \neg Pv_1, v_0 \equiv v_1\}$.

In addition, we can use the connection built in 4.4 to transfer properties of first-order logic to propositional logic. We show this for the Compactness Theorem (for a purely propositional proof see Exercise 4.11).

4.5 Compactness Theorem for Propositional Logic. *A set of propositional formulas is satisfiable if and only if each of its finite subsets is satisfiable.*

Proof. We set $S := \{P\}$ with unary P and define π_0 on $A^S = \{Pv_i \mid i \in \mathbf{N}\}$ by $\pi_0(Pv_i) := p_i$ for $i \in \mathbf{N}$. Then the following holds for arbitrary $\Delta \subset PF$:

Sat Δ iff Sat $\pi^{-1}(\Delta)$ (by 4.4(a))
iff for every finite subset Φ_0 of $\pi^{-1}(\Delta)$, Sat Φ_0
(by the Compactness Theorem for first-order logic)
iff for every finite subset Δ_0 of Δ, Sat Δ_0
(by 4.4(a)). □

In a similar way, Exercise 4.10 should encourage the reader to transfer the Theorems on the Disjunctive and on the Conjunctive Normal Form to propositional logic. A propositional formula is in *disjunctive normal form* (written: in DNF), if it is a disjunction of conjunctions of propositional variables or negated propositional variables; it is in *conjunctive normal form* (written: in CNF), if it is a conjunction of disjunctions of propositional variables or negated propositional variables. For example, the formulas

$$(p \vee q \vee (\neg r \wedge q \wedge \neg p)) \text{ and } ((\neg p \wedge r) \vee (q \wedge \neg r \wedge \neg q) \vee r)$$

are in disjunctive normal form, and the formula

$$((p \vee \neg r) \wedge (\neg q \vee r \vee q))$$

is in conjunctive normal form.

We prove the Theorems on the Disjunctive and on the Conjunctive Normal Form for propositional logic directly. We do so by discussing the question raised in III.2 and showing that every extensional connective can be defined by means of $\neg$ and $\vee$ within propositional logic.

The connective "and" is defined by the formula $\alpha := \neg(\neg p_0 \vee \neg p_1)$ (and hence by $\neg$ and $\vee$) in the sense that

$$\text{for all } b_0, b_1 \in \{T, F\}: \quad \dot{\wedge}(b_0, b_1) = \alpha[b_0, b_1].$$

The same is true for every extensional connective:

4.6 Theorem. *Let $n \geq 0$. For every truth-function h: $\{T, F\}^{n+1} \to \{T, F\}$ there is a formula $\alpha \in PF_{n+1}$ such that*

$$h(b_0, \ldots, b_n) = \alpha[b_0, \ldots, b_n] \quad \textit{for all } b_0, \ldots, b_n \in \{T, F\}.$$

The formula can be chosen to be in DNF *or in* CNF *– as desired –.*

Proof. First, we explain the idea of the proof for the example of the binary truth-function h with the truth-table

		h
T	T	F
T	F	T
F	T	F
F	F	T

We get a formula in DNF defining h as follows: The second and the fourth row of the table give the truth-value T; the arguments there are described by the conjunctions $(p_0 \wedge \neg p_1)$ and $(\neg p_0 \wedge \neg p_1)$, respectively. Their disjunction

$$(p_0 \wedge \neg p_1) \vee (\neg p_0 \wedge \neg p_1)$$

is a formula in DNF defining h.

The first and the third row of the table give the truth-value F; the formulas $(\neg p_0 \vee \neg p_1)$ and $(p_0 \vee \neg p_1)$ say that these arguments are excluded. Their conjunction

$$(\neg p_0 \vee \neg p_1) \wedge (p_0 \vee \neg p_1)$$

is a formula in CNF defining h.

Now let $h\colon \{T,F\}^{n+1} \to \{T,F\}$ be an arbitrary truth-function. We set $-T := F$ and $-F := T$. For a propositional variable p let $p^T := p$ and $p^F := \neg p$. Finally, for "arguments" $b_0, \ldots, b_n \in \{T,F\}$ let

$\alpha^{b_0,\ldots,b_n} := p_0^{b_0} \wedge \ldots \wedge p_n^{b_n}$

("we are in the row with the arguments $b_0, \ldots, b_n$"),

$\beta^{b_0,\ldots,b_n} := p_0^{-b_0} \vee \ldots \vee p_n^{-b_n}$

("we are not in the row with the arguments $b_0, \ldots, b_n$").

Then the following holds for all $b'_0, \ldots, b'_n \in \{T,F\}$:

(1) $\alpha^{b_0,\ldots,b_n}[b'_0, \ldots, b'_n] = T$ iff $b_0 = b'_0$ and $\ldots$ and $b_n = b'_n$

and

(2) $\beta^{b_0,\ldots,b_n}[b'_0, \ldots, b'_n] = T$ iff $b_0 \neq b'_0$ or $\ldots$ or $b_n \neq b'_n$.

The following formulas α_{D} and α_{C} in DNF and CNF, respectively, define h:

$$\alpha_{\mathrm{D}} := \begin{cases} p_0 \wedge \neg p_0, & \text{if } h(b_0, \ldots, b_n) = F \text{ for all } b_0, \ldots, b_n \in \{T,F\}, \\ \bigvee\{\alpha^{b_0,\ldots,b_n} \mid b_0, \ldots, b_n \in \{T,F\}, \quad h(b_0, \ldots, b_n) = T\}, & \\ & \text{otherwise;} \end{cases}$$

$$\alpha_{\mathrm{C}} := \begin{cases} p_0 \vee \neg p_0, & \text{if } h(b_0, \ldots, b_n) = T \text{ for all } b_0, \ldots, b_n \in \{T,F\}, \\ \bigwedge\{\beta^{b_0,\ldots,b_n} \mid b_0, \ldots, b_n \in \{T,F\}, \quad h(b_0, \ldots, b_n) = F\}, & \\ & \text{otherwise.} \end{cases}$$

We show this for the formula α_{D}, i.e. we prove:

$$\text{for all } b_0, \ldots, b_n \in \{T,F\}, \;\; h(b_0, \ldots, b_n) = \alpha_{\mathrm{D}}[b_0, \ldots, b_n].$$

If $h(b_0, \ldots, b_n) = T$ then $\alpha^{b_0,\ldots,b_n}$ is a member of the disjunction α_{D}. By (1) we have $\alpha^{b_0,\ldots,b_n}[b_0, \ldots, b_n] = T$, therefore $\alpha_{\mathrm{D}}[b_0, \ldots, b_n] = T$. Conversely, if $\alpha_{\mathrm{D}}[b_0, \ldots, b_n] = T$ then (by definition of α_{D}) there are truth-values $b'_0, \ldots, b'_n \in \{T, F\}$ with $h(b'_0, \ldots, b'_n) = T$ and $\alpha^{b'_0,\ldots,b'_n}[b_0, \ldots, b_n] = T$. By (1) it follows that $b'_0 = b_0, \ldots, b'_n = b_n$ and so $h(b_0, \ldots, b_n) = T$. □

As a corollary we easily obtain:

4.7 Theorem on the Disjunctive and on the Conjunctive Normal Form. *Every propositional formula is logically equivalent to a formula in disjunctive normal form and to a formula in conjunctive normal form.*

Proof. Let α be a propositional formula in PF_{n+1}. We choose the truth-function $h\colon \{T, F\}^{n+1} \to \{T, F\}$ with $h(b_0, \ldots, b_n) = \alpha[b_0, \ldots, b_n]$ for $b_0, \ldots, b_n \in \{T, F\}$. By 4.6 there are a formula in DNF and a formula in CNF, each of which defines h and hence is logically equivalent to α. □

4.8 Corollary. *For $n \geq 0$ there are exactly $2^{(2^{n+1})}$ pairwise non logically equivalent formulas in PF_{n+1}.*

Proof. Two formulas α and β in PF_{n+1} are logically equivalent if and only if $\alpha[b_0, \ldots, b_n] = \beta[b_0, \ldots, b_n]$ for all $b_0, \ldots, b_n \in \{T, F\}$, i.e. if they define the same $(n+1)$-ary truth-function. By 4.6 the number of pairwise non logically equivalent formulas in PF_{n+1} is equal to the number of truth-functions $h\colon \{T, F\}^{n+1} \to \{T, F\}$, hence equal to $2^{(2^{n+1})}$. □

4.9 Exercise. In 4.6 we have shown that every truth-function can be defined with $\dot{\neg}$ and $\dot{\vee}$. Prove the corresponding statement if $\dot{\neg}$ and $\dot{\vee}$ are replaced by
(a) $\dot{\neg}$ and $\dot{\wedge}$;
(b) $\dot{|}\colon \{T, F\} \times \{T, F\} \to \{T, F\}$ with truth-table

		$\dot{\vert}$
T	T	F
T	F	T
F	T	T
F	F	T

We say that the sets $\{\dot{\neg}, \dot{\vee}\}$, $\{\dot{\neg}, \dot{\wedge}\}$, $\{\dot{|}\}$ are *functionally complete.*

4.10 Exercise. Transfer the theorems about DNF and CNF of first-order logic to propositional logic using 4.4.

4.11 Exercise. Prove the Compactness Theorem of propositional logic directly. Hint: Let $\Delta \subset PF$, and assume that every finite subset of Δ is satisfiable. Call a sequence $(b_0, \ldots, b_n)$ of truth-values *good* if every finite subset of Δ has a model b with $b(p_i) = b_i$ for $i \leq n$. Show that there are arbitrarily long good sequences and infer the existence of an assignment b satisfying every finite subset of Δ, and hence Δ itself.

4.12 Exercise. Let the sequent calculus $\mathfrak{S}_a$ of propositional logic consist of the rules analogous to (Assm), (Ant), (PC), (Ctr), ($\vee$A) and ($\vee$S). For the resulting derivation relation $\vdash_a$ of propositional logic show the following Adequacy Theorem: For all $\Delta \subset PF$ and all $\alpha \in PF$: $\Delta \vdash_a \alpha$ iff $\Delta \models \alpha$.

§5. Propositional Resolution

In this section we study techniques for "quickly" testing whether certain propositional formulas are satisfiable. Partly these techniques are preliminary versions of methods in logic programming to be considered in the next section.

If we want to test whether $\alpha \in PF_{n+1}$ is satisfiable using the definition of the relation of satisfaction, in the worst case we have to calculate the truth-value $\alpha[b_0, \ldots, b_n]$ for 2^{n+1} tuples $(b_0, \ldots, b_n) \in \{T, F\}^{n+1}$. For $n = 5, 10, 20$ these are already 64, 2048, 2 097 152 tuples. It is still open whether there are "better" algorithms which work faster. As we mentioned in X.3, the following question is equivalent to the "$\mathcal{P} = \mathcal{NP}$"-problem of theoretical computer science: Is it possible to test the satisfiability of propositional formulas with a register program which, for suitable $k \in \mathbf{N}$, gives the answer for formulas of length $\leq n$ in at most n^k steps?

For subclasses of formulas one can give fast algorithms. For instance, one can easily test the satisfiability of formulas in DNF: For $\alpha = (\beta_0 \vee \ldots \vee \beta_r)$, α is satisfiable if and only if for some i with $0 \leq i \leq r$ the formula β_i is satisfiable. For $\beta_i = (\lambda_0 \wedge \ldots \wedge \lambda_s)$, where the λ_j are propositional variables or negations of propositional variables, β_i is satisfiable if and only if for no propositional variable p, both p and $\neg p$ occur among $\lambda_0, \ldots, \lambda_s$.

Since a formula α is valid if and only if $\neg\alpha$ is not satisfiable, every algorithm for proving satisfiability gives an algorithm for proving validity. Furthermore, for each α in CNF or in DNF one can immediately give a DNF or CNF, respectively, for $\neg\alpha$. For instance, for the formula in CNF

$$\alpha = (p \vee \neg q \vee r) \wedge (\neg p \vee s \vee t \vee r) \wedge q,$$

the negation $\neg\alpha$ is logically equivalent to

$$(\neg p \wedge q \wedge \neg r) \vee (p \wedge \neg s \wedge \neg t \wedge \neg r) \vee \neg q.$$

Therefore the fast test for satisfiability of formulas in DNF mentioned above gives a fast test for validity of formulas in CNF .

We now show that there is a fast test for satisfiability also of special formulas in CNF, the so-called propositional Horn formulas. For the following definition, see III.4.16 or 2.2.

5.1 Definition. The formulas which can be obtained by means of the following calculus are called (*propositional*) *Horn formulas*.

(1) $\dfrac{}{(\neg q_1 \vee \ldots \vee \neg q_n \vee q)}$ for $n \in \mathbf{N}$,

(2) $\dfrac{}{(\neg q_0 \vee \ldots \vee \neg q_n)}$ for $n \in \mathbf{N}$,

(3) $\dfrac{\alpha, \beta}{(\alpha \wedge \beta)}$.

Every Horn formula is a formula in conjunctive normal form, where the members of the conjunction are of the form (1) or (2). If, in (1), we distinguish the cases $n = 0$ and $n > 0$, every member of the conjunction has the form (PH1), (PH2) or (PH3):

(PH1) q
(PH2) $(q_0 \wedge \ldots \wedge q_n \rightarrow q)$
(PH3) $(\neg q_0 \vee \ldots \vee \neg q_n)$.

Horn formulas of the form (PH1) or (PH2) are called *positive*, those of the form (PH3) *negative*.

Henceforth, let Δ be a set of positive Horn formulas. Δ is satisfiable: The assignment b with $b(q) = T$ for all propositional variables q is a model of Δ. In addition to this maximal assignment satisfying Δ (maximal in the sense that a maximal number of propositional variables get the truth-value T) we want to give a minimal assignment b^Δ satisfying Δ. For this purpose we interpret the formulas of the form (PH1) and (PH2) as rules.

(PH1) requires: "T is assigned to q",
(PH2) requires: "If T is assigned to $q_0, \ldots, q_n$, then also to q".

And we use this dynamic interpretation of formulas to "construct" b^Δ. So we consider the calculus with the rules

(T1) $\dfrac{}{q}$, if $q \in \Delta$

(T2) $\dfrac{q_0, \ldots, q_n}{q}$, if $(q_0 \wedge \ldots \wedge q_n \rightarrow q) \in \Delta$

and for a propositional variable p we set:

$b^\Delta(p) = T$:iff p is derivable in the calculus with the rules (T1) and (T2).

5.2 Lemma. *b^Δ is a minimal model of Δ, i.e.*
(a) *b^Δ is a model of Δ.*
(b) *For every assignment b which is a model of Δ and for every propositional variable q:*

$$\textit{If } b^\Delta(q) = T \textit{ then } b(q) = T.$$

Proof. (a): If, for example, the formula $(q_0 \wedge \ldots \wedge q_n \rightarrow q)$ is in Δ and $(q_0 \wedge \ldots \wedge q_n)[b^\Delta] = T$ then, by definition of b^Δ, the variables $q_0, \ldots, q_n$ are derivable in the calculus and so (cf. (T2)) q as well; therefore $b^\Delta(q) = T$.

(b): Let b be a model of Δ. By definition of b^Δ it suffices to show that $b(q) = T$ for every derivable q. This can easily be proved by induction over the calculus: For instance, if we get q by rule (T1) then $q \in \Delta$ and so $b(q) = T$, since b is a model of Δ. □

We drop the hypothesis that Δ is a set of positive Horn formulas and show:

5.3 Theorem. *Let Δ be a set of Horn formulas of the form* (PH1), (PH2), *or* (PH3), *and let Δ^+ and Δ^- be the set of positive and negative formulas in Δ, respectively. Then the following are equivalent:*
(a) *Δ is satisfiable.*
(b) *For all $\alpha \in \Delta^-$, $\Delta^+ \cup \{\alpha\}$ is satisfiable.*
(c) *b^{Δ^+} is a model of Δ.*

Proof. The directions from (a) to (b) and from (c) to (a) are trivial. We show how to get (c) from (b). By 5.2(a), b^{Δ^+} is a model of Δ^+. Let $\alpha \in \Delta^-$, say, $\alpha = (\neg q_0 \vee \ldots \vee \neg q_n)$. As we assume (b), there is an assignment b which is a model of $\Delta^+ \cup \{\neg q_0 \vee \ldots \vee \neg q_n\}$, and therefore there is some $i \in \{0, \ldots, n\}$ with $b(q_i) = F$. Since b is a model of Δ^+, 5.2(b) shows that $b^{\Delta^+}(q_i) = F$ and hence $(\neg q_0 \vee \ldots \vee \neg q_n)[b^{\Delta^+}] = T$. □

Now we are ready to give a fast algorithm for testing satisfiability of Horn formulas, the *underlining algorithm*.

Let α be a Horn formula. By the remarks following 5.1, α is a conjunction of formulas β of the form (PH1), (PH2), or (PH3). Let Δ be the set of these β, i.e. the set of members of the conjunction α.

The rules (U1) and (U2) of the underlining algorithm correspond to the rules (T1) and (T2) from above:

(U1) Underline in α all occurrences of a propositional variable q which is itself a member of the conjunction α.

(U2) If in a member $(q_0 \wedge \ldots \wedge q_n \rightarrow q)$ of the conjunction α the propositional variables $q_0, \ldots, q_n$ are already underlined, then underline all occurrences of q in α.

The algorithm terminates when none of the two rules can be applied anymore. If α contains, say, r distinct propositional variables, this happens after at most r steps. Then just those variables q are underlined for which $b^{\Delta^+}(q) = T$. Hence (cf. 5.3) α is satisfiable if and only if in no member $(\neg q_0 \vee \ldots \vee \neg q_n)$ of the conjunction all propositional variables are underlined.

We illustrate the algorithm with two examples. First, let

$$\alpha = (\neg p \vee \neg q) \wedge (p \to q) \wedge (p \wedge r \to q) \wedge r.$$

With (U1) we get:

$$(\neg p \vee \neg q) \wedge (p \to q) \wedge (p \wedge \underline{r} \to q) \wedge \underline{r}.$$

Now we cannot apply any of the rules (U1), (U2). Therefore, α is satisfiable and the minimal assignment b for α is given by

$$b(s) = \begin{cases} T & \text{for } s = r \\ F & \text{otherwise.} \end{cases}$$

Now let

$$\alpha = (\neg p \vee \neg q \vee \neg s) \wedge \neg t \wedge (r \to p) \wedge r \wedge q \wedge (u \to s) \wedge u$$

with propositional variables p, q, r, s, t, and u. Step by step we get:

$$\begin{array}{ll}
(\neg p \vee \neg q \vee \neg s) \wedge \neg t \wedge (\underline{r} \to p) \wedge \underline{r} \wedge q \wedge (u \to s) \wedge u & \text{(with(U1))} \\
(\neg \underline{p} \vee \neg q \vee \neg s) \wedge \neg t \wedge (\underline{r} \to \underline{p}) \wedge \underline{r} \wedge q \wedge (u \to s) \wedge u & \text{(with(U2))} \\
(\neg \underline{p} \vee \neg \underline{q} \vee \neg s) \wedge \neg t \wedge (\underline{r} \to \underline{p}) \wedge \underline{r} \wedge \underline{q} \wedge (u \to s) \wedge u & \text{(with(U1))} \\
(\neg \underline{p} \vee \neg \underline{q} \vee \neg s) \wedge \neg t \wedge (\underline{r} \to \underline{p}) \wedge \underline{r} \wedge \underline{q} \wedge (\underline{u} \to s) \wedge \underline{u} & \text{(with(U1))} \\
(\neg \underline{p} \vee \neg \underline{q} \vee \neg \underline{s}) \wedge \neg t \wedge (\underline{r} \to \underline{p}) \wedge \underline{r} \wedge \underline{q} \wedge (\underline{u} \to \underline{s}) \wedge \underline{u} & \text{(with(U2)).}
\end{array}$$

So α is not satisfiable, since all variables in $(\neg p \vee \neg q \vee \neg s)$ are underlined. In fact, not even the formula

$$\alpha_0 = (\neg p \vee \neg q \vee \neg s) \wedge (r \to p) \wedge r \wedge q \wedge (u \to s) \wedge u,$$

which can be obtained from α by keeping only $(\neg p \vee \neg q \vee \neg s)$ from the negative members of the conjunction, is satisfiable.

In the procedure which we will study later under the name *Horn resolution,* the underlining algorithm is run "backwards": For example, let α be a Horn formula with only one negative member $(\neg q_0 \vee \ldots \vee \neg q_n)$ in the conjunction; if we want to prove that α is not satisfiable using the underlining algorithm, we have to show that all variables in $\{\neg q_0, \ldots, \neg q_n\}$ (i.e. $q_0, \ldots, q_n$) will finally be underlined. If $(r_0 \wedge \ldots \wedge r_j \to q)$ (or if q) is a member of the conjunction α and if $q = q_i$, by rule (T2) (or by rule (T1)) it suffices to show that each variable in

$$(*) \qquad \{\neg q_0, \ldots, \neg q_{i-1}, \neg r_0, \ldots, \neg r_j, \neg q_{i+1}, \ldots, \neg q_n\}$$
$$\text{(or } \{\neg q_0, \ldots, \neg q_{i-1}, \neg q_{i+1}, \ldots, \neg q_n\}\text{)}$$

ends up being underlined.

Now this argument can be repeated and applied to the set in $(*)$. It will turn out that α is not satisfiable if in this way one can reach the empty set in finitely many steps. (Then none of the variables remains to be shown to be underlined.) In the case of

$$\alpha_0 = (\neg p \vee \neg q \vee \neg s) \wedge (r \to p) \wedge r \wedge q \wedge (u \to s) \wedge u$$

we can reach the empty set as follows:

$$\begin{array}{ll}
\{\neg p, \neg q, \neg s\} & \\
\{\neg p, \neg q, \neg u\} & (\text{since } (u \to s) \in \Delta^+) \\
\{\neg p, \neg q\} & (\text{since } u \in \Delta^+) \\
\{\neg p\} & (\text{since } q \in \Delta^+) \\
\{\neg r\} & (\text{since } (r \to p) \in \Delta^+) \\
\emptyset & (\text{since } r \in \Delta^+).
\end{array}$$

The idea underlying this procedure can be extended to arbitrary formulas in CNF; in this way one arrives at the *resolution method* due to J. A. Robinson (1965). There, formulas in CNF are given in set theoretic notation. For instance, one identifies a disjunction $(\alpha_0 \vee \ldots \vee \alpha_n)$ with the set $\{\alpha_0, \ldots, \alpha_n\}$ of its members. In this way the formulas $(\neg p_0 \vee p_1 \vee \neg p_0)$, $(\neg p_0 \vee \neg p_0 \vee p_1)$, and $(p_1 \vee \neg p_0)$ coincide with the set $\{\neg p_0, p_1\}$. Obviously, disjunctions which lead to the same set are logically equivalent. We introduce the notation in a more precise way.

A *literal* is a formula of the form p or $\neg p$. For literals we write $\lambda, \lambda_1, \ldots$. A finite, possibly empty set of literals is called a *clause*. We use the letters $K, L, M, \ldots$ for clauses and $\mathfrak{K}, \ldots$ for (not necessarily finite) sets of clauses.

For a formula α in CNF,

$$\alpha = (\lambda_{00} \vee \ldots \vee \lambda_{0n_0}) \wedge \ldots \wedge (\lambda_{k0} \vee \ldots \vee \lambda_{kn_k}),$$

let

$$\mathfrak{K}(\alpha) := \{\{\lambda_{00}, \ldots, \lambda_{0n_0}\}, \ldots, \{\lambda_{k0}, \ldots, \lambda_{kn_k}\}\}$$

be the set of clauses associated with α.

This transition from a formula to its set of clauses motivates the following definitions:

5.4 Definition. Let b be an assignment, K a clause and $\mathfrak{K}$ a set of clauses.
(a) *b satisfies K* (or *K holds under b*) :iff there is $\lambda \in K$ with $\lambda[b] = T$.
(b) K is *satisfiable* :iff there is an assignment which satisfies K.
(c) *b satisfies $\mathfrak{K}$* :iff b satisfies K for all $K \in \mathfrak{K}$.
(d) *$\mathfrak{K}$ is satisfiable* :iff there is an assignment which satisfies $\mathfrak{K}$.

Thus, an assignment b satisfies a clause $\{\lambda_0, \ldots, \lambda_n\}$ iff $(\lambda_0 \vee \ldots \vee \lambda_n)[b] = T$, and the empty clause is not satisfiable. Therefore, if $\emptyset \in \mathfrak{K}$, $\mathfrak{K}$ is not satisfiable. On the other hand, the empty set of clauses is satisfiable.

Furthermore, we see immediately: If $\emptyset \notin \mathfrak{K}$ and $\mathfrak{K} \neq \emptyset$ then b satisfies the set $\mathfrak{K}$ if and only if b is a model of $\bigwedge_{K \in \mathfrak{K}} \bigvee_{\lambda \in K} \lambda$. Consequently, a formula α in CNF and its set of clauses $\mathfrak{K}(\alpha)$ hold under the same assignments.

With the resolution method one can check whether a set $\mathfrak{K}$ of clauses (and therefore, whether a formula in CNF) is satisfiable. This method is based on a single rule and, therefore, has certain advantages for computer implementation. The rule allows the formation of so-called resolvents.

We extend the notation $p^F := \neg p$ to literals by setting $(\neg p)^F := p$.

5.5 Definition. Let K, K_1 and K_2 be clauses. K is called a *resolvent* of K_1 and K_2 :iff there is a literal λ with $\lambda \in K_1$ and $\lambda^F \in K_2$ such that $(K_1 \setminus \{\lambda\}) \cup (K_2 \setminus \{\lambda^F\}) \subset K \subset K_1 \cup K_2$. [3]

For $K_1 = \{\neg r, p, \neg q, s, t\}$ and $K_2 = \{p, q, \neg s\}$, $\{\neg r, p, s, t, \neg s\}$ is a resolvent of K_1 and K_2, as are $\{\neg r, p, \neg q, t, q\}$ and $\{\neg r, p, \neg q, s, t, q, \neg s\}$.

Adding a resolvent to a set of clauses does not change its satisfiability:

5.6 Resolution Lemma. *Let $\mathfrak{K}$ be a set of clauses, $K_1, K_2 \in \mathfrak{K}$, and K a resolvent of K_1 and K_2. Then for every assignment b the following holds:*

$$b \text{ satisfies } \mathfrak{K} \cup \{K\} \quad \text{iff} \quad b \text{ satisfies } \mathfrak{K}.$$

Proof. The direction from left to right is trivial. For the other direction let b satisfy the set $\mathfrak{K}$. We have to show that b satisfies the clause K. Since K is a resolvent of K_1 and K_2, there is a literal λ with $\lambda \in K_1$, $\lambda^F \in K_2$, and $(K_1 \setminus \{\lambda\}) \cup (K_2 \setminus \{\lambda^F\}) \subset K \subset K_1 \cup K_2$. There are two cases:

$\lambda[b] = F$: Since K_1 holds under b, there is $\lambda' \in K_1$, $\lambda \neq \lambda'$, with $\lambda'[b] = T$. Since $\lambda' \in K$, K is satisfied by b.

$\lambda[b] = T$: Then $\lambda^F[b] = F$, and we argue similarly with K_2 and λ^F. □

We now show that an arbitrary set $\mathfrak{K}$ of clauses is not satisfiable if and only if, by forming resolvents and starting from the clauses in $\mathfrak{K}$, one can get to the empty clause in finitely many steps. For this purpose we introduce for $i \in \mathbf{N}$ the set $\mathrm{Res}_i(\mathfrak{K})$ of clauses, which can be obtained from $\mathfrak{K}$ in at most i steps.

5.7 Definition. For a set $\mathfrak{K}$ of clauses let

$$\begin{aligned}\mathrm{Res}(\mathfrak{K}) := \mathfrak{K} \cup \{K \mid\ & \text{there are } K_1, K_2 \in \mathfrak{K} \text{ such that} \\ & K \text{ is a resolvent of } K_1 \text{ and } K_2\ \}.\end{aligned}$$

For $i \in \mathbf{N}$ define $\mathrm{Res}_i(\mathfrak{K})$ inductively by

$$\begin{aligned}\mathrm{Res}_0(\mathfrak{K}) &:= \mathfrak{K} \\ \mathrm{Res}_{i+1}(\mathfrak{K}) &:= \mathrm{Res}(\mathrm{Res}_i(\mathfrak{K})).\end{aligned}$$

Finally, set

$$\mathrm{Res}_\infty(\mathfrak{K}) := \bigcup_{i \in \mathbf{N}} \mathrm{Res}_i(\mathfrak{K}).$$

Now the result which was already announced several times can be phrased as follows:

---

[3] The results that follow below remain valid if in addition we require that $K = (K_1 \setminus \{\lambda\}) \cup (K_2 \setminus \{\lambda^F\})$. For the purposes of logic programming, however, it is better to give the definitions as done above.

5.8 Resolution Theorem. *For a set $\mathfrak{K}$ of clauses,*

$$\mathfrak{K} \text{ is satisfiable} \quad \text{iff} \quad \emptyset \notin \mathrm{Res}_\infty(\mathfrak{K}).$$

Proof. First, let $\mathfrak{K}$ be satisfiable. Then, by the Resolution Lemma, $\mathrm{Res}(\mathfrak{K})$ is satisfiable as well. From this we get immediately by induction that $\mathrm{Res}_i(\mathfrak{K})$ is satisfiable for all i and therefore $\emptyset \notin \mathrm{Res}_i(\mathfrak{K})$. Hence $\emptyset \notin \mathrm{Res}_\infty(\mathfrak{K})$.

Conversely, assume for contradiction that $\emptyset \notin \mathrm{Res}_\infty(\mathfrak{K})$ and $\mathfrak{K}$ is not satisfiable. As a set of nonempty clauses, $\mathfrak{K}$ is not satisfiable if and only if $\{\bigvee_{\lambda \in K} \lambda \mid K \in \mathfrak{K}\}$ is not satisfiable. By the Compactness Theorem 4.5 we can assume that $\mathfrak{K}$ is finite. For $m \in \mathbf{N}$ we set

$$\mathfrak{R}_m := \{K \in \mathrm{Res}_\infty(\mathfrak{K}) \mid K \subset PF_m\}.$$

In particular, $\mathfrak{R}_0 = \emptyset$ or $\mathfrak{R}_0 = \{\emptyset\}$; but $\emptyset \notin \mathrm{Res}_\infty(\mathfrak{K})$ and therefore $\mathfrak{R}_0 = \emptyset$. We choose $n \in \mathbf{N}$ such that $K \subset PF_n$ for all $K \in \mathfrak{K}$, i.e. in the clauses of $\mathfrak{K}$ only the propositional variables $p_0, \ldots, p_{n-1}$ and their negations occur. Since this property is preserved by forming resolvents we easily obtain, by induction on i, that $K \subset PF_n$ for all $K \in \mathrm{Res}_i(\mathfrak{K})$, i.e. for all $K \in \mathrm{Res}_\infty(\mathfrak{K})$. In particular $\mathfrak{K} \subset \mathrm{Res}_\infty(\mathfrak{K}) = \mathfrak{R}_n$ and therefore $\mathfrak{R}_n$ is not satisfiable (as $\mathfrak{K}$ was assumed to be unsatisfiable).

