FORMAL LOGIC

Formal logic originates with Aristotle and concerns the activity of drawing inferences. Of course, by the time language developed, humans had deduced conclusions from premises, but Aristotle inaugurated the systematic study of the rules involved in the construction of valid reasoning. The first important discovery of this approach was that the logical structure of sentences and deductions is given by some relations between signs in abstraction from their meaning. This aspect explains the attribute *formal.* Since the mid-nineteenth century, modern logic has emphasized this aspect by developing logic notational systems. In this sense it is also referred to as *symbolic logic,* or *mathematical logic,* inasmuch as the emergence of the symbolic perspective was stimulated by certain trends within mathematics, namely, the generalization of algebra, the development of the axiomatic method, especially in geometry, and the tendency, above all in analysis, to find basic concepts for a foundation of mathematics. The elaboration of the formal method in modern logic was pioneered by Leibniz (1646–1716) and was given its fundamental basis in the works of De Morgan (1806–1871), Boole (1815–1864), Peirce (1839–1914), Schöreder (1841–1902), Frege (1848–1925), Peano (1858–1932), Hilbert (1862–1943), Russell (1872–1970), Löwenheim (1878–1957), Skolem (1887–1963), Post (1897–1954), Tarski (1901–1983), Church (1903–?), Gödel (1906–1978), Herbrand (1908–1931), Gentzen (1909–1945), Kleene (1909–?), and Turing (1912–1954). Church's book (1) is a source of information for the history of logic before 1956; another valuable, and more recent, historical survey is Moore's (2); some fundamental works of modern formal logic are collected in (3,4).

In logic, the essential aspect of the formal method consists of a clear distinction between *syntax* and *semantics.* This is an intrinsic feature of any *formal language* as opposed to a natural language. Syntax establishes which (linear) arrangements of symbols of a specified alphabet should be considered as well-formed expressions, the categories in which they are classified, and the symbolic rules following which some relations between expressions are defined. Semantics establishes how to define the general concepts of interpretation, satisfiability, truth, consequence, and independence. This distinction does not mean that syntax and semantics are opposed, but rather complementary. In fact, syntax, defined separately from semantics can elaborate formulae by using only symbolic information which, by virtue of its nature, can be encoded in physical states of machines, and thus *calculated* or *mechanized.* Semantics, which deals with no particular interpretation of symbols, can formally define logical validity, which is conceived, according to Leibniz's definition, as truth in all possible worlds.

However, it took many decades for modern logic to make a clear and rigorous distinction between syntax and semantics. Such a distinction originated with the Warsaw School of Logic, and the first steps were made by Tarski, in the 1930s, toward its notion of interpretation of logical languages.

Syntactical and semantical methods are the approaches from which the two main branches of modern mathematical logic stem: *proof theory* and *model theory.* Proof theory is strictly related to the theory of effective processes, and thus it is connected to the notion of computation and algorithm. This field grew as an autonomous theory after the seminal work of Turing (1936), where the first mathematical model of a computing machine was introduced. Computability, or recursion theory in a more abstract perspective, was developed chiefly by Turing, Post, Gödel, Church, Kleene, Curry, and von Neumann, in connection with automata and formal languages theory (5).

Since its inception, model theory has been strictly related to the foundational theories of mathematics: set theory and arithmetic, along with many classical algebraic and geometric theories. Moreover, analysis too was able to benefit from the model-theoretic perspective: *Nonstandard analysis,* due to Abraham Robinson in the 1960s, gives, in purely logical terms, a rigorous foundation to the infinitesimal method of early (pre-Cauchy and pre-Weierstrass) analysis, as developed by Leibniz.

The extension of model- and proof-theoretic approaches to fields other than mathematics has become, especially since the 1960s, an important area of investigation, related to old problems in philosophical logic and to alternative approaches in the foundation of mathematics (6). In fact, many interesting situations require the formalization of concepts that are beyond the scope of typical mathematical problems—for example, constructive reasoning, modal notions, spatiotemporal relations, epistemological states, knowledge representation, natural language comprehension, and computational processes. All logical systems that deal with these subjects constitute the realm of *nonclassical,* or *alternative,* logics. The following are some nonclassical logics (and thinkers): *intuitionistic logic* (Brower, Heyting), *modal logic* (Lewis, Langford), *temporal logic* (Prior, Fine), *intensional logic* (Kripke, Montague), and *linear logic* (Girard). Comprehensive essays in these fields can be found in Refs. 5–9.

This wide spectrum of applications indicates the centrality and vitality of formal logic; moreover, the logical nature of computability (10a), the search for automated deduction systems (10), and the importance of almost all nonclassical logics for computer science (5,7,11) show that the connection between formal logic and computer science is so deep that it can be compared to the relationship between classical mathematics and physics.

First-order or (elementary) predicate logic is the basic logical system on which proof theory and model theory are built. It is also the basis for a deep understanding of advanced logical systems. The following sections present the fundamental results of predicate logic. Let us take a preliminary look at the symbolization process in logic. Consider seven *logical symbols:* $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$, $\forall$, $\exists$. Assume for them these intuitive meanings: $\neg\varphi$ asserts the *negation* of the sentence φ; $\varphi \wedge \psi$ asserts the *conjunction* of the two sentences φ and ψ; $\varphi \vee \psi$ asserts their *disjunction,* that is, that at least one of them holds; $\varphi \rightarrow \psi$ asserts that φ implies ψ; and $\varphi \leftrightarrow \psi$ asserts the *equivalence* between φ and ψ, that is, that either both sentences hold, or neither holds. Moreover, let us fix a *domain* where variables range; then $\forall x\varphi$ means that φ holds for any value of x, while $\exists x\varphi$ means that there exists at least one value of x that makes the sentence φ true. Let us consider an example. We use (i) three variables x, y, and z ranging on the individuals of some biological population with sexual reproduction and (ii) four *predicate symbols* P, A, M, F such that $P(x, y)$ means x *is a parent of* y, $A(x, y)$ means x *is an ancestor of* y, $M(x)$ means x *is a male,* and $F(x)$ means x *is a female.* By using logical symbols, predicate symbols, variables, and parentheses, we can put into formulae many sentences expressing some common facts about sexual reproduction:

- $\forall x(\exists y(P(y, x) \wedge M(y)))$ (Everybody has a father).
- $\forall x(\exists y(P(y, x) \wedge F(y)))$ (Everybody has a mother).
- $\forall x(M(x) \vee F(x))$ (Everybody is either male or female).
- $\forall x(\neg(A(x, x)))$ (Nobody is a self-ancestor).
- $\forall x(\forall y((P(x, y) \rightarrow A(x, y))))$ (Parents are ancestors).
- $\forall x(\forall y(\forall z((A(x, y) \wedge P(y, z)) \rightarrow A(x, z))))$ (The ancestors of parents are ancestors too).

These sentences constitute the *axioms* of a *theory.* Can we interpret them in a different domain, with different meanings for predicate symbols, in such a way that they could be true in this new interpretation as well? With a more detailed analysis, we discover that these axioms cannot be fulfilled by real biological populations, inasmuch as it is possible to show that they require a domain (if nonempty) with infinitely many individuals. On the other hand, we can interpret these formulae on natural numbers. However, how can we prove that a father's uniqueness is not a consequence of the given axioms? And, in what sense is $\neg\exists x(\forall y((A(x, y))))$—that is, the nonexistence of a common ancestor for all individuals—a logical consequence of them? Is there an algorithm for generating all the logical consequences of these axioms? The theory developed in the following sections will provide general answers to these questions.

In mathematical logic the relationship between mathematics and logic is twofold: On the one hand, mathematics provides tools and methods in order to find rigorous formulations and solutions to old logical problems; on the other hand, the logical analysis of mathematical concepts (after Hilbert, *metamathematics*) tries to define general notions and notations where all mathematical theories can be expressed. These two aspects have been strictly related since the early development of mathematical logic. Indeed, one of the most important results of the twentieth century was the definition of a foundational framework, essentially common to almost all mathematical theories. This framework relies on two theories which can be briefly depicted by two evocative expressions: *Cantor's Paradise,* according to a famous definition of set theory by Hilbert, and *Peano's Paradise,* an analogous expression adopted to indicate induction principles. Sets and induction, besides their enormous foundational aspect, are also the basis for the syntax and the semantics of predicate logic which will be presented below.

Hereafter the basic notation and concepts of set theory and arithmetic will be assumed: *membership, inclusion, classes, sets* (i.e., classes which belong to other classes), the set ω of *natural numbers* $\{0, 1, 2, \ldots\}$; *operations, sequences, relations, functions* (or maps); *equivalence* and *ordering relations; countable* (finite or denumerable) and *more than countable cardinalities; graphs* and *trees* with *König's tree lemma* (if an infinite tree has a positive but finite number of nodes at any level, then the tree has an infinite branch); and finally, *induction principles* for proving statements and for defining sets, functions, or relations.

The next seven sections describe the basic concepts and results in predicate logic. The final two sections outline some aspects centered around the notion of logical representability. This is the basis for many applications of formal logic and for a logical analysis of computability, which is the core discipline of theoretical computer science. In Refs. 1 and 12–16 there are some valuable presentations of predicate logic, along with introductions to the main branches of mathematical logic; logical representability is studied in depth in Refs. 17 and 18; and many important developments and applications of mathematical logic are presented in Refs. 6, 7, and 11.

THE SYNTAX OF PREDICATE LOGIC

A signature Σ is a set of *symbols* for denoting functions and relations. Each symbol is equipped with a number expressing its *arity.* Relation symbols of arity 0 are called *propositional symbols;* function symbols with arity 0 are called (*individual*) *constants.* Let us indicate by Σ_n^{fun} the set of n-ary function symbols of Σ, and by Σ_n^{rel} the set of n-ary relation symbols of Σ. Let V be a set of symbols for *individual variables* (usually letters from the end of the alphabet, with or without subscripts). The sets $T_\Sigma(V)$ of Σ-terms of variables V, along with the sets $F_\Sigma(V)$ of Σ-formulae of variables V, consist of sequences of symbols in the alphabet:

$$\Sigma \cup V \cup \{\neg, \wedge, \rightarrow, \leftrightarrow, \forall, \exists, =, (,)\}$$

defined by the following inductive conditions (where $\Rightarrow$ is the usual *if–then* implication):

- $c \in \Sigma_0^{\text{fun}} \Rightarrow c \in T_\Sigma(V)$
- $v \in V \Rightarrow v \in T_\Sigma(V)$
- $n > 0,\ f \in \Sigma_n^{\text{fun}},\ t_1, \ldots, t_n \in T_\Sigma(V) \Rightarrow f(t_1, \ldots, t_n) \in T_\Sigma(V)$
- $Q \in \Sigma_0^{\text{rel}} \Rightarrow Q \in F_\Sigma(V)$
- $n > 0,\ p \in \Sigma_n^{\text{rel}},\ t_1, \ldots, t_n \in T_\Sigma(V) \Rightarrow p(t_1, \ldots, t_n) \in F_\Sigma(V)$
- $t_1, t_2 \in T_\Sigma(V) \Rightarrow (t_1 = t_2) \in F_\Sigma(V)$
- $\varphi \in F_\Sigma(V) \Rightarrow (\neg\varphi) \in F_\Sigma(V)$
- $\varphi, \psi \in F_\Sigma(V) \Rightarrow (\varphi \wedge \psi) \in F_\Sigma(V)$
- $\varphi, \psi \in F_\Sigma(V) \Rightarrow (\varphi \vee \psi) \in F_\Sigma(V)$
- $\varphi, \psi \in F_\Sigma(V) \Rightarrow (\varphi \rightarrow \psi) \in F_\Sigma(V)$
- $\varphi, \psi \in F_\Sigma(V) \Rightarrow (\varphi \leftrightarrow \psi) \in F_\Sigma(V)$
- $v \in V,\ \varphi \in F_\Sigma(V) \Rightarrow (\forall v\varphi) \in F_\Sigma(V)$
- $v \in V,\ \varphi \in F_\Sigma(V) \Rightarrow (\exists v\varphi) \in F_\Sigma(V)$

A (*predicate*) *formula,* or simply a *predicate,* is a formula of $F_\Sigma(V)$ for some signature Σ and for some set V of variables. A *propositional formula* is a predicate formula built on propositional symbols and connectives. Letters $\varphi, \psi, \ldots$ (from the end of the Greek alphabet) stand for *predicate variables,* that is, *meta*-variables ranging over predicate formulae. The expression $\varphi(x, y, \ldots)$ denotes a predicate where variables among $x, y, \ldots$ may occur. In this case, if $t, t', \ldots$ are terms, then $\varphi(t, t', \ldots)$ denotes the formula $\varphi(x, y, \ldots)$ after replacing all the occurrences of $x, y, \ldots$ by $t, t', \ldots$, respectively. A formula where symbols do not belong to a specific signature is considered to be a *predicate schema.* A predicate schema built on variables for propositional symbols is a *propositional schema.* A formula is said to be *atomic* if no connectives or quantifiers occur in it. The set var(t) of variables occurring in a term t can easily be defined by induction. In the formulae $\forall v\varphi$, $\exists v\varphi$ the formula φ is said to be the *scope* of the quantifiers $\forall$ and $\exists$, respectively. In this case the occurrence of variable v is said to be *bound* or *apparent.* An occurrence that is not bound is said to be *free.* A formula which does not contain free occurrences of variables is said to be a *sentence;* F_Σ is the set of sentences on the signature Σ; and T_Σ is the set of Σ-terms without variables, also called *closed terms.* The set *free*(φ) of variables having free occurrences in the formula φ can easily be defined by induction. The notions of *subterm* and *subformula,* the replacement of variables by terms, the replacement of subterms by other terms, and the replacement of subformulae by other formulae can easily be defined by induction. When a term t replaces a variable x in a formula φ, t is assumed to be *free in φ with respect to* (*w.r.t.*) x; that is, no variable of t will be bound after the replacement.

Parentheses are usually omitted, provided that there is no ambiguity, or if any ambiguity which is thereby introduced is irrelevant. Parentheses are also omitted by assuming that unary logical symbols are connected to a formula in the rightmost order, for example, $\forall v \neg \varphi$ stands for $(\forall v(\neg \varphi))$; unary logical symbols precede binary connectives, for example, $\neg \varphi \vee \psi$ stands for $((\neg \varphi) \vee \psi)$; and connectives $\wedge$, $\vee$ tie the constituent formulae more closely than $\rightarrow$, $\leftrightarrow$. Finally, $\forall u \forall v \varphi$ is usually abbreviated to $\forall uv\varphi$ (and $\exists u \exists v \varphi$ to $\exists uv\varphi$).

THE SEMANTICS OF PREDICATE LOGIC

Given a signature Σ such that

1. $\Sigma_0^{\mathrm{fun}} = \{a, b, \ldots\}$
2. $\bigcup_{n>0} \Sigma_n^{\mathrm{fun}} = \{f, g, \ldots\}$
3. $\bigcup_{n>0} \Sigma_n^{\mathrm{rel}} = \{p, q, \ldots\}$

a Σ structure $\mathscr{M}$ defined as

$$\mathscr{M} = \langle A, a^{\mathscr{M}}, b^{\mathscr{M}}, \ldots, f^{\mathscr{M}}, g^{\mathscr{M}}, \ldots, p^{\mathscr{M}}, q^{\mathscr{M}}, \ldots \rangle$$

consists of: (a) a nonempty set A, called the *domain* of $\mathscr{M}$, where some elements $a^{\mathscr{M}}, b^{\mathscr{M}}, \ldots$ belong to A; (b) some operations $f^{\mathscr{M}}, g^{\mathscr{M}}, \ldots$ on A whose arities are those of $f, g, \ldots$ respectively (an n-ary operation on A is a function from the n-sequences of A in A); and (c) some relations $p^{\mathscr{M}}, q^{\mathscr{M}}, \ldots$ on A whose arities are those of $p, q, \ldots$, respectively (an n-ary relation on A is a set of sequences of n elements of A). We identify relations of arity 0 with two elements called *truth values,* denoted by 1, 0 (*true, false*). The domain of $\mathscr{M}$ will be denoted by $|\mathscr{M}|$.

For example, the structure $\mathscr{AR}$ of standard arithmetic has the signature $\{0, 1, +, \times, \leq\}$, where 0, 1 are constants, $+$, $\times$ are binary operation symbols, and $\leq$ is a binary relation symbol. We indicate it by

$$\mathscr{AR} = \langle \omega, 0^{\mathscr{AR}}, 1^{\mathscr{AR}}, +^{\mathscr{AR}}, \times^{\mathscr{AR}}, \leq^{\mathscr{AR}} \rangle$$

where $0^{\mathscr{AR}}, 1^{\mathscr{AR}}, +^{\mathscr{AR}}, \times^{\mathscr{AR}}, \leq^{\mathscr{AR}}$ are the usual meanings associated with the corresponding symbols. In the following the superscripts are dropped; that is, we use ambiguously the same notation for symbols of a signature Σ and for their meanings in a Σ structure. The context will indicate the sense of the notation used.

Let us define set-theoretic semantics for predicate logic. Let Σ be a signature, V a set of variables, and $\mathscr{M}$ a Σ structure. First, we extend (by induction) the interpretation $c \mapsto c^{\mathscr{M}}$, of constants of Σ into the domain of $\mathscr{M}$, to the set T_Σ of closed Σ terms. To this end, it is sufficient to put

$$(f(t_1, \ldots, t_n))^{\mathscr{M}} = f^{\mathscr{M}}(t_1^{\mathscr{M}}, \ldots, t_n^{\mathscr{M}})$$

We will denote by $\mathscr{M}_a$ the structure obtained by $\mathscr{M}$ by adding to it an element $a \in |\mathscr{M}|$ as a new constant such that $a^{\mathscr{M}} = a$ (Σ_a will denote the signature of $\mathscr{M}_a$). Let MOD_Σ be the class of all Σ structures. The following conditions define the *satisfaction* relation $\models$ between a model of MOD_Σ and a sentence of F_Σ. If $\mathscr{M} \models \varphi$, we say that the Σ structure $\mathscr{M}$ *satisfies* the Σ sentence φ (φ *holds* in $\mathscr{M}$). We assume that $Q \in \Sigma_0^{\mathrm{rel}}$, $p \in \Sigma_n^{\mathrm{rel}}$ with $n > 0$, $t_1, t_2, \ldots, t_n \in T_\Sigma$, $\varphi, \psi \in F_\Sigma$, $v \in V$. We will ambiguously use $=$ for the equality symbol of predicate logic, the equality between individuals of a model, and the equality between sets; moreover, $\Leftrightarrow$, $\not\models$, $\neq$ will denote the equivalence between assertions, the nonsatisfaction relation, and the nonequality relation, respectively; a comma between assertions will indicate their conjunction.