We set

$$l := \min\{m \mid \mathfrak{R}_m \text{ is not satisfiable}\}$$

and distinguish two cases:

Case $l = 0$: Then $\mathfrak{R}_0$ is not satisfiable which contradicts $\mathfrak{R}_0 = \emptyset$.

Case $l = k + 1$: By minimality of l, the set $\mathfrak{R}_k$ is satisfiable. Since in $\mathfrak{R}_k$ only the variables $p_0, \ldots, p_{k-1}$ occur, there are $b_0, \ldots, b_{k-1} \in \{T, F\}$ with

(1) $$(b_0, \ldots, b_{k-1}) \text{ satisfies } \mathfrak{R}_k.$$

Since $\mathfrak{R}_{k+1}$ is not satisfiable there exists a clause K_T for the assignment $(b_0, \ldots, b_{k-1}, T)$ such that

(2) $$K_T \in \mathfrak{R}_{k+1} \text{ and } (b_0, \ldots, b_{k-1}, T) \text{ does not satisfy } K_T,$$

and for the assignment $(b_0, \ldots, b_{k-1}, F)$ there is a clause K_F such that

(3) $$K_F \in \mathfrak{R}_{k+1} \text{ and } (b_0, \ldots, b_{k-1}, F) \text{ does not satisfy } K_F.$$

By (2) and (3) we have

(4) $$p_k \notin K_T \text{ and } \neg p_k \notin K_F.$$

We show

(5) $$\neg p_k \in K_T \text{ and } p_k \in K_F.$$

Namely, if $\neg p_k \notin K_T$ then (with (4)) $K_T \subset PF_k$ and therefore $K_T \subset \mathfrak{R}_k$. But with $(b_0, \ldots, b_{k-1})$ also $(b_0, \ldots, b_{k-1}, T)$ would satisfy the clause K_T – a contradiction to (2). Similarly one can show that $p_k \in K_F$.

By (5), $K := (K_T \setminus \{\neg p_k\}) \cup (K_F \setminus \{p_k\})$ is a resolvent of K_T and K_F, which belongs to $\mathfrak{R}_k$ by (4). By (1), $(b_0, \ldots, b_{k-1})$ satisfies the clause K, i.e. $(b_0, \ldots, b_{k-1})$ satisfies a literal from $(K_T \setminus \{\neg p_k\}) \cup (K_F \setminus \{p_k\})$, which contradicts (2) or (3). $\square$

We illustrate the resolution method by an example, introducing a transparent notation at the same time. Let

$$\alpha = (q \vee \neg r) \wedge \neg p \wedge (p \vee r) \wedge (\neg q \vee p \vee \neg r).$$

Then

$$\mathfrak{K}(\alpha) = \{\{q, \neg r\}, \{\neg p\}, \{p, r\}, \{\neg q, p, \neg r\}\}.$$

The "resolution tree" in Figure XI.1 shows that $\mathfrak{K}(\alpha)$ and therefore α is not satisfiable: The nodes with no upper neighbours are clauses from $\mathfrak{K}(\alpha)$, the remaining nodes are resolvents of their respective upper neighbours.

FIGURE XI.1.

If every clause in $\mathfrak{K}$ contains only literals from $\{p_0, \ldots, p_{n-1}\} \cup \{\neg p_0, \ldots, \neg p_{n-1}\}$ then in every resolvent at most these literals occur. From this we easily get for such $\mathfrak{K}$ (we leave the details to the reader): $\mathrm{Res}_{2^{2n}}(\mathfrak{K}) = \mathrm{Res}_\infty(\mathfrak{K})$. Therefore, if $\mathfrak{K}$ is finite, we get an answer to the question whether $\mathfrak{K}$ is satisfiable in finitely many steps.

On the other hand, if $\mathfrak{K}$ is infinite, it is possible that infinitely many resolvents can be formed by passing from $\mathrm{Res}_i(\mathfrak{K})$ to $\mathrm{Res}_{i+1}(\mathfrak{K})$ or that

$$\mathrm{Res}_0(\mathfrak{K}) \subsetneq \mathrm{Res}_1(\mathfrak{K}) \subsetneq \ldots.$$

In these cases, if $\mathfrak{K}$ is satisfiable, we can form infinitely many resolvents without getting an answer to the question whether $\mathfrak{K}$ is satisfiable or not. For instance, the satisfiable set of clauses

$$\{\{p_0\}\} \cup \{\{\neg p_i, p_{i+1}\} \mid i \in \mathbf{N}\}$$

admits the resolution tree in Figure XI.2.

FIGURE XI.2.

Even for unsatisfiable infinite $\mathfrak{K}$ we may obtain the empty clause (and with it the answer "$\mathfrak{K}$ is not satisfiable") in finitely many steps only by an appropriate choice of resolvents. For example, Figure XI.2 also is a resolution tree for the unsatisfiable set of clauses

$$\{\{p_0\}, \{\neg p_0\}\} \cup \{\{\neg p_i, p_{i+1}\} \mid i \in \mathbf{N}\}$$

in which $\emptyset$ does not occur.

Now we return to the special case of Horn formulas, which was, in fact, the starting point of our considerations.

We call a clause of the form $\{q\}$ or $\{\neg q_0, \ldots, \neg q_n, q\}$ *positive*, one of the form $\{\neg q_1, \ldots, \neg q_n\}$ *negative*. A negative clause can be empty, a positive one cannot. Positive clauses correspond to positive Horn formulas and nonempty negative clauses to negative Horn fomulas. For negative clauses we use the letters $N, N_1, \ldots$.

In the following we only deal with a single negative clause at a time. Because of 5.3 this is not an essential restriction.

5.9 Definition. Let $\mathfrak{P}$ be a set of positive clauses and let N be negative.

(a) A sequence $N_0, \ldots, N_k$ of negative clauses is a *Horn-* (short: *H-*) *resolution of* $\mathfrak{P}$ *and* N :iff there are $K_0, \ldots, K_{k-1} \in \mathfrak{P}$ so that $N = N_0$ and N_{i+1} is a resolvent of K_i and N_i for $i < k$.

(b) A negative clause N' is called *H-derivable* from $\mathfrak{P}$ and N :iff there is an H-resolution $N_0, \ldots, N_k$ of $\mathfrak{P}$ and N with $N' = N_k$.

We often represent the H-resolution in (a) as in Figure XI.3.

As motivated by our treatment of the "backwards" version of the underlining algorithm, we get:

FIGURE XI.3.

5.10 Theorem on the H-Resolution. *For a set $\mathfrak{P}$ of positive clauses and a negative clause N the following are equivalent:*
(a) *$\mathfrak{P} \cup \{N\}$ is satisfiable.*
(b) *$\emptyset$ is not H-derivable from $\mathfrak{P}$ and N.*

Proof. First, let b be an assignment satisfying $\mathfrak{P} \cup \{N\}$. By the Resolution Lemma 5.6 we have for every H-resolution $N_0, \ldots, N_k$ of $\mathfrak{P}$ and N:

$$b \text{ satisfies } N_0,\ b \text{ satisfies } N_1,\ \ldots,\ b \text{ satisfies } N_k;$$

therefore in particular $N_k \neq \emptyset$. Hence $\emptyset$ is not H-derivable from $\mathfrak{P}$ and N.

The direction from (b) to (a): The clauses in $\mathfrak{P}$ correspond to a set Δ of positive Horn formulas. We show:

(*) If $k \in \mathbf{N}$ and $b^\Delta(q_1) = \ldots = b^\Delta(q_k) = T$ then $\emptyset$ is H-derivable from $\mathfrak{P}$ and $\{\neg q_1, \ldots, \neg q_k\}$.

Then we are done: In fact, if $\emptyset$ is not H-derivable from $\mathfrak{P}$ and N and if, say, $N = \{\neg q_1, \ldots, \neg q_k\}$, then (*) shows that there is an i with $b^\Delta(q_i) = F$. So b^Δ is a model of $\mathfrak{P} \cup \{N\}$.

We obtain (*) by proving inductively on l that (*) holds provided each q_i can be obtained in $\leq l$ steps by means of the calculus with the rules (T1), (T2) associated with Δ (cf. the considerations leading to 5.2): Suppose the last step in the derivation of q_i is of the form $\dfrac{r_{i1} \ldots r_{ij_i}}{q_i}$ (i.e. a step according to (T1) if $j_i = 0$, and according to (T2) if $j_i > 0$). In particular, the clauses $\{\neg r_{i1}, \ldots, \neg r_{ij_i}, q_i\}$ belong to $\mathfrak{P}$. Furthermore, by definition of b^Δ, $b^\Delta(r_{is}) = T$ for $i = 1, \ldots, k$ and $s = 1, \ldots, j_i$. By the induction hypothesis, $\emptyset$ is H-derivable from $\mathfrak{P}$ and $N' := \{\neg r_{11}, \ldots, \neg r_{1j_1}, \ldots, \neg r_{k1}, \ldots, \neg r_{kj_k}\}$.

Let $\underset{\emptyset}{\nabla}$ denote such a derivation. Then Figure XI.4 represents an H-derivation of $\emptyset$ from $\mathfrak{P}$ and $\{\neg q_1, \ldots, \neg q_k\}$. □

FIGURE XI.4.

For an application in Section 7 we rephrase the previous theorem in a form which is closer to the Resolution Theorem 5.8. For this purpose we modify the operation Res so that only those resolvents are included which are of the form as permitted in Theorem 5.10:

For a set $\mathfrak{K}$ of clauses let

$$\mathrm{HRes}(\mathfrak{K}) := \mathfrak{K} \cup \{N \mid N \text{ is a negative clause and there are a positive } K_1 \in \mathfrak{K} \text{ and a negative } N_1 \in \mathfrak{K} \text{ such that } N \text{ is a resolvent of } K_1 \text{ and } N_1\}.$$

Again let $\mathrm{HRes}_0(\mathfrak{K}) := \mathfrak{K}$, $\mathrm{HRes}_{i+1}(\mathfrak{K}) := \mathrm{HRes}(\mathrm{HRes}_i(\mathfrak{K}))$, $\mathrm{HRes}_\infty(\mathfrak{K}) := \bigcup_{i \in \mathbf{N}} \mathrm{HRes}_i(\mathfrak{K})$. Then 5.10 can be phrased as follows:

5.11 Theorem. *For a set $\mathfrak{P}$ of positive clauses and a negative clause N,*

$$\mathfrak{P} \cup \{N\} \textit{ is satisfiable} \quad \textit{iff} \quad \emptyset \notin \mathrm{HRes}_\infty(\mathfrak{P} \cup \{N\}).$$

Proof. An easy induction on $i \in \mathbf{N}$ shows that for a negative clause N':

$$N' \in \mathrm{HRes}_i(\mathfrak{P} \cup \{N\}) \quad \text{iff} \quad \text{there is an H-derivation of } N' \text{ of } \mathfrak{P} \text{ and } N \text{ of length} \leq i.$$

From this we get the claim immediately with 5.10. □

5.12 Exercise. For $\mathfrak{K} := \{\{p_0, p_1, p_2\}\} \cup \{\{\neg p_i\} \mid i \geq 1\}$ show:
(a) $\mathrm{Res}_\infty(\mathfrak{K}) = \mathrm{Res}_2(\mathfrak{K})$;
(b) $\mathrm{Res}_2(\mathfrak{K}) \setminus \mathrm{Res}_1(\mathfrak{K})$ and $\mathrm{Res}_1(\mathfrak{K}) \setminus \mathfrak{K}$ are finite.
(c) $\mathfrak{K}$ is satisfiable.

§6. First-Order Resolution (without Unification)

To conclude this chapter, we transfer to first-order language the resolution methods which we have introduced for propositional logic. Thereby, Herbrand's Theorem will play an important role. As expected, it will turn out that the corresponding algorithms are more complex, since, in addition to the propositional structure, term instantiations also have to be considered. In the present section we prove that in principle this transfer is possible. In the next section, we learn how to carry out the term instantiations in a goal-directed and efficient manner. We will be led to an analogue of the propositional Horn resolution. It forms the core of the procedure taken by a computer which runs a program written in PROLOG. We shall not go into refinements of the method or details of the implementation which should increase efficiency; for such details see [1], [26] or [31]. Essential limitations of the method are indicated in Exercise X.4.4.

At the end of Section 2 we mentioned that a programmer, who wants to write a program in PROLOG for a certain type of problem, has to formalize the assumptions as universal Horn formulas and the "queries" as existential formulas. The following examples illustrate this approach.

First, we give a very simple example. Let the relation symbols M, F and D be unary and $S := \{M, F, D\}$. Let an S-structure $\mathfrak{A}$ be given. We interpret the elements of A as inhabitants of a town, M^A and F^A as the subsets of male and female inhabitants respectively, and finally, let $D^A a$ mean that a has a driver's license. Then we consider the question

(1) Are there male inhabitants which have a driver's license?

For each $a \in A$ we choose a constant c_a. Then the following set Φ of atomic Horn sentences contains the "positive" information about $\mathfrak{A}$:

$$\Phi := \{Mc_a \mid a \in M^A\} \cup \{Fc_a \mid a \in F^A\} \cup \{Dc_a \mid a \in D^A\}.$$

We show that question (1) is equivalent to

$$\Phi \vdash \exists x(Mx \wedge Dx) \quad ? \tag{2}$$

Hence, it can be written in a form which, by the introductory remarks, can be translated into a logic program (which, in case of a positive answer, should be able to list all male inhabitants with driver's license).

To show the equivalence of (1) and (2) it suffices to prove

$$\mathfrak{A} \models \exists x(Mx \wedge Dx) \quad \text{iff} \quad \Phi \vdash \exists x(Mx \wedge Dx). \tag{3}$$

Because of $(\mathfrak{A}, (a)_{a \in A}) \models \Phi$ the direction from right to left holds. The definition of Φ immediately gives

(4) If $M'^A, F'^A, D'^A \subset A$ and $(A, M'^A, F'^A, D'^A, (a)_{a \in A}) \models \Phi$, then $M^A \subset M'^A, F^A \subset F'^A$ und $D^A \subset D'^A$.

If we identify the term c_a with a, (4) says that $(\mathfrak{A}, (a)_{a\in A})$ is the minimal Herbrand model of Φ, so, by 3.8, it is the term structure $\mathfrak{T}_0^\Phi$ of Φ. Therefore, from $(\mathfrak{A}, (a)_{a\in A}) \models \exists x(Mx \wedge Dx)$ we get, by 3.9, that $\Phi \vdash \exists x(Mx \wedge Dx)$.

An example from graph theory: In a directed graph $\mathfrak{G} = (G, R^G)$ we call two vertices $a, b \in G$ *connected* if there are $n \in \mathbf{N}$ and $a_0, \ldots, a_n \in G$ with

$$a = a_0, b = a_n \quad \text{and} \quad R^G a_i a_{i+1} \text{ for } i < n.$$

We set

$$C^G := \{(a, b) \mid a \text{ and } b \text{ are connected in } \mathfrak{G}\}.$$

If, say, G is the set of towns of a country and $R^G ab$ means that a certain airline offers service from a to b without stopover, then $C^G ab$ holds if and only if it is possible to fly from a to b with this airline (all stopovers lying in the home country). Let agents of a company live in the towns a and b who can use this airline free of charge. We show how, for instance, the questions "Is it possible for the agent living in a to fly to b free of charge?" and "Is there a town to which both agents can get free of charge?" can be written as logic programs. So, we are dealing with the following two questions:

$$(G, R^G, C^G) \models Cxy[a, b] \quad ?$$
$$(G, R^G, C^G) \models \exists z(Cxz \wedge Cyz)[a, b] \quad ?$$

For each $a \in G$ we introduce a constant c_a and let Φ_0 be the "positive" atomic information of the structure $(G, R^G, (a)_{a\in G})$:

$$\Phi_0 := \{Rc_a c_b \mid a, b \in G, R^G ab\}.$$

Furthermore, we set

$$\Phi_1 := \Phi_0 \cup \{\forall x Cxx, \forall x \forall y \forall z(Cxy \wedge Ryz \to Cxz)\}.$$

Then Φ_1 is a set of universal Horn sentences. We show that the questions from above can be phrased in the form

$$\Phi_1 \vdash Cc_a c_b \quad ? \quad \text{and} \quad \Phi_1 \vdash \exists z(Cc_a z \wedge Cc_b z) \quad ?$$

i.e., in a form, in which they can (by the introductory remarks) be written as logic programs. We set $\mathfrak{G}_1 := (G, R^G, C^G, (a)_{a\in G})$. Then we have to show

(1) $$\mathfrak{G}_1 \models Cc_a c_b \quad \text{iff} \quad \Phi_1 \vdash Cc_a c_b.$$

(2) $$\mathfrak{G}_1 \models \exists z(Cc_a z \wedge Cc_b z) \quad \text{iff} \quad \Phi_1 \vdash \exists z(Cc_a z \wedge Cc_b z).$$

We argue similarly to the previous example: Because of $\mathfrak{G}_1 \models \Phi_1$ the left-hand sides in (1) and (2) follow immediately from the right-hand sides. We now prove the other directions and note first:

(3) If $R'^G, C'^G \subset G \times G$ and $(G, R'^G, C'^G, (a)_{a\in G}) \models \Phi_1$ then $R^G \subset R'^G$ and $C^G \subset C'^G$.

Indeed, the definition of Φ_0 immediately gives $R^G \subset R'^G$. Furthermore, by definition of C^G, we have to show for $n \in \mathbf{N}$ and $a_0, \ldots, a_n \in G$ with $R^G a_i a_{i+1}$ for $i < n$ that $C'^G a_0 a_n$. This is easily obtained from the axioms in Φ_1 by induction on n.

Now, if for $a \in G$ we identify the term c_a with a, then (3) together with 3.8 shows that $\mathfrak{G}_1$ is the Herbrand structure $\mathfrak{T}_0^{\Phi_1}$. Therefore, by 3.9, the right-hand sides in (1) and (2) follow from the left-hand sides.

Of course, one normally expects not only an answer to the question whether a and b are connected in (G, R^G), but, in the positive case, also a specification of the paths from a to b. We indicate how this can be realized.

For this purpose, we consider the symbol set $S := \{R, P, f\} \cup \{c_a \mid a \in G\}$, where P is ternary and f is binary. For $a, b, d, e \in G$ with $R^G ab$, $R^G bd$, $R^G da$, and $R^G ae$ say, the term $fffffc_a c_b c_d c_a c_e$ represents in an obvious manner the path from a over b, d, and a to e. In general, let $Pxyv$ say that v represents a path from x to y. We set

$$\Phi_2 := \Phi_0 \cup \{\forall x\, Pxxx, \forall x \forall y \forall u \forall z (Pxyu \wedge Ryz \to Pxzfuz)\}.$$

The reader should verify (as above in the proof of (1) and (2)) that the following holds for any term $t \in T_0^S$:

$$\Phi_2 \vdash Pc_a c_b t \quad \text{iff} \quad t \text{ represents a path from } a \text{ to } b \text{ in } (G, R^G).$$

Now we expect that, given the question "$\Phi_2 \vdash \exists v\, Pc_a c_b v$?", a logic program provides all terms $t \in T_0^S$ which represent paths from a to b.

In the examples, as in most applications of logic programming, the equality symbol does not occur. *Therefore, in the remainder of this chapter we restrict ourselves to equality-free formulas without emphasizing this explicitly in each case.* (Exercise 6.11 shows how to make use of the results and techniques also for formulas *with* equality.)

In order to transfer the propositional resolution methods to the first-order language we make use of the connection given by 4.4 between propositional logic and quantifier-free first-order formulas, and of Herbrand's Theorem. First, however, we need some more terminology.

Throughout let S be an at most countable symbol set containing a constant.

6.1 Definition.

(a) Let φ be a formula of the form $\forall x_1 \ldots \forall x_n \psi$ with quantifier-free ψ. Then for arbitrary (!) pairwise distinct variables $y_1, \ldots, y_l$ and terms $t_1, \ldots, t_l$, the formula $\psi(\overset{l}{y} \mid \overset{l}{t})$ is called an *instance* of φ. If $\psi(\overset{l}{y} \mid \overset{l}{t})$ is a sentence we also call it a *ground instance* of φ.

(b) Let $\mathrm{GI}(\varphi)$ be the set of ground instances of φ.

(c) For a set Φ of formulas φ of the form above let $\mathrm{GI}(\Phi) := \bigcup_{\varphi \in \Phi} \mathrm{GI}(\varphi)$.

Note that for a sentence $\varphi := \forall x_1 \ldots \forall x_m \psi$ with quantifier-free ψ and terms $t_1, \ldots, t_m \in T_0^S$ the formula $\psi(\overset{m}{x} \mid \overset{m}{t})$ is a ground instance of φ.

We choose a bijection $\pi_0 \colon A^S \to \{p_i \mid i \in \mathbf{N}\}$ from the set of (equality-free) atomic formulas onto the set of propositional variables. Let π be the extension of π_0 to the set of quantifier-free formulas given before 4.3.

6.2 Definition. A set Ψ of quantifier-free formulas is called *propositionally satisfiable* :iff $\pi(\Psi)$ is satisfiable.

By 4.4 the following obviously holds:

6.3 Lemma. *For a set Ψ of quantifier-free formulas the following are equivalent:*
(a) *Ψ is satisfiable.*
(b) *Ψ is propositionally satisfiable.* □

Herbrand's Theorem in the form of 1.3 yields the following (for simplicity, we restrict ourselves to sentences):

6.4 Theorem. *For a set Φ of equality-free sentences of the form $\forall x_1 \ldots \forall x_m \psi$ with quantifier-free ψ the following are equivalent:*
(a) *Φ is satisfiable.*
(b) *$\mathrm{GI}(\Phi)$ is propositionally satisfiable.*

Proof. We only have to notice that the ground instances of $\forall x_1 \ldots \forall x_m \psi$ can be written in the form $\psi(\overset{m}{x} \mid \overset{m}{t})$ with $t_1, \ldots, t_m \in T_0^S$. □

In the situation of the previous theorem we can apply the resolution method to the set of formulas given in (b). Note, however, that in general the set $\mathrm{GI}(\forall x_1 \ldots \forall x_m \psi)$ of formulas is infinite. (The limitations of the resolution method for infinite sets have been discussed at the end of the previous section.)

We give a few examples. For the sake of clarity and legibility, we work here and in the following with clauses consisting of atomic and negated atomic first-order formulas, and we do not pass to their images under π. In fact, Lemma 6.3 says that we can deal with atomic formulas as we do with propositional variables. We transfer the notations and terminology in a natural manner. So literals are now atomic or negated atomic formulas; and for a literal ψ we have

$$\psi^F = \begin{cases} \neg\psi & \text{if } \psi \text{ is atomic,} \\ \varphi & \text{if } \psi = \neg\varphi. \end{cases}$$

For a clause K let

$$K^F := \{\psi^F \mid \psi \in K\}.$$

6.5 Example. Let $S := \{R, g, c\}$ with binary R and unary g. The satisfiability of the sentence

$$\forall z \forall y (Rcy \wedge \neg Rzgz)$$

is equivalent to the propositional satisfiability of

$$\{Rct_1 \wedge \neg Rt_2gt_2 \mid t_1, t_2 \in T_0^S\},$$

i.e. to the satisfiability of the set of clauses

$$\big\{\{Rct\} \mid t \in T_0^S\big\} \cup \big\{\{\neg Rtgt\} \mid t \in T_0^S\big\}.$$

Thus, the resolution tree

$$\begin{array}{ccc} \{Rcgc\} & & \{\neg Rcgc\} \\ & \diagdown \quad \diagup & \\ & \emptyset & \end{array}$$

shows that $\forall z \forall y (Rcy \wedge \neg Rzgz)$ is not satisfiable.

6.6 Example. Let $S := \{Q, R, g, c\}$ with unary Q and R, g, c as in the previous example. The sentence

$$\forall x \forall y ((Rxy \vee Qx) \wedge \neg Rxgx \wedge \neg Qy)$$

is not satisfiable, since its set of clauses

$$\big\{\{Rt_1t_2, Qt_1\} \mid t_1, t_2 \in T_0^S\big\} \cup \big\{\{\neg Rtgt\} \mid t \in T_0^S\big\} \cup \big\{\{\neg Qt\} \mid t \in T_0^S\big\}$$

admits the resolution tree in Figure XI.5 leading to $\emptyset$.

$$\begin{array}{ccccc} & \{Rcgc, Qc\} & & \{\neg Rcgc\} & \\ & & \diagdown \quad \diagup & & \\ \{\neg Qc\} & & \{Qc\} & & \\ & \diagdown \quad \diagup & & & \\ & \emptyset & & & \end{array}$$

FIGURE XI.5.

It is clear that we could also have chosen the ground instances corresponding to $x := ggc$ and $y := gggc$ and then, in a similar way, we would have obtained the tree in Figure XI.6.

In complicated cases it is certainly important to use terms as simple as possible. We consider corresponding methods and heuristics in the next section.

Theorem 6.4 refers only to universal formulas in prenex normal form. However, in VIII.4.5 we saw how to assign to an arbitrary formula a universal formula in prenex normal form equivalent to it for satisfiability (Theorem

FIGURE XI.6.

on the Skolem Normal Form). In this way the resolution method becomes applicable to arbitrary (equality-free) formulas. We illustrate this by the following example.

6.7 Example. Let $S := \{R\}$ with binary R and

$$\varphi := (\exists x \forall y\, Rxy \wedge \forall z \exists u \neg Rzu).$$

φ is logically equivalent to the formula in prenex normal form

$$\exists x \forall z \exists u \forall y (Rxy \wedge \neg Rzu).$$

Choosing a unary function symbol g and a constant c we get the $\{R, g, c\}$-sentence (cf. the proof of VIII.4.5)

$$\forall z \forall y (Rcy \wedge \neg Rzgz),$$

which is equivalent to φ for satisfiability, and which we found out to be unsatisfiable using the resolution method (see Example 6.5). So φ is not satisfiable.

The Horn resolution for propositional logic given in the previous section can be transferred to universal Horn formulas.

By 2.3(a), universal Horn formulas are logically equivalent to a conjunction of formulas of the form (H1), (H2) or (H3):

(H1) $\forall x_1 \ldots \forall x_m \varphi$

(H2) $\forall x_1 \ldots \forall x_m (\varphi_0 \wedge \ldots \wedge \varphi_n \rightarrow \varphi)$

(H3) $\forall x_1 \ldots \forall x_m (\neg\varphi_0 \vee \ldots \vee \neg\varphi_n)$

with atomic φ and φ_i.

Horn formulas of the form (H1) or (H2) are called *positive*, those of the form (H3) *negative*. So the positive and negative propositional Horn formulas correspond, by virtue of π, to the quantifier-free positive and negative Horn formulas, respectively. For a set Φ of universal Horn formulas let Φ^+ and Φ^- stand for the subsets of positive and negative formulas, respectively. Since instances of positive and negative Horn formulas are again positive and negative, respectively, we have $\mathrm{GI}(\Phi^+) = (\mathrm{GI}(\Phi))^+$.

6.8 Lemma. *Let Φ be a satisfiable set of universal Horn sentences of the form* (H1), (H2) *or* (H3), *and let $\exists x_1 \ldots \exists x_m(\psi_0 \wedge \ldots \wedge \psi_l)$ be a sentence with atomic $\psi_0, \ldots, \psi_l$.*

(a) *For $t_1, \ldots, t_m \in T_0^S$ the following are equivalent:*

(i) $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{m}{x} \mid \overset{m}{t})$

(ii) $\Phi^+ \vdash (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{m}{x} \mid \overset{m}{t})$.

(b) *The following are equivalent:*

(i) $\Phi \vdash \exists x_1 \ldots \exists x_m(\psi_0 \wedge \ldots \wedge \psi_l)$

(ii) $\Phi^+ \vdash \exists x_1 \ldots \exists x_m(\psi_0 \wedge \ldots \wedge \psi_l)$.

Proof. (a): For $t_1, \ldots, t_m \in T_0^S$ we get the equivalence of the following statements:

(1) $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{m}{x} \mid \overset{m}{t})$.

(2) $\Phi \cup \{(\neg\psi_0 \vee \ldots \vee \neg\psi_l)(\overset{m}{x} \mid \overset{m}{t})\}$ is not satisfiable.

(3) $\mathrm{GI}(\Phi) \cup \{(\neg\psi_0 \vee \ldots \vee \neg\psi_l)(\overset{m}{x} \mid \overset{m}{t})\}$ is not propositionally satisfiable (cf. 6.4).

(4) $\mathrm{GI}(\Phi^+) \cup \{(\neg\psi_0 \vee \ldots \vee \neg\psi_l)(\overset{m}{x} \mid \overset{m}{t})\}$ is not propositionally satisfiable.

(By 6.4, as Φ is satisfiable so is $\mathrm{GI}(\Phi)$. Hence, we get the equivalence of (3) and (4) immediately from $(\mathrm{GI}(\Phi))^+ = \mathrm{GI}(\Phi^+)$ with 5.3.) If we choose the set Φ^+ for Φ and note that $(\Phi^+)^+ = \Phi^+$, then the equivalence of (1) and (4) shows that statement (4) is equivalent to

(5) $\Phi^+ \vdash (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{m}{x} \mid \overset{m}{t})$.

By 3.9, (b) follows immediately from (a). □

In the following considerations we restrict ourselves to sets Φ of *positive* universal Horn sentences; the previous lemma shows that this is not an essential restriction. For this case, the Horn resolution can easily be transferred to first-order language.

6.9 Theorem. *Let Φ be a set of positive universal Horn sentences and $\exists x_1 \ldots \exists x_m(\psi_0 \wedge \ldots \wedge \psi_l)$ a sentence with atomic $\psi_0, \ldots, \psi_l$.*

(a) *For $t_1, \ldots, t_m \in T_0^S$ the following are equivalent:*

(i) $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{m}{x} \mid \overset{m}{t})$

(ii) *There is an H-derivation of the empty clause $\emptyset$ from* $\mathrm{GI}(\Phi)$ *and* $(\neg\psi_0 \vee \ldots \vee \neg\psi_l)(\overset{m}{x} \mid \overset{m}{t})$ *(more exactly: from the clauses corresponding to* $\mathrm{GI}(\Phi)$ *and the clause* $\{\neg\psi_0(\overset{m}{x} \mid \overset{m}{t}), \ldots, \neg\psi_l(\overset{m}{x} \mid \overset{m}{t})\}$*).*

(b) *The following are equivalent:*

(i) $\Phi \vdash \exists x_1 \ldots \exists x_m(\psi_0 \wedge \ldots \wedge \psi_l)$

(ii) *There are $t_1, \ldots, t_m \in T_0^S$ such that there exists an H-derivation of $\emptyset$ from* $\mathrm{GI}(\Phi)$ *and* $(\neg\psi_0 \vee \ldots \vee \neg\psi_l)(\overset{m}{x} \mid \overset{m}{t})$.