Definition 1

$$\begin{aligned}
\mathscr{M} \models Q &\Longleftrightarrow Q^{\mathscr{M}} = 1 \\
\mathscr{M} \models p(t_1, \ldots, t_n) &\Longleftrightarrow \langle t_1^{\mathscr{M}}, \ldots, t^{\mathscr{M}} \rangle \in p^{\mathscr{M}} \\
\mathscr{M} \models (t_1 = t_2) &\Longleftrightarrow t_1^{\mathscr{M}} = t_2^{\mathscr{M}} \\
\mathscr{M} \models \neg\varphi &\Longleftrightarrow \mathscr{M} \not\models \varphi \\
\mathscr{M} \models (\varphi \wedge \psi) &\Longleftrightarrow \mathscr{M} \models \varphi, \mathscr{M} \models \psi \\
\mathscr{M} \not\models (\varphi \vee \psi) &\Longleftrightarrow \mathscr{M} \not\models \varphi, \mathscr{M} \not\models \psi \\
\mathscr{M} \not\models (\varphi \rightarrow \psi) &\Longleftrightarrow \mathscr{M} \models \varphi, \mathscr{M} \not\models \psi \\
\mathscr{M} \models (\varphi \leftrightarrow \psi) &\Longleftrightarrow \mathscr{M} \models (\varphi \rightarrow \psi), \mathscr{M} \models (\psi \rightarrow \varphi) \\
\mathscr{M} \models \forall v\varphi(v) &\Longleftrightarrow \{a | \mathscr{M}_a \models \varphi(a)\} = |\mathscr{M}| \\
\mathscr{M} \models \exists v\varphi(v) &\Longleftrightarrow \{a | \mathscr{M}_a \models \varphi(a)\} \neq \emptyset
\end{aligned}$$

The semantics given for Σ sentences allows us to interpret a predicate $\varphi(x_1, \ldots, x_k)$, with k free variables, into the k-ary relation $(\varphi(x_1, \ldots, x_k))^{\mathscr{M}}$ on the domain of a Σ model $\mathscr{M}$. In fact, let $a_1, \ldots, a_k \in |\mathscr{M}|$, and let $\Sigma_{a_1, \ldots, a_k}$ be the signature Σ extended with the elements $a_1, \ldots, a_k$ as constants. We denote by $\mathscr{M}_{a_1, \ldots, a_k}$ the $\Sigma_{a_1, \ldots, a_k}$ model which extends $\mathscr{M}$, where $a_i^{\mathscr{M}} = a_i$ for $1 \leq i \leq k$; therefore we can define

$$\langle a_1, \ldots, a_k \rangle \in (\varphi(x_1, \ldots, x_k))^{\mathscr{M}} \Longleftrightarrow \mathscr{M}_{a_1, \ldots, a_k} \models \varphi(a_1, \ldots, a_k)$$

We put

$$\mathrm{TH}(\mathscr{M}) = \{\varphi \in F_\Sigma | \mathscr{M} \models \varphi\}$$
$$\mathrm{MOD}_\Sigma(\varphi) = \{\mathscr{M} \in \mathrm{MOD}_\Sigma | \mathscr{M} \models \varphi\}$$

Let $\mathscr{M}_\Delta$ be the model which extends $\mathscr{M}$ with all the elements of its domain $|\mathscr{M}|$ as self-referential constants ($a^{\mathscr{M}} = a$ for all $a \in |\mathscr{M}|$). The set $\mathrm{DIAG}(\mathscr{M})$, called diagram of $\mathscr{M}$, is constituted by the atomic formulae, or the negations of atomic formulae which belong to $\mathrm{TH}(\mathscr{M}_\Delta)$. It is easy to verify that a model $\mathscr{M}$ is completely identified by its diagram. Two Σ models $\mathscr{M}$ and $\mathscr{M}'$ are *elementary equivalent* if $\mathrm{TH}(\mathscr{M}) = \mathrm{TH}(\mathscr{M}')$.

When a sentence holds in a model, we also say that its truth value is *true* (*false* otherwise). Thus, the semantics of connectives can be expressed by the so-called *truth tables,*—that is, by giving the truth value of composite formulae in correspondence to the truth value of the constituent formulae. For example, if 1, 0 stands for *true* and *false,* respectively, we can express truth tables by the following equations:

$$\begin{aligned}
1 &= (\neg 0) = (1 \vee 0) = (0 \vee 1) = (1 \vee 1) = (1 \wedge 1) = (0 \rightarrow 1) \\
&= (1 \rightarrow 1) = (0 \rightarrow 0) = (1 \leftrightarrow 1) = (0 \leftrightarrow 0) \\
0 &= (\neg 1) = (1 \wedge 0) = (0 \wedge 1) = (0 \wedge 0) = (0 \vee 0) = (1 \leftrightarrow 0) \\
&= (0 \leftrightarrow 1)
\end{aligned}$$

A model of a propositional formula is completely determined by the truth value assigned to the propositional symbols—

that is, by a function called the *Boolean valuation* of propositional symbols.

A Σ *theory* is a set Φ of Σ-sentences. A Σ-structure $\mathscr{M}$ is a *model* of a Σ theory Φ if all the sentences of Φ hold in $\mathscr{M}$. The set $\mathrm{MOD}_\Sigma(\Phi)$ is so defined:

$$\mathrm{MOD}_\Sigma(\Phi) = \bigcap_{\varphi\in\Phi} \mathrm{MOD}_\Sigma(\varphi)$$

A Σ theory Φ is *satisfiable* if it has a model, that is, $\mathrm{MOD}_\Sigma(\Phi) \neq \emptyset$; otherwise it is *unsatisfiable*.

A Σ sentence φ is *logically valid* if it is valid in any Σ structure:

$$\mathrm{MOD}_\Sigma(\varphi) = \mathrm{MOD}_\Sigma$$

In this case it represents a *logical law,* and we also write

$$\models \varphi$$

A propositional Σ formula which is logically valid is called a *tautology*. A Σ sentence φ is a *logical consequence* of a Σ theory Φ if any model of Φ is also a model of φ:

$$\mathrm{MOD}_\Sigma(\Phi) \subseteq \mathrm{MOD}_\Sigma(\varphi)$$

In this case we also write

$$\Phi \models \varphi$$

Of course, $\mathrm{MOD}_\Sigma = \mathrm{MOD}_\Sigma(\emptyset)$, and therefore $\models \varphi$ is equivalent to saying that φ is a logical consequence of the empty set (of sentences). The notation introduced above gives rise to two different, though related, meanings for the symbol $\models$: (a) satisfaction of a sentence in a model and (b) logical consequence of a sentence from a theory.

If SR are the axioms of the theory of sexual reproduction, considered in the introduction, then a father's uniqueness is not a logical consequence of SR; that is, $\mathrm{SR} \not\models \forall xyz(P(x, z) \wedge P(y, z) \wedge M(x) \wedge M(y) \rightarrow x = y)$. Indeed, we can define a model $\mathscr{N}$ for SR on the domain ω of natural numbers by putting

$$P^{\mathscr{N}}(n, m) \Longleftrightarrow A^{\mathscr{N}}(n, m) \Longleftrightarrow n > m$$
$$M^{\mathscr{N}} = F^{\mathscr{N}} = \omega$$

Therefore, $\mathscr{N} \models \mathrm{SR}$, but $\mathscr{N} \not\models \forall xyz(P(x, z) \wedge P(y, z) \wedge M(x) \wedge M(y) \rightarrow x = y)$.

Example 1 Important Logical Laws ($* \in \{\wedge, \vee\}$, $Q \in \{\forall, \exists\}$):

1. $(\varphi * \psi) \leftrightarrow (\psi * \varphi)$ (Commutativity)
2. $(\varphi * (\psi * \chi)) \leftrightarrow ((\varphi * \psi) * \chi)$ (Associativity)
3. $(\varphi \wedge (\psi \vee \chi)) \leftrightarrow ((\varphi \wedge \psi) \vee (\varphi \wedge \chi))$ (Distributivity of $\wedge$ w.r.t. $\vee$)
4. $(\varphi \vee (\psi \wedge \chi)) \leftrightarrow ((\varphi \vee \psi) \wedge (\varphi \vee \chi))$ (Distributivity of $\vee$ w.r.t. $\wedge$)
5. $(\varphi \wedge \psi) \vee \varphi \leftrightarrow \varphi$ ($\wedge\vee$ Absorption)
6. $(\varphi \vee \psi) \wedge \varphi \leftrightarrow \varphi$ ($\vee\wedge$ Absorption)
7. $\varphi * \varphi \leftrightarrow \varphi$ (Idempotence)
8. $\neg\neg\varphi \leftrightarrow \varphi$ (Double negation)
9. $\varphi \vee \neg\varphi$ (Excluded middle)
10. $\varphi \wedge \psi \leftrightarrow \neg(\neg\varphi \vee \neg\psi)$ ($\wedge$ De Morgan)
11. $\varphi \vee \psi \leftrightarrow \neg(\neg\varphi \wedge \neg\psi)$ ($\vee$ De Morgan)
12. $\varphi \wedge (\varphi \rightarrow \psi) \rightarrow \psi$ (Detachment)
13. $(\varphi \rightarrow \psi) \leftrightarrow (\neg\psi \rightarrow \neg\varphi)$ (Contraposition)
14. $(\varphi \rightarrow (\psi \rightarrow \chi)) \leftrightarrow (\varphi \wedge \psi \rightarrow \chi)$ (Exportation)
15. $((\varphi \rightarrow \psi) \wedge (\psi \rightarrow \chi)) \rightarrow (\varphi \rightarrow \chi)$ (Syllogism)
16. $((\varphi \vee \psi) \wedge (\neg\varphi \vee \chi)) \rightarrow (\psi \vee \chi)$ (Resolution)
17. $(\varphi \rightarrow \psi) \leftrightarrow \neg\varphi \vee \psi$ (Implication by $\vee$, $\neg$)
18. $(\varphi \leftrightarrow \psi) \leftrightarrow (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$ (Equivalence by $\rightarrow$, $\wedge$)
19. $\neg\forall x\varphi \leftrightarrow \exists x\neg\varphi$ ($\forall$ Negation)
20. $\forall x\varphi(x) \rightarrow \varphi(t)$ ($\forall$ Elimination)
21. $\varphi(t) \rightarrow \exists x\varphi(x)$ ($\exists$ Introduction)
22. $QxQy\varphi \leftrightarrow QyQx\varphi$ (Q Repeating)
23. $Qx\varphi(x) \leftrightarrow Qy\varphi(y)$ (Q Renaming)
24. $(Qx\varphi(x) * \psi) \leftrightarrow Qx(\varphi(x) * \psi)$ (Q Prefixing w.r.t. $*$, $x \notin \mathrm{var}(\psi)$)
25. $(\forall x\varphi \rightarrow \psi) \leftrightarrow \exists x(\varphi \rightarrow \psi)$ ($\forall$ Prefixing w.r.t. $\rightarrow$, $x \notin \mathrm{var}(\psi)$)
26. $(\exists x\varphi \rightarrow \psi) \leftrightarrow \forall x(\varphi \rightarrow \psi)$ ($\exists$ Prefixing w.r.t. $\rightarrow$, $x \notin \mathrm{var}(\psi)$).

If (nonlogical) symbols occurring in a predicate schema can be instantiated by symbols of some signature Σ, then the schema can be interpreted in a Σ-structure as if it were a Σ-formula. In this case it is logically valid if it holds in any Σ-structure, and of course, any instance of it is a logically valid formula. A propositional schema that is logically valid is called a *tautological schema.* Given a set *AX* of sentences or predicate schemata, an *axiomatic theory* of axioms *AX* is the set of all sentences that are logical consequences of *AX*, which in this context are also called *theorems* of the theory.

Example 2 (Peano's Arithmetic). The theory PA has the usual arithmetical signature $\{\leq, +, \times, 0, 1\}$, and consists of the following axioms. PA is an infinite theory because its last axiom is the axiom schema of the induction principle ($\varphi(x)$ ranges on predicates with a free variable x).

- $\forall x\neg(0 = x + 1)$
- $\forall xy(x + 1 = y + 1 \rightarrow x = y)$
- $\forall x(x + 0 = x)$
- $\forall xy(x + (y + 1) = (x + y) + 1)$
- $\forall x(x \times 0 = 0)$
- $\forall xy(x \times (y + 1) = (x \times y) + x)$
- $\forall xy(x \leq y \leftrightarrow \exists z(x + z = y))$
- $(\varphi(0) \wedge \forall x(\varphi(x) \rightarrow \varphi(x + 1))) \rightarrow \forall x\varphi(x)$

Another important arithmetical theory is Raphael Robinson's theory RR (17) which essentially coincides with $\mathrm{DIAG}(\mathscr{AR})$.

A very interesting theory, which we call SS (an acronym for *standard syntax*), is the diagram of the following structure:

$$\mathscr{SQ} = (\omega^*, 0, \lambda, --, \|, \leq, \preceq)$$

where ω^* is the set of finite sequences of natural numbers (0 is the number zero and λ the empty sequence), $--$ is the concatenation of sequences (usually indicated by juxtaposition), || is the length of a sequence (where numbers are sequences of unitary length), $\leq$ is the usual ordering relation on numbers, and $\preceq$ is the substring inclusion. Many basic relations on finite strings can be logically *encoded* by predicates within theories which extend SS (see later).

The implication connective needs to be considered with considerable attention in order to explain its counterintuitive behaviors, known as *paradoxes of material implication.* In fact, according to its formal semantics, an implication $\varphi \rightarrow \psi$ is true when φ is false or ψ is true. Therefore, the proposition "If 1 = 0, then there are infinite prime numbers" is true, although it appears to make no sense. Moreover, given a model $\mathscr{M}$, we can verify that

$$\mathscr{M} \models \varphi \rightarrow \psi \iff (\mathscr{M} \models \varphi \Rightarrow \mathscr{M} \models \psi)$$

but the implication $\Rightarrow$ which appears in this equivalence does not mean that we can *prove* the validity of ψ in $\mathscr{M}$, from the validity of φ in $\mathscr{M}$. In fact, given two sentences φ and ψ, it is easy to verify that, in any model, at least one of the two implications $\varphi \rightarrow \psi$ or $\psi \rightarrow \varphi$ has to be true. Nevertheless, there are models where we cannot prove either the validity of ψ from the validity of φ, or that φ holds because ψ holds. One of the great merits of formal logic is the rigorous definition of two forms of implication: $\rightarrow$ (*material implication*) and $\models$ (*formal implication*). These two forms select two specific meanings of $\Rightarrow$ and allow us to avoid the intrinsic vagueness related to the psychological content of the ordinary *if–then.* Although these implications are adequate for the usual needs of mathematical formalization, the search for other rigorous forms of implication is nevertheless a central issue in constructive and alternative logics.

The difference between material implication and formal implication relies on the two different notions they are based on: truth and proof, respectively. A proposition or its negation has to be true, but, as we will see, there are axiomatic theories where for some sentence φ, neither sentence φ nor sentence $\neg\varphi$ is a logical consequence of the axioms.

PROPOSITIONAL LOGIC

A *literal* is an atomic formula or the negation of an atomic formula. A formula constituted by the disjunction of conjunctions of literals is said to be a *disjunctive normal form.* Likewise, the conjunction of disjunctions of literals is a *conjunctive normal form.*

Proposition 1 (Disjunctive Normal Forms). Any propositional formula is equivalent to some disjunctive normal form.

Proof. Let us set $1\varphi = \varphi$ and $0\varphi = \neg\varphi$. Let $P_1, \ldots, P_k$ be the propositional symbols of φ and suppose that $h^1, \ldots, h^m$ are the Boolean valuations for which the φ results are true ($m > 0$, otherwise the proposition is trivial). Then, according to the semantics of $\wedge$ and $\vee$, we get ($h^i_{P_j}$ is the truth value that h^i assigns to the propositional symbol P_j, for $1 \leq i \leq m$ and $1 \leq j \leq k$):

$$\varphi \leftrightarrow (h^1_{P_1}P_1 \wedge \cdots \wedge h^1_{P_k}P_k) \vee \cdots \vee (h^m_{P_1}P_1 \wedge \cdots \wedge h^m_{P_k}P_k)$$

From De Morgan laws it follows that any propositional formula can be put in the conjunctive normal form.