Proof. (a): We argue as in the proof of 6.8 and get the equivalence of the following statements for $t_1, \ldots, t_m \in T_0^S$:

(1) $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{m}{x} \mid \overset{m}{t})$.

(2) $\Phi \cup \{(\neg\psi_0 \vee \ldots \vee \neg\psi_l)(\overset{m}{x} \mid \overset{m}{t})\}$ is not satisfiable.

(3) $\mathrm{GI}(\Phi) \cup \{(\neg\psi_0 \vee \ldots \vee \neg\psi_l)(\overset{m}{x} \mid \overset{m}{t})\}$ is not propositionally satisfiable.

(4) There is an H-derivation of the empty clause from $\mathrm{GI}(\Phi)$ and $(\neg\psi_0 \vee \ldots \vee \neg\psi_l)(\overset{m}{x} \mid \overset{m}{t})$ (cf. 5.10: since Φ is a set of positive universal Horn sentences, the clauses corresponding to the sentences from $\mathrm{GI}(\Phi)$ are positive!).

Again, (b) follows, by 3.9, from (a). □

6.10 Example. Let $S := \{R, T, a, b, c, d, e\}$ with binary R, T and constants a, b, c, d, e, and let

$$\begin{aligned}\Phi := \{&Rab, Rcb, Rbd, Rde,\\ &\forall x\forall y(Rxy \to Txy),\ \forall x\forall y(Txy \wedge Tyz \to Txz)\}.\end{aligned}$$

Then $\Phi \vdash \exists x(Rcx \wedge Rax \wedge Txe)$, since $\mathrm{GI}(\Phi) \cup \{\neg Rcb \vee \neg Rab \vee \neg Tbe\}$ is not propositionally satisfiable, as is shown by the H-resolution tree in Figure XI.7 on the next page.

6.11 Exercise. This exercise shows how to extend the results of this section to formulas containing the equality symbol: Let the binary relation symbol E not occur in the symbol set S. We set $S' := S \cup \{E\}$. To each S-formula φ we assign the S'-formula φ^*, which we get from φ by replacing all atomic subformulas $t_1 \equiv t_2$ by Et_1t_2. Furthermore, let $\Psi_E \subset L_0^{S'}$ be the set of *axioms for equality*,

$$\begin{aligned}\Psi_E := \{&\forall xExx, \forall x\forall y(Exy \to Eyx), \forall x\forall y\forall z(Exy \wedge Eyz \to Exz)\}\\ &\cup \{\forall x_1 \ldots \forall x_n \forall y_1 \ldots \forall y_n(Ex_1y_1 \wedge \ldots \wedge Ex_ny_n \wedge Rx_1 \ldots x_n\\ &\qquad\qquad \to Ry_1 \ldots y_n) \mid R \in S\ n\text{-ary}\}\\ &\cup \{\forall x_1 \ldots \forall x_n \forall y_1 \ldots \forall y_n(Ex_1y_1 \wedge \ldots \wedge Ex_ny_n\\ &\qquad\qquad \to E\,fx_1 \ldots x_n\,fy_1 \ldots y_n) \mid f \in S\ n\text{-ary}\}.\end{aligned}$$

So Ψ_E consists of universal Horn sentences. If the $S \cup \{E\}$-structure $(\mathfrak{A}, E^A)$ is a model of Ψ_E define the S-structure $\mathfrak{A}/_E$, the *quotient structure* of $(\mathfrak{A}, E^A)$ by E^A, as follows:

$A/_E := \{\overline{a} \mid a \in A\}$,
where $\overline{a}$ denotes the equivalence class of a mod E^A;

$R^{A/E} := \{(\overline{a_1}, \ldots, \overline{a_n}) \mid a_1, \ldots, a_n \in A,\ R^A a_1 \ldots a_n\}$;

$f^{A/E}(\overline{a_1}, \ldots, \overline{a_n}) := \overline{f^A(a_1, \ldots, a_n)}$.

For an assignment β in $(\mathfrak{A}, E^A)$ let $\beta/_E$ be the assignment in $\mathfrak{A}/_E$ with

FIGURE XI.7.

$\beta/_E(x) := \overline{\beta(x)}$. Show:
(a) For every $\varphi \in L^S$: $((\mathfrak{A}, E^A), \beta) \models \varphi^*$ iff $(\mathfrak{A}/_E, \beta/_E) \models \varphi$.
Conclude:
(b) For $\Phi \cup \{\psi\} \subset L^S$: $\Phi \vdash \psi$ iff $\Phi \cup \Psi_E \vdash \psi^*$.

§7. Logic Programming

Consider the situation given by the hypotheses of Theorem 6.9: If $\Phi \vdash \exists x_1 \ldots \exists x_m(\psi_0 \wedge \ldots \wedge \psi_l)$ then an algorithm, which systematically produces all H-derivations from GI(Φ) and $(\neg\psi_0 \vee \ldots \vee \neg\psi_l)(\overset{m}{x} \mid \overset{m}{t})$ for all terms $t_1, \ldots, t_m \in T_0^S$, will finally yield an H-derivation of $\emptyset$ from GI(Φ) and $(\neg\psi_0 \vee \ldots \vee \neg\psi_l)(\overset{m}{x} \mid \overset{m}{t})$ for certain terms $t_1, \ldots, t_m \in T_0^S$. Then $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_l)(\overset{m}{x} \mid \overset{m}{t})$, i.e. we have found a "solution" $t_1, \ldots, t_m$ for the existential formula.

PROLOG-programs do not simply work through the terms from T_0^S in a fixed order, independently of the problem, but they search for suitable terms in a "goal-directed" manner, at the same time aiming at efficient substitutions as indicated after Example 6.6. The guiding idea here is to choose the terms "as general as possible and as special as necessary." First, we express the notion of substitution in a suitable form.

7.1 Definition. A *substitutor* is a map $\sigma: V \to T^S$ from the set V of variables to the set of S-terms such that $\sigma(x) = x$ for almost all x.

For a substitutor σ there are $n \in \mathbf{N}$ and pairwise distinct variables $x_1, \ldots, x_n$ with $\sigma(x) = x$ for all $x \neq x_1, \ldots, x_n$. We write $t_i := \sigma(x_i)$ for $i = 1, \ldots, n$. For $t \in T^S$ and $\varphi \in L^S$ we set

$$t\sigma := t\frac{t_1 \ldots t_n}{x_1 \ldots x_n} \quad \text{and} \quad \varphi\sigma := \varphi\frac{t_1 \ldots t_n}{x_1 \ldots x_n} \; (= \varphi(\overset{n}{x} \mid \overset{n}{t}))$$

(by III.8.4, $t\sigma$ and $\varphi\sigma$ are well-defined) and, accordingly, we sometimes write $\frac{t_1 \ldots t_n}{x_1 \ldots x_n}$ for σ. In particular $\sigma(x) = x\sigma$.

Let ι be the substitutor with $\iota(x) = x$ for all x, the so-called *identity* substitutor. For substitutors σ and τ let $\sigma\tau: V \to T^S$ denote the substitutor with $x(\sigma\tau) := (x\sigma)\tau$. For a clause K (of atomic or negated atomic formulas) let $K\sigma := \{\varphi\sigma \mid \varphi \in K\}$. From simple properties of the substitution we immediately obtain:

7.2. (a) *$t\iota = t$ and $\varphi\iota = \varphi$ for all $t \in T^S$ and $\varphi \in L^S$.*
(b) *$t(\sigma\tau) = (t\sigma)\tau$ and $\varphi(\sigma\tau) = (\varphi\sigma)\tau$ for all $t \in T^S$ and quantifier-free $\varphi \in L^S$.*
(c) *$(\rho\sigma)\tau = \rho(\sigma\tau)$ for substitutors ρ, σ, τ.*

(b) and (c) justify the use of parenthesis free notations such as $\varphi\rho\sigma\tau$.

We call a substitutor ξ a *renaming* if ξ is a bijective map from V onto V. If ξ is a renaming, so is the inverse map $\xi^{-1}: V \to V$, and we have $\xi\xi^{-1} = \xi^{-1}\xi = \iota$.

7.3 Definition. Let K_1 and K_2 be clauses. A renaming ξ is called a *separator of K_1 and K_2* :iff $\operatorname{free}(K_1\xi) \cap \operatorname{free}(K_2) = \emptyset$.

For example, $\xi = \frac{v_4 v_5 v_2 v_3}{v_2 v_3 v_4 v_5}$ is a separator of the clauses $\{Pv_0v_2, Pv_3v_2\}$ and $\{Qv_1, Pv_2v_3\}$.

The following example will serve to explain the strategy of carefully handling term instantiations; it anticipates, in a concrete case, the general considerations, which form the subject of the remainder of this section. It also describes the course we take and the goal we want to reach.

7.4 Example. Let $S := \{P, R, f, g, c\}$ with ternary P, binary R and unary f, g and let

$$\Phi := \{\forall x \forall y (Pxyc \to Rygfx), \forall x \forall y\, Pfxyc\}.$$

We look for a proof that $\Phi \vdash \exists x \exists y\, Rfxgy$, as well as for a solution (all solutions) t_1 and t_2 of this existential problem. To apply the method of H-resolutions from 6.9 in a more goal-directed manner and keep the term instantiations as general as possible, we first represent Φ by the "unsubstituted" clauses

$$K_1 := \{\neg Pxyc, Rygfx\} \quad \text{and} \quad K_2 := \{Pfxyc\}$$

and $\exists x \exists y\, Rfxgy$ by the clause

$$N_1 := \{\neg Rfxgy\}.$$

Then, we try to prepare K_1 and N_1 for resolution by a specialization (as weak as possible) of the occurring terms.

For this purpose we choose a separator of K_1 and N_1, say $\xi_1 := \frac{uvxy}{xyuv}$. Then

$$K_1' := K_1\xi_1 = \{\neg Puvc, Rvgfu\}.$$

With the substitutor $\sigma_1 := \frac{fx\ fu}{v\ \ y}$ we get

$$K_1'\sigma_1 = \{\neg Pufxc, Rfxgfu\} \quad \text{and} \quad N_1\sigma_1 = \{\neg Rfxgfu\}.$$

Now $N_2 := \{\neg Pufxc\}$ is a resolvent of $K_1'\sigma_1$ and $N_1\sigma_1$.

For the separator $\xi_2 := \frac{zx}{xz}$ of K_2 and N_2 we have

$$K_2' := K_2\xi_2 = \{Pfzyc\},$$

and with the substitutor $\sigma_2 := \frac{fz\ fx}{u\ \ y}$ we get

$$K_2'\sigma_2 = \{Pfzfxc\} \quad \text{and} \quad N_2\sigma_2 = \{\neg Pfzfxc\},$$

and $\emptyset$ is a resolvent of $K_2'\sigma_2$ and $N_2\sigma_2$.

In Figure XI.8 this derivation is represented schematically.

FIGURE XI.8.

Now $[Rfxgy]\sigma_1\sigma_2 = Rfxgffz$. Indeed we have (and the following considerations will show this in general):

$$\Phi \vdash Rfxgffz$$

and therefore

$$\Phi \vdash \forall x \forall z\, Rfxgffz.$$

So the existential problem $\exists x \exists y\, Rfxgy$ has "$x\sigma_1\sigma_2$ and $y\sigma_1\sigma_2$", i.e. "x and ffz" as a "family of solutions." In particular, $x = gc$ and $y = ffc$ (for $x = gc$ and $z = c$) is a solution in T_0^S.

By the substitutor σ_1, the formulas $Rvgfu$ in K_1' and $\neg Rfxgy$ in N_1 were made equal (except for $\neg$) "in the most efficient manner." In the sense of the following considerations, σ_1 can be called a *general* unifier of $\{Rvgfu, Rfxgy\}$ (cf. Example 7.7(a)).

7.5 Definition. Let K be a clause. K is called *unifiable* :iff there is a substitutor σ for which $K\sigma$ has a single element. In this case σ is called a *unifier* of K.

So the empty clause is not unifiable.

7.6 Lemma on the Unifier. *The following algorithm, the so-called* unification algorithm, *decides for every clause K whether it is unifiable and, in the positive case, yields a* general unifier of K, *i.e. a unifier η of K for which the following holds:*

If σ is a unifier of K then there is a substitutor τ with $\sigma = \eta\tau$.

We call the general unifier, which is the output of the algorithm, the *general unifier of K.*

Starting with (UA1), *the unification algorithm is carried out step by step.*

(UA1) *If K is empty or K contains atomic as well as negated atomic formulas or if the formulas in K do not all contain the same relation symbol, then stop with the answer "K is not unifiable."*

(UA2) *Set $i := 0$ and $\sigma_0 := \iota$.*

(UA3) *If $K\sigma_i$ contains a single element, stop with the answer "K is unifiable and σ_i is a general unifier."*

(UA4) *If $K\sigma_i$ contains more than one element, let ψ_1 and ψ_2 be two distinct literals in $K\sigma_i$ (say the first two with respect to a fixed order, e.g. the lexicographic order). Determine the first place where the words ψ_1 and ψ_2 differ. Let $\S_1$ and $\S_2$ be the letters at this place in ψ_1 and ψ_2, respectively.*

(UA5) *If the (different) letters $\S_1$ and $\S_2$ are function symbols or constants, stop with the answer "K is not unifiable."*

(UA6) *One of the letters $\S_1, \S_2$ is a variable x, say $\S_1$. Determine the term t which starts with $\S_2$ in ψ_2 (t can be a variable; by* II.4.9 *t exists and is uniquely determined).*

(UA7) *If x occurs in t, stop with the answer "K is not unifiable."*

(UA8) *Set* $\sigma_{i+1} := \sigma_i \frac{t}{x}$ *and* $i := i + 1$.

(UA9) *Go to* (UA3).

Proof. We have to show that the unification algorithm stops for every clause K and gives the right answer to the question "Is K unifiable?", and, in the positive case, yields a general unifier.

If the algorithm stops at (UA1) then obviously K is not unifiable. Therefore we may assume that K is a nonempty clause whose literals are all atomic or all negated atomic formulas that, moreover, contain the same relation symbol.

The algorithm will stop for K after finitely many steps: Since applying (UA8) causes the variable x to disappear (x does not occur in t!), the only possible loop (UA3)–(UA9) can be passed through only as often as there are different variables in K.

If the algorithm stops at (UA3), K is unifiable. Therefore, if K is not unifiable, it can stop only after (UA5) or (UA7). Thus the algorithm yields the right answer in case K is not unifiable.

Now let K be unifiable. We will show:

(*) If τ is a unifier of K then for every value i reached by the algorithm there is τ_i with $\sigma_i \tau_i = \tau$.

Then we are done: If k is the last value of i then the clause $K\sigma_k$ is unifiable since $K\sigma_k\tau_k = K\tau$; so the algorithm cannot end with (UA5) or (UA7). (If it would end, e.g., with (UA7), there would be two different literals in $K\sigma_k$ of the form $\ldots x \sim$ and $\ldots t$ __ where $t \neq x$ and x occurs in t; after any substitutions are carried out, there would always be terms of different length starting at the places corresponding to x and t respectively, and hence K would not be unifiable.) Therefore the algorithm must end with (UA3), i.e. σ_k is a unifier and by (*) a general unifier of K.

We prove (*) by induction on i.

For $i = 0$ we set $\tau_0 := \tau$. Then $\sigma_0\tau_0 = \iota\tau_0 = \tau$. In the induction step let $\sigma_i\tau_i = \tau$ and suppose the value $i + 1$ has been reached. By (UA8) we have $\sigma_{i+1} = \sigma_i \frac{t}{x}$. Next, we observe ($K\sigma_i\tau_i$ has a single element!):

$$x\tau_i = t\tau_i. \tag{1}$$

We define τ_{i+1} by

$$y\tau_{i+1} := \begin{cases} y\tau_i & \text{if } y \neq x, \\ x & \text{if } y = x. \end{cases}$$

Since x does not occur in t, we have

$$t\tau_{i+1} = t\tau_i. \tag{2}$$

Now $\frac{t}{x}\tau_{i+1} = \tau_i$; namely, if $y \neq x$, then $y(\frac{t}{x}\tau_{i+1}) = y\tau_{i+1} = y\tau_i$, and if $y = x$ we have $y(\frac{t}{x}\tau_{i+1}) = t\tau_{i+1} = t\tau_i = x\tau_i = y\tau_i$ (cf. (1) and (2)).

Altogether:

$$\sigma_{i+1}\tau_{i+1} = (\sigma_i\tfrac{t}{x})\tau_{i+1} = \sigma_i(\tfrac{t}{x}\tau_{i+1}) = \sigma_i\tau_i = \tau$$

and we have finished the induction step. □

7.7 Examples. Let S be as in 7.4.
(a) Let $K := \{Rvgfu, Rfxgy\}$. The unification algorithm yields successively $\sigma_0 = \iota$, $\sigma_1 = \frac{fx}{v}$, $\sigma_2 = \frac{fx}{v}\frac{fu}{y}$ and the answer: "K is unifiable and $\frac{fx}{v}\frac{fu}{y}$ is a general unifier."

(b) Let $K := \{Pfzyc, Pufxc\}$. The algorithm yields $\sigma_0 = \iota$, $\sigma_1 = \frac{fz}{u}$, $\sigma_2 = \frac{fz}{u}\frac{fx}{y}$ and the answer: "K is unifiable and $\frac{fz}{u}\frac{fx}{y}$ is a general unifier."

(c) Let $K := \{Ryfy, Rzz\}$. We get $\sigma_0 = \iota$, $\sigma_1 = \frac{z}{y}$ (or $\sigma_1 = \frac{y}{z}$) and the answer: "K is not unifiable."

The crux of the resolution in Example 7.4 is expressed in the following notion:

7.8 Definition. Let K, K_1, and K_2 be clauses. K is a *unification resolvent* (written: *U-resolvent*) of K_1 and K_2 :iff there is a separator ξ of K_1 and K_2, and there are $M_1, L_1 \subset K_1$ and $M_2, L_2 \subset K_2$ with the following properties:

(i) L_1 and L_2 are not empty.

(ii) $L_1\xi \cup L_2^F$ is unifiable.

(iii) $K_1 = M_1 \cup L_1$, $K_2 = M_2 \cup L_2$, and $K = (M_1\xi \cup M_2)\eta$,

where η is the general unifier of $L_1\xi \cup L_2^F$.

Schematically, we represent this "U-resolution" by

$$\begin{tikzcd} K_1 \arrow[dr, dash, "\xi"'] & & K_2 \arrow[dl, dash, "\eta"] \\ & K & \end{tikzcd}$$

Since substitutions do not change anything in ground clauses (i.e. in variable free clauses) and since a unifiable ground clause has only one element (with ι as general unifier), we see immediately:

7.9 Remark. *For ground clauses K, K_1, and K_2 the following holds: K is a resolvent of K_1 and K_2 iff K is a U-resolvent of K_1 and K_2.* □

In Example 7.4 we had $K_1 = \{\neg Pxyc, Rygfx\}$ and $N_1 = \{\neg Rfxgy\}$. Let ξ_1 be the separator $\frac{uvxy}{xyuv}$ of K_1 and N_1 chosen there; hence $K_1\xi_1 = \{\neg Puvc, Rvgfu\}$. Then for $L_1 := \{Rygfx\}$ and $L_2 := \{\neg Rfxgy\}$ the clause $L_1\xi_1 \cup L_2^F$ $(= \{Rvgfu, Rfxgy\})$ is unifiable, and $\sigma_1 = \frac{fx\ fu}{v\ \ y}$ is its general unifier. Thus $N_2 = \{\neg Pufxc\}$ is a U-resolvent of K_1 and N_1.

With the following lemma we establish the connection between resolvents and U-resolvents. It gives us the key to Theorem 7.14 on the U-resolution.

7.10 Compatibility Lemma. *Let K_1 and K_2 be clauses. Then:*

(a) *Every resolvent of a ground instance of K_1 and a ground instance of K_2 is a ground instance of a U-resolvent of K_1 and K_2.*

(b) *Every ground instance of a U-resolvent of K_1 and K_2 is a resolvent of a ground instance of K_1 and a ground instance of K_2.*

Proof. (a) Let $K_i\sigma_i$ be a ground instance of K_i $(i = 1, 2)$ and K a resolvent of $K_1\sigma_1$ and $K_2\sigma_2$, i.e. for suitable M_1, M_2 and φ_0

$$K_1\sigma_1 = M_1 \cup \{\varphi_0\}, \quad K_2\sigma_2 = M_2 \cup \{\varphi_0^F\}, \quad K = M_1 \cup M_2.$$

We set

$$\begin{aligned} M_i' &:= \{\varphi \in K_i \mid \varphi\sigma_i \in M_i\} \quad (i = 1, 2), \\ L_1 &:= \{\varphi \in K_1 \mid \varphi\sigma_1 = \varphi_0\},\ L_2 := \{\varphi \in K_2 \mid \varphi\sigma_2 = \varphi_0^F\}. \end{aligned}$$

Then we have

$$\begin{aligned} & K_i = M_i' \cup L_i \quad (i = 1, 2), \\ (*) \quad & M_i'\sigma_i = M_i \quad (i = 1, 2), \\ & L_1\sigma_1 = L_2^F\sigma_2 = \{\varphi_0\}. \end{aligned}$$

Let ξ be a separator of K_1 and K_2 and σ the substitutor with

$$x\sigma := \begin{cases} x\xi^{-1}\sigma_1 & \text{if } x \in \text{free}(K_1\xi), \\ x\sigma_2 & \text{otherwise.} \end{cases}$$

As $\text{free}(K_1\xi) \cap \text{free}(K_2) = \emptyset$ we obtain

$$(+) \qquad \varphi\sigma = \varphi\sigma_2 \text{ for } \varphi \in K_2.$$

Therefore,

$$(L_1\xi \cup L_2^F)\sigma = L_1\xi\xi^{-1}\sigma_1 \cup L_2^F\sigma = L_1\sigma_1 \cup L_2^F\sigma_2 = \{\varphi_0\},$$

hence σ is a unifier of $L_1\xi \cup L_2^F$. Let η be the general unifier and $\sigma = \eta\tau$. Then $K^* := (M_1'\xi \cup M_2')\eta$ is a U-resolvent of K_1 and K_2. Finally, K is a ground instance of K^*; namely $K^*\tau = (M_1'\xi \cup M_2')\sigma \overset{(+)}{=} M_1'\sigma_1 \cup M_2'\sigma_2 \overset{(*)}{=} M_1 \cup M_2 = K$.

So we proved (a). For later purposes we note the following strengthening: Since, for a given finite set Y of variables, we can choose the separator ξ of K_1 and K_2 such that $\text{free}(K_1\xi) \cap Y = \emptyset$, we have shown:

$(\substack{**})$ If K_1 and K_2 are clauses and $K_1\sigma_1$ and $K_2\sigma_2$ ground instances of K_1 and K_2 respectively, and if

$$\begin{tikzcd} K_1\sigma_1 \arrow[dr, dash] & & K_2\sigma_2 \arrow[dl, dash] \\ & K & \end{tikzcd}$$

is a resolution, then for every finite set Y of variables there are K^*, ξ, η and τ so that

$$\begin{tikzcd} K_1 \arrow[dr, dash, "\xi"'] & & K_2 \arrow[dl, dash, "\eta"] \\ & K^* & \end{tikzcd}$$

is a U-resolution and $K = K^*\tau$ as well as $y\eta\tau = (y\sigma =)y\sigma_2$ for $y \in Y$.

(b) Let K be a U-resolvent of K_1 and K_2, say $K = (M_1\xi \cup M_2)\eta$, $K_i = M_i \cup L_i$ $(i = 1, 2)$ and $(L_1\xi \cup L_2^F)\eta = \{\varphi_0\}$, where ξ is a separator of K_1 and K_2, and η the general unifier of $L_1\xi \cup L_2^F$.

Furthermore, let $K\sigma$ be a ground instance of K. We set

$$\sigma_1 := \xi\eta\sigma \quad \text{and} \quad \sigma_2 := \eta\sigma.$$

We can assume that $K_1\sigma_1$ and $K_2\sigma_2$ are ground clauses (otherwise replace σ by $\sigma\tau$ where $\tau(x) \in T_0^S$ for $x \in \text{free}(K_1\sigma_1 \cup K_2\sigma_2)$, and note that $K\sigma\tau = K\sigma$, since $K\sigma$ is a ground instance). Hence the claim follows from

$$K\sigma \text{ is a resolvent of } K_1\sigma_1 \text{ and } K_2\sigma_2.$$

In fact, we only have to note that

$$\begin{aligned} K_1\sigma_1 &= M_1\sigma_1 \cup L_1\sigma_1 = M_1\sigma_1 \cup \{\varphi_0\sigma\}, \\ K_2\sigma_2 &= M_2\sigma_2 \cup L_2\sigma_2 = M_2\sigma_2 \cup \{\varphi_0^F\sigma\} \end{aligned}$$

and

$$M_1\sigma_1 \cup M_2\sigma_2 = (M_1\xi \cup M_2)\eta\sigma = K\sigma. \qquad \square$$

As for the resolution, we now introduce the sets $\text{URes}_i(\mathfrak{K})$ of clauses which can be obtained from clauses in $\mathfrak{K}$ by forming U-resolvents i times.

7.11 Definition. For a set $\mathfrak{K}$ of clauses let

$$\text{URes}(\mathfrak{K}) := \mathfrak{K} \cup \{K \mid \text{ there are } K_1, K_2 \in \mathfrak{K} \text{ such that } K \text{ is a U-resolvent of } K_1 \text{ and } K_2\}.$$

For $i \in \mathbf{N}$ let $\text{URes}_i(\mathfrak{K})$ be defined inductively by

$$\begin{aligned} \text{URes}_0(\mathfrak{K}) &:= \mathfrak{K} \\ \text{URes}_{i+1}(\mathfrak{K}) &:= \text{URes}(\text{URes}_i(\mathfrak{K})). \end{aligned}$$

Finally,

$$\mathrm{URes}_\infty(\mathfrak{K}) := \bigcup_{i\in\mathbf{N}} \mathrm{URes}_i(\mathfrak{K}).$$

First, we want to establish a relationship between the operations URes and Res. For this purpose we extend the notion of ground instance: For a clause $K = \{\varphi_1, \ldots, \varphi_l\}$ with $\mathrm{free}(\varphi_i) \subset \{x_1, \ldots, x_m\}$ for $1 \le i \le l$ let

$$\mathrm{GI}(K) := \left\{\{\varphi_1(\overset{m}{x} \mid \overset{m}{t}), \ldots, \varphi_l(\overset{m}{x} \mid \overset{m}{t})\} \mid t_1, \ldots, t_m \in T_0^S\right\}$$

be the set of *ground instances of* K, and for a set $\mathfrak{K}$ of clauses let

$$\mathrm{GI}(\mathfrak{K}) := \bigcup_{K\in\mathfrak{K}} \mathrm{GI}(K).$$

Since, by the Compatibility Lemma, the operations of forming ground instances and forming U-resolvents can be interchanged, we obtain:

7.12 Lemma. *For a set $\mathfrak{K}$ of clauses the following holds:*
(a) *For all $i \in \mathbf{N}$:* $\mathrm{Res}_i(\mathrm{GI}(\mathfrak{K})) = \mathrm{GI}(\mathrm{URes}_i(\mathfrak{K}))$.
(b) $\mathrm{Res}_\infty(\mathrm{GI}(\mathfrak{K})) = \mathrm{GI}(\mathrm{URes}_\infty(\mathfrak{K}))$.

Proof. (b) follows immediately from (a). We show (a) by induction on i. For $i = 0$ we have

$$\mathrm{Res}_0(\mathrm{GI}(\mathfrak{K})) = \mathrm{GI}(\mathfrak{K}) = \mathrm{GI}(\mathrm{URes}_0(\mathfrak{K})).$$

In the induction step we conclude as follows:

$$\begin{aligned}\mathrm{Res}_{i+1}(\mathrm{GI}(\mathfrak{K})) &= \mathrm{Res}(\mathrm{Res}_i(\mathrm{GI}(\mathfrak{K}))) \\ &= \mathrm{Res}(\mathrm{GI}(\mathrm{URes}_i(\mathfrak{K}))) && \text{(by induction hypothesis)} \\ &= \mathrm{GI}(\mathrm{URes}(\mathrm{URes}_i(\mathfrak{K}))) && \text{(by 7.10)} \\ &= \mathrm{GI}(\mathrm{URes}_{i+1}(\mathfrak{K})).\end{aligned}$$

□

Since ($\emptyset \in \mathrm{GI}(\mathrm{URes}_\infty(\mathfrak{K}))$ iff $\emptyset \in \mathrm{URes}_\infty(\mathfrak{K})$) we get from 7.12:

7.13 Main Lemma on the U-Resolution. *For a set $\mathfrak{K}$ of clauses the following holds:*

$$\emptyset \in \mathrm{Res}_\infty(\mathrm{GI}(\mathfrak{K})) \quad \textit{iff} \quad \emptyset \in \mathrm{URes}_\infty(\mathfrak{K}).$$

□

We translate the result to sets of universal sentences. For a universal sentence φ of the form

$$\forall x_1 \ldots \forall x_m\, ((\varphi_{00} \vee \ldots \vee \varphi_{0l_0}) \wedge \ldots \wedge (\varphi_{s0} \vee \ldots \vee \varphi_{sl_s}))$$

with literals φ_{ij} let

$$\mathfrak{K}(\varphi) := \{\{\varphi_{00}, \ldots, \varphi_{0l_0}\}, \ldots, \{\varphi_{s0}, \ldots, \varphi_{sl_s}\}\}$$

be the set of "unsubstituted" clauses belonging to φ, and for a set Φ of such sentences let $\mathfrak{K}(\Phi)$ be the set of clauses $\bigcup_{\varphi\in\Phi} \mathfrak{K}(\varphi)$. Note that

$$(*) \qquad \mathrm{GI}(\mathfrak{K}(\Phi)) = \mathfrak{K}(\mathrm{GI}(\Phi)).$$

We show:

7.14 Theorem on the U-Resolution. *For a set Φ of universal sentences of the form $\forall x_1 \ldots \forall x_m \varphi$ with quantifier-free φ in* CNF *the following holds:*

$$\Phi \textit{ is satisfiable} \quad \textit{iff} \quad \emptyset \notin \mathrm{URes}_\infty(\mathfrak{K}(\Phi)).$$

Proof. Φ is satisfiable

iff	$\mathrm{GI}(\Phi)$ is propositionally satisfiable	(by 6.4)
iff	$\mathfrak{K}(\mathrm{GI}(\Phi))$ is satisfiable	(by 5.4)
iff	$\mathrm{GI}(\mathfrak{K}(\Phi))$ is satisfiable	(by $(*)$)
iff	$\emptyset \notin \mathrm{Res}_\infty(\mathrm{GI}(\mathfrak{K}(\Phi)))$	(by 5.8)
iff	$\emptyset \notin \mathrm{URes}_\infty(\mathfrak{K}(\Phi))$	(by 7.13). $\square$

To illustrate 7.14, we once again consider Example 6.6: We show the unsatisfiability of the formula

$$\varphi := \forall x \forall y((Rxy \vee Qx) \wedge \neg Rxgx \wedge \neg Qy)$$

using the U-resolution. Compare also with the procedure in 6.6!

First we proceed from φ to the set of clauses

$$\mathfrak{K}(\varphi) = \{\{Rxy, Qx\}, \{\neg Rxgx\}, \{\neg Qy\}\}.$$

Then the U-resolution tree in Figure XI.9 shows the unsatisfiability of φ.

FIGURE XI.9.

We reach the core of logic programming by combining the Horn-resolution with the unification algorithm. First, we transfer the definition 5.9 of H-derivability:

7.15 Definition. Let $\mathfrak{P}$ be a set of positive (first-order) clauses and let N be negative.

(a) A sequence $N_0, \ldots, N_k$ of negative clauses is *a UH-resolution from $\mathfrak{P}$ and N* :iff there are $K_0, \ldots, K_{k-1} \in \mathfrak{P}$ such that $N_0 = N$ and N_{i+1} is a U-resolvent of K_i and N_i for $i < k$.

(b) A negative clause N' is said to be *UH-derivable* from $\mathfrak{P}$ and N :iff there is a UH-Resolution $N_0, \ldots, N_k$ from $\mathfrak{P}$ and N with $N' = N_k$.

(c) For a set $\mathfrak{K}$ of clauses let

$$\mathrm{UHRes}(\mathfrak{K}) := \mathfrak{K} \cup \{N \mid N \text{ is a negative clause, and there is a positive } K_1 \in \mathfrak{K} \text{ and a negative } N_1 \in \mathfrak{K} \text{ such that } N \text{ is a U-resolvent of } K_1 \text{ and } N_1\ \}.$$

Furthermore, set

$$\begin{aligned} \mathrm{UHRes}_0(\mathfrak{K}) &:= \mathfrak{K}, \\ \mathrm{UHRes}_{i+1}(\mathfrak{K}) &:= \mathrm{UHRes}(\mathrm{UHRes}_i(\mathfrak{K})) \text{ and} \\ \mathrm{UHRes}_\infty(\mathfrak{K}) &:= \textstyle\bigcup_{i\in\mathbf{N}} \mathrm{UHRes}_i(\mathfrak{K}). \end{aligned}$$

7.16 Main Lemma on the UH-Resolution. *For a set $\mathfrak{P}$ of positive clauses and a negative clause N the following holds:*

$$\emptyset \in \mathrm{HRes}_\infty(\mathrm{GI}(\mathfrak{P} \cup \{N\})) \quad \textit{iff} \quad \emptyset \in \mathrm{UHRes}_\infty(\mathfrak{P} \cup \{N\}).$$

Proof. Again with the Compatibility Lemma 7.10 one shows that $\mathrm{HRes}_\infty(\mathrm{GI}(\mathfrak{P}\cup\{N\})) = \mathrm{GI}(\mathrm{UHRes}_\infty(\mathfrak{P}\cup\{N\}))$. From this the claim follows immediately. □

Similarly to 7.14, we now obtain:

7.17 Theorem on the UH-Resolution. *Let Φ be a set of positive universal Horn sentences and φ a negative universal Horn sentence. Then:*

$$\Phi \cup \{\varphi\} \textit{is satisfiable} \quad \textit{iff} \quad \emptyset \textit{ is not UH-derivable from } \mathfrak{K}(\Phi) \textit{ and } \mathfrak{K}(\varphi).$$

Proof. Note first that $\mathrm{GI}(\mathfrak{K}(\Phi))$ consists of positive and $\mathrm{GI}(\mathfrak{K}(\varphi))$ of negative clauses. The following statements are equivalent:

(1) $\Phi \cup \{\varphi\}$ is satisfiable.

(2) $\mathrm{GI}(\mathfrak{K}(\Phi) \cup \mathfrak{K}(\varphi)) = \mathrm{GI}(\mathfrak{K}(\Phi)) \cup \mathrm{GI}(\mathfrak{K}(\varphi))$ is propositionally satisfiable.

(3) $\emptyset \notin \mathrm{HRes}_\infty(\mathrm{GI}(\mathfrak{K}(\Phi)) \cup \mathrm{GI}(\mathfrak{K}(\varphi)))$.

(4) $\emptyset \notin \mathrm{UHRes}_\infty(\mathfrak{K}(\Phi) \cup \mathfrak{K}(\varphi))$.

(5) $\emptyset$ is not UH-derivable from $\mathfrak{K}(\Phi)$ and $\mathfrak{K}(\varphi)$.

To verify these equivalences, we give the following remarks: The equivalence of (1) and (2) corresponds to the equivalence of the first and fourth statement in the proof of 7.14; from (3) to (2) we get with 5.11 by using 5.3, from (2) to (3) with the Resolution Theorem 5.8, since $\mathrm{HRes}_\infty(\ldots) \subset \mathrm{Res}_\infty(\ldots)$. The equivalence of (3) and (4) follows with 7.16, the one of (4) and (5) by Definition 7.15. □

For illustration we consider a previous example, namely 7.4. Let

$$\Phi := \{\forall x \forall y (Pxyc \to Rygfx), \forall x \forall y\, Pfxyc\}.$$

To show that

$$(*) \qquad \Phi \vdash \exists x \exists y\, Rfxgy,$$

i.e. that $\Phi \cup \{\forall x \forall y \neg Rfxgy\}$ is unsatisfiable, it suffices to prove that there is a UH-derivation of $\emptyset$ from $\{\{\neg Pxyc, Rygfx\}, \{Pfxyc\}\}$ and $\{\neg Rfxgy\}$ (cf. 7.17). Indeed, the resolution tree in Figure XI.8 on page 228 represents such a UH-derivation. In 7.4 we also mentioned that this derivation yields a solution for the existential proposition (*). Our last aim is to show this in general and, at the same time, prove that we get *all* solutions of the existential problem in this way. Thus we will have reached our goal.

7.18 Theorem on Logic Programming. *Let Φ be a set of positive universal Horn sentences of the form*

(1) $\forall y_1 \ldots \forall y_l \varphi$ *or* (2) $\forall y_1 \ldots \forall y_l (\varphi_0 \wedge \ldots \wedge \varphi_s \rightarrow \varphi)$

with atomic $\varphi, \varphi_0, \ldots, \varphi_s$, and let $\exists x_1 \ldots \exists x_m (\psi_0 \wedge \ldots \wedge \psi_r)$ be a sentence with atomic $\psi_0, \ldots, \psi_r$. Finally, set

$$N := \{\neg\psi_0, \ldots, \neg\psi_r\} \text{ and } \mathfrak{P} := \mathfrak{K}(\Phi)$$

(hence $\mathfrak{P}$ contains the clause $\{\varphi\}$ for sentences in Φ of the form (1) *and the clause $\{\neg\varphi_0, \ldots, \neg\varphi_s, \varphi\}$ for sentences of the form* (2)*). Then the following holds:*

(a) Adequacy:

$$\Phi \vdash \exists x_1 \ldots \exists x_m (\psi_0 \wedge \ldots \wedge \psi_r) \text{ iff } \emptyset \text{ is UH-derivable from } \mathfrak{P} \text{ and } N.$$

(b) Correctness: *If*

is a UH-derivation of $\emptyset$ from $\mathfrak{P}$ and N then

$$\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_r)\eta_1 \ldots \eta_k.$$

(c) Completeness: *If for $t_1, \ldots, t_m \in T_0^S$*

$$\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_r)(\overset{m}{x} \mid \overset{m}{t})$$

then there is a UH-derivation of $\emptyset$ from $\mathfrak{P}$ and N of the form given in (b) *and a substitutor τ with*

$$t_i = x_i \eta_1 \ldots \eta_k \tau, \quad \text{for } i = 1, \ldots, m.$$

If in part (b) exactly the variables $z_1, \ldots, z_s$ occur in the formula $(\psi_0 \wedge \ldots \wedge \psi_r)\eta_1 \ldots \eta_k$ then $\Phi \vdash \forall z_1 \ldots \forall z_s[(\psi_0 \wedge \ldots \wedge \psi_r)\eta_1 \ldots \eta_k]$; therefore $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_r)\eta_1 \ldots \eta_k \tau$ for every substitutor τ. Thus (b) and (c) show that the variable-free terms $t_1, \ldots, t_m$ with $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_r)(\overset{m}{x} \mid \overset{m}{t})$, i.e. the solutions of the existential problem, are exactly the "specializations" of the "families of solutions" $x_1\eta_1 \ldots \eta_k, \ldots, x_m\eta_1 \ldots \eta_k$ given by the UH-derivations.

Proof. Since $\Phi \vdash \exists x_1 \ldots \exists x_m(\psi_0 \wedge \ldots \wedge \psi_r)$ iff not Sat $\Phi \cup \{\forall x_1 \ldots \forall x_m(\neg\psi_0 \vee \ldots \vee \neg\psi_r)\}$, (a) follows immediately from 7.17.

(b): The proof is by induction on the length k of the derivation. For $k = 1$ we have

$$\begin{tikzcd} K_1 \arrow[dr, dash, "\xi_1"'] & & N_1 = N \arrow[dl, dash, "\eta_1"] \\ & \emptyset & \end{tikzcd}$$

Therefore, $K_1\xi_1\eta_1 = N^F\eta_1$, so there must be a sentence $\forall y_1 \ldots \forall y_l\varphi \in \Phi$ such that $K_1 = \{\varphi\}$ and $\varphi\xi_1\eta_1 = \psi_i\eta_1$ for $i = 0, \ldots, r$. Since $\Phi \vdash \forall y_1 \ldots \forall y_l\varphi$ we have $\Phi \vdash \varphi\xi_1\eta_1$ and hence $\Phi \vdash \psi_i\eta_1$ for $i = 0, \ldots, r$, i.e. $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_r)\eta_1$.

For the induction step let $k > 1$ and say $N_2 = \{\neg\chi_0, \ldots, \neg\chi_t\}$ (N_2 is not empty!). The induction hypothesis, applied to the derivation starting with K_2 and N_2, gives

$$\Phi \vdash (\chi_0 \wedge \ldots \wedge \chi_t)\eta_2 \ldots \eta_k. \tag{1}$$

Let $i \leq r$. We show

$$\Phi \vdash \psi_i\eta_1 \ldots \eta_k, \tag{$*$}$$

thus getting our claim $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_r)\eta_1 \ldots \eta_k$. We distinguish two cases:

If $\neg\psi_i\eta_1 \in N_2$, we get $(*)$ immediately from (1).

Now suppose $\neg\psi_i\eta_1 \notin N_2$. Then we have to "lose" $\neg\psi_i\eta_1$ in the resolution step leading to N_2. So in Φ there is a sentence $\forall y_1 \ldots \forall y_l(\varphi_1 \wedge \ldots \wedge \varphi_s \to \varphi)$ (i.e. $\forall y_1 \ldots \forall y_l\varphi$ in case $s = 0$) with $K_1 = \{\neg\varphi_1, \ldots, \neg\varphi_s, \varphi\}$ and

$$\varphi\xi_1\eta_1 = \psi_i\eta_1, \tag{2}$$

$$\neg\varphi_j\xi_1\eta_1 \in N_2 \quad \text{for } 1 \leq j \leq s, \tag{3}$$

therefore by (3) and (1):

$$\Phi \vdash \varphi_j\xi_1\eta_1\eta_2 \ldots \eta_k \quad \text{for } 1 \leq j \leq s. \tag{4}$$

Since $\Phi \vdash \forall y_1 \ldots \forall y_l(\varphi_1 \wedge \ldots \wedge \varphi_s \to \varphi)$ we get

$$\Phi \vdash (\neg\varphi_1 \vee \ldots \vee \neg\varphi_s \vee \varphi)\xi_1\eta_1\eta_2 \ldots \eta_k,$$

thus by (4)

$$\Phi \vdash \varphi\xi_1\eta_1\eta_2\ldots\eta_k,$$

and with (2) this leads to $(*)$.

(c): For technical reasons we make a slightly weaker assumption on the terms:

For $t_1,\ldots,t_m \in T^S$ set $\rho_1 := \frac{t_1\ldots t_m}{x_1\ldots x_m}$ and $N_1 := N = \{\neg\psi_0,\ldots,\neg\psi_r\}$; suppose that $\Phi \vdash (\psi_0 \wedge \ldots \wedge \psi_r)\rho_1$ and that $N' := N_1\rho_1$ is a ground clause.

Then, by 6.4, $\mathfrak{K}(\mathrm{GI}(\Phi)) \cup \{N_1\rho_1\}$ is not propositionally satisfiable. So by 5.10 there is an H-derivation of $\emptyset$ from $\mathfrak{K}(\mathrm{GI}(\Phi))$ and N_1' as shown in Figure XI.10.

FIGURE XI.10.

Here the K_j' and the N_j' are ground clauses and, say, $K_j' = K_j\sigma_j$ with suitable clauses $K_j \in \mathfrak{P}$. We show: For every finite set X of variables there is a UH-derivation as in Figure XI.11 of $\emptyset$ from $\mathfrak{P} = \mathfrak{K}(\Phi)$ and $N = N_1$ such that there exists a substitutor τ with

$$x\eta_1\ldots\eta_k\tau = x\rho_1 \quad \text{for } x \in X.$$

Then, for $X := \{x_1,\ldots,x_m\}$ we get

$$x_i\eta_1\ldots\eta_k\tau = t_i \quad (1 \le i \le m),$$

and we are done.

We show the existence of a corresponding UH-derivation by induction on the length of k.

FIGURE XI.11.

For $k = 1$ we have the derivation

The claim follows immediately from $\binom{*}{*}$ in the proof of 7.10 by setting

$$K_2 := N_1, \quad \sigma_2 := \rho_1, \quad K := \emptyset \quad \text{and} \quad Y := X.$$

In the induction step let $k \geq 2$. For the first step of the H-derivation in Figure XI.10 we choose, again with $\binom{*}{*}$ in 7.10, ξ_1, η_1, N_2 and ρ_2 so that

and

$$(*) \qquad x\eta_1\rho_2 = x\rho_1 \text{ for } x \in X$$

as well as $N_2' = N_2\rho_2$. We apply the induction hypothesis to the part of the H-derivation in Figure XI.11 starting with K_2' and N_2' and to

$$Y := \text{var}(\{x\eta_1 \mid x \in X\}).$$

Then we get the UH-derivation in Figure XI.12 and a substitutor τ for which

$$y\eta_2 \ldots \eta_k\tau = y\rho_2 \text{ for } y \in Y,$$

FIGURE XI.12.

hence by $(*)$ and the definition of Y,

$$x\eta_1\eta_2\ldots\eta_k\tau = x\eta_1\rho_2 = x\rho_1 \text{ for } x \in X.$$

Thus, everything is proved. □

XII

An Algebraic Characterization of Elementary Equivalence

The greater part of our exposition so far has been devoted to the development and investigation of first-order logic. We can justify the dominant role assumed by first-order logic in several ways:

(a) First-order logic is in principle sufficient for mathematics.

(b) The intuitive concept of proof and the consequence relation can be adequately described by a formal notion of proof, which is given by means of a calculus.

(c) A number of semantic results such as the Compactness Theorem or the Löwenheim-Skolem Theorem leads to an enrichment of mathematical methods.

However, in contrast to these positive aspects, one also has to take into account that the limited expressive power of first-order language often requires clumsy formulations. In particular, it forces us to make explicit reference to set theory to an extent not usual in mathematical practice. For this reason we were led to seek other systems with greater expressive power but still satisfying conditions (b) and (c).

In Chapter IX we introduced a number of extensions of first-order logic ($\mathcal{L}_{\mathrm{II}}$, $\mathcal{L}_{\mathrm{II}}^{w}$, $\mathcal{L}_{\omega_1\omega}$, $\mathcal{L}_Q$) and investigated their semantic properties. In each case we found that not all the properties mentioned in (c) are available.

In the previous chapter we obtained negative results of a more syntactic nature. For example, we saw that for $\mathcal{L}_{\mathrm{II}}$ and for $\mathcal{L}_{\mathrm{II}}^{w}$ there is no possibility of adequately describing the notion of proof by means of a calculus, and not even of listing the valid formulas; hence in these cases we also have to make concessions concerning (b).

The discussion in the last two chapters will show that these negative results have a deeper reason: Having made precise the concept of a "logical system" we shall prove in Chapter XIII that no logical system with more expressive power than first-order logic can meet the conditions of (b) and (c).

In the present chapter we introduce a useful tool for these investigations. Recall that two structures are elementarily equivalent if they satisfy the

same first-order sentences. We now present a purely algebraic characterization of elementary equivalence. This characterization is useful not only for our present purpose, but also in other contexts. For example, in many cases it can serve to verify that two given structures $\mathfrak{A}$ and $\mathfrak{B}$ are elementarily equivalent, in a simpler way than by proving directly that $\mathfrak{A}$ and $\mathfrak{B}$ satisfy the same first-order sentences. This establishes one of the most important methods to prove the completeness of theories. At the same time, one obtains a tool to show that certain properties are not expressible in first-order language.

§1. Finite and Partial Isomorphisms

In this section we provide the concepts we need in order to formulate the algebraic characterization of elementary equivalence. We refer to a fixed symbol set S. The domain of a map p is denoted by $\mathrm{dom}(p)$, its range, that is the set $\{p(x) \mid x \in \mathrm{dom}(p)\}$, by $\mathrm{rg}(p)$.

1.1 Definition. Let $\mathfrak{A}$ and $\mathfrak{B}$ be (S-)structures and let p be a map. We call p a *partial isomorphism from* $\mathfrak{A}$ *to* $\mathfrak{B}$ if and only if $\mathrm{dom}(p) \subset A, \mathrm{rg}(p) \subset B$, and p has the following properties:

(a) p is injective.

(b) p is homomorphic in the following sense:

(1) For n-ary $P \in S$ and $a_1, \ldots, a_n \in \mathrm{dom}(p)$,

$$P^{\mathfrak{A}} a_1 \ldots a_n \quad \text{iff} \quad P^{\mathfrak{B}} p(a_1) \ldots p(a_n).$$

(2) For n-ary $f \in S$ and $a_1, \ldots, a_n, a \in \mathrm{dom}(p)$,

$$f^{\mathfrak{A}}(a_1, \ldots, a_n) = a \quad \text{iff} \quad f^{\mathfrak{B}}(p(a_1), \ldots, p(a_n)) = p(a).$$

(3) For $c \in S$ and $a \in \mathrm{dom}(p)$,

$$c^{\mathfrak{A}} = a \quad \text{iff} \quad c^{\mathfrak{B}} = p(a).$$

We write $\mathrm{Part}(\mathfrak{A}, \mathfrak{B})$ for the set of partial isomorphisms from $\mathfrak{A}$ to $\mathfrak{B}$.

1.2 Examples and Comments. (a) The empty map, i.e., the map with empty domain, is a partial isomorphism from $\mathfrak{A}$ to $\mathfrak{B}$.

(b) The map p with $\mathrm{dom}(p) = \{2, 3\}$ and $p(2) = 2$, $p(3) = 6$ is a partial isomorphism from the additive group $(\mathbf{R}, +, 0)$ of real numbers to the additive group $(\mathbf{Z}, +, 0)$ of integers. On the other hand, the map q with $\mathrm{dom}(q) = \{2, 3\}$ and $q(2) = 1$, $q(3) = 2$ is not a partial isomorphism from $(\mathbf{R}, +, 0)$ to $(\mathbf{Z}, +, 0)$, because $2 + 2 \neq 3$ but $q(2) + q(2) = q(3)$.

(c) If S is relational, that is, if S contains only relation symbols, then for $a_0, \ldots, a_{r-1} \in A$ and $b_0, \ldots, b_{r-1} \in B$ the following statements are equivalent:

(*) By setting

$$p(a_i) := b_i \text{ for } i < r$$

a partial isomorphism from $\mathfrak{A}$ to $\mathfrak{B}$ is determined (where $\mathrm{dom}(p) = \{a_0, \ldots, a_{r-1}\}$ and $\mathrm{rg}(p) = \{b_0, \ldots, b_{r-1}\}$).

(**) For every atomic formula $\psi \in L_r^S$,

$$\mathfrak{A} \models \psi[a_0, \ldots, a_{r-1}] \quad \text{iff} \quad \mathfrak{B} \models \psi[b_0, \ldots, b_{r-1}].$$

Proof. First we note that for $i, j < r$

$$\begin{aligned} a_i = a_j \quad &\text{iff} \quad \mathfrak{A} \models v_i \equiv v_j[a_0, \ldots, a_{r-1}], \\ b_i = b_j \quad &\text{iff} \quad \mathfrak{B} \models v_i \equiv v_j[b_0, \ldots, b_{r-1}], \end{aligned} \tag{1}$$

and that for n-ary $P \in S$ and $i_1, \ldots, i_n < r$

$$\begin{aligned} P^{\mathfrak{A}} a_{i_1} \ldots a_{i_n} \quad &\text{iff} \quad \mathfrak{A} \models P v_{i_1} \ldots v_{i_n}[a_0, \ldots, a_{r-1}], \\ P^{\mathfrak{B}} b_{i_1} \ldots b_{i_n} \quad &\text{iff} \quad \mathfrak{B} \models P v_{i_1} \ldots v_{i_n}[b_0, \ldots, b_{r-1}]. \end{aligned} \tag{2}$$

Now, if (**) holds, then by (1) and the fact that

$$\mathfrak{A} \models v_i \equiv v_j[a_0, \ldots, a_{r-1}] \quad \text{iff} \quad \mathfrak{B} \models v_i \equiv v_j[b_0, \ldots, b_{r-1}],$$

p is well-defined and injective. Since

$$\mathfrak{A} \models P v_{i_1} \ldots v_{i_n}[a_0, \ldots, a_{r-1}] \quad \text{iff} \quad \mathfrak{B} \models P v_{i_1} \ldots v_{i_n}[b_0, \ldots, b_{r-1}]$$

and by (2), p is also homomorphic.

Similarly, one can use (1) and (2) to deduce (**) from (*). □

(d) Note that the equivalence in (c) is no longer true if S contains function symbols and constants. For example, for the partial isomorphism p in (b)

$$\text{not} \quad (\mathbf{R}, +, 0) \models v_0 + (v_0 + v_0) \equiv v_1[2, 3],$$

but on the other hand

$$(\mathbf{Z}, +, 0) \models v_0 + (v_0 + v_0) \equiv v_1[p(2), p(3)].$$

(e) The following example shows that even for relational S a partial isomorphism does not in general preserve the validity of formulas with quantifiers.

Let $S = \{<\}$ and let q_0 be the partial isomorphism from $(\mathbf{R}, <)$ to $(\mathbf{Z}, <)$ such that $\mathrm{dom}(q_0) = \{2, 3\}$ and $q_0(2) = 3$, $q_0(3) = 4$. Then

$$(\mathbf{R}, <) \models \exists v_2(v_0 < v_2 \wedge v_2 < v_1)[2, 3],$$

but

$$\text{not} \quad (\mathbf{Z}, <) \models \exists v_2(v_0 < v_2 \wedge v_2 < v_1)[q_0(2), q_0(3)].$$

If p is a partial isomorphism from $(\mathbf{R}, <)$ to $(\mathbf{Z}, <)$ such that $\mathrm{dom}(p) = \{a, b\}$ and $a < b$, then we always have

$$(\mathbf{R}, <) \models \exists v_2(v_0 < v_2 \wedge v_2 < v_1)[a, b],$$

since, for example,

$$(\mathbf{R}, <) \models (v_0 < v_2 \wedge v_2 < v_1)[a, b, \tfrac{a+b}{2}].$$

In this case the validity of

$$(+) \qquad (\mathbf{Z}, <) \models \exists v_2(v_0 < v_2 \wedge v_2 < v_1)[p_0(a), p_0(b)]$$

is equivalent to the existence of a partial isomorphism q from $(\mathbf{R}, <)$ to $(\mathbf{Z}, <)$ which extends p and has $\frac{a+b}{2}$ in its domain. For, if such a q exists, then $(+)$ holds, since

$$(\mathbf{Z}, <) \models (v_0 < v_2 \wedge v_2 < v_1)[q(a), q(b), q(\tfrac{a+b}{2})];$$

conversely, if $(+)$ is satisfied and, say,

$$(\mathbf{Z}, <) \models (v_0 < v_2 \wedge v_2 < v_1)[p(a), p(b), d],$$

then the extension q of p with $\mathrm{dom}(q) = \{a, b, \frac{a+b}{2}\}$ and $q(\frac{a+b}{2}) = d$ is such a partial isomorphism.

This argument indicates that the truth of formulas with quantifiers is preserved under partial isomorphisms, provided that these admit certain extensions. It embodies the basic idea behind the algebraic characterization of elementary equivalence: The elementary equivalence of structures amounts to the existence of extensions of certain partial isomorphisms.

In the following definitions we introduce the algebraic notions we need. For maps we use the set-theoretical notation, i.e., we identify a map p with its graph $\{(a, p(a)) \mid a \in \mathrm{dom}(p)\}$. Then, for example, $p \subset q$ means that q is an extension of p.

1.3 Definition. $\mathfrak{A}$ and $\mathfrak{B}$ are said to be *finitely isomorphic* (written: $\mathfrak{A} \cong_f \mathfrak{B}$) :iff there is a sequence $(I_n)_{n \in \mathbf{N}}$ with the following properties:

(a) Every I_n is a nonempty set of partial isomorphisms from $\mathfrak{A}$ to $\mathfrak{B}$.
(b) (*Forth-property*) For every $p \in I_{n+1}$ and $a \in A$ there is $q \in I_n$ such that $q \supset p$ and $a \in \mathrm{dom}(q)$.
(c) (*Back-property*) For every $p \in I_{n+1}$ and $b \in B$ there is $q \in I_n$ such that $q \supset p$ and $b \in \mathrm{rg}(q)$.

Informally we can express (b) and (c) as follows: partial isomorphisms in I_{n+1} can be extended $(n+1)$ times; the corresponding extensions lie in I_n, $I_{n-1}, \ldots, I_1$, and I_0, respectively.

If $(I_n)_{n \in \mathbf{N}}$ has the properties (a), (b), and (c), we write $(I_n)_{n \in \mathbf{N}} : \mathfrak{A} \cong_f \mathfrak{B}$.

1.4 Definition. $\mathfrak{A}$ and $\mathfrak{B}$ are said to be *partially isomorphic* (written: $\mathfrak{A} \cong_p \mathfrak{B}$) :iff there is a set I such that

(a) I is a nonempty set of partial isomorphisms from $\mathfrak{A}$ to $\mathfrak{B}$.
(b) (*Forth-property*) For every $p \in I$ and $a \in A$ there is $q \in I$ such that $q \supset p$ and $a \in \mathrm{dom}(q)$.
(c) (*Back-property*) For every $p \in I$ and $b \in B$ there is $q \in I$ such that $q \supset p$ and $b \in \mathrm{rg}(q)$.

Thus, the conditions (a), (b), and (c) amount to $(I)_{n\in\mathbf{N}}: \mathfrak{A} \cong_f \mathfrak{B}$ for the constant sequence $(I)_{n\in\mathbf{N}}$.

If (a), (b), and (c) are satisfied for I we write $I: \mathfrak{A} \cong_p \mathfrak{B}$.

The following lemma lists the relations between the various notions of isomorphism.

1.5 Lemma. (a) *If* $\mathfrak{A} \cong \mathfrak{B}$ *then* $\mathfrak{A} \cong_p \mathfrak{B}$.
(b) *If* $\mathfrak{A} \cong_p \mathfrak{B}$ *then* $\mathfrak{A} \cong_f \mathfrak{B}$.
(c) *If* $\mathfrak{A} \cong_f \mathfrak{B}$ *and* A *is finite then* $\mathfrak{A} \cong \mathfrak{B}$.
(d) *If* $\mathfrak{A} \cong_p \mathfrak{B}$ *and* A *and* B *are at most countable then* $\mathfrak{A} \cong \mathfrak{B}$.

Proof. (a) If $\pi : \mathfrak{A} \cong \mathfrak{B}$, then $I: \mathfrak{A} \cong_p \mathfrak{B}$ for $I = \{\pi\}$.

(b) If $I : \mathfrak{A} \cong_p \mathfrak{B}$ then $(I)_{n\in\mathbf{N}}: \mathfrak{A} \cong_f \mathfrak{B}$.

(c) Suppose $(I_n)_{n\in\mathbf{N}}: \mathfrak{A} \cong_f \mathfrak{B}$, and suppose A has exactly r elements, say, $A = \{a_1, \ldots, a_r\}$. We choose $p \in I_{r+1}$. If we suitably apply the forth-property r times we obtain a $q \in I_1$ such that $a_1, \ldots, a_r \in \mathrm{dom}(q)$, i.e. $\mathrm{dom}(q) = A$. If $\mathrm{rg}(q) \neq B$ and $b \in B$ with $b \notin \mathrm{rg}(q)$, then by the back-property there would be a proper extension q' of q in I_0 such that $b \in \mathrm{rg}(q')$. Since $\mathrm{dom}(q) = A$, this is not possible. Therefore $\mathrm{rg}(q) = B$ and thus $q : \mathfrak{A} \cong \mathfrak{B}$.

(d) Suppose $I : \mathfrak{A} \cong_p \mathfrak{B}$, $A = \{a_0, a_1, \ldots\}$ and $B = \{b_0, b_1, \ldots\}$. Starting from an arbitrary $p_0 \in I$, by repeated application of the back- and forth-properties, we obtain extensions $p_1, p_2, \ldots$ in I such that $a_0 \in \mathrm{dom}(p_1)$, $b_0 \in \mathrm{rg}(p_2)$, $a_1 \in \mathrm{dom}(p_3)$, $b_1 \in \mathrm{rg}(p_4)$, $\ldots$, that is, a sequence $(p_n)_{n\in\mathbf{N}}$ of partial isomorphisms in I such that for all n

(1) $p_n \subset p_{n+1}$;

(2) if n is odd, say $n = 2r + 1$, then $a_r \in \mathrm{dom}(p_n)$;

(3) if n is even, say $n = 2r + 2$, then $b_r \in \mathrm{rg}(p_n)$.

By (1), $p := \bigcup_{n\in\mathbf{N}} p_n$ is a partial isomorphism from $\mathfrak{A}$ to $\mathfrak{B}$. As $\mathrm{dom}(p) = A$ (by (2)) and $\mathrm{rg}(p) = B$ (by (3)), we have $p: \mathfrak{A} \cong \mathfrak{B}$. □

Part (d) of 1.5 is an abstract version of the following theorem of Cantor.