Tautologies can be determined not only by means of truth tables, but also with a calculus based on Boole's axioms (essentially the logical laws on $\neg, \wedge, \vee$ considered in the example in the previous section). Given a sentence $\varphi(\psi)$, where ψ occurs as a subformula, let us indicate by $\varphi(\chi)$ a sentence where in $\varphi(\psi)$ some occurrences of ψ are replaced by χ. As a simple consequence of the way truth tables are constructed, we have

$$\models \psi \leftrightarrow \chi \Rightarrow \varphi(\psi) \leftrightarrow \varphi(\chi)$$

Boole's calculus is an algebraic calculus in the usual sense, based on the replacement of equivalent subexpressions. Two propositional formulae φ, ψ are *equivalent* according to this calculus when, after changing the formula φ into φ', so that connectives $\rightarrow, \leftrightarrow$ are expressed in terms of $\wedge, \vee, \neg$, it is possible to find a sequence of formulae starting with φ' and ending with ψ, where at any step a subformula α is replaced by a formula β if $\alpha \leftrightarrow \beta$ is a Boole's axiom. For example,

$$\varphi \rightarrow (\psi \rightarrow \theta) \Rightarrow \neg\varphi \vee (\neg\psi \vee \theta) \Rightarrow (\neg\varphi \vee \neg\psi) \vee \theta$$
$$\Rightarrow \neg(\varphi \wedge \psi) \vee \theta \Rightarrow \varphi \wedge \psi \rightarrow \theta$$

Boole's axioms are sufficient to transform any propositional formula in normal disjunctive form. Therefore, since a truth table is uniquely determined by a normal disjunctive form, we get the following proposition.

Proposition 2 (Completeness of Boole's propositional calculus). Two formulae φ, ψ are equivalent according to Boole's calculus iff (if and only if) they have the same truth table.

Propositional logic is strictly connected to the theory of combinatorial circuits in the logical design of computer systems. Relevant aspects in this regard are: the correspondence between propositional formulae and combinatorial circuits, the search for connectives that can express all propositional formulae, and the techniques for minimizing some complexity parameters in circuit design. For example, there are 16 different binary connectives and in general 2^{2^n} n-ary connectives. Moreover, disjunctive (or conjunctive) normal forms and De Morgan laws tell us that any propositional formula is equivalent to a formula where only the connectives $\wedge, \neg$ or only $\vee, \neg$ occur. If we express $\vee$ (or $\wedge$) by means of $\neg \rightarrow$, we obtain an analogous result for the pair of connectives $\rightarrow, \neg$; moreover, if we set P nand $Q = \neg(P \wedge Q)$, then any propositional formula can be equivalently expressed only in terms of the connective nand (likewise for nor defined as P nor $Q = \neg(P \vee Q)$).

In propositional logic we can state one of the most challenging problems in theoretical computer science: *Given a propositional formula, does there exist a deterministic Turing machine* (*see later*) *which can decide whether the formula is satisfiable* (belongs to *SAT*), *in a number of steps that is a polynomial function on the number of occurrences of propositional symbols?* This problem (5) is a sort of *mother* problem, because a great number of combinatorial problems on graphs, trees, strings, automata, and finite sets can be translated into particular instances of it. If this problem were solved, it would lead to the striking conclusion that problems solvable in *polynomial time* by means of nondeterministic algorithms could also be solved in polynomial time in a deterministic way. This would imply the coincidence of the two classes of problems usually indicated by *P* and *NP*.

Let us conclude this section with a fundamental theorem.

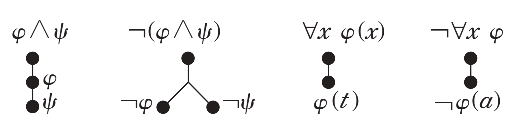
Theorem 1 (Compactness of Propositional Logic). A denumerable theory Φ of propositional formulae is satisfiable iff (if and only if) any finite subset of Φ is satisfiable.

Proof. If Φ is satisfiable, then obviously any subset of Φ is satisfiable. Therefore, let us prove the reverse implication. Assume that any finite subset of Φ is satisfiable. First, let us suppose that Φ has a finite set of propositional symbols. In this case, all the possible Boolean valuations of Φ constitute a finite set $\{f_1, \ldots, f_k\}$. We claim that one of them satisfies Φ. In fact, suppose that no Boolean valuation could satisfy Φ. For any $1 \leq i \leq k$, let φ_{n_i} be a formula of Φ such that f_i does not satisfy φ_{n_i}. By hypothesis, $\{\varphi_{n_1}, \ldots, \varphi_{n_k}\}$ has to be satisfiable (it is finite), but this means that for some j, $1 \leq j \leq k$, f_j satisfies $\{\varphi_{n_1}, \ldots, \varphi_{n_k}\}$; thus in particular, f_j would satisfy φ_{n_j}, against the definition of φ_{n_j}.

When the propositional symbols of Φ are a denumerable set, we consider an enumeration of them ($P_1, P_2, \ldots, P_n, \ldots$, and construct the following labelled tree. We put at the root of the tree the empty valuation of propositional symbols. Then, given a node at level n, we add a son to it and label it with an assignment of a truth value to the propositional symbol $P_{n+1,}$ only if this assignment, together with the assignments associated with the ancestors of the current node, does not make unsatisfiable the first $n + 1$ proposition of Φ. According to our hypothesis, any finite set of Φ is satisfiable. Therefore, at any level, we can assign a truth value to a new propositional symbol. This implies that the tree is infinite, thus for König's lemma, it has an infinite branch. This branch leads to a Boolean valuation that satisfies Φ.

COMPLETENESS AND COMPACTNESS

This section presents a method for establishing whether a sentence φ is a logical consequence of a theory Φ. For the sake of brevity, we use only logical symbols $\{\neg, \wedge, \forall\}$ (the others can easily be expressed in terms of these symbols). The method is mainly based on the following rules expressed by labeled trees and called $\wedge$ rule, $\neg\wedge$ rule, $\forall$ rule, $\neg\forall$ rule:



Given a Σ theory Φ, a Φ tableaux is a tree with nodes labeled by Σ sentences, according to the following (inductive) definition, where T is any Φ tableaux (when no confusion arises, nodes are identified by their labels).

1. Any tree with only one node which is labelled by a sentence of Φ is a Φ tableaux.
2. If we add a leaf to a branch of T and assign to it a label $\varphi \in \Phi$ (Introduction Rule), or a label which already occurs in the branch (Coping Rule), then we get a new Φ tableaux.
3. If a label $\neg\neg\varphi$ occurs in T, it can be replaced by φ (Double Negation Rule).
4. If a leaf of T coincides with the root of one of the $\wedge$, $\neg\wedge$, $\forall$, $\neg\forall$ rules, then we get a new Φ tableaux by replacing in T the leaf with the entire rule (Proper Tableaux rules).
5. If $\varphi(t_1)$ and $t_1 = t_2$ are labels occurring in a branch of T, then we get a new Φ tableaux by adding to the branch $\varphi(t_2)$ (where t_2 replaces t_1) or $t_2 = t_1$ (Replacement Rule and Symmetry Rule).

In the $\forall$ rule, t indicates any Σ term without variables, while in the $\neg\forall$ rule a indicates an individual constant that is uniquely determined by the formula where it is introduced. This constant is also called a *witness* of the formula or even its *Henkin constant.* The uniqueness of this constant implies that the $\neg\forall$ rule can be only applied once (apart from irrelevant repetitions).

A branch of a Φ tableaux T is said to be *closed* when some formula and its negation occur in T; if all the branches of T are closed, then also T is said to be closed.

We say that the formula φ derives from Φ according to the tableaux method, if there exists a $\Phi \cup \{\neg\varphi\}$ tableaux which is closed. In this case we write

$$\Phi \vdash_T \varphi$$

(avoiding the subscript when it is arguable).

Example 3. The following is a closed $\{\forall x \neg(\varphi(x) \wedge \neg\psi(x)) \wedge (\forall x\varphi(x) \wedge \neg\forall x\psi(x))\}$ tableaux. In fact (only Introduction and Coping rules are indicated),

$\forall x\neg(\varphi(x) \wedge \neg\psi(x)) \wedge (\forall x\varphi(x) \wedge \neg\forall x\psi(s))$ Introduction

$\forall x\neg(\varphi(x) \wedge \neg\psi(x))$

$\forall x\varphi(x) \wedge \neg\forall x\psi(x)$

$\forall x\varphi(x)$

$\neg\forall x\psi(x)$

$\neg\psi(a)$

$\forall x\varphi(x)$ Coping

$\varphi(a)$

$\forall x\neg(\varphi(x) \wedge \neg\psi(x))$ Coping

$\neg(\varphi(a) \wedge \neg\psi(a))$

$\bigwedge$

$\neg\varphi(a)$ $\neg\neg\psi(a)$

$\psi(a)$

A theory Φ is (*tableaux*) *consistent* iff no closed Φ tableaux exists. It is easy to understand that if some closed ($\Phi \cup \{\neg\varphi\}$) tableaux exists, then $\Phi \cup \{\neg\varphi\}$ is unsatisfiable, and that $\Phi \cup \{\neg\varphi\}$ is unsatisfiable iff $\Phi \vDash \varphi$:

$$\Phi \vdash \varphi \Rightarrow \mathrm{MOD}_{\Sigma}(\Phi \cup \{\neg\varphi\}) = \emptyset \Leftrightarrow \Phi \models \varphi$$

This implies the inclusion $\vdash \subseteq \vDash$, that is, the *soundness* of the $\vdash$ relation:

$$\Phi \vdash \varphi \Rightarrow \Phi \models \varphi$$

The reverse implication is a consequence of the completeness theorem which we will show, after introducing the concept of *systematic* tableaux. Intuitively, this is a Φ tableaux where all formulae of Φ occur in any nonclosed branch and where, if a tableaux rule can be applied, then it is applied in all the possible ways.