1.6 Theorem. *Any two countable dense orderings (without endpoints) are isomorphic.*

Here a *dense ordering* is a $\{<\}$-structure which is a model of Φ_{dord}. Φ_{dord} contains the ordering axioms (compare III.6.4) together with the following sentences ("density"):

$$\forall x \forall y (x < y \to \exists z (x < z \wedge z < y)), \quad \forall x \exists y\, x < y, \quad \forall x \exists y\, y < x.$$

$(\mathbf{R}, <)$ and $(\mathbf{Q}, <)$ are dense orderings. By contrast, $(\mathbf{Z}, <)$ is not a dense ordering.

Cantor's theorem follows from 1.5(d) and

1.7 Lemma. *If $\mathfrak{A} = (A, <^A)$ and $\mathfrak{B} = (B, <^B)$ are dense orderings, then $I: \mathfrak{A} \cong_p \mathfrak{B}$ for $I = \{p \mid p \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B}), \mathrm{dom}(p)$ is finite$\}$.*

Proof. Since $p = \emptyset$ is in I, $I \neq \emptyset$. I satisfies the forth-property. For, if $p \in I$, $\mathrm{dom}(p) = \{a_1, \ldots, a_n\}$ and $a \in A$, then because $\mathfrak{B}$ is dense there is an element $b \in B$ which is related to $p(a_1), \ldots, p(a_n)$ in the ordering $\mathfrak{B}$ in the same manner as a is related to $a_1, \ldots, a_n$ in the ordering $\mathfrak{A}$. Then the map $q := p \cup \{(a, b)\}$ is an extension of p which is defined for a and lies in I. The back-property follows analogously, using the fact that $\mathfrak{A}$ is dense. □

1.8 Example. Suppose $S = \{\boldsymbol{\sigma}, 0\}$ and let Φ_σ consist of the "successor axioms"

$$\forall x(\neg x \equiv 0 \leftrightarrow \exists y\, \boldsymbol{\sigma} y \equiv x), \quad \forall x \forall y(\boldsymbol{\sigma} x \equiv \boldsymbol{\sigma} y \to x \equiv y),$$
$$\text{and for every } m \geq 1: \quad \forall x \neg \underbrace{\boldsymbol{\sigma} \ldots \boldsymbol{\sigma}}_{m-\text{times}} x \equiv x.$$

The structure $\mathfrak{N}_\sigma$ (cf. III.7.3(2)) is a model of Φ_σ. We show that any two models of Φ_σ are finitely isomorphic. First, we fix the following notation: For a model $\mathfrak{A}$ of Φ_σ and $a \in A$ we set $a^{(m)} := \underbrace{\boldsymbol{\sigma}^A \ldots \boldsymbol{\sigma}^A}_{m-\text{times}}(a)$. For every $n \in \mathbf{N}$ we define a "distance function" d_n on $A \times A$ by

$$d_n(a, a') := \begin{cases} m & \text{if } a^{(m)} = a' \text{ and } m \leq 2^n \\ -m & \text{if } a'^{(m)} = a \text{ and } m \leq 2^n \\ \infty & \text{otherwise.} \end{cases}$$

Now suppose $\mathfrak{A}$ and $\mathfrak{B}$ are models of Φ_σ. We show that $(I_n)_{n \in \mathbf{N}} : \mathfrak{A} \cong_f \mathfrak{B}$, where

$$I_n := \{p \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B}) \mid \mathrm{dom}(p) \text{ is finite}, \quad 0^A \in \mathrm{dom}(p),$$
$$\text{and for all } a, a' \in \mathrm{dom}(p),\ d_n(a, a') = d_n(p(a), p(a'))\}.$$

Thus, a partial isomorphism in I_n preserves the "d_n-distances". First, we have $I_n \neq \emptyset$ since $(0^A, 0^B) \in I_n$. We sketch a proof of the forth-property for $(I_n)_{n \in \mathbf{N}}$ (the back-property can be proved analogously). Suppose $p \in I_{n+1}$ and $a \in A$. We distinguish two cases, depending on whether or not the condition

$$(*) \qquad \text{There is an } a' \in \mathrm{dom}(p) \text{ such that } |d_n(a', a)| \leq 2^n$$

holds. If $(*)$ holds and, say, $a' \in \mathrm{dom}(p)$ with $|d_n(a', a)| \leq 2^n$, then we choose the $b \in B$ with $d_n(p(a'), b) = d_n(a', a)$. From $p \in I_{n+1}$ it follows easily that $q := p \cup \{(a, b)\}$ is a partial isomorphism preserving the d_n-distances, hence $q \in I_n$. If $(*)$ does not hold, we choose an arbitrary element b such that $d_n(p(a'), b) = \infty$ for all $a' \in \mathrm{dom}(p)$ (such an element b must exist since every model Φ_σ is infinite!). Now it is easy to see that $q := p \cup \{(a, b)\} \in I_n$. □

1.9 Exercise. Let $S = \emptyset$. Show that any two infinite S-structures are partially isomorphic.

1.10 Exercise. (a) Give an example of structures which are partially isomorphic but not isomorphic.
(b) Give an example of structures which are finitely isomorphic but not partially isomorphic.

1.11 Exercise. Give an uncountable model of the system Φ_σ of axioms in 1.8.

1.12 Exercise. Define $\mathfrak{A}$ to be *finitely embeddable* in $\mathfrak{B}$ (written: $\mathfrak{A} \to_f \mathfrak{B}$) if there is a sequence $(I_n)_{n\in\mathbf{N}}$ with the properties 1.3(a) and (b). Analogously define $\mathfrak{A}$ to be *partially embeddable* in $\mathfrak{B}$, written: $\mathfrak{A} \to_p \mathfrak{B}$. Show:
(a) If $\mathfrak{A} \to_f \mathfrak{B}$ and A is finite then $\mathfrak{A}$ is embeddable in $\mathfrak{B}$, i.e. $\mathfrak{A}$ is isomorphic to a substructure of $\mathfrak{B}$.
(b) If $\mathfrak{A} \to_p \mathfrak{B}$ and A is at most countable then $\mathfrak{A}$ is embeddable in $\mathfrak{B}$.
(c) If $\mathfrak{A}$ is an ordering and $\mathfrak{B}$ is a dense ordering then $\mathfrak{A} \to_p \mathfrak{B}$.

§2. Fraïssé's Theorem

Using the concepts introduced in Section 1, we now formulate the main result of this chapter.

2.1 Fraïssé's Theorem. *Let S be a finite symbol set and $\mathfrak{A}$, $\mathfrak{B}$ S-structures. Then*

$$\mathfrak{A} \equiv \mathfrak{B} \quad \textit{iff} \quad \mathfrak{A} \cong_f \mathfrak{B}.$$

Note that Fraïssé's Theorem provides us with a characterization of elementary equivalence which does not refer to the first-order language.

Before proving the theorem (in the next section) we give several examples showing how it can be used to check the elementary equivalence of structures and the completeness of theories.

2.2 Proposition. (a) *Any two dense orderings are elementarily equivalent. In particular,* $(\mathbf{R}, <) \equiv (\mathbf{Q}, <)$.
(b) *Any two $\{\sigma, 0\}$-structures satisfying the axioms in* 1.8 *are elementarily equivalent.*

Proof. (a) follows from 2.1, since (cf. 1.7) any two dense orderings are partially isomorphic, and thus, also finitely isomorphic; (b) follows analogously by means of 1.8. □

For applications on completeness of theories we need the following simple criterion.

2.3 Lemma. *For a theory $T \subset L_0^S$ the following are equivalent:*
(a) *T is complete, i.e., for every S-sentence φ either $\varphi \in T$ or $\neg\varphi \in T$.*
(b) *Any two models of T are elementarily equivalent.*

Proof. Suppose first that (a) holds, and let $\mathfrak{A}$ and $\mathfrak{B}$ be models of T. For any S-sentence φ either $\varphi \in T$ or $\neg\varphi \in T$. If $\varphi \in T$ then $\mathfrak{A} \models \varphi$ and $\mathfrak{B} \models \varphi$; if $\neg\varphi \in T$ then $\mathfrak{A} \models \neg\varphi$ and $\mathfrak{B} \models \neg\varphi$. Thus $\mathfrak{A} \models \varphi$ iff $\mathfrak{B} \models \varphi$.

Conversely, let φ be an S-sentence and suppose $\varphi \notin T$. Since T is a theory, $T \models \varphi$ does not hold, and therefore, there is a model $\mathfrak{A}$ of $T \cup \{\neg\varphi\}$. By (b) every model of T is elementarily equivalent to $\mathfrak{A}$, and thus is a model of $\neg\varphi$. Hence $T \models \neg\varphi$ and, since T is a theory, $\neg\varphi \in T$. □

From 2.2, with the aid of 2.3 and X.6.5, we obtain

2.4 Proposition. (a) *The theory $\Phi_{\text{dord}}^{\models}$ of dense orderings is complete and R-decidable. (Thus, for example, $\Phi_{\text{dord}}^{\models} = \text{Th}(\mathbf{R}, <)$.)*
(b) *The theory $\Phi_{\sigma}^{\models}$ of successor structures is complete and R-decidable. (Thus, for example, $\Phi_{\sigma}^{\models} = \text{Th}(\mathbf{N}, \sigma)$.)*

Preparatory to the proof of Fraïssé's Theorem we show that we can restrict ourselves to relational symbol sets. (A direct proof for arbitrary finite symbol sets is sketched in Exercise 3.15.)

Let S be an arbitrary symbol set. As done before VIII.1.5, we choose, for each n-ary $f \in S$, a new $(n+1)$-ary relation symbol F and, for each $c \in S$, a new unary relation symbol C. Let S^r consist of the relation symbols from S together with the new relation symbols. S^r is relational. For an S-structure $\mathfrak{A}$, let $\mathfrak{A}^r$ be the S^r-structure obtained from $\mathfrak{A}$, replacing functions and constants by their graphs (as in VIII.1).

When defining partial isomorphisms we treated functions and constants in such a way (cf. 1.1) that for arbitrary structures $\mathfrak{A}$ and $\mathfrak{B}$,

$$\text{Part}(\mathfrak{A}, \mathfrak{B}) = \text{Part}(\mathfrak{A}^r, \mathfrak{B}^r).$$

From this we obtain

$$(*) \qquad \mathfrak{A} \cong_f \mathfrak{B} \quad \text{iff} \quad \mathfrak{A}^r \cong_f \mathfrak{B}^r.$$

In VIII.1.4 we showed that

$$(**) \qquad \mathfrak{A} \equiv \mathfrak{B} \quad \text{iff} \quad \mathfrak{A}^r \equiv \mathfrak{B}^r.$$

Thus, in proving Fraïssé's Theorem, we can restrict to *relational* symbol sets. For, if $\mathfrak{A}$ and $\mathfrak{B}$ are given, it follows from

$$\mathfrak{A}^r \equiv \mathfrak{B}^r \quad \text{iff} \quad \mathfrak{A}^r \cong_f \mathfrak{B}^r$$

by $(*)$ and $(**)$ that

$$\mathfrak{A} \equiv \mathfrak{B} \quad \text{iff} \quad \mathfrak{A} \cong_f \mathfrak{B}.$$

2.5 Exercise. Show that for $S = \emptyset$ the theory $\{\varphi_{\geq n} \mid n \geq 2\}^{\models}$ of infinite sets is complete and R-decidable.

2.6 Exercise. Let $S = \{P_n \mid n \in \mathbf{N}\}$ be a set of unary relation symbols. Define the S-structures $\mathfrak{A}$ and $\mathfrak{B}$ as follows: $A := \mathbf{N}$, $B := \mathbf{N} \cup \{\infty\}$, $P_n^{\mathfrak{A}} := \{m \mid m \in \mathbf{N}, m \geq n\}$, $P_n^{\mathfrak{B}} := \{m \mid m \in \mathbf{N}, m \geq n\} \cup \{\infty\}$. Show that $\mathfrak{A} \equiv \mathfrak{B}$ but not $\mathfrak{A} \cong_f \mathfrak{B}$. Thus Fraïssé's Theorem is, in general, not true for infinite symbol sets. Note, on the other hand, that for arbitrary S and S-structures $\mathfrak{A}, \mathfrak{B}$ we have ($\mathfrak{A} \equiv \mathfrak{B}$ iff for every finite $S_0 \subset S$, $\mathfrak{A}|_{S_0} \equiv \mathfrak{B}|_{S_0}$), and therefore ($\mathfrak{A} \equiv \mathfrak{B}$ iff for every finite $S_0 \subset S$, $\mathfrak{A}|_{S_0} \cong_f \mathfrak{B}|_{S_0}$).

§3. Proof of Fraïssé's Theorem

In the sequel we prove Fraïssé's Theorem. Let S be a fixed *finite*, *relational* symbol set.

For a formula φ we define the *quantifier rank* $\mathrm{qr}(\varphi)$ of φ to be the maximum number of nested quantifiers occurring in it:

$$\begin{aligned}
\mathrm{qr}(\varphi) &:= 0, \text{ if } \varphi \text{ atomic;} \\
\mathrm{qr}(\neg\varphi) &:= \mathrm{qr}(\varphi); \\
\mathrm{qr}(\varphi \vee \psi) &:= \max\{\mathrm{qr}(\varphi), \mathrm{qr}(\psi)\}; \\
\mathrm{qr}(\exists x\varphi) &:= \mathrm{qr}(\varphi) + 1.
\end{aligned}$$

For example, the formula $\neg\exists x(\forall y Rxz \wedge Qy) \wedge \forall z Qz$ has quantifier rank 2. The formulas of quantifier rank zero are the quantifier-free formulas.

One direction of Fraïssé's Theorem follows from

3.1. *If* $\mathfrak{A} \cong_f \mathfrak{B}$ *then* $\mathfrak{A} \equiv \mathfrak{B}$.

In order to prove 3.1 we must show for every S-sentence φ that

$$\mathfrak{A} \models \varphi \quad \text{iff} \quad \mathfrak{B} \models \varphi.$$

We obtain this by applying the following lemma, taking $r = 0$, $n = \mathrm{qr}(\varphi)$, and an arbitrary $p \in I_n$ (note that $I_n \neq \emptyset$!).

3.2 Lemma. *Let* $(I_n)_{n\in\mathbf{N}}: \mathfrak{A} \cong_f \mathfrak{B}$. *Then for every formula* φ:

$$(*) \quad \begin{array}{l} \textit{If } \varphi \in L_r^S,\ \mathrm{qr}(\varphi) \leq n,\ p \in I_n \textit{ and } a_0, \ldots, a_{r-1} \in \mathrm{dom}(p), \textit{ then} \\ \mathfrak{A} \models \varphi[a_0, \ldots, a_{r-1}] \quad \textit{iff} \quad \mathfrak{B} \models \varphi[p(a_0), \ldots, p(a_{r-1})]. \end{array}$$

Informally, 3.2 says that partial isomorphisms from I_n preserve formulas of quantifier rank $\leq n$. It makes precise the idea discussed in 1.2(e) that formulas with quantifiers are preserved under partial isomorphisms, provided these isomorphisms admit certain extensions.

Proof of 3.2. We show (*) by induction on formulas φ. Suppose $\varphi \in L_r^S$, $\mathrm{qr}(\varphi) \leq n$, $p \in I_n$ and $a_0, \ldots, a_{r-1} \in \mathrm{dom}(p)$.

(i) For atomic φ the result was proved in 1.2(c).

(ii) If $\varphi = \neg\psi$ then $\mathfrak{A} \models \varphi[a_0, \ldots, a_{r-1}]$
iff not $\mathfrak{A} \models \psi[a_0, \ldots, a_{r-1}]$
iff not $\mathfrak{B} \models \psi[p(a_0), \ldots, p(a_{r-1})]$ (by induction hypothesis)
iff $\mathfrak{B} \models \varphi[p(a_0), \ldots, p(a_{r-1})]$.

(iii) For $\varphi = \psi_0 \vee \psi_1$ the argument is analogous.

(iv) Suppose $\varphi = \exists x\psi$. Since $\varphi \in L_r^S$, v_r does not occur free in φ. Thus $\models \exists x\psi \leftrightarrow \exists v_r \psi \frac{v_r}{x}$, and therefore, we may assume that $x = v_r$. Because $\mathrm{qr}(\varphi) = \mathrm{qr}(\exists x\psi) \leq n$, we have $\mathrm{qr}(\psi) \leq n-1$. The claim is now obtained from the following chain of equivalent statements:

(a) $\mathfrak{A} \models \varphi[a_0, \ldots, a_{r-1}]$.

(b) There is $a \in A$ such that $\mathfrak{A} \models \psi[a_0, \ldots, a_{r-1}, a]$.

(c) There is $a \in A$ and $q \in I_{n-1}$ such that $q \supset p$, $a \in \mathrm{dom}(q)$, and $\mathfrak{A} \models \psi[a_0, \ldots, a_{r-1}, a]$.

(d) There is $a \in A$ and $q \in I_{n-1}$ such that $q \supset p$, $a \in \mathrm{dom}(q)$, and $\mathfrak{B} \models \psi[p(a_0), \ldots, p(a_{r-1}), q(a)]$.

(e) There is $b \in B$ and $q \in I_{n-1}$ such that $q \supset p$, $b \in \mathrm{rg}(q)$, and $\mathfrak{B} \models \psi[p(a_0), \ldots, p(a_{r-1}), b]$.

(f) There is $b \in B$ such that $\mathfrak{B} \models \psi[p(a_0), \ldots, p(a_{r-1}), b]$.

(g) $\mathfrak{B} \models \varphi[p(a_0), \ldots, p(a_{r-1})]$.

To prove the equivalence of (b) and (c) and of (e) and (f), respectively, one uses the back- and the forth-property of the sequence $(I_n)_{n \in \mathbf{N}}$. The equivalence of (c) and (d) follows from the induction hypothesis. □

From the foregoing proof we can extract another result:

Two structures $\mathfrak{A}$ and $\mathfrak{B}$ are said to be *m-isomorphic* (written: $\mathfrak{A} \cong_m \mathfrak{B}$) if there is a sequence $I_0, \ldots, I_m$ of nonempty sets of partial isomorphisms from $\mathfrak{A}$ to $\mathfrak{B}$ with the back- and the forth-property, i.e.,

> for $n+1 \leq m$, $p \in I_{n+1}$ and $a \in A$ (resp. $b \in B$), there is $q \in I_n$ such that $q \supset p$ and $a \in \mathrm{dom}(q)$ (resp. $b \in \mathrm{rg}(q)$).

In this case, we write $(I_n)_{n \leq m} : \mathfrak{A} \cong_m \mathfrak{B}$.

In case $(I_n)_{n \leq m} : \mathfrak{A} \cong_m \mathfrak{B}$, the proof of 3.2 shows that each $p \in I_n$ (with $n \leq m$) preserves the validity of formulas of quantifier rank $\leq n$. If we write $\mathfrak{A} \equiv_m \mathfrak{B}$ in case $\mathfrak{A}$ and $\mathfrak{B}$ satisfy the same sentences of quantifier rank $\leq m$, we thus have

3.3 Corollary. *If* $\mathfrak{A} \cong_m \mathfrak{B}$ *then* $\mathfrak{A} \equiv_m \mathfrak{B}$. □

The following considerations lead to the converse of 3.1.

For an S-structure $\mathfrak{B}$, finitely many elements $b_0, \ldots, b_{r-1} \in B$, written: $\overset{r}{b} \in B$, and $n \in \mathbf{N}$ we introduce a formula $\varphi^n_{\mathfrak{B},\overset{r}{b}} \in L^S_r$: $\varphi^0_{\mathfrak{B},\overset{r}{b}}$ describes the "isomorphism type" of the substructure $[b_0, \ldots, b_{r-1}]^{\mathfrak{B}}$; for $n > 0$, $\varphi^n_{\mathfrak{B},\overset{r}{b}}$ indicates to which isomorphism types $\overset{r}{b}$ can be extended in $\mathfrak{B}$ by adding n times one element respectively. We shall have that $\mathfrak{B} \models \varphi^n_{\mathfrak{B},\overset{r}{b}}[\overset{r}{b}]$; and if $\mathfrak{A} \models \varphi^n_{\mathfrak{B},\overset{r}{b}}[\overset{r}{a}]$ for an S-structure $\mathfrak{A}$ and $\overset{r}{a} \in A$, then the map given by $a_i \mapsto b_i$ $(i < r)$ will be a partial isomorphism which can be extended "back and forth n times". For $n > 0$ we also allow the case $r = 0$, i.e., the case of the empty sequence $\emptyset$ of elements from $\mathfrak{B}$, and we write $\varphi^n_{\mathfrak{B}}$ for $\varphi^n_{\mathfrak{B},\emptyset}$.

We now give an exact definition. As an abbreviation we set

$$\Phi_r := \{\varphi \in L^S_r \mid \varphi \text{ is atomic or negated atomic}\}.$$

Since S contains only relation symbols, Φ_r is finite and Φ_0 is empty.

For an S-structure $\mathfrak{B}$ we define the formula $\varphi^n_{\mathfrak{B},\overset{r}{b}}$ by induction on n for all r ($r > 0$, if $n = 0$) and all $\overset{r}{b} \in B$ as follows (afterwards we shall show that the conjunctions and disjunctions occurring in the definition are finite):

$$\varphi^0_{\mathfrak{B},\overset{r}{b}} := \bigwedge\{\varphi \in \Phi_r \mid \mathfrak{B} \models \varphi[\overset{r}{b}]\},$$

$$\varphi^{n+1}_{\mathfrak{B},\overset{r}{b}} := \forall v_r \bigvee\{\varphi^n_{\mathfrak{B},\overset{r}{b}b} \mid b \in B\} \wedge \bigwedge\{\exists v_r \varphi^n_{\mathfrak{B},\overset{r}{b}b} \mid b \in B\}.$$

Here, $\overset{r}{b}b$ abbreviates $(b_0, \ldots, b_{r-1}, b)$.

Since Φ_r is finite for all r, by induction on n we obtain easily:

3.4. *The set* $\{\varphi^n_{\mathfrak{B},\overset{r}{b}} \mid \mathfrak{B}$ *is an* S*-structure and* $\overset{r}{b} \in B\}$ *is finite.* □

The conjunctions and disjunctions occurring in the definition are therefore all finite, and hence the $\varphi^n_{\mathfrak{B},\overset{r}{b}}$ are first-order formulas.

3.5. (a) $\varphi^n_{\mathfrak{B},\overset{r}{b}} \in L^S_r$ *and* $\mathrm{qr}(\varphi^n_{\mathfrak{B},\overset{r}{b}}) = n$.
(b) $\mathfrak{B} \models \varphi^n_{\mathfrak{B},\overset{r}{b}}[\overset{r}{b}]$.

Proof for (a) and (b) by induction on n. We consider (b). For $n = 0$ (and $r > 0$) the claim follows immediately from the definition of $\varphi^0_{\mathfrak{B},\overset{r}{b}}$. For the step from n to $n + 1$, the induction hypothesis yields for all $b' \in B$,

$$\mathfrak{B} \models \varphi^n_{\mathfrak{B},\overset{r}{b}b'}[\overset{r}{b}, b'],$$

hence, for all $b' \in B$,

$$\mathfrak{B} \models \bigvee\{\varphi^n_{\mathfrak{B},\overset{r}{b}b} \mid b \in B\}[\overset{r}{b}, b'] \quad \text{and} \quad \mathfrak{B} \models \exists v_r \varphi^n_{\mathfrak{B},\overset{r}{b}b'}[\overset{r}{b}].$$

Thus

$$\mathfrak{B} \models \forall v_r \bigvee\{\varphi^n_{\mathfrak{B},\overset{r}{b}b} \mid b \in B\}[\overset{r}{b}] \quad \text{and} \quad \mathfrak{B} \models \bigwedge\{\exists v_r \varphi^n_{\mathfrak{B},\overset{r}{b}b'} \mid b' \in B\}[\overset{r}{b}]$$

and therefore, $\mathfrak{B} \models \varphi^{n+1}_{\mathfrak{B},\overset{r}{b}}[\overset{r}{b}]$. □

Let $\overset{r}{b} \in B$. If $\mathfrak{A}$ is also an S-structure and $\overset{r}{a} \in A$ then 1.2(c) shows:

3.6. $\mathfrak{A} \models \varphi^0_{\mathfrak{B},\overset{r}{b}}[\overset{r}{a}]$ *iff* $p(a_i) = b_i$ *for* $i < r$ *defines a partial isomorphism from* $\mathfrak{A}$ *to* $\mathfrak{B}$ (*written:* $\overset{r}{a} \mapsto \overset{r}{b} \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$).

We generalize the direction from left to right:

3.7. *If* $\mathfrak{A} \models \varphi^n_{\mathfrak{B},\overset{r}{b}}[\overset{r}{a}]$ *then* $\overset{r}{a} \mapsto \overset{r}{b} \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$.

Proof by induction on n. For $n = 0$ the claim follows from 3.6. Let $\mathfrak{A} \models \varphi^{n+1}_{\mathfrak{B},\overset{r}{b}}[\overset{r}{a}]$. We choose $a \in A$ arbitrarily. Since $\mathfrak{A} \models \forall v_r \bigvee\{\varphi^n_{\mathfrak{B},\overset{r}{b}b} \mid b \in B\}[\overset{r}{a}]$, there is some $b \in B$ such that $\mathfrak{A} \models \varphi^n_{\mathfrak{B},\overset{r}{b}b}[\overset{r}{a}, a]$. By induction hypothesis, $\overset{r}{a}a \mapsto \overset{r}{b}b \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$, hence $\overset{r}{a} \mapsto \overset{r}{b} \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$. □

We fix two S-structures $\mathfrak{A}$ and $\mathfrak{B}$. For $n \in \mathbf{N}$ we set

$$J_n := \{\overset{r}{a} \mapsto \overset{r}{b} \mid r \in \mathbf{N}, \overset{r}{a} \in A, \overset{r}{b} \in B \text{ and } \mathfrak{A} \models \varphi^n_{\mathfrak{B},\overset{r}{b}}[\overset{r}{a}]\}.$$

Then we obtain:

3.8. (a) $J_n \subset \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$;
(b) $(J_n)_{n \in \mathbf{N}}$ *has the back- and the forth-property;*
(c) *If* $n > 0$ *and* $\mathfrak{A} \models \varphi^n_{\mathfrak{B}}$ $(= \varphi^n_{\mathfrak{B},\emptyset})$ *then* $\emptyset \in J_n$, *hence* $J_n \neq \emptyset$.

Proof. Since (a) follows immediately from 3.7 and (c) from the definition of J_n, we only have to prove (b). First we show the forth-property. Let $p = \overset{r}{a} \mapsto \overset{r}{b} \in J_{n+1}$ and $a \in A$. Then $\mathfrak{A} \models \varphi^{n+1}_{\mathfrak{B},\overset{r}{b}}[\overset{r}{a}]$; in particular, $\mathfrak{A} \models \forall v_r \bigvee\{\varphi^n_{\mathfrak{B},\overset{r}{b}b} \mid b \in B\}[\overset{r}{a}]$. Therefore there is a $b \in B$ such that $\mathfrak{A} \models \varphi^n_{\mathfrak{B},\overset{r}{b}b}[\overset{r}{a}, a]$. Then $\overset{r}{a}a \mapsto \overset{r}{b}b$ is a partial isomorphism in J_n which extends p and whose domain contains a. – Since also $\mathfrak{A} \models \bigwedge\{\exists v_r \varphi^n_{\mathfrak{B},\overset{r}{b}b} \mid b \in B\}[\overset{r}{a}]$, for each $b \in B$ there is $a \in A$ such that $\mathfrak{A} \models \varphi^n_{\mathfrak{B},\overset{r}{b}b}[\overset{r}{a}, a]$ and hence $\overset{r}{a}a \mapsto \overset{r}{b}b \in J_n$, i.e., there is a partial isomorphism in J_n which extends p and has b in its range. This proves the back-property. □

With 3.8 we easily obtain the direction of Fraïssé's Theorem, which was still open: If $\mathfrak{A} \equiv \mathfrak{B}$ then $\mathfrak{A} \cong_f \mathfrak{B}$. So let $\mathfrak{A} \equiv \mathfrak{B}$. Since, for $n \geq 1$, $\mathfrak{B} \models \varphi^n_{\mathfrak{B}}$ (cf. 3.5(b)), we have $\mathfrak{A} \models \varphi^n_{\mathfrak{B}}$. By 3.8(c) we get $J_n \neq \emptyset$ and therefore $(J_n)_{n \in \mathbf{N}} : \mathfrak{A} \cong_f \mathfrak{B}$ (cf. 3.8(a), (b)). □

From the preceding considerations we instantly obtain:

3.9 Theorem. *Let S be a finite relational symbol set, and let $\mathfrak{A}$ and $\mathfrak{B}$ be S-structures. Then the following are equivalent:*

(a) $\mathfrak{A} \equiv \mathfrak{B}$.

(b) $\mathfrak{A} \models \varphi^n_{\mathfrak{B}}$ *for* $n \geq 1$.

(c) $(J_n)_{n\in\mathbf{N}}\colon \mathfrak{A} \cong_f \mathfrak{B}$.

(d) $\mathfrak{A} \cong_f \mathfrak{B}$. □

Since $\mathrm{qr}(\varphi^m_{\mathfrak{B}}) = m$ for $m \geq 1$, we further get:

3.10 Theorem. *Let S be a finite and relational symbol set, and let $\mathfrak{A}$ and $\mathfrak{B}$ be S-structures. Then the following are equivalent for $m \geq 1$:*

(a) $\mathfrak{A} \equiv_m \mathfrak{B}$.

(b) $\mathfrak{A} \models \varphi^m_{\mathfrak{B}}$.

(c) $(J_n)_{n\leq m}\colon \mathfrak{A} \cong_m \mathfrak{B}$.

(d) $\mathfrak{A} \cong_m \mathfrak{B}$.

For the direction from (b) to (c) notice that, because of the back- and the forth-property of the sequence $(J_n)_{n\in\mathbf{N}}$, if $J_n \neq \emptyset$, then also $J_m \neq \emptyset$ for $m < n$. □

In VI.3 we have shown that some classes are not Δ-elementary. The arguments involved the Compactness Theorem and used infinite structures. The preceding considerations provide a method for showing that certain properties cannot be expressed by a first-order sentence, even if we restrict ourselves to *finite* structures. We explain the approach treating, as an example, the connectedness of finite graphs (in VI.3.6 we considered the connectedness for the class of *all* graphs). A further example is contained in Exercise 3.16(b).

3.11 Theorem. *Let R be a binary relation symbol. There is no $\{R\}$-sentence whose finite models are the finite connected graphs. (Hence, in particular, the class of connected graphs is not elementary).*

Proof. For $k \geq 0$ let $\mathfrak{G}_k$ be the graph corresponding to the regular $(k+1)$-gon with the vertices $0, \ldots, k$, i.e.,

$$\mathfrak{G}_k = \big(\{0, \ldots, k\}, R^{G_k}\big)$$

where

$$R^{G_k} = \{(i, i+1) \mid i < k\} \cup \{(i, i-1) \mid 1 \leq i \leq k\} \cup \{(0, k), (k, 0)\},$$

and let $\mathfrak{H}_k$ consist of two disjoint copies of $\mathfrak{G}_k$, say,

$$\mathfrak{H}_k = \big(\{0, \ldots, k\} \times \{0, 1\}, R^{H_k}\big)$$

with

$$R^{H_k} = \{((i, 0), (j, 0)) \mid (i, j) \in R^{G_k}\} \cup \{((i, 1), (j, 1)) \mid (i, j) \in R^{G_k}\}.$$

We claim:

$$(*) \qquad \text{For } k \geq 2^m\colon\ \mathfrak{G}_k \cong_m \mathfrak{H}_k.$$

Then we are done. In fact, let φ be an $\{R\}$-sentence and $m = \mathrm{qr}(\varphi)$. Then $(*)$ says that $\mathfrak{G}_{2^m} \cong_m \mathfrak{H}_{2^m}$, i.e. $\mathfrak{G}_{2^m} \equiv_m \mathfrak{H}_{2^m}$ and therefore ($\mathfrak{G}_{2^m} \models \varphi$ iff $\mathfrak{H}_{2^m} \models \varphi$). Since $\mathfrak{G}_{2^m}$ is connected, but $\mathfrak{H}_{2^m}$ is not, the class of finite models of φ cannot be identical with the class of all finite connected graphs.

For the proof of $(*)$ we define, for fixed $k \geq 2^m$ and $n \geq 0$, similarly to 1.8, "distance functions" d_n on $G_k \times G_k$ and $d_n{}'$ on $H_k \times H_k$) as follows:

$$d_n(a,b) := \begin{cases} \text{length of the shortest path connecting } a \text{ and } b \text{ in } \mathfrak{G}_k, \\ \qquad\qquad\qquad\qquad \text{if this length is} \leq 2^m; \\ \infty, \qquad\qquad\qquad\qquad \text{otherwise;} \end{cases}$$

$$d_n{}'((a,i),(b,j)) := \begin{cases} d_n(a,b) & \text{if } i = j; \\ \infty & \text{otherwise.} \end{cases}$$

We set

$$I_n := \{p \in \mathrm{Part}(\mathfrak{G}_k, \mathfrak{H}_k) \mid \mathrm{dom}(p) \text{ is finite, and for all } a,b \in G_k, \\ d_n(a,b) = d_n{}'(p(a),p(b))\}.$$

Similarly to 1.8, one can easily show now that $(I_n)_{n \leq m} \colon \mathfrak{G}_k \cong_m \mathfrak{H}_k$. □

3.12 Exercise. Let S be finite and relational, and let the S-structure $\mathfrak{B}$ contain exactly n elements. Then for every S-structure $\mathfrak{A}$:

$$\mathfrak{A} \cong \mathfrak{B} \quad \text{iff} \quad \mathfrak{A} \models \varphi^{n+1}_{\mathfrak{B}},$$

i.e. $\varphi^{n+1}_{\mathfrak{B}}$ characterizes $\mathfrak{B}$ up to isomorphism.

3.13 Exercise. Let S be finite and relational and let $\mathfrak{B}$ be an S-structure. Show that for all n and r (with $n+r > 0$) and for all $\overset{r}{b} \in B$, $\models \varphi^{n+1}_{\mathfrak{B},\overset{r}{b}} \to \varphi^{n}_{\mathfrak{B},\overset{r}{b}}$.

3.14 Exercise. Again, let S be finite and relational. For an S-structure $\mathfrak{B}$ and $\overset{r}{b} \in B$ let $\psi^{n+1}_{\mathfrak{B},\overset{r}{b}}$ (for $n + r > 0$) be defined by $\psi^{0}_{\mathfrak{B},\overset{r}{b}} := \varphi^{0}_{\mathfrak{B},\overset{r}{b}}$ and $\psi^{n+1}_{\mathfrak{B},\overset{r}{b}} := \forall v_r \bigvee\{\psi^{n}_{\mathfrak{B},\overset{r}{b}b} \mid b \in B\}$. Show:

(a) $\psi^{n}_{\mathfrak{B},\overset{r}{b}}$ is a universal formula.

(b) For an S-structure $\mathfrak{A}$ the following are equivalent:

(1) $\mathfrak{A}$ satisfies every universal S-sentence which holds in $\mathfrak{B}$.

(2) $\mathfrak{A} \models \psi^{n}_{\mathfrak{B}}$ for all $n \geq 1$.

(3) $\mathfrak{A} \to_f \mathfrak{B}$ (for notation see 1.12).

(c) Part (b) corresponds to Theorem 3.9. Formulate and prove the version analogous to 3.10.

3.15 Exercise. Transfer the results of this section to arbitrary finite symbol sets. For this purpose define a *modified rank* mrk for terms and formulas as follows:

$$\begin{aligned}
\mathrm{mrk}(x) &= 0, \quad \mathrm{mrk}(c) = 1 \\
\mathrm{mrk}(ft_1 \ldots t_n) &= 1 + \mathrm{mrk}(t_1) + \ldots + \mathrm{mrk}(t_n) \\
\mathrm{mrk}(Rt_1 \ldots t_n) &= \mathrm{mrk}(t_1) + \ldots + \mathrm{mrk}(t_n) \\
\mathrm{mrk}(t_1 \equiv t_2) &= \max\{0, \mathrm{mrk}(t_1) + \mathrm{mrk}(t_2) - 1\} \\
\mathrm{mrk}(\neg\varphi) &= \mathrm{mrk}(\varphi) \\
\mathrm{mrk}(\varphi \vee \psi) &= \max\{\mathrm{mrk}(\varphi), \mathrm{mrk}(\psi)\} \\
\mathrm{mrk}(\exists x\varphi) &= 1 + \mathrm{mrk}(\varphi).
\end{aligned}$$

Furthermore, for $r \geq 0$ set

$$\Phi_r{}' := \{\varphi \in L_r^S \mid \varphi \text{ is atomic or negated atomic and } \mathrm{mrk}(\varphi) = 0\}.$$

Show:
(a) Theorems 3.9 and 3.10 and the considerations leading to them remain valid if we replace the quantifier rank everywhere by the modified rank and Φ_r by $\Phi_r{}'$. The same holds for the preceding exercises.
(b) If S is relational, then $\Phi_r = \Phi_r{}'$ and $\mathrm{qr}(\varphi) = \mathrm{mrk}(\varphi)$ for all $\varphi \in L^S$.

3.16 Exercise. Let $S := \{<, R\}$ with binary relation symbols $<$ and R. For $k \in \mathbf{N}$ let $\mathfrak{A}_k := (\{0, \ldots, k\}, <^{A_k}, R^{A_k})$, where $<^{A_k}$ is the natural order on $\{0, \ldots, k\}$ and R^{A_k} is the successor relation, i.e. $R^{A_k} = \{(i, i{+}1) \mid i < k\}$.
Show:
(a) For $k, l, m \in \mathbf{N}$ with $k, l \geq 2^m$: $\mathfrak{A}_k \cong_m \mathfrak{A}_l$.
Hint: As in 1.8 define "distance functions" d_n on $\mathbf{N} \times \mathbf{N}$ by

$$d_n(a, b) := \begin{cases} b - a & \text{if } |b - a| \leq 2^n \\ \infty & \text{otherwise.} \end{cases}$$

For $k, l \geq 2^m$ set

$$\begin{aligned} I_n := \{p \in \mathrm{Part}(\mathfrak{A}_k, \mathfrak{A}_l) \mid \mathrm{dom}(p) &\text{ is finite, } p(0) = 0,\ p(k) = l, \text{ and} \\ &\text{for all } a, b \in \mathrm{dom}(p),\quad d_n(a, b) = d_n(p(a), p(b))\} \end{aligned}$$

and show (as in 1.8) that $(I_n)_{n \leq m}\colon \mathfrak{A}_k \cong_m \mathfrak{A}_l$.
(b) There is no $\varphi \in L_0^S$ such that $(\mathfrak{A}_k \models \varphi$ iff k is even).

3.17 Exercise. Let $S := \{P_1, \ldots, P_r\}$ with unary P_i. Show: For each S-structure $\mathfrak{A}$ and each $m \geq 1$ there is a structure $\mathfrak{B}$ with $\mathfrak{A} \cong_m \mathfrak{B}$ containing at most $m \cdot 2^r$ elements. (Hint: Consider the 2^r many subsets of A of the form $A_1 \cap \ldots \cap A_r$, where $A_i = P_i^A$ or $A_i = A \setminus P_i^A$. For $\mathfrak{B}$ take a structure in which the corresponding sets have the same number of elements, if this number is $< m$, and m elements otherwise.)

3.18 Exercise. Again, let $S = \{P_1, \ldots, P_r\}$ with unary P_i, $m \geq 1$ and let $\varphi \in L_0^S$ be a sentence of quantifier rank $\leq m$. Show:
(a) If φ is satisfiable, then φ is satisfiable already over a domain with at most $m \cdot 2^r$ elements.
(b) $\{\psi \mid \psi \in L_0^S,\ \psi \text{ valid}\}$ is R-decidable.

§4. Ehrenfeucht Games

The algebraic description of elementary equivalence is well-suited for many purposes. However, it lacks the intuitive appeal of a game-theoretical characterization due to Ehrenfeucht, which we describe in the present section.

Let S be an arbitrary symbol set and let $\mathfrak{A}$ and $\mathfrak{B}$ be S-structures. To simplify the formulation we assume $A \cap B = \emptyset$. The *Ehrenfeucht game* $\mathrm{G}(\mathfrak{A}, \mathfrak{B})$ corresponding to $\mathfrak{A}$ and $\mathfrak{B}$ is played by two players, I and II, according to the following rules:

Each *play* of the game begins with player I choosing a natural number $r \geq 1$; r is the number of subsequent moves each player has to make in the course of the play. These subsequent moves are begun by player I, and both players move alternately. Each move consists of choosing an element from $A \cup B$. If player I chooses an element $a_i \in A$ in his ith move, then player II must choose an element $b_i \in B$ in his ith move. If player I chooses an element $b_i \in B$ in his ith move, then player II must choose an element $a_i \in A$. After the rth move of player II the play is completed. Altogether some number $r \geq 1$, elements $a_1, \ldots, a_r \in A$ and $b_1, \ldots, b_r \in B$ have been chosen. Player II has won the play iff by $p(a_i) := b_i$ for $i = 1, \ldots, r$ a partial isomorphism from $\mathfrak{A}$ to $\mathfrak{B}$ is defined.

We say that player II has a winning strategy in $\mathrm{G}(\mathfrak{A}, \mathfrak{B})$ and write "II wins $\mathrm{G}(\mathfrak{A}, \mathfrak{B})$", if it is possible for him to win each play. (We omit an exact definition of the notion of "winning stategy.")

4.1 Lemma. $\mathfrak{A} \cong_f \mathfrak{B}$ *iff* II *wins* $\mathrm{G}(\mathfrak{A}, \mathfrak{B})$.

This lemma, together with Fraïssé's Theorem, yields the desired game-theoretical characterization of elementary equivalence:

4.2 Ehrenfeucht's Theorem. *Let S be a finite symbol set. Then for any S-structures $\mathfrak{A}$ and $\mathfrak{B}$:*

$$\mathfrak{A} \equiv \mathfrak{B} \quad \textit{iff} \quad \text{II } \textit{wins } \mathrm{G}(\mathfrak{A}, \mathfrak{B}).$$

Proof of 4.1. Suppose $(I_n)_{n \in \mathbf{N}} : \mathfrak{A} \cong_f \mathfrak{B}$. Then also $(I'_n)_{n \in \mathbf{N}} : \mathfrak{A} \cong_f \mathfrak{B}$, where $I'_n := \{p \mid \text{there is } q \in I_n \text{ such that } p \subset q\}$. We describe a winning strategy for player II:

If player I chooses the number r at the beginning of a $\mathrm{G}(\mathfrak{A}, \mathfrak{B})$-play, then for $i = 1, \ldots, r$ player II should choose the elements a_i (or respectively b_i) so that by $p_i(a_j) := b_j$ for $1 \leq j \leq i$ one obtains a partial isomorphism $p_i : \{a_1, \ldots, a_i\} \to \{b_1, \ldots, b_i\}$ with $p_i \in I'_{r-i}$. This is always possible because of the extension properties of partial isomorphisms in $(I'_n)_{n \in \mathbf{N}}$. For $i = r$ it follows that player II has a winning strategy for the game.

Conversely, suppose that player II has a winning strategy in $G(\mathfrak{A}, \mathfrak{B})$. We define a sequence $(I_n)_{n\in\mathbf{N}}$ as follows:

For $n \in \mathbf{N}$ let

$p \in I_n$:iff $p \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$ and there are $j \in \mathbf{N}$ and $a_1, \ldots, a_j \in A$ such that
(i) $\mathrm{dom}(p) = \{a_1, \ldots, a_j\}$;
(ii) there is an $m \geq n$ and a $G(\mathfrak{A}, \mathfrak{B})$-play which II plays according to his winning strategy, which player I opens by choosing the number $m + j$, and where in the first j moves the elements $a_1, \ldots, a_j \in A$ and $p(a_1), \ldots, p(a_j) \in B$ are chosen.

From the rules of the game we immediately obtain that $(I_n)_{n\in\mathbf{N}}: \mathfrak{A} \cong_f \mathfrak{B}$. □

4.3 Exercise. For $r \geq 1$ let $G_r(\mathfrak{A}, \mathfrak{B})$ be the game obtained from the Ehrenfeucht game $G(\mathfrak{A}, \mathfrak{B})$ by fixing r to be the number player I has to choose first. Show: $\mathfrak{A} \cong_r \mathfrak{B}$ iff II wins $G_r(\mathfrak{A}, \mathfrak{B})$.

XIII

Lindström's Theorems

In this final chapter we present some results, due to Lindström [25], which we have already mentioned several times. They show that first-order logic occupies a unique place among logical systems. Indeed, we shall prove:

(a) There is no logical system with more expressive power than first-order logic, for which both the Compactness Theorem and the Löwenheim-Skolem Theorem hold (Section 3).

(b) There is no logical system with more expressive power than first-order logic, for which the Löwenheim-Skolem Theorem holds and for which the set of valid sentences is enumerable (Section 4).

§1. Logical Systems

In the following definition of a "logical system" we collect several properties which are shared by the logics we have considered so far. As we are mainly interested in semantic aspects, we shall speak of a logical system as soon as we have the following: We are given, for every symbol set S, an "abstract" set whose elements play the role of S-sentences, and in addition, a relationship between structures and such sentences which corresponds to the satisfaction relation, and determines whether an "abstract" sentence holds in a structure.

1.1 Definition. A *logical system* $\mathcal{L}$ consists of a function L and a binary relation $\models_{\mathcal{L}}$. L associates with every symbol set S a set $L(S)$, the *set of S-sentences* of $\mathcal{L}$. The following properties are required:

(a) If $S_0 \subset S_1$ then $L(S_0) \subset L(S_1)$.
(b) If $\mathfrak{A} \models_{\mathcal{L}} \varphi$ (i.e. if $\mathfrak{A}$ and φ are related under $\models_{\mathcal{L}}$), then, for some S, $\mathfrak{A}$ is an S-structure and $\varphi \in L(S)$.
(c) (*Isomorphism property*) If $\mathfrak{A} \models_{\mathcal{L}} \varphi$ and $\mathfrak{A} \cong \mathfrak{B}$ then $\mathfrak{B} \models_{\mathcal{L}} \varphi$.
(d) (*Reduct property*) If $S_0 \subset S_1$, $\varphi \in L(S_0)$ and $\mathfrak{A}$ is an S_1-structure then

$$\mathfrak{A} \models_{\mathcal{L}} \varphi \quad \text{iff} \quad \mathfrak{A}|_{S_0} \models_{\mathcal{L}} \varphi.$$

$\mathcal{L}_{\mathrm{I}}, \mathcal{L}_{\mathrm{II}}, \mathcal{L}_{\mathrm{II}}^{w}, \mathcal{L}_{\omega_1\omega}, \mathcal{L}_Q$ are logical systems. For instance, in the case of $\mathcal{L}_{\mathrm{I}}$ we choose L_{I} to be the function which assigns to a symbol set S the set $L_{\mathrm{I}}(S) := L_0^S$ of first-order S-sentences, and we take $\models_{\mathcal{L}_{\mathrm{I}}}$ to be the usual satisfaction relation between structures and first-order sentences.

If $\mathcal{L}$ is a logical system and $\varphi \in L(S)$, let

$$\mathrm{Mod}_{\mathcal{L}}^S(\varphi) := \{\mathfrak{A} \mid \mathfrak{A} \text{ is an } S\text{-structure and } \mathfrak{A} \models \varphi\}.$$

In case S is clear from the context we just write $\mathrm{Mod}_{\mathcal{L}}(\varphi)$.

$\mathrm{Mod}_{\mathcal{L}}^S(\varphi)$ can be regarded as a mathematically precise counterpart to the *meaning* of φ. It suggests the following definition of when a logical system $\mathcal{L}'$ has at least the expressive power as $\mathcal{L}$, namely, if for every $\mathcal{L}$-sentence φ there is an $\mathcal{L}'$-sentence ψ with the same meaning:

1.2 Definition. Let $\mathcal{L}$ and $\mathcal{L}'$ be logical systems.

(a) Let S be a symbol set, $\varphi \in L(S)$ and $\psi \in L'(S)$. φ and ψ are said to be *logically equivalent* :iff $\mathrm{Mod}_{\mathcal{L}}^S(\varphi) = \mathrm{Mod}_{\mathcal{L}'}^S(\psi)$.

(b) *$\mathcal{L}'$ is at least as strong as $\mathcal{L}$* (written: $\mathcal{L} \leq \mathcal{L}'$) :iff for every S and every $\varphi \in L(S)$ there is a $\psi \in L'(S)$ such that φ and ψ are logically equivalent.

(c) *$\mathcal{L}$ and $\mathcal{L}'$ are equally strong* (written: $\mathcal{L} \sim \mathcal{L}'$) :iff $\mathcal{L} \leq \mathcal{L}'$ and $\mathcal{L}' \leq \mathcal{L}$.

Examples. $\mathcal{L}_{\mathrm{I}} \leq \mathcal{L}_{\mathrm{II}}^{w}$; $\mathcal{L}_{\mathrm{II}}^{w} \leq \mathcal{L}_{\mathrm{II}}$; not $\mathcal{L}_{\mathrm{II}} \leq \mathcal{L}_{\mathrm{II}}^{w}$ (cf. IX.1.7); $\mathcal{L}_{\mathrm{II}}^{w} \leq \mathcal{L}_{\omega_1\omega}$ (cf. IX.2.7).

On our abstract level we now formulate some properties of logical systems $\mathcal{L}$, which are known to hold for the systems we have considered so far.

We fix the following conventions:

Boole($\mathcal{L}$) ("$\mathcal{L}$ contains propositional ("Boolean") connectives"):

(1) Given S and $\varphi \in L(S)$, there is a $\chi \in L(S)$ such that for every S-structure $\mathfrak{A}$:

$$\mathfrak{A} \models_{\mathcal{L}} \chi \quad \text{iff} \quad \text{not } \mathfrak{A} \models_{\mathcal{L}} \varphi.$$

(2) Given S and $\varphi, \psi \in L(S)$, there is a $\chi \in L(S)$ such that for every S-structure $\mathfrak{A}$:

$$\mathfrak{A} \models_{\mathcal{L}} \chi \quad \text{iff} \quad (\mathfrak{A} \models_{\mathcal{L}} \varphi \text{ or } \mathfrak{A} \models_{\mathcal{L}} \psi).$$

If Boole($\mathcal{L}$) holds then $\neg\varphi$ and $(\varphi \vee \psi)$ stand for formulas χ in the sense of (1) and (2), respectively. $(\varphi \wedge \psi), (\varphi \to \psi), \ldots$ are used analogously.

Rel($\mathcal{L}$) ("$\mathcal{L}$ permits relativization"):

For S, $\varphi \in L(S)$ and unary U there is a $\psi \in L(S \cup \{U\})$ such that

$$(\mathfrak{A}, U^A) \models_{\mathcal{L}} \psi \quad \text{iff} \quad [U^A]^{\mathfrak{A}} \models_{\mathcal{L}} \varphi$$

for all S-structures $\mathfrak{A}$ and all S-closed subsets U^A of A. ($[U^A]^{\mathfrak{A}}$ is the substructure of $\mathfrak{A}$ with domain U^A.)

If Rel($\mathcal{L}$) holds let φ^U be a formula ψ with the above property.

Repl($\mathcal{L}$) ("$\mathcal{L}$ permits replacement of function symbols and constants by relation symbols"):

If S is a symbol set and S^r is chosen as for VIII.1.3 – the function symbols and constants from S are replaced by relation symbols for their graphs –, then for any $\varphi \in L(S)$ there is a $\psi \in L(S^r)$ such that for all S-structures $\mathfrak{A}$:

$$\mathfrak{A} \models_{\mathcal{L}} \varphi \quad \text{iff} \quad \mathfrak{A}^r \models_{\mathcal{L}} \psi.$$

(For the definition of $\mathfrak{A}^r$ see also VIII.1). If Repl($\mathcal{L}$), we write φ^r for a formula ψ with the above property.

1.3 Definition. A logical system $\mathcal{L}$ is said to be *regular* if it satisfies the properties Boole($\mathcal{L}$), Rel($\mathcal{L}$) and Repl($\mathcal{L}$).

All logical systems which we have hitherto considered are regular. In the case of $\mathcal{L}_{\mathrm{I}}$ we verified Rel($\mathcal{L}_{\mathrm{I}}$) and Repl($\mathcal{L}_{\mathrm{I}}$) in VIII.1,2. The arguments given there can also be applied without difficulty to the other logical systems.

We tacitly adopt some semantic notions whose definitions can be extended from $\mathcal{L}_{\mathrm{I}}$ to other logical systems $\mathcal{L}$ in a straightforward manner. For example, $\varphi \in L(S)$ is said to be *satisfiable* if $\mathrm{Mod}^S_{\mathcal{L}}(\varphi) \neq \emptyset$, and *valid* if $\mathrm{Mod}^S_{\mathcal{L}}(\varphi)$ is the class of all S-structures. If $\Phi \subset L(S)$ then $\Phi \models_{\mathcal{L}} \varphi$ means that every model of Φ (in the sense of $\models_{\mathcal{L}}$) is a model of φ. Note that these definitions refer to a fixed symbol set S. However, using the reduct property 1.1(d) one can argue that they do not depend on S. In the sequel similar applications of the reduct property will be made without explicit mention.

We introduce the following abbreviations:

LöSko($\mathcal{L}$) ("The Löwenheim-Skolem Theorem holds for $\mathcal{L}$"):

If $\varphi \in L(S)$ is satisfiable, then there is a model of φ whose domain is at most countable.

and

Comp($\mathcal{L}$) ("The Compactness Theorem holds for $\mathcal{L}$"):

If $\Phi \subset L(S)$ and if every finite subset of Φ is satisfiable, then Φ itself is satisfiable.

In this terminology the result of Lindström mentioned in the introduction to the present chapter under (a) reads as follows:

If $\mathcal{L}$ is a regular logical system such that $\mathcal{L}_{\mathrm{I}} \leq \mathcal{L}$, LöSko($\mathcal{L}$), and Comp($\mathcal{L}$), then $\mathcal{L} \sim \mathcal{L}_{\mathrm{I}}$.

We shall use the following result to restrict ourselves to a relational S in the proofs of Lindström's Theorems.

1.4 Lemma. *Let $\mathcal{L}$ be a regular logical system. If, for all relational symbol sets S, every $L(S)$-sentence is logically equivalent to a first-order sentence, then $\mathcal{L} \leq \mathcal{L}_{\mathrm{I}}$.*

Proof. We prove the claim using Repl($\mathcal{L}$) and the results from VIII.1: Let S be an arbitrary symbol set and $\psi \in L(S)$. With Repl($\mathcal{L}$) we choose the $L(S^r)$-sentence ψ^r (for the definition of S^r see Repl($\mathcal{L}$)). Since S^r is relational, by assumption there is a first-order sentence $\varphi \in L_{\mathrm{I}}(S^r)$ which is logically equivalent to ψ^r. For φ we choose the $L_{\mathrm{I}}(S)$-sentence φ^{-r} according to VIII.1.3. Then the following holds for every S-structure $\mathfrak{A}$:

$$\begin{aligned} \mathfrak{A} \models_{\mathcal{L}} \psi \quad & \text{iff} \quad \mathfrak{A}^r \models_{\mathcal{L}} \psi^r \\ & \text{iff} \quad \mathfrak{A}^r \models \varphi \\ & \text{iff} \quad \mathfrak{A} \models \varphi^{-r}. \end{aligned}$$

Hence ψ and φ^{-r} are logically equivalent. □

1.5 Exercise. Let $\mathcal{L}$ be given by
(i) $L(S) := \{\varphi \mid \varphi$ is an L^S_{II}-sentence of the form $\exists X_1 \ldots \exists X_n \psi$, where ψ does not contain a second-order quantifier$\}$.
(ii) For $\varphi \in L(S)$ and S-structures $\mathfrak{A}$, $\mathfrak{A} \models_{\mathcal{L}} \varphi$ iff $\mathfrak{A} \models_{\mathcal{L}_{\mathrm{II}}} \varphi$.
Show
(a) $\mathcal{L}$ is a logical system.
(b) LöSko($\mathcal{L}$), Comp($\mathcal{L}$), Rel($\mathcal{L}$) and Repl($\mathcal{L}$) hold.
(c) Boole($\mathcal{L}$) does not hold.
(d) $\mathcal{L}_{\mathrm{I}} \leq \mathcal{L}$, but not $\mathcal{L} \leq \mathcal{L}_{\mathrm{I}}$.
(e) The set of valid $L(S_{\mathrm{ar}})$-sentences is not enumerable. (Hint: Note that Th($\mathfrak{N}$) is not enumerable and use the axiom system Π given in III.7.5.)
The logical system $\mathcal{L}$ shows that Boole($\mathcal{L}$) is necessary in Lindström's First Theorem 3.5.

1.6 Exercise. Show: $\mathcal{L}_Q \leq \mathcal{L}_{\mathrm{II}}$, not $\mathcal{L}^w_{\mathrm{II}} \leq \mathcal{L}_Q$, not $\mathcal{L}_Q \leq \mathcal{L}^w_{\mathrm{II}}$.

§2. Compact Regular Logical Systems

Before proving Lindström's Theorems we derive some properties of logical systems for which the Compactness Theorem holds.

In the following, $\mathcal{L}$ is a *regular logical system such that* $\mathcal{L}_{\mathrm{I}} \leq \mathcal{L}$. For a first-order S-sentence φ, let φ^* be a sentence in $L(S)$ logically equivalent to φ. For a set Φ of first-order S-sentences define $\Phi^* := \{\varphi^* \mid \varphi \in \Phi\}$.

As usual, Comp($\mathcal{L}$), the Compactness Theorem for satisfaction, yields the

Compactness Theorem for the consequence relation:

2.1 Lemma. *Suppose* Comp($\mathcal{L}$), *and let* $\Phi \cup \{\varphi\} \subset L(S)$ *and* $\Phi \models_{\mathcal{L}} \varphi$. *Then there is a finite subset* Φ_0 *of* Φ *such that* $\Phi_0 \models_{\mathcal{L}} \varphi$.

Proof. Choose $\neg\varphi$ by Boole($\mathcal{L}$). Then $\Phi \cup \{\neg\varphi\}$ is not satisfiable. By Comp($\mathcal{L}$) there is a finite subset Φ_0 of Φ so that $\Phi_0 \cup \{\neg\varphi\}$ is not satisfiable, i.e. $\Phi_0 \models_{\mathcal{L}} \varphi$. □

If Comp($\mathcal{L}$) holds, the meaning of an $L(S)$-sentence only depends on finitely many symbols from S:

2.2 Lemma. *Suppose* Comp($\mathcal{L}$) *and* $\psi \in L(S)$. *Then there is a finite subset* S_0 *of* S *such that for all* S*-structures* $\mathfrak{A}$ *and* $\mathfrak{B}$:

$$\textit{If } \mathfrak{A}|_{S_0} \cong \mathfrak{B}|_{S_0} \textit{ then} \quad (\mathfrak{A} \models_{\mathcal{L}} \psi \quad \textit{iff} \quad \mathfrak{B} \models_{\mathcal{L}} \psi).$$

Proof. We restrict ourselves to the case where S is relational (the case we shall subsequently need). There is no difficulty in extending the proof to arbitrary symbol sets.