Given a countable, consistent Σ theory Φ, the systematic Φ tableaux is defined in the following way.

Let us consider an enumeration $\{\varphi_i | i \in \omega\}$ of all Σ sentences where every formula occurs infinitely many times (it is easy to define such an enumeration). We proceed by a succession of steps indexed by natural numbers. We start with a tree constituted by only one node labeled by $\forall x(x = x)$. At any step $i \in \omega$ consider φ_i and a nonclosed branch B of the tableaux obtained so far, then apply the following procedure.

If the sentence φ_i does not belong to Φ, or is not yet in B, then we do not alter B. Otherwise, we add φ_i as a leaf of B. If some rule can be applied to φ_i, we apply this rule exhaustively. That is, if φ is $\forall x \psi(x)$, we add $\psi(t)$ to B for every closed term t occurring in B; and if φ is $t_1 = t_2$, we add $t_2 = t_1$, and $\chi(t_2)$ to B, for every $\chi(t_1)$ which occurs in B. We repeat all this for any other nonclosed branch, and then we go to the next step with φ_{i+1}.

The systematic Φ tableaux described here is nonclosed, otherwise Φ would be inconsistent. The labels of a non closed branch constitute a theory, usually called a *Hintikka set.*

Theorem 2 (Completeness). Any countable consistent Σ theory Φ is satisfiable.

Proof. We can apply the previous construction and get the systematic Φ tableaux, which is nonclosed. Therefore, by König's lemma, this tableaux has a nonclosed branch H. We prove that H determines a model for its labels and thus for Φ. First, let us assume that in Φ neither the equality symbol nor function symbols occur.

Let us define a Σ model where the domain is the set of all constants occurring in H (each constant interpreted into itself) and where any n-ary relation symbol p is interpreted as the relation $[\![p]\!]$ such that $\langle t_1, \ldots, t_k\rangle \in [\![p]\!]$ if the atomic formula $p(t_1, \ldots, t_k)$ occurs in H. In this model if $\varphi \in H$, then φ is true. We verify this statement by induction on the number of occurrences of symbols $\wedge$, $\forall$.

An atomic formula $\varphi \in H$ is true by virtue of the given interpretation. If $\varphi = \neg\psi$ and ψ is atomic, then ψ cannot belong to H because in this case H would be closed; therefore ψ is false, and thus φ is true.

If $\varphi \wedge \psi \in H$, then since H is a branch of the systematic tableaux, $\varphi \in H$ and $\psi \in H$, so by induction hypothesis both φ and ψ are true; thus $\varphi \wedge \psi$ is also true.

If $\neg(\varphi \wedge \psi) \in H$, then, again by *systematicity,* $\neg\varphi \in H$ or $\neg\psi \in H$; that is, by induction hypothesis at least one of these two formulae is true, and thus $\neg(\varphi \wedge \psi)$ is true as well.

If $\forall x\varphi(x) \in H$, by systematicity, for every constant a of H, $\varphi(a) \in H$; therefore by induction hypothesis, these formulae are true; but these constants are the individuals of our domain, and therefore $\forall x\varphi(x)$ is true.

If $\neg\forall x\varphi(x) \in H$, by systematicity, there exists a constant a such that $\neg\varphi(a) \in H$; but by induction, $\varphi(a)$ is not true, and therefore $\forall x\varphi(x)$ is not true, that is, $\neg\forall x\varphi(x)$ is true.

If equality symbols or function symbols also occur in Φ, the previous model has to be modified in the following manner. Let us consider an equivalence relation $\equiv$ such that $t \equiv t'$ iff $(t = t') \in H$. Then, we put as domain the set $T_\Sigma/\equiv$ of equivalence classes of closed terms occurring in H, and we interpret (a) any constant c into its equivalence class $[c]$ w.r.t. $\equiv$, that is, $[\![c]\!] = [c]$, (b) any n-ary function symbol f into the function $[\![f]\!]$ such that, for every $[\![t_1]\!], \ldots, [\![t_n]\!]$ we have

$$[\![f]\!]([\![t_1]\!], \ldots, [\![t_n]\!]) = [f(t_1, \ldots, t_n)]$$

and (c) any n-ary relation symbol p into the n-ary relation $[\![p]\!]$ such that, for every $[\![t_1]\!], \ldots, [\![t_n]\!]$ we have

$$\langle [\![t_1]\!], \ldots, [\![t_n]\!]\rangle \in [\![p]\!] \Leftrightarrow p(t_1, \ldots, t_n) \in H$$

In this manner the previous proof can be extended to the more general case.

The completeness theorem can be generalized: Any consistent theory of any cardinality is satisfiable.

Completeness is equivalent to the inclusion $\models\ \subseteq\ \vdash$. In fact,

$$\Phi \nvdash \varphi \Rightarrow \mathrm{MOD}_\Sigma(\Phi \cup \{\neg\varphi\}) \neq \emptyset$$
(by definition of $\vdash$ and completeness)

$$\mathrm{MOD}_\Sigma(\Phi \cup \{\neg\varphi\}) \neq \emptyset \Rightarrow \Phi \nvDash \varphi \qquad \text{(by definition of } \models)$$

$$\Phi \models \varphi \Rightarrow \Phi \vdash \varphi \qquad \text{(by transitivity and contraposition)}$$

In conclusion, we can assert the following propositions.

Proposition 3 (Equivalence between $\models$ and $\vdash$)

$$\Phi \models \varphi \Longleftrightarrow \Phi \vdash \varphi$$

Proof. The inclusion $\vdash\ \subseteq\ \models$ holds for soundness, while the inclusion $\models\ \subseteq\ \vdash$ holds for completeness.

The completeness theorem implies two other important properties: *finiteness* and *compactness.*

Proposition 4 (Finiteness)

$$\Phi \models \varphi \Longleftrightarrow \Delta \models \varphi$$

for some finite subset Δ of Φ.

Proof. The verse $\Rightarrow$ is trivial. By the equivalence theorem

$$\Phi \models \varphi \Longleftrightarrow \Phi \vdash \varphi$$

Moreover, $\Phi \vdash \varphi$ if there exists a closed $\Phi \cup \{\neg\varphi\}$-tableaux; obviously, such a tableaux is finite, and thus only a finite number of sentences of Φ occur in it, that is, φ is a logical consequence of them.

The compactness property owes its name to a topological space naturally definable on the set MOD_Σ, which is compact (in the standard topological sense). The following is the usual formulation of compactness in predicate logic.

Proposition 5 (Compactness). A theory Φ is satisfiable iff every finite subset of Φ is satisfiable.

Proof. If every finite subset of Φ is satisfiable, due to the soundness property, no closed Φ tableaux can exist; therefore

Φ is consistent, and thus for the completeness theorem it is satisfiable.

Compactness implies important consequences for predicate logic. Consider the theory DEN = $\{\psi_n | n \in \omega\}$ of denumerability, where for every natural number n, ψ_n is a sentence which asserts the existence of at least n different individuals (it can be constructed with $\neg$, $=$, $\exists$, and n variables). By using this theory we can see that any theory Φ with models of any finite cardinality has a model with infinite cardinality. It is sufficient to consider $\Phi \cup$ DEN. Clearly, any finite subset of this new theory has a model; therefore, by compactness, $\Phi \cup$ DEN has a model, which is a model of Φ, but it is necessarily infinite because it is a model of DEN too. As a direct consequence, no first-order theory can be satisfied by all and only finite models. With similar reasonings we could show that no predicate theory of well orderings can exist. Indeed, there are well orderings with descending chains of any length; therefore any theory for these models, by compactness, should also satisfy the existence of an infinite descending chain, which is exactly the opposite of the well ordering definition.

LÖWENHEIM–SKOLEM THEOREMS

The essence of Löwenheim–Skolem theorems is in the relationship between first-order theories and cardinalities. According to these theorems, any countable theory with an infinite model also has a denumerable model (Löwenheim–Skolem Downward Theorem), and it even has models of any infinite cardinality (Löwenheim–Skolem Upward Theorem). This means that predicate logic is not good at discerning the cardinalities of structures. This produces *pathological* effects (usual referred to as *Skolem's paradox*)—for example, denumerable models of first-order theory of real numbers, but, at the same time, models of Peano arithmetic with more than denumerable domains. Technically, the Löwenheim–Skolem Downward Theorem is a simple consequence of the systematic tableaux construction used in the proof of the completeness theorem. In fact, let Φ be a countable Σ theory with an infinite model; thus the theory $\Phi \cup$ DEN (DEN being the theory of denumerability) has a model, because Φ has an infinite model. If we consider a model obtained by a systematic $\Phi \cup$ DEN tableaux, then it has at most a denumerable set of individuals, because Φ is a countable Σ theory, and thus the set T_Σ of closed terms is denumerable. However, these individuals must be a denumerable set because this model has to satisfy DEN.