Choose new unary symbols U, V and f. Let Φ consist of the following first-order $S \cup \{U, V, f\}$-sentences, which say that f is an isomorphism between the substructure induced on U and the substructure induced on V:

$$\begin{aligned}
&\exists x Ux,\ \exists x Vx,\\
&\forall x(Ux \to Vfx),\ \forall y(Vy \to \exists x(Ux \wedge fx \equiv y)),\\
&\forall x \forall y((Ux \wedge Uy \wedge fx \equiv fy) \to x \equiv y),
\end{aligned}$$

and, for every $R \in S$, R n-ary:

$$\forall x_1 \ldots \forall x_n((Ux_1 \wedge \ldots \wedge Ux_n) \to (Rx_1 \ldots x_n \leftrightarrow Rfx_1 \ldots fx_n)).$$

Then, firstly (note that $\mathcal{L}_{\mathrm{I}} \leq \mathcal{L}$!),

$$\Phi^* \models_{\mathcal{L}} \psi^U \leftrightarrow \psi^V. \tag{1}$$

In fact, if $\mathfrak{A}$ is an S-structure and $(\mathfrak{A}, U^A, V^A, f^A) \models_{\mathcal{L}} \Phi^*$, i.e. if we have $(\mathfrak{A}, U^A, V^A, f^A) \models \Phi$, then U^A and V^A are nonempty and $f^A|_{U^A}$ is an isomorphism from $[U^A]^{\mathfrak{A}}$ to $[V^A]^{\mathfrak{A}}$. By the isomorphism property (cf. 1.1(c)) we have

$$[U^A]^{\mathfrak{A}} \models_{\mathcal{L}} \psi \quad \text{iff} \quad [V^A]^{\mathfrak{A}} \models_{\mathcal{L}} \psi,$$

that is, by Rel($\mathcal{L}$),

$$(\mathfrak{A}, U^A) \models_{\mathcal{L}} \psi^U \quad \text{iff} \quad (\mathfrak{A}, V^A) \models_{\mathcal{L}} \psi^V.$$

Using the reduct property and Boole($\mathcal{L}$), we obtain

$$(\mathfrak{A}, U^A, V^A, f^A) \models_{\mathcal{L}} \psi^U \leftrightarrow \psi^V.$$

Thus (1) is proved. By Comp($\mathcal{L}$) there is a finite subset Φ_0 of Φ such that

$$\Phi_0^* \models_{\mathcal{L}} \psi^U \leftrightarrow \psi^V. \tag{2}$$

Since Φ_0 consists of first-order sentences, we may choose a finite subset S_0 of S such that Φ_0 consists of S_0-sentences. We show that S_0 has the desired properties. Suppose $\mathfrak{A}$ and $\mathfrak{B}$ are S-structures and $\pi\colon \mathfrak{A}|_{S_0} \cong \mathfrak{B}|_{S_0}$, where we assume $A \cap B = \emptyset$. (Otherwise, we can take an isomorphic copy of $\mathfrak{B}$ and use the isomorphism property.) We define over $C := A \cup B$ an $S \cup \{U, V, f\}$-structure $(\mathfrak{C}, U^C, V^C, f^C)$ as follows (note that S is relational!):

$$\begin{aligned} &R^C := R^A \cup R^B \text{ for } R \in S, \\ &U^C := A, \quad V^C := B, \\ &f^C \text{ such that } f^C|_{U^C} = \pi. \end{aligned}$$

Then $(\mathfrak{C}, U^C, V^C, f^C)$ is a model of Φ_0, i.e., $(\mathfrak{C}, U^C, V^C, f^C) \models_{\mathcal{L}} \Phi_0^*$. Hence by (2),

$$(\mathfrak{C}, U^C, V^C, f^C) \models_{\mathcal{L}} \psi^U \leftrightarrow \psi^V,$$

and therefore, using $[U^C]^{\mathfrak{C}} = \mathfrak{A}$ and $[V^C]^{\mathfrak{C}} = \mathfrak{B}$,

$$\mathfrak{A} \models_{\mathcal{L}} \psi \quad \text{iff} \quad \mathfrak{B} \models_{\mathcal{L}} \psi. \qquad \square$$

§3. Lindström's First Theorem

In the following, let $\mathcal{L}$ be a regular logical system with $\mathcal{L}_{\mathrm{I}} \leq \mathcal{L}$. Furthermore, let S be a relational symbol set and ψ an $L(S)$-sentence which is not logically equivalent to any first-order sentence. To prepare for Lindström's Theorems we first show that there are structures $\mathfrak{A}$ and $\mathfrak{B}$ with $\mathfrak{A} \models_{\mathcal{L}} \psi$ and $\mathfrak{B} \models_{\mathcal{L}} \neg\psi$, which are nearly identical with respect to the first-order language.

For a first-order S-sentence φ, let φ^* again be a logically equivalent sentence in $L(S)$.

3.1 Lemma. *Let S be a relational symbol set and ψ an $L(S)$-sentence which is not logically equivalent to any first-order sentence. Then, for every finite $S_0 \subset S$ and every $m \in \mathbf{N}$, there are S-structures $\mathfrak{A}$ and $\mathfrak{B}$ such that:*

$$(+) \qquad \mathfrak{A}|_{S_0} \cong_m \mathfrak{B}|_{S_0}, \ \ \mathfrak{A} \models_{\mathcal{L}} \psi, \ \ \textit{and} \ \ \mathfrak{B} \models_{\mathcal{L}} \neg\psi.$$

Proof. Let S_0 be a finite subset of S and, without loss of generality, $m \geq 1$. We set, using the formulas $\varphi^m_{\mathfrak{B}}$ of XII.3,

$$\varphi := \bigvee\{\varphi^m_{\mathfrak{A}|_{S_0}} \mid \mathfrak{A} \text{ is an } S\text{-structure and } \mathfrak{A} \models_{\mathcal{L}} \psi\}.$$

By XII.3.4, this disjunction is finite, hence φ is a first-order sentence. Obviously $\psi \to \varphi^*$ is valid. Since, by assumption, ψ is not logically equivalent to φ, and hence not to φ^*, there is an S-structure $\mathfrak{B}$ such that $\mathfrak{B} \models_{\mathcal{L}} \varphi^*$ and $\mathfrak{B} \models_{\mathcal{L}} \neg\psi$. Since $\mathfrak{B} \models \varphi$, there is an S-structure $\mathfrak{A}$ such that $\mathfrak{A} \models_{\mathcal{L}} \psi$ and $\mathfrak{B} \models \varphi^m_{\mathfrak{A}|_{S_0}}$. Therefore, we also have $\mathfrak{A}|_{S_0} \cong_m \mathfrak{B}|_{S_0}$ (cf. XII.3.10). $\square$

In the proofs of Lindström's Theorems we shall essentially use the fact that 3.1 can be formulated in $\mathcal{L}$. Let us turn to such a formulation, using the terminology of partial isomorphisms as introduced in XII.3. For $m \in \mathbf{N}$ we choose ψ, $\mathfrak{A}$, $\mathfrak{B}$, S_0 and $(I_n)_{n\leq m}$ such that $(I_n)_{n\leq m}\colon \mathfrak{A}|_{S_0} \cong_m \mathfrak{B}|_{S_0}$, $\mathfrak{A} \models_{\mathcal{L}} \psi$, $\mathfrak{B} \models_{\mathcal{L}} \neg\psi$. We may assume that $A \cap B = \emptyset$ (otherwise we take an isomorphic copy of $\mathfrak{B}$). Let the symbol set S^+ be obtained from S by adding the following new symbols: a constant c, a unary function symbol f, and relation symbols P, U, V, W (unary), $<, I$ (binary) and G (ternary). We define an S^+-structure $\mathfrak{C}$ which contains $\mathfrak{A}$ and $\mathfrak{B}$ and which allows to describe the m-isomorphism property $(I_n)_{n\leq m}\colon \mathfrak{A}|_{S_0} \cong_m \mathfrak{B}|_{S_0}$, by including the partial isomorphisms from I_n as elements of its domain. More exactly:

(a) $C = A \cup B \cup \{0, \ldots, m\} \cup \bigcup_{n\leq m} I_n$;

(b) $U^C = A$ and $[U^C]^{\mathfrak{C}|S} = \mathfrak{A}$;

(c) $V^C = B$ and $[V^C]^{\mathfrak{C}|S} = \mathfrak{B}$;

((b) and (c) are possible since $A \cap B = \emptyset$ and since S is relational.)

(d) $W^C = \{0, \ldots, m\}$, $<^C$ is the natural ordering relation on $\{0, \ldots, m\}$, $c^C = m$ and $f^C|_{W^C}$ is the predecessor function on W^C, i.e. $f^C(n+1) = n$ for $n < m$, and, say, $f^C(0) = 0$;

(e) $P^C = \bigcup_{n\leq m} I_n$;

(f) $I^C np$ iff $n \leq m$ and $p \in I_n$;

(g) $G^C pab$ iff $P^C p$, $a \in \mathrm{dom}(p)$ and $p(a) = b$.

Figure XIII.1 gives an illustration.

FIGURE XIII.1.

$\mathfrak{C}$ is then a model of the conjunction χ of the following finite set of sentences of $L(S^+)$, which yields the desired formulation of (+). (Note that χ does not depend on m. Here and later we use first-order sentences as an intuitive notation for the corresponding sentences of $\mathcal{L}$.)

(i) $\forall p(Pp \to \forall x \forall y(Gpxy \to (Ux \wedge Vy)))$.

(ii) $\forall p(Pp \to \forall x \forall x' \forall y \forall y'((Gpxy \wedge Gpx'y') \to (x \equiv x' \leftrightarrow y \equiv y')))$.

(iii) For every $R \in S_0$, R n-ary:

$$\forall p(Pp \to \forall x_1 \ldots \forall x_n \forall y_1 \ldots \forall y_n((Gpx_1y_1 \wedge \ldots \wedge Gpx_ny_n) \to (Rx_1 \ldots x_n \leftrightarrow Ry_1 \ldots y_n))).$$

((i), (ii) and (iii) say that for a fixed $p \in P$, $Gp \cdot\cdot$ describes the graph of a partial isomorphism of the S_0-substructure induced on U to the S_0-substructure induced on V.)

(iv) The axioms of Φ_{pord} (for partially defined orderings) and the sentences

$\forall x(Wx \leftrightarrow (x \equiv c \vee \exists y(y < x \vee x < y))) \wedge \forall x(Wx \to (x < c \vee x \equiv c))$

($<$ is the empty relation and $W = \{c\}$, or W is the field of $<$ and c is the greatest element; in both cases we say that W is the field of $<$),

$\forall x(\exists y\, y < x \to (fx < x \wedge \neg \exists z(fx < z \wedge z < x)))$

(f is the predecessor function).

(v) $\forall x(Wx \to \exists p(Pp \wedge Ixp)$

(if x is in the field of $<$ then $I_x = \{p \mid Pp \wedge Ixp\}$ is nonempty).

(vi) $\forall x \forall p \forall u((fx < x \wedge Ixp \wedge Uu) \to \exists q \exists v(Ifxq \wedge Gquv \wedge \forall x' \forall y'(Gpx'y' \to Gqx'y')))$

(the forth-property).

(vii) An analogous sentence for the back-property.

(viii) $\exists x Ux \wedge \exists y Vy \wedge \psi^U \wedge (\neg\psi)^V$

(note that $U^C = A$, $V^C = B$, $\mathfrak{A} \models_{\mathcal{L}} \psi$, $\mathfrak{B} \models_{\mathcal{L}} \neg\psi$).

We have:

3.2. *For every $m \in \mathbf{N}$ there is a model $\mathfrak{C}$ of χ in which the field W^C of $<^C$ consists of exactly $(m+1)$ elements.* □

We now show:

3.3. *Assume* LöSko($\mathcal{L}$). *Then one of the following conditions* (a) *or* (b) *holds:*

(a) *There are S-structures $\mathfrak{A}$ and $\mathfrak{B}$ such that*

$$\mathfrak{A} \models_{\mathcal{L}} \psi, \ \ \mathfrak{B} \models_{\mathcal{L}} \neg\psi, \text{ and } \mathfrak{A}|_{S_0} \cong \mathfrak{B}|_{S_0}.$$

(b) *In all models $\mathfrak{D}$ of χ, the field W^D of $<^D$ is finite.*

Proof. First we show:

(∘) If the S^+-structure $\mathfrak{D}$ is a model of χ, in which the field W^D of $<^D$ is infinite, then the U-part and the V-part of $\mathfrak{D}$ are domains of S-substructures $\mathfrak{A} := [U^D]^{\mathfrak{D}|S}$ and $\mathfrak{B} := [V^D]^{\mathfrak{D}|S}$ such that
$$\mathfrak{A} \models_{\mathcal{L}} \psi, \quad \mathfrak{B} \models_{\mathcal{L}} \neg\psi \quad \text{and} \quad \mathfrak{A}|_{S_0} \cong_p \mathfrak{B}|_{S_0}.$$

Indeed: Since $\mathfrak{D}$ satisfies the sentences in (viii), $U^D \neq \emptyset$ and $V^D \neq \emptyset$; and since S is relational, U^D and V^D are domains of S-substructures. Again by (viii), $\mathfrak{D} \models_{\mathcal{L}} \psi^U, \mathfrak{D} \models_{\mathcal{L}} (\neg\psi)^V$, and therefore
$$\mathfrak{A} \models_{\mathcal{L}} \psi, \quad \mathfrak{B} \models_{\mathcal{L}} \neg\psi.$$

From (i), (ii), (iii) we know that every $p \in P^D$ corresponds, via G^D, to a partial isomorphism from $\mathfrak{A}|_{S_0}$ to $\mathfrak{B}|_{S_0}$, which we also denote by p. We extract from P^D a subset I in the following way: Let $f^0c, f^1c, f^2c, \ldots$ be abbreviations for $c, fc, ffc, \ldots$. Since W^D is infinite and c^D is the last element of $<^D$, $<^D$ has an infinite descending chain (f is the predecessor function, cf. (iv)):
$$\ldots <^D (f^2c)^D <^D (fc)^D <^D c^D.$$

We set
$$I := \{p \mid \text{ there is an } n \text{ with } I^D(f^nc)^Dp\}$$
and show
$$I\colon \mathfrak{A}|_{S_0} \cong_p \mathfrak{B}|_{S_0}.$$

Indeed, by (v) we get that $I \neq \emptyset$, and by (vi) and (vii) that I has the forth- and the back-property. For example, for the forth-property we conclude as follows: If $p \in I$, say $I^D(f^nc)^Dp$, and $a \in A = U^D$, then by (vi) there is a q such that $I^D(f^{n+1}c)^Dq$ (thus $q \in I$), $q \supset p$, and $a \in \mathrm{dom}(q)$. Hence, (∘) is proved.

Now we return to the proof of 3.3 and suppose that (b) does not hold. So there is a model of χ, in which the field W of $<$ is infinite. We show below, using LöSko($\mathcal{L}$), that we may assume the domain of this model to be countable. Then, by (∘), the following holds for the U-part $\mathfrak{A}$ and the V-part $\mathfrak{B}$:
$$\mathfrak{A} \models_{\mathcal{L}} \psi, \quad \mathfrak{B} \models_{\mathcal{L}} \neg\psi \quad \text{and} \quad \mathfrak{A}|_{S_0} \cong_p \mathfrak{B}|_{S_0};$$

Since $\mathfrak{A}|_{S_0}$ and $\mathfrak{B}|_{S_0}$ are partially isomorphic and at most countable, they are isomorphic (cf. XII.1.5(d)). Hence part (a) in 3.3 is satisfied.

It remains to justify the transition to a countable model. So let $\mathfrak{D}$ be a model of χ with infinite field W^D of $<^D$. As mentioned before, to obtain from $\mathfrak{D}$ a countable model of χ with an infinite field, we use LöSko($\mathcal{L}$). Since LöSko only holds for sentences, not for infinite sets of sentences, we have to ensure by a single sentence that the field of $<$ is infinite. This is done as follows: $<^D$ has an infinite descending chain (see above),
$$\ldots <^D (f^2c)^D <^D (fc)^D <^D c^D.$$

Let Q be a new unary relation symbol and let ϑ be the $L(S^+ \cup \{Q\})$-sentence

$$\vartheta = Qc \wedge \forall x(Qx \to (fx < x \wedge Qfx))$$

("Q contains c, and every element in Q contains an immediate $<$-predecessor which also belongs to Q.")

With $Q^D := \{(f^n c)^D \mid n \in \mathbf{N}\}$ we have:

$$(\mathfrak{D}, Q^D) \models_{\mathcal{L}} \chi \wedge \vartheta.$$

So, since $\chi \wedge \vartheta$ is satisfiable, by LöSko($\mathcal{L}$) there exists an at most countable model $(\mathfrak{E}, Q^E)$ of $\chi \wedge \vartheta$. $\mathfrak{E}$ is an at most countable model of χ, and the field W^E of $<^E$ is infinite. □

For the following applications we summarize our considerations (of 3.2 and 3.3):

3.4 Main Lemma. *Let $\mathcal{L}$ be a regular logical system with $\mathcal{L}_{\mathrm{I}} \leq \mathcal{L}$ and* LöSko($\mathcal{L}$). *Furthermore, let S be a relational symbol set, and let ψ be a sentence in $L(S)$, which is not logically equivalent to any first-order sentence. Then* (a) *or* (b) *holds:*

(a) *For all finite symbol sets S_0 with $S_0 \subset S$ there are S-structures $\mathfrak{A}$ and $\mathfrak{B}$ such that*

$$\mathfrak{A} \models_{\mathcal{L}} \psi, \quad \mathfrak{B} \models_{\mathcal{L}} \neg\psi \quad \text{and} \quad \mathfrak{A}|_{S_0} \cong \mathfrak{B}|_{S_0}.$$

(b) *For a unary relation symbol W and a suitable symbol set S^+ with $S \cup \{W\} \subset S^+$ and finite $S^+ \setminus S$, there is an $L(S^+)$-sentence χ such that*

  (i) *In every model $\mathfrak{C}$ of χ, W^C is finite and nonempty.*

  (ii) *For every $m \geq 1$ there is a model $\mathfrak{C}$ of χ, in which W^C has exactly m elements.*

Now we show:

3.5 Lindström's First Theorem. *For a regular logical system $\mathcal{L}$ with $\mathcal{L}_{\mathrm{I}} \leq \mathcal{L}$ the following holds:*

If LöSko($\mathcal{L}$) *and* Comp($\mathcal{L}$), *then* $\mathcal{L} \sim \mathcal{L}_{\mathrm{I}}$.

Proof. Assume, for contradiction, that ψ is a sentence in $L(S)$ which is not logically equivalent to any first-order sentence. By 1.4 we may assume that S is relational. Since Comp($\mathcal{L}$) holds, by 2.2 the meaning of ψ depends only on finitely many symbols. So we can choose a finite subset S_0 of S such that for all S-structures $\mathfrak{A}, \mathfrak{B}$:

$$\text{If} \quad \mathfrak{A}|_{S_0} \cong \mathfrak{B}|_{S_0} \quad \text{then } (\mathfrak{A} \models_{\mathcal{L}} \psi \quad \text{iff} \quad \mathfrak{B} \models_{\mathcal{L}} \psi).$$

Hence the condition (a) in 3.4 is not satisfied, and therefore (b) must hold, i.e., there is an $\mathcal{L}$-sentence χ which satisfies (i) and (ii) in 3.4(b). But this contradicts Comp($\mathcal{L}$): By (i), the set of sentences

$$\{\chi\} \cup \{\text{“}W \text{ contains at least } n \text{ elements”} \mid n \in \mathbf{N}\}$$

is not satisfiable, but by (ii), every finite subset has a model. □

To clarify the role of the conditions LöSko($\mathcal{L}$) and Comp($\mathcal{L}$) in Lindström's First Theorem, we describe the main idea of the proof once again:

Starting with the assumption that ψ is an $\mathcal{L}$-sentence which is not logically equivalent to any first-order sentence, for any $m \geq 1$ we obtain structures $\mathfrak{A}$ and $\mathfrak{B}$ with

(1) $$\mathfrak{A} \models_{\mathcal{L}} \psi \text{ and } \mathfrak{B} \models_{\mathcal{L}} \neg\psi$$

(2) $$\mathfrak{A} \cong_m \mathfrak{B}.$$

By Comp($\mathcal{L}$) we get structures $\mathfrak{A}$ and $\mathfrak{B}$ with

(1) $$\mathfrak{A} \models_{\mathcal{L}} \psi \text{ and } \mathfrak{B} \models_{\mathcal{L}} \neg\psi$$

(2′) $$\mathfrak{A} \cong_p \mathfrak{B}.$$

LöSko($\mathcal{L}$) allows to find countable structures which satisfy (1) and (2′) and therefore

(1) $$\mathfrak{A} \models_{\mathcal{L}} \psi \text{ and } \mathfrak{B} \models_{\mathcal{L}} \neg\psi$$

(2″) $$\mathfrak{A} \cong \mathfrak{B},$$

a contradiction. In (2), (2′) and (2″) we do not explicitly refer to a finite symbol set; however, this is not important since, by Comp($\mathcal{L}$), the sentence ψ depends only on finitely many symbols (cf. 2.2).

Lindström's Theorem 3.5 characterizes first-order logic in the following sense: Among the regular logical systems there is none of greater expressive power which still satisfies the Compactness Theorem and the Löwenheim-Skolem Theorem.

If one considers the defining properties of regular logical systems $\mathcal{L}$, the properties Rel($\mathcal{L}$) and Repl($\mathcal{L}$) do not seem as fundamental as the others. An analysis of the proof of 3.3 shows that both these properties were used to speak about two structures $\mathfrak{A}$ and $\mathfrak{B}$ in $\mathcal{L}$ by placing them together in the structure $\mathfrak{C}$. There are alternative properties that can be used instead of Rel($\mathcal{L}$) and Repl($\mathcal{L}$): For given structures $\mathfrak{A}$ and $\mathfrak{B}$, say $\mathfrak{A} = (A, P^{\mathfrak{A}})$ and $\mathfrak{B} = (B, P^{\mathfrak{B}})$ with $A = B$ (one can reduce to the case where both domains are the same), we consider the structure $\mathfrak{C} = (A, P^{\mathfrak{C}}, Q^{\mathfrak{C}})$ with $P^{\mathfrak{C}} = P^{\mathfrak{A}}$ and $Q^{\mathfrak{C}} = P^{\mathfrak{B}}$. If $\mathcal{L}$ is one of the logical systems considered in Chapter IX, then it is possible to talk about $\mathfrak{A}$ in $\mathfrak{C}$, since $\mathfrak{A} = \mathfrak{C}|_{\{P\}}$, and about $\mathfrak{B}$, since for every $\varphi \in L(\{P\})$ there exists a $\varphi' \in L(\{Q\})$ which says the same in $\mathfrak{C}$ as φ does in $\mathfrak{B}$. (φ' is obtained from φ by replacing P by Q.) In the proof of Lindström's First Theorem, we can eliminate the use of Rel($\mathcal{L}$) and Repl($\mathcal{L}$), if $\mathcal{L}$ permits this kind of replacements. But if there

is no substitute for $\mathrm{Rel}(\mathcal{L})$ and $\mathrm{Repl}(\mathcal{L})$, there are counterexamples to 3.5 (cf. [3]).

The method in the proof of 3.5 can also be used to prove properties of first-order logic. The following exercises contain a few examples.

3.6 Exercise. Show that a first-order sentence whose class of models is closed under substructures is logically equivalent to a universal sentence. (For the converse, see III.5.8.) Hint: Let S be finite, $\psi \in L_{\mathrm{I}}(S)$, $\mathrm{Mod}^S(\psi)$ be closed under substructures. For $m \geq 1$ set $\varphi^m := \bigvee\{\psi^m_{\mathfrak{B}} \mid \mathfrak{B}$ is an S-structure and $\mathfrak{B} \models \psi\}$ (cf. XII.3.14 for the definition of $\psi^m_{\mathfrak{B}}$). Then $\models \psi \to \varphi^m$, and φ^m is universal. Suppose ψ is not logically equivalent to any φ^m. As in the proof of 3.5, find S-structures $\mathfrak{A}$ and $\mathfrak{B}$ such that $\mathfrak{A} \models \neg\psi$, $\mathfrak{B} \models \psi$, and $\mathfrak{A}$ is embeddable in $\mathfrak{B}$. So $\mathfrak{B}$ has a substructure isomorphic to $\mathfrak{A}$, which is not a model of ψ, a contradiction.

3.7 Exercise. Let P be a k-ary relation symbol which is not in the symbol set S, and let Φ be a set of $(S \cup \{P\})$-sentences. Φ *defines* P *implicitly* if for every S-structure $\mathfrak{A}$ and $P^1, P^2 \subset A^k$ the following holds:

$$\text{If } (\mathfrak{A}, P^1) \models \Phi \text{ and } (\mathfrak{A}, P^2) \models \Phi \text{ then } P^1 = P^2.$$

Φ *defines* P *explicitly* if there is a $\psi \in L^S_k$ such that

$$\Phi \models \forall v_0 \ldots \forall v_{k-1}(Pv_0 \ldots v_{k-1} \leftrightarrow \psi).$$

Show the equivalence of (1) and (2), i.e., *Beth's Definability Theorem*:

(1) Φ defines P explicitly. (2) Φ defines P implicitly.

Hint: The direction from (1) to (2) follows immediately from the definitions. For the direction from (2) to (1) consider, for $n \geq 0$, the following formula:

$$\chi^n := \bigvee\{\varphi^n_{\mathfrak{A},\overset{k}{a}} \mid \mathfrak{A} \text{ is an } S\text{-structure}, (\mathfrak{A}, P^A) \models \Phi, \text{ and } P^A\overset{k}{a}\}.$$

Using the methods developed in this section, show that there is an n for which $\Phi \models \forall v_0 \ldots \forall v_{k-1}(Pv_0 \ldots v_{k-1} \leftrightarrow \chi^n)$.

§4. Lindström's Second Theorem

In our considerations of logical systems we now pay special attention to syntactic aspects. In this context we recall the following properties of first-order logic: For a decidable symbol set S

- the S-sentences are concrete finite symbol strings each of which contains only finitely many symbols from S,
- the set of S-sentences is decidable,

- operations such as negation, relativization and the replacement of function symbols can be carried out effectively,
- there exists an adequate proof calculus, and therefore the set of valid S-sentences is enumerable.

We shall consider these aspects for logical systems in general, thereby arriving at the concept of an effective logical system. Within this framework we can then formulate and prove the result of Lindström mentioned in the introduction to this chapter under (b).

When speaking of a decidable set, we understand it to be a set of words over a suitable alphabet that is R-decidable in the sense of X.2.5.

4.1 Definition. Let $\mathcal{L}$ be a logical system. $\mathcal{L}$ is called an *effective logical system* if for every decidable symbol set S the set $L(S)$ is decidable, and for every $\varphi \in L(S)$ there is a finite subset S_0 of S such that $\varphi \in L(S_0)$.

4.2 Definition. Let $\mathcal{L}$ and $\mathcal{L}'$ be effective logical systems.

(a) $\mathcal{L} \leq_{\text{eff}} \mathcal{L}'$:iff for every decidable S there is a computable function * which associates with every $\varphi \in L(S)$ a sentence $\varphi^* \in L'(S)$ such that $\text{Mod}^S_{\mathcal{L}}(\varphi) = \text{Mod}^S_{\mathcal{L}'}(\varphi^*)$.

(b) $\mathcal{L} \sim_{\text{eff}} \mathcal{L}'$:iff $\mathcal{L} \leq_{\text{eff}} \mathcal{L}'$ and $\mathcal{L}' \leq_{\text{eff}} \mathcal{L}$.

$\mathcal{L}_{\text{I}}$, $\mathcal{L}^w_{\text{II}}$, $\mathcal{L}_{\text{II}}$, and $\mathcal{L}_Q$ are effective logical systems, but $\mathcal{L}_{\omega_1\omega}$ is not. We have, for instance, $\mathcal{L}_{\text{I}} \leq_{\text{eff}} \mathcal{L}^w_{\text{II}}$, $\mathcal{L}^w_{\text{II}} \leq_{\text{eff}} \mathcal{L}_{\text{II}}$.

4.3 Definition. Let $\mathcal{L}$ be a logical system. $\mathcal{L}$ is said to be *effectively regular* if $\mathcal{L}$ is effective and if the following effective analogues of Boole($\mathcal{L}$), Rel($\mathcal{L}$) and Repl($\mathcal{L}$) hold.

For every decidable symbol set S:

(i) There is a computable function which assigns to every $\varphi \in L(S)$ a formula $\neg\varphi$, and, in addition, a computable function which assigns to any φ and ψ from $LL(S)$ a formula $(\varphi \vee \psi)$. (Here $\neg\varphi$, for instance, denotes an $L(S)$-sentence ψ such that $\mathfrak{A} \models_{\mathcal{L}} \psi$ iff not $\mathfrak{A} \models_{\mathcal{L}} \varphi$.)

(ii) For every unary U, there is a computable function which associates with every $\varphi \in L(S)$ a formula φ^U.

(iii) There is a computable function which associates with every $\varphi \in L(S)$ a formula $\varphi^r \in L(S^r)$ (where S^r is chosen as a decidable symbol set).

$\mathcal{L}_{\text{I}}$, $\mathcal{L}^w_{\text{II}}$, $\mathcal{L}_{\text{II}}$, and $\mathcal{L}_Q$ are effectively regular logical systems.

Let $\mathcal{L}$ be an effectively regular logical system. We say that *for $\mathcal{L}$ the set of valid sentences is enumerable* if for every decidable S, the set

$$\{\varphi \in L(S) \mid \models_{\mathcal{L}} \varphi\}$$

is enumerable.

Clearly, if $\mathcal{L}$ has an adequate proof calculus, then for $\mathcal{L}$ the set of valid sentences is enumerable. In particular, for $\mathcal{L}_{\mathrm{I}}$ and for $\mathcal{L}_Q$ the set of valid sentences is enumerable.

Lindström's Second Theorem tells us that among the effectively regular logical systems with LöSko($\mathcal{L}$) there is no system which is both properly stronger than $\mathcal{L}_{\mathrm{I}}$ and has an adequate proof calculus.

4.4 Lindström's Second Theorem. *Let $\mathcal{L}$ be an effectively regular logical system such that $\mathcal{L}_{\mathrm{I}} \leq_{\mathrm{eff}} \mathcal{L}$. If* LöSko($\mathcal{L}$) *and if for $\mathcal{L}$ the set of valid sentences is enumerable, then $\mathcal{L}_{\mathrm{I}} \sim_{\mathrm{eff}} \mathcal{L}$.*

Proof. Let $\mathcal{L}$ satisfy the hypotheses of the theorem. We prove that $\mathcal{L} \leq_{\mathrm{eff}} \mathcal{L}_{\mathrm{I}}$ in two steps.

First, we show:

(+) For every decidable S and for every $\psi \in L(S)$, there is a logically equivalent first-order S-sentence φ.

Then we prove that the transition from ψ to φ can be carried out effectively: Given a decidable S, we set up a *procedure* which yields for every $\psi \in L(S)$ a first-order S-sentence with the same models.

Since $\mathcal{L}$ is an effective logical system, we only need to give a proof of (+) for *finite* decidable S (cf. 4.1). Since (the effective variant of) Repl($\mathcal{L}$) holds for $\mathcal{L}$, we can assume S to be relational by an argument similar to that in the proof of 1.4.

Therefore, let S be decidable, finite, and relational.

To prove (+) we assume, for contradiction, that $\psi \in L(S)$ is a sentence which is not logically equivalent to any first-order sentence. Then (a) or (b) in 3.4 holds. Part (a) says for $S_0 := S$ (note that S is finite) that there are S-structures $\mathfrak{A}$ and $\mathfrak{B}$ such that $\mathfrak{A} \cong \mathfrak{B}$, $\mathfrak{A} \models \psi$ and $\mathfrak{B} \models \neg\psi$. Since this contradicts the isomorphism property in Definition 1.1 of a logical system, part (b) in 3.4 holds; that is, for a suitable finite symbol set S^+, containing S and a unary relation symbol W, there is a sentence χ in $L(S^+)$ with (i) and (ii):

(i) In every model $\mathfrak{C}$ of χ, W^C is finite and nonempty.

(ii) For every $m \geq 1$ there is a model $\mathfrak{C}$ of χ such that W^C has exactly m elements.

Thus, as $\mathfrak{C}$ ranges over the models of χ, W^C ranges over the finite sets (isomorphism property!). We shall now see that we can use (i) and (ii), together with Trahtenbrot's Theorem, to conclude that for $\mathcal{L}$ the set of valid sentences is not enumerable, in contradiction to our assumption on $\mathcal{L}$. We argue as in the proof of the incompleteness of second-order logic (cf. X.5.5).

By Trahtenbrot's Theorem there is a decidable symbol set S_1 such that the set of fin-valid first-order S_1-sentences is not enumerable. We may assume that S_1 is relational and disjoint from S^+.