The second (Upward) Löwenheim–Skolem theorem is a simple consequence of compactness. In fact, let Φ be a countable Σ theory with an infinite model, then we can find for Φ a model of any infinite cardinality α. To this end, we extend the signature Σ with a set C of constants of cardinality α and with the set of sentences $\{\neg(c = c') | c, c' \in C\}$, which also has cardinality α. By compactness this theory is satisfiable, and thus it is easy to extract from it a Σ model for Φ with cardinality α.

Löwenheim–Skolem theorems can be generalized: Any theory of infinite cardinality α which has an infinite model also has a model of cardinality β, for any $\beta \geq \alpha$.

Let us extend Peano arithmetic with a constant c greater than any natural number (i.e., such that formulae $c > n$ are added to the theory). By compactness, this theory has a model that is also a model of PA, nevertheless, in this model we have a *nonstandard* number inasmuch as it expresses a sort of infinite quantity. This kind of phenomenon, strictly connected to Skolem's paradox, not only can be considered a limitative result of the expressibility of predicate logic, but is also the basis for the powerful application of formal logic in the analysis of infinite and infinitesimal quantities. In fact, nonstandard analysis, founded by Robinson, who elaborated on this idea, gives a rigorous treatment of *actual* (versus *potential*) infinitely big and infinitely small (real) numbers, in terms of nonstandard elements. On this basis, the Cauchy–Weierstrass *ϵ–δ theory of convergence* was reformulated, which gave rise to new important research fields.

SKOLEM FORMS AND HERBRAND EXPANSIONS

Every predicate can be put in an equivalent *prenex normal form:*

$$Q_1x_1Q_2x_2 \ldots Q_kx_k \quad \mu$$

where $Q_1\, Q_2 \,.\,.\,.\, Q_k$ is a sequence of quantifiers called *prefix,* and μ is a formula without quantifiers, called the *matrix* of the form.

This is a simple consequence of the logical laws on quantifiers considered in the section entitled "the Semantics of Predicate Logic" (essentially $\forall$ negation, Q renaming, and Q prefixing).

It is easy to see that a Σ formula $\forall x \exists y \varphi(x, y)$, in prenex form, has a model iff the formula $\forall x \varphi(x, f(x))$, built in the signature that extends Σ with the function symbol f, has a model. This result is generalized by the following proposition.

Proposition 6. For every prenex Σ sentence φ where $\exists$ occurs, we can effectively find a sentence φ' with no occurrence of $\exists$ which is built in a signature $\Sigma' \supset \Sigma$ with new constants and/or function symbols, and such that φ' is satisfiable iff φ is satisfiable.

The formula φ' of the previous proposition is said to be the *Skolem form* of φ. The construction of φ' is the following: If no universal quantification precedes the quantification $\exists x$ in the prefix of φ, then $\exists x$ is removed from the prefix and a constant c is replaced at every occurrence of x in the matrix of φ. If this quantification is after the sequence $Q_1x1 \,.\,.\,.\, Q_jx_j$, then $\exists x$ is removed and the term $f(x_1, \,.\,.\,., x_j)$ is replaced at every occurrence of x in the matrix of φ. The constant c and the function symbol f, called a *Skolem function* symbol, depend uniquely on the formula φ and on the existential quantifications that are eliminated when they are introduced.

Of course, if we apply prenex form and Skolem form transformations to all the sentences of a Σ theory Φ, we can find a Σ' theory $\Phi'(\Sigma \subset \Sigma')$ where any formula is in Skolem normal form and which is *co-satisfiable* with Φ; that is, it has a model iff Φ has a model. In this case we say that the set $T_{\Sigma'}$ of closed Σ' terms is the *Herbrand Universe* of Φ'. From Skolem forms and the Herbrand universe, we get a propositional theory Ψ which is co-satisfiable with Φ:

$$\Psi = \{\varphi(t_1, \ldots, t_k) \mid \forall x_1, \ldots, x_k \varphi(x_1, \ldots, x_k) \in \Phi', t_1, \ldots, t_k \in T_{\Sigma'}\}$$

Ψ is said to be the *Herbrand expansion* of Φ'. From its definition and from the systematic tableaux construction it follows that Ψ has a model iff Φ has a model.

In Skolem forms the universal quantification is not usually indicated; that is, all the variables are implicitly assumed to be universally quantified.

Let us consider a Skolem form with a matrix in normal conjunctive form. We may collect all the literals of its disjunctions into sets of literals called *clauses.* Thus, if a clause is considered to be true when it contains some true literal, then the initial Skolem form is equivalent to a set of clauses. This *clause representation* can obviously be extended to an entire theory of Skolem forms.

LOGICAL CALCULI

A logical calculus is an effective method which defines a deduction relation $\vdash$ between a Σ theory Φ and a Σ sentence φ. The first logical calculi for predicate logic were developed by Frege and Hilbert. They can be classified as *axiomatic* calculi, because they derive logically valid formulae (logical theorems), starting from some axioms and applying some *inference rules.* Concise formulations of such calculi have a few axioms and *modus ponens* as the only inference rule. For example, if we do not consider equality axioms, a possible set of axiom schemata ($\varphi, \psi, \chi \in F_\Sigma(V)$, $t \in T_\Sigma(V)$) is (13)

$$\varphi \to (\psi \to \varphi)$$
$$((\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi)))$$
$$((\neg\varphi \to \psi) \to ((\neg\varphi \to \neg\psi) \to \varphi))$$
$$\forall x(\varphi \to \psi) \to (\forall x\varphi \to \forall x\psi)$$
$$\varphi \to \forall x\varphi$$
$$\forall x\varphi(x) \to \varphi(t)$$

Moreover, axioms are (logical) theorems, and if φ and $\varphi \to \psi$ are theorems, then ψ is a theorem too (modus ponens). The Frege–Hilbert deduction relation $\vdash_{\mathrm{FH}}$ holds between a theory Φ and a sentence φ if $\varphi_1 \wedge \ldots \wedge \varphi_k \to \varphi$ is a theorem of this calculus for some $\varphi_1, \ldots \varphi_k \in \Phi$.

Although such calculi are very simple and elegant, it is very difficult to construct complex deductions within them. A very significant result, discovered independently by Herbrand and Tarski (1930) and known as the *Deduction Theorem,* states that

$$\Phi \cup \{\psi\} \vdash \varphi \Rightarrow \Phi \vdash \psi \to \phi$$

This implication was the starting point for some important research, begun by Gentzen, that led to a novel idea of formal deduction, strictly connected to basic mechanisms of mathematical proofs: *natural deduction.* According to this approach, any logical operator (connective or quantifier) determines rules which express its deductive meaning. For example, if sentences φ and ψ are derived, then also the sentence $\varphi \wedge \psi$ can be deduced ($\wedge$ *introduction rule*), while from $\varphi \wedge \psi$ both φ and ψ can be deduced ($\wedge$ *elimination rules*). A natural deduction of a sentence φ from a theory Φ is a sequence of formulae ending with φ and such that every formula in the sequence belongs to Φ or derives, according to some inference rule, from some preceding formulae.

The *sequent* calculi (19), also due to Gentzen, are strictly related to natural deduction. In these calculi, inference rules are directly stated in terms of the deduction relation. For example, a sequent style formulation of $\wedge$ introduction could be the following:

$$\Phi \vdash \varphi, \Phi \vdash \psi \Rightarrow \Phi \vdash \varphi \wedge \psi$$

A central issue in sequent calculi, related to very significant results in proof theory, is their *cut-freeness.* This can be paraphrased by saying that any proof obtained by means of a lemma can be also constructed directly:

$$\Phi \vdash \varphi, \Phi \cup \{\varphi\} \vdash \psi \Rightarrow \Phi \vdash \psi$$

Sequent calculi can be viewed as a sort of reversed tableaux method, that is, a direct formulation of tableaux rules that are indirect or confutative, because they try to establish the unsatisfiability of theories.

Frege–Hilbert calculi, natural deduction calculi, and sequent calculi are sound and complete calculi, and therefore their deduction relations are equivalent to the logical consequence relation.

Another deduction method, due essentially to Skolem and Herbrand, and confutative like the tableaux method, is based on the compactness of propositional logic and on the co-satisfiability between a theory Φ and the Herbrand expansion of its Skolem forms. Suppose we want verify if $\Phi \vDash \varphi$. In order to get a positive conclusion, it is sufficient to prove the unsatisfiability of the theory $\Phi \cup \{\neg\varphi\}$. Thus, let us consider a Herbrand expansion Φ' of this theory, which is co-satisfiable with it. By (propositional) compactness, Φ' is unsatisfiable iff some finite subset of Φ' is such. Our task can then be reduced to enumerating all the finite subsets of Φ' and to testing their satisfiability (e.g., by truth tables).

A more efficient method based on the same idea is the so-called *Resolution Method.* In this case, in order to prove that $\Phi \vDash \varphi$, we consider Skolem forms of $\Phi \cup \{\neg\varphi\}$ and put it in clause form—that is, as a set C of clauses, where each clause is a set of literals. We try then to get the unsatisfiability of C by adding new clauses to C, by means of two rules (only one rule in concise formulations), until we get the empty clause (an absurdity). Let σ be a substitution—that is, a function from variables into terms (possibly with variables). The two basic rules on which the Resolution Method relies are as follows:

- *Substitution:* Given a clause $\Gamma \in C$, add to C the clause Γ_σ obtained by replacing in Γ every occurrence of the variable x with the term $\sigma(x)$.
- *Resolution:* If $\Gamma \cup \{\varphi\} \in C$ and $\Psi \cup \{\neg\varphi\} \in C$, add to C their *resolvent* $\Gamma \cup \Psi$.

The completeness of this method is a consequence of (a) the co-satisfiability between a theory and any clause representation of it and (b) the completeness of resolution rule for propositional logic.

The Resolution Method is the basic tool of *logic programming* (5), where clauses represent particular implications called *Horn formulae* (which have only one positive literal). In this case there are specific resolution strategies which provide a particular efficiency.

The following simple proposition allows us to relate different deduction relations.

Proposition 7. Let $\vdash_1$ and $\vdash_2$ be two deduction relations. If $\vdash_1 \subseteq \vdash_2$ and $\vdash_1$ is complete, then also $\vdash_2$ is complete. (In fact, $\Phi \models \varphi \Rightarrow \Phi \vdash_1 \varphi \Rightarrow \Phi \vdash_2 \varphi$.)

LOGICAL REPRESENTABILITY

Given a signature Σ, along with a Σ model $\mathscr{M}$ with domain D, a set $A \subseteq D$ is representable within $\mathscr{M}$ if there exists a Σ formula $\varphi(x)$ such that for any elements $a \in D$ the following equivalence holds:

$$a \in A \Longleftrightarrow \mathscr{M} \models \varphi(a)$$

($\varphi(a)$ is a formula in the signature Σ_a which extends Σ with the elements a as a self-referential constant). In this case we say that φ represents logically A in $\mathscr{M}$.

Likewise, a subset A of T_Σ is representable within a Σ theory Φ if there exists a formula $\varphi(x)$ such that for any closed Σ term t we have

$$t \in A \Longleftrightarrow \Phi \models \varphi(t)$$

In this case we say that φ represents A within Φ.

A set $A \subseteq T_\Sigma$ is axiomatically represented by a finite set of axioms AX, within the theory Φ, if A is represented within $\Phi \cup$ AX (if Φ is empty we say simply that A is axiomatically represented within AX).

A relation, viewed as a particular set, can be logically represented (in models or theories) by a formula with a number of free variables equal to its arity.

Usual arithmetical sets and relations are representable within the arithmetical model $\mathscr{AR}$, or within the theories *PA*, *RR*, or *SS*. Likewise, arithmetical and syntactical relations can be naturally represented in the model $\mathscr{SEQ}$ and in the theory SS.

For example, we can represent in SS the sum of natural numbers; in fact,

$$\mathscr{AR} \models n + m = k \Longleftrightarrow \text{SS} \models \exists uw(|u| = n \wedge |w| = m \wedge |uw| = k)$$

With a more complex formula we could show that also the product on natural numbers can be represented in SS. Many concepts in the field of formal languages theory can be illustrated in terms of logical representability, producing interesting perspectives in the logical analysis of complex syntactical systems.

Let us give a very simple example. The language of sequences of zeros followed by the same number of ones—that is, $\{0^n1^n | n \in \omega\}$—is represented in SS by the formula $\varphi(w)$:

$$\begin{aligned} \exists uv(w = uv \wedge |u| = |v| \\ \wedge \forall xyz(u = xyz \wedge |y| = |0| \rightarrow y = 0) \\ \wedge \forall xyz(v = xyz \wedge |y| = |0| \rightarrow y = 1)) \end{aligned}$$

The same language is represented by the formula $L(x)$ within SS plus the axioms:

$$L(\lambda)$$
$$\forall x(L(x) \rightarrow L(0x1))$$

The usual symbolic devices for defining formal languages (grammars, automata, rewriting systems) can easily be *translated* into logical theories. In the following we limit ourselves to giving an important example of logical representability, which helps us to understand the logical nature of computability.

We assume that the reader is familiar with the notion of a *Turing Machine* (Turing's original paper also appears in Ref. 3). The theory TT, which we will now present, is a logical description of Turing machines. Its signature consists of (a) four constant symbols a_0, q_0, $>$, $<$, for the blank symbol, the initial state, the right move, and the left move; (b) a unary function symbol s for generating, from a_0 and q_0, other symbols and other states; (c) a binary function symbol for concatenation (indicated by juxtaposition); and (d) seven unary predicate symbols I, T, N, O, F, S, C such that: $I(\alpha q \beta)$ means that the machine is in the state q, its control unit is reading the first symbol of β, and $\alpha\beta$ fills a portion of the tape outside which there are only blank symbols; $T(\alpha)$ means that α fills a portion of the tape in a final configuration (when a final state has been reached), and outside α there are only blank symbols; $O(\alpha)$ means that α is an output, that is, the longest string in the tape of a final configuration such that α begins and ends with symbols that are different from a_0; $F(q)$ means that q is a final state; $S(a)$ means that a is an input symbol (a symbol different from a_0), and $C(a)$ means that a is a character, that is, an input or a blank symbol. Finally, in the signature of TT we have a binary predicate symbol R for expressing the instructions of Turing machines: $R(qx, q'y >)$ means that when in the state q the symbol x is read, then the state q' is reached, the symbol x is replaced by y, and the control moves to the next symbol to the right (likewise for the left move if $<$ occurs instead of $>$).

The following axioms (where universal quantification is tacitly assumed) allow us to derive all the possible initial configurations, the way the instructions change configurations, and the way an output string is recovered:

1. $C(a_0) \wedge I(q_0)$
2. $I(q_0w) \wedge S(x) \rightarrow I(q_0wx)$
3. $(uv)w = u(vw)$
4. $S(x) \rightarrow C(x)$
5. $R(qx, q'y >) \wedge I(wqxz) \rightarrow I(wyq'z)$
6. $R(qx, q'y >) \wedge I(wqx) \rightarrow I(wyq'a_0)$
7. $R(qx, q'y <) \wedge I(wuqxz) \wedge C(u) \rightarrow I(wq'uyz)$
8. $R(qx, q'y <) \wedge I(qxw) \rightarrow I(q'a_0yw)$
9. $I(wqv) \wedge F(q) \rightarrow T(wv)$
10. $T(a_0w) \rightarrow T(w)$
11. $T(wa_0) \rightarrow T(w)$
12. $T(xwy) \wedge S(x) \wedge S(y) \rightarrow O(xwy)$.

In order to simulate a particular Turing machine M, we must add other specific axioms to *TT*, say *AX*(M), which express the input symbols of M, the instructions of M, and the final states of M (M is *deterministic* if $R(qx, t), R(qx, t') \in AX(M) \Rightarrow t = t'$; otherwise M is *non-deterministic*).

It is important to note that these axioms are Horn formulae; therefore Horn formulae can be considered as the *computable part* of predicate logic.

Assume a fixed, but arbitrary, finite alphabet A, included in $\{a_0, s(a_0), s(s(a_0)), \ldots\}$ (or, without loss of generality, a finite subset of ω). A language on A is a subset of the set A^* of strings of A; moreover, the language $L(M)$ generated by a Turing machine M is constituted by all the strings which are the output of M in correspondence to all possible input strings.

A language L is said to be *recursively enumerable* (or *semi-decidable*) if $L = L(M)$ for some Turing machine M. *Church's thesis* (1936) can be formulated by saying that any language generated by some algorithmic procedure is a recursively enumerable language (3).

A language L is said to be *decidable* if L and its complementary $\overline{L} = A^*/L$ are both recursively enumerable. This is in fact the same as having an effective method for deciding whether a string of A^* belongs to L. Let us enumerate (without repetitions) $A^* = \{\alpha_1, \alpha_2, \ldots\}$ and the set $\mathrm{TM}(A) = \{M_1, M_2, \ldots\}$ of all the Turing machines with A as the alphabet of its input symbols (it is equivalent to enumerating $\{\mathrm{AX}(M) | M \in \mathrm{TM}(A)\}$. A famous example of recursively enumerable language is $K = \{\alpha_i | \alpha_i \in L(M_i)\}$. This language is not decidable because its complementary $\overline{K}$ is not recursively enumerable. We could prove the nonrecursive enumerability of $\overline{K}$ by means of the same diagonal argument of Cantor's theorem (on the nondenumerability of real numbers) or of Russell's paradox.

The notions of recursive enumerability and of decidability can be naturally extended to theories, if we consider their sentences as strings of suitable alphabets.

The following proposition is a direct consequence of the construction of the theory TT; it tells us that any recursively enumerable language can be axiomatically represented in the theory TT.

Proposition 8. For every $\alpha \in A^*$, $TT \cup AX(M) \vDash O(\alpha) \Leftrightarrow \alpha \in L(M)$.

UNDECIDABILITY AND INCOMPLETENESS

The main limitation of predicate logic is a direct consequence of its capability to represent recursively enumerable sets.

Proposition 9 (Church). The logical consequence $\vDash$ of predicate logic is not a decidable relation