Let * be a computable function which associates with every first-order S_1-sentence φ a sentence $\varphi^* \in L(S_1)$ which has the same models. Then for $\varphi \in L_0^{S_1}$ we have

$$(\circ) \qquad \varphi \text{ is fin-valid} \quad \text{iff} \quad \models_{\mathcal{L}} \chi \to (\varphi^*)^W.$$

To prove this, we assume first that φ is fin-valid. If $\mathfrak{A}$ is an $(S^+ \cup S_1)$-structure such that $\mathfrak{A} \models_{\mathcal{L}} \chi$, then W^A is finite and nonempty by (i), and thus $[W^A]^{\mathfrak{A}|_{S_1}} \models \varphi$. But then $[W^A]^{\mathfrak{A}|_{S_1}} \models_{\mathcal{L}} \varphi^*$, and hence $\mathfrak{A} \models_{\mathcal{L}} (\varphi^*)^W$. The converse is obtained similarly by applying (ii).

The equivalence $(\circ)$ enables us to obtain from an enumeration procedure $\mathfrak{P}$ for the set of valid $L(S^+ \cup S_1)$-sentences an enumeration procedure $\mathfrak{Q}$ for the fin-valid first-order S_1-sentences, thus yielding a contradiction to Trahtenbrot's Theorem. $\mathfrak{Q}$ proceeds as follows: For $n = 1, 2, 3, \ldots$ the (lexicographically) first n first-order S_1-sentences $\varphi_1, \ldots, \varphi_n$ are generated, and the $L(S^+ \cup S_1)$-sentences $\chi \to (\varphi_1^*)^W, \ldots, \chi \to (\varphi_n^*)^W$ are formed. (Note that the map * is computable and that the operations of relativization and implication are effective.) Then, using $\mathfrak{P}$, one generates the first n valid $L(S^+ \cup S_1)$-sentences, listing those φ_i for which $\chi \to (\varphi_i^*)^W$ occurs. This finishes the proof of $(+)$.

Now, given a decidable S, we describe an effective procedure which associates with every sentence $\psi \in L(S)$ a first-order S-sentence with the same models. Let $\mathfrak{P}$ be an enumeration procedure for the set of valid $L(S)$-sentences, and * a computable function which assigns to every first-order S-sentence φ an $L(S)$-sentence φ^* with the same models.

Given ψ, proceed as follows: For $n = 1, 2, 3, \ldots$ use $\mathfrak{P}$ to generate the first n valid sentences $\psi_1, \ldots, \psi_n$ from $L(S)$; then generate the (lexicographically) first n first-order S-sentences $\varphi_1, \ldots, \varphi_n$, and finally, form the $L(S)$-sentences $\psi \leftrightarrow \varphi_1^*, \ldots, \psi \leftrightarrow \varphi_n^*$. Check when there are i and j for the first time such that $\psi_i = \psi \leftrightarrow \varphi_j^*$ (by $(+)$ this must eventually happen!). Then let φ_j be the φ associated with ψ. $\square$

Lindström's results initiated a series of investigations of properties of logical systems and relations between them, in a general setting (cf. [3]). In this way it is possible to bring important aspects of such properties into better perspective, thus gaining new insights into concrete logical systems, and even into first-order logic. We illustrate this briefly, taking the Compactness Theorem as an example.

An ordering $(A, <^A)$ that contains no infinite descending chain

$$\ldots <^A a_2 <^A a_1 <^A a_0$$

is said to be a *well-ordering*. All finite orderings are well-orderings, as are $(\mathbf{N}, <^{\mathbf{N}})$, and the ordering which results when $(\mathbf{N}, <^{\mathbf{N}})$ is extended by adding an isomorphic copy. On the other hand, $(\mathbf{Z}, <^{\mathbf{Z}})$ and $(\mathbf{Q}, <^{\mathbf{Q}})$ are not well-orderings.

For the following discussion let $\mathcal{L}$ be a regular logical system such that $\mathcal{L}_{\mathrm{I}} \leq \mathcal{L}$. A well-ordering $(A, <^A)$ is said to be *$\mathcal{L}$-accessible* if there is an S with $< \in S$ and a satisfiable $L(S)$-sentence ψ such that

(a) in every model $\mathfrak{B}$ of ψ, (field $<^B, <^B$) is a well-ordering;

(b) there is a model $\mathfrak{B}$ of ψ such that $(A, <^A) \subset$ (field $<^B, <^B$).

Since $\mathcal{L}_{\mathrm{I}} \leq \mathcal{L}$, all finite well-orderings are $\mathcal{L}$-accessible. If Comp($\mathcal{L}$) holds then no infinite well-ordering is $\mathcal{L}$-accessible. For if a sentence ψ has a model $\mathfrak{A}$, where (field $<^A, <^A$) is an infinite well-ordering, then one can show, by a method similar to that used in Exercise VI.4.11, that ψ has a model $\mathfrak{B}$ in which (field $<^B, <^B$) has an infinite descending chain.

If one assumes LöSko($\mathcal{L}$) and strengthens the regularity conditions slightly, for example, by demanding the relativizations to hold in a suitable way also for relation symbols of larger arities[1], the following two statements are, in fact, equivalent:

(i) not Comp($\mathcal{L}$).

(ii) $(\mathbf{N}, <^{\mathbf{N}})$ is $\mathcal{L}$-accessible.

These considerations motivate us to look beyond the simple dichotomy "Comp($\mathcal{L}$) – not Comp($\mathcal{L}$)", and to make finer distinctions: the more (infinite) $\mathcal{L}$-accessible well-orderings there are, the more the Compactness Theorem is violated. As a measure for the violation one can take the "smallest" well-ordering which is not $\mathcal{L}$-accessible, the so-called *well-ordering number* of $\mathcal{L}$. The study of well-ordering numbers has led to a series of fruitful investigations (cf. [3]). In particular, it has turned out that for certain logical systems one can use arguments involving the well-ordering number to compensate for the absence of the compactness property.

---

[1]For further details see [3]. Here we only mention that the systems discussed in Chapter IX satisfy these strengthened regularity conditions.

References

For textbooks we state the year of publication of the first edition.

[1] K. R. Apt: Logic Programming. In: J. v. Leeuwen (Editor): Handbook of Theoretical Computer Science, vol. B. Elsevier, Amsterdam-New York-Oxford-Tokyo 1990

[2] J. Barwise: Admissible Sets and Structures. Springer-Verlag, Berlin-Heidelberg-New York 1975

[3] J. Barwise and S. Feferman (Editors): Model-Theoretic Logics. Springer-Verlag, Berlin-Heidelberg-New York-Tokyo 1985

[4] P. Benacerraf and H. Putnam (Editors): Philosophy of Mathematics. Selected Readings. Cambridge University Press, Cambridge ^{2}1983

[5] B. Bolzano: Wissenschaftslehre, vol. II of the four-volume edition. J. E. von Seidel, Sulzbach 1837

[6] C. C. Chang and H. J. Keisler: Model Theory. North-Holland Publishing Company, Amsterdam-London 1973

[7] A. Church: A Note on the Entscheidungsproblem. *The Journal of Symbolic Logic* **1** (1936)

[8] N. Cutland: Computability. Cambridge University Press, Cambridge 1980

[9] G. Frege: Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens. Louis Nebert, Halle 1879

[10] M. R. Garey and D. S. Johnson: Computers and Intractability. A Guide to the Theory of NP-Completeness. W. H. Freeman and Company, San Francisco 1979.

[11] K. Gödel: Die Vollständigkeit der Axiome des logischen Funktionenkalküls. *Monatshefte für Mathematik und Physik* **37** (1930)

[12] K. Gödel: Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatshefte für Mathematik und Physik* **38** (1931)

[13] L. Henkin: The Completeness of the First-Order Functional Calculus. *The Journal of Symbolic Logic* **14** (1949)

[14] J. M. Henle and E. M. Kleinberg: Infinitesimal Calculus. MIT Press, Cambridge, MA 1979

[15] H. Hermes: Enumerability, Decidability, Computability. Springer-Verlag, Berlin-Heidelberg-New York 1965

[16] H. Hermes: Introduction to Mathematical Logic. Springer-Verlag, Berlin-Heidelberg-New York 1973

[17] A. Heyting: Intuitionism. An Introduction. North-Holland Publishing Company, Amsterdam 1961

[18] D. Hilbert u P. Bernays: Grundlagen der Mathematik I, II. Springer-Verlag, Berlin-Heidelberg 1934/1939

[19] W. Hodges: Model Theory. Cambridge University Press 1993

[20] J. E. Hopcroft and J. D. Ullman: Introduction to Automata Theory, Languages, and Computation. Addison-Wesley Publishing Company, Reading 1979

[21] H. J. Keisler: Logic with the Quantifier "There exist uncountably many". *Annals of Mathematical Logic* **1** (1970)

[22] H. J. Keisler: Model Theory for Infinitary Logic. North-Holland Publishing Company, Amsterdam-London 1971

[23] K. Kunen: Set Theory. An Introduction to Independence Proofs. North-Holland Publishing Company, Amsterdam-New York-Oxford 1980

[24] A. Levy: Basic Set Theory. Springer-Verlag, Berlin-Heidelberg-New York 1979

[25] P. Lindström: On Extensions of Elementary Logic. *Theoria* **35** (1969)

[26] J. W. Lloyd: Foundations of Logic Programming. Springer-Verlag, Berlin-Heidelberg-New York 1984

[27] G. Peano: Arithmetices Principia, Novo Methodo Exposita. Fratres Bocca, Turin 1889

[28] A. Robinson: Non-Standard Analysis. North-Holland Publishing Company, Amsterdam-London 1966

[29] H. Scholz and G. Hasenjaeger: Grundzüge der mathematischen Logik. Springer-Verlag, Berlin-Göttingen-Heidelberg 1961

[30] R. M. Smullyan: First-Order Logic. Springer-Verlag, Berlin-Heidelberg-New York 1968

[31] L. Sterling and E. Y. Shapiro: The Art of PROLOG. MIT Press, Cambridge, MA 1986

[32] A. Tarski: Der Wahrheitsbegriff in den formalisierten Sprachen. *Studia Philosophica* **1** (1936)

[33] A. Tarski, A. Mostowski and R. M. Robinson: Undecidable Theories. North-Holland Publishing Company, Amsterdam 1953

[34] A. Turing: On Computable Numbers, with an Application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society* **42** (1936/37) and **43** (1937)

For further reading we refer to the *Handbook of Mathematical Logic*, edited by J. Barwise, North-Holland Publishing Company, Amsterdam-New York-Oxford 1977.

English translations of parts of [5] and of [9], [11], and [12] can be found in *From Frege to Gödel*, edited by J. van Heijenoort, Harvard University Press, Cambridge, MA, 1967. [11] and [12] are translated in Kurt Gödel: Collected Works, Volume I (edited by S. Feferman a.o.), Oxford University Press, New York 1986.

A list of all the literature in mathematical logic until 1985 is contained in the six volumes of the *Ω-Bibliography of Mathematical Logic*, edited by G. H. Müller, Springer-Verlag, Berlin-Heidelberg-New York 1987.

Symbol Index

$\mathbf{R}$	4	$\mathfrak{I}(t)$	32
$\mathbf{Z}$	5	$\mathfrak{I} \models \varphi$	32
$\mathcal{A}^*$	11	$\Phi \models \varphi$	34
$\square$	11	Φ_{gr}	34
$\mathbf{N}$	12	$\models \varphi$	35
$\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$	14	Sat φ, Sat Φ	35
$\forall$, $\exists$	14	$\varphi \mathrel{=\!\!|\!\!\models} \psi$	35
$\equiv$	14	$\mathfrak{A} \models \varphi[a_0, \ldots, a_{n-1}]$	37
$\mathcal{A}_S$	14	$t^{\mathfrak{A}}[a_0, \ldots, a_{n-1}]$	37
S	14	$\mathfrak{A} \models \varphi$	37
S_{gr}, S_{eq}, S_∞	14	$\mathfrak{A}'\|_S$	38
T^S	15	$\prod_{i \in I} \mathfrak{A}_i$	39
L^S	16	$\pi\colon \mathfrak{A} \cong \mathfrak{B}$	40
$\mathrm{var}(t)$	23	$\mathfrak{A} \cong \mathfrak{B}$	40
$\mathrm{SF}(\varphi)$	23	$\mathfrak{A} \subset \mathfrak{B}$	42
$\mathrm{free}(\varphi)$	25	$[X]^{\mathfrak{B}}$	42
L_n^S	25	Φ_{grp}	43
S_{ar}, $S_{\mathrm{ar}}^<$	29	$\varphi_{\geq n}$	46
$\mathfrak{N}$, $\mathfrak{N}^<$, $\mathfrak{R}$, $\mathfrak{R}^<$	29	Φ_{ord}	46
$\beta\frac{a}{x}$, $\mathfrak{I}\frac{a}{x}$	29	field $<^{\mathfrak{A}}$	46
$\mathfrak{A} \times \mathfrak{B}$	30	Φ_{pord}	46
$\dot{\vee}$, $\dot{\wedge}$, $\dot{\rightarrow}$, $\dot{\leftrightarrow}$, $\dot{\neg}$	31	Φ_{fd}, Φ_{ofd}	47

Φ_{dgph}, Φ_{gph}	47
$\mathfrak{N}_\sigma$	50
$t\frac{t_0\ldots t_r}{x_0\ldots x_r}$, $\varphi\frac{t_0\ldots t_r}{x_0\ldots x_r}$	53
$\beta\frac{a_0\ldots a_r}{x_0\ldots x_r}$, $\mathfrak{I}\frac{a_0\ldots a_r}{x_0\ldots x_r}$	54
$\mathrm{rk}(\varphi)$	56
$\exists^{=1}x\varphi$	57
$\Phi^{\models}$	59
$\vdash \Gamma\varphi$	61
(Ant), (Assm), (PC)	62
(Ctr), ($\vee$A), ($\vee$S), (TND)	63
(Ctr'), (Ch), (Cp)	64
($\exists$S), ($\exists$A)	66
($\equiv$), (Sub)	67
$\mathfrak{S}$	69
$\Phi \vdash \varphi$	70
Con Φ, Inc Φ	72
$\mathfrak{I}^\Phi$, $\mathfrak{T}^\Phi$, β^Φ	76
$\mathrm{free}(\Phi)$	80
$\mathrm{Mod}^S\Phi$	91
$\mathfrak{A} \equiv \mathfrak{B}$	94
$\mathrm{Th}(\mathfrak{A})$	94
$\boldsymbol{n}$	95
ZFC	107
$\psi(y_1,\ldots,y_n)$	108
ω	109
CH	110
S^r, $\mathfrak{A}^r$	116
S_{g}, Φ_{g}	118
Φ_{rg}	119
L^S_{II}	138
$\mathcal{L}_{\mathrm{II}}$, $\mathcal{L}_{\mathrm{I}}$	138
$\mathcal{L}^w_{\mathrm{II}}$	142
$\mathcal{L}_{\omega_1\omega}$	143
$\bigvee\Phi$, $\bigwedge\Phi$	143
$\mathcal{L}_Q$	148
$l(\zeta)$	153
$\mathrm{P}\colon\zeta \to \mathrm{halt}$	159
$\mathrm{P}\colon\zeta \to \infty$	159
$\mathrm{P}\colon\zeta \to \eta$	159
ξ_{P}	163
Π, Π_{halt}	163
SAT	166
$\mathcal{P}$, $\mathcal{NP}$	166
Φ_{fs}, Φ_{fv}	170
Φ_{PA}	173
Der_Φ	185
$\varphi(\overset{n}{x}\,|\,\overset{n}{t})$	190
T^S_k, T^Φ_k, $\mathfrak{T}^\Phi_k$	190
β^Φ_k, $\mathfrak{I}^\Phi_k$	190
$\mathrm{pvar}(\alpha)$	201

PF_n	201
$\alpha[b]$	201
A^S	202
DNF, CNF	204
p^T, p^F	205
b^Δ	208
$\mathrm{Res}(\mathfrak{K})$, $\mathrm{Res}_\infty(\mathfrak{K})$	212
$\mathrm{HRes}(\mathfrak{K})$, $\mathrm{HRes}_\infty(\mathfrak{K})$	217
$\mathrm{GI}(\varphi)$, $\mathrm{GI}(\Phi)$	220
ψ^F	221
Φ^+, Φ^-	223
$t\sigma$, $\varphi\sigma$	227
ι	227
$\mathrm{URes}(\mathfrak{K})$, $\mathrm{URes}_\infty(\mathfrak{K})$	233
$\mathrm{GI}(K)$, $\mathrm{GI}(\mathfrak{K})$	234
$\mathrm{UHRes}(\mathfrak{K})$, $\mathrm{UHRes}_\infty(\mathfrak{K})$	236
$\mathrm{dom}(p)$, $\mathrm{rg}(p)$	244
$\mathrm{Part}(\mathfrak{A}, \mathfrak{B})$	244
$\mathfrak{A} \cong_f \mathfrak{B}$	246
$(I_n)_{n\in\mathbf{N}}\colon \mathfrak{A} \cong_f \mathfrak{B}$	246
$\mathfrak{A} \cong_p \mathfrak{B}$	246
Φ_{dord}	247
Φ_σ	248
$\mathfrak{A} \to_f \mathfrak{B}$, $\mathfrak{A} \to_p \mathfrak{B}$	249
$\mathrm{qr}(\varphi)$	251
$\mathfrak{A} \cong_m \mathfrak{B}$, $\mathfrak{A} \equiv_m \mathfrak{B}$	252
$\varphi^n_{\mathfrak{B}}$, $\varphi^n_{\mathfrak{B},\vec{b}}$	253
$\vec{a} \mapsto \vec{b}$	254
$\mathfrak{A} \models_{\mathcal{L}} \varphi$	261
$\mathrm{Mod}^S_L(\varphi)$	262
$\mathcal{L} \leq \mathcal{L}'$, $\mathcal{L} \sim \mathcal{L}'$	262
$\mathrm{Boole}(\mathcal{L})$, $\mathrm{Rel}(\mathcal{L})$	262
$\mathrm{Repl}(\mathcal{L})$	262
$\mathrm{LöSko}(\mathcal{L})$, $\mathrm{Comp}(\mathcal{L})$	263
$\mathcal{L} \leq_{\mathrm{eff}} \mathcal{L}'$	273
$\mathcal{L} \sim_{\mathrm{eff}} \mathcal{L}'$	273

Subject Index

add-instruction, 157
Adequacy of the Sequent Calculus, 85
algorithm, 153
alphabet, 11, 153
  of a language, 14
and, 13, 32, 35
antecedent, 60
Aristotle, 3, 85
arithmetic, 29, 173, 175
  nonstandard model of, 96
  Peano, 173
  Theorem on the Undecidability of, 176
  truth in, 184
arithmetical, 180
assignment, 29
  propositional, 201
  second-order, 138
automorphism, 44
axiomatic method, 186
axiomatizable
  finitely, 174
  register-, 174
axioms
  independent system of, 94
  system of, 93

back-property, 246, 247
Barwise Compactness Theorem, 145
β-function, 177
Beth's Definability Theorem, 272
Bolzano, 40
Boole, 3
bound occurrence, 24
bounded in time
  t-, 166
  polynomially, 166

calculus, 15, 19, 60, 172
calculus of terms, 15
Cantor, 110, 141, 247
cardinality
  of the same, 110
CH, 141
chain
  descending, 98
chain rule, 64
characteristic
  of a field, 91
characterize
  up to isomorphism, 50, 95
Church, 161, 169
Church's Thesis, 161, 166
class
  Δ-elementary, 91
  elementary, 91
  of models, 91
clause, 211
  negative, 215
  positive, 215
  unifiable, 229
CNF, *see* conjunctive normal form, 204
Cobham, 166
Cohen, 110
Coincidence Lemma, 36
  of Propositional Calculus, 201
colloquial speech, 31
Compactness Theorem, 89, 96, 140, 142, 144
  $\mathcal{L}_Q$-, 149
  Barwise, 145
  for Propositional Logic, 204
complete theory, 174
completeness, 75
completeness axiom, 88
Completeness Theorem, 85, 99, 113

complexity theory, 166
computable, 156
    register-, 161
computer science, 3, 167
configuration, 168
conjunction, 16, 144
conjunctive normal form, 133, 204, 206
connective, 8, 31, 204
consequence relation, 5, 7, 33, 40, 202
consistency of mathematics, 111, 186
consistent, 72
constant, 14
containing witnesses, 78
continuum hypothesis, 109, 141
contraposition, 64
correct rule, 61
correct sequent, 61
Correctness of the Sequent Calculus, 70
countable, 12
    at most, 12

decidable, 152, 153
    register-, 160
decide, 160
decision problem, 169
decision procedure, 153
declarative, 197
Dedekind's Theorem, 50
definition, 126
    explicit, 272
    extension by, 127
    implicit, 272
Δ-elementary class, 91
derivable, 61
    H-, 215
    UH-, 235
derivable in a calculus, 15
derivation, 15
diagonal argument, 167
direct product, 30, 39
disjunction, 16, 143
disjunctive normal form, 130, 204, 206
DNF, *see* disjunctive normal form204
domain, 29

Edmonds, 166
effective procedure, 153
Ehrenfeucht game, 257
Ehrenfeucht's Theorem, 258
elementarily equivalent, 94, 246
elementary class, 91
embeddable, 249
    finitely, 249
    partially, 249
Entscheidungsproblem, 169
enumerable, 151, 154
    register-, 160
enumerate, 160
enumeration procedure, 154
epistemology, 3, 102
equality, 13, 139
    axioms for, 225
equation, 196
equivalence relation, 5, 45
equivalence structure, 5, 91
equivalent
    for satisfaction, 131
    logically, 34, 35, 202, 262
everyday language, 7
expansion, 38
extensional, 31

field, 47, 91
    algebraically closed, 175
    archimedean, 95
    archimedean ordered, 143
    ordered, 38, 47
finitely axiomatizable, 174
finitely embeddable, 249
finitely isomorphic, 246
finitistic, 186
first-order language, 9
first-order object, 9
Fixed Point Theorem, 183

follows from, 5, 7
for all, 13, 32, 36
formal proof, 8
formalization, 44
formally provable, 60, 61
formula, 7, 16
    atomic, 16
    equality-free, 198
    existential, 44
    Horn, 39, 47
    positive, 33
    propositional, 201
    term reduced, 115
    universal, 43
forth-property, 246
FORTRAN, 161
Fraïssé's Theorem, 249
Fraenkel, 107
free model, 193, 194
free occurrence, 24
Frege, 3, 18
function, 28
    arithmetical, 180
    partial, 48
function symbol, 14
function variable, 139
functionally complete, 206

general unifier, 229
Gödel, 85, 110, 172, 177, 181
Gödel numbering, 163, 182
Gödel's Completeness Theorem, 8, 85, 99, 113
Gödel's First Incompleteness Theorem, 184
Gödel's Second Incompleteness Theorem, 111, 185
graph, 47, 91, 219
    connected, 93, 144, 255
    directed, 47
    of a function, 48, 105, 116
ground clause, 231
ground instance, 220, 234
group, 4, 91, 118
    free, 196
    free abelian, 196
    simple, 147
    torsion, 50, 92, 143
group theory, 4, 151

halt-instruction, 158
halting problem, 163
Henkin, 85
Henkin's Theorem, 78
Herbrand model, 200
    minimal, 200
Herbrand structure, 199
Herbrand's Theorem, 192
Hilbert, 3, 156
Hilbert's program, 3, 111, 185
Hilbert's 10. Problem, 156
hold, 32
homomorphism, 194
Horn formula, 39, 47
    negative, 208, 223
    positive, 208, 223
    propositional, 208
    universal, 194
Horn sentence, 40
    universal, 170, 195

identitas indiscernibilium, 139
if and only if, 13, 35
if-then, 13, 32, 35
iff, 20
implication, 16
incompleteness
    of second-order logic, 172
Incompleteness Theorem
    Gödel's First, 184
    Gödel's Second, 111, 185
inconsistent, 72
independent, 39, 94
induction
    on formulas, 20
    on terms, 20
    over a calculus, 19
induction axiom, 50, 96
induction schema, 174
inductive definition

on formulas, 22
on terms, 22
inductive proof, 19
inductive set, 109
inference, 7, 59
infinitary language, 143
infinitesimal, 97
input, 152
instance, 220
ground, 220
integers, 119
intensional, 32
interpretation, 29
syntactic, 120
intuitionist, 102
isomorphic, 40
m-, 252
finitely, 246
partially, 246
isomorphism, 40
partial, 244
Isomorphism Lemma, 40
isomorphism property, 261

jump-instruction, 158

label, 157
language
alphabet of a, 14
everyday, 7
first-order, 9, 16
formal, 7
infinitary, 143
many-sorted, 49
second-order, 9, 138
Leibniz, 3, 85, 139, 170
length, 153
lexicographic order, 154
liar paradox, 182
Lindström, 261
Lindström's First Theorem, 270
Lindström's Second Theorem, 274
literal, 211
Llull, 85, 170
Löb axioms, 186
logic
first-order, 17
mathematical, 3
second-order, 137, 172
weak second-order, 142, 147, 172
logic programming, 189, 197, 237
logical system, 261
effective, 273
effectively regular, 273
regular, 263
logically equivalent, 34, 35, 202, 262
Löwenheim, Skolem and Tarski, Theorem of, 90
Löwenheim-Skolem Theorem, 87, 140, 142, 144, 145
downward, 88
upward, 89

mathematics, 3
classical, 102
consistency of, 111, 186
intuitionistic, 102
set-theoretical setup of, 103
Matijasevič, 180, 186
matrix, 130
metalanguage, 18
minimal model, 193, 208
model, 32, 38, 201
free, 193, 194
minimal, 193, 208
model theory, 90
Modus ponens, 65

natural numbers, 109
negation, 16
negation complete, 78
non-deterministic register program, 166
nonstandard analysis, 97
nonstandard model of arithmetic, 96
normal form

conjunctive, 133, 204
disjunctive, 130, 204
prenex, 130, 222
Skolem, 132, 223
not, 13, 32
notion of proof, 59

object
first-order, 9
second-order, 9
object language, 18
operation, syntactic, 151
or, 13, 32
ordering, 46, 118
ω_1-like, 148
dense, 247, 249
field of a, 46
partial, 46
partially defined, 46, 91
well-, 275
output, 152

"$\mathcal{P} = \mathcal{NP}$"-problem, 167, 207
paradox
liar, 182
Skolem's, 107
parameter, 24
partially embeddable, 249
partially isomorphic, 246
PASCAL, 161
Peano, 18
Peano arithmetic, 173
Peano axioms, 50, 96, 103
Peano structure, 103, 109
philosophy, 3
philosophy of science, 3
platonism, 102
Polish notation, 24
polynomially bounded in time, 166
predicate calculus
first-order, 17
prefix, 130
prenex normal form, 130, 222
print-instruction, 158
procedural, 197
procedure
decision, 153
effective, 153
enumeration, 154
process, 153
program, 158
non-deterministic, 166
PROLOG, 197, 218, 227
proof, 3, 7, 34
notion of, 62, 70, 99
proposition, 7
propositional logic, 166, 200
language of, 201
propositional variable, 200
provable, 7
formally, 61, 70

quantifier
for all, 8
number, 131
restricted, 45
there are at least countably many, 149
there are uncountably many, 148
there exists, 8
there exists exactly one, 57
quantifier rank, 251
quantifier-free, 42, 130
quotient structure, 225

R-, *see* register-160
rank, 56
modified, 256
recursion theory, 152
recursive, 161
recursively enumerable, 161
reduct, 38
reduct property, 261
register, 157
register machine, 157
register program, 158
register-axiomatizable, 174
register-computable, 161

register-decidable, 160
register-enumerable, 160
regular logical system, 263
relation, 5, 28
    arithmetical, 180
relation symbol, 14
relation variable, 138
relational, 116
relativization, 119, 124
representable, 181
resolution, 212
    H-, 215
    U-, 231, 235
    UH-, 235, 236
Resolution Lemma, 212
resolution method, 211
Resolution Theorem, 213
resolution tree, 214
resolvent, 212
    U-, 231
    unification, 231
ring, 119
    of integers, 119
Robinson, 211
rule, 19
    connective, 62
    correct, 61
    derivable, 64
    equality, 62, 66
    list of rules in $\mathfrak{S}$, 69
    quantifier, 62, 66
    structural, 62
Russell, 3, 85

S-closed, 42
SAT, 166
satisfaction
    equivalent for, 131
satisfaction relation, 32, 262
satisfiable, 34, 263
    clause, 211
    fin-, 170
    formula, 35
    propositional formula, 202
    propositionally, 221
    set of clauses, 211
    set of formulas, 35
satisfy, 32, 201
second-order language, 9
second-order object, 9
self-referential, 182
semantic, 28
sentence, 25
    of $\mathcal{L}$, 261
separator, 227
sequent, 60
    correct, 61
sequent calculus, 61, 100
    of propositional logic, 207
    Theorem on the Adequacy of, 85
set
    concept of, 104, 110
set theory, 186
    background, 106
    object, 107
    system of axioms for, 104
    Zermelo-Fraenkel axioms for, 107
Skolem, 107
Skolem normal form, 132, 223
Skolem's paradox, 107
Skolem's Theorem, 96
sort, 48
sort reduction, 49
spectrum, 47
statement
    cardinality, 46
    self-referential, 182
string, 11
    empty, 11
strong
    at least as, 262
    equally, 262
structure, 4, 28
    many-sorted, 48
    quotient, 225
substitution
    simultaneous, 52
Substitution Lemma, 54

substitutor, 227
substructure, 42, 272
    generated, 42
Substructure Lemma, 43
subtract-instruction, 157
succedent, 60
symbol, 11
symbol set, 14
    relational, 116
syntactic, 28
syntactic interpretation, 120
    associated, 127
syntactic operation, 60, 101
system of axioms, 6, 93

Tarski, 40
Tarski's Theorem, 184
term, 15
term interpretation, 77, 189
term-reduced, 115
term structure, 76, 189
Tertium non datur, 63
theory, 173
    complete, 174
    of a structure, 94
there exists, 13, 32
time complexity, 166
torsion group, 50, 92, 143
Trahtenbrot's Theorem, 171
truth, 183
truth-function, 204
truth-value, 31
Turing, 157

uncountable, 12
undecidability
    of arithmetic, 175, 176
    of first-order logic, 167
    of the halting problem, 163
underlining algorithm, 209
unifiable, 229
unification algorithm, 229
unification resolvent, 231
unifier, 229
    general, 229
    Lemma on the, 229
unit
    groups of, 119
    in a ring, 119
universal, 43, 194
universe, 29, 103
urelement, 104, 109

valid, 35, 202, 263
    fin-, 170
variable, 14
    function-, 139
    propositional, 200
    relation, 138
vector space, 49

well-ordering, 275
well-ordering number, 276
Whitehead, 85
witness, 78
word, 11
    length of, 11

Zermelo, 107
Zorn's Lemma, 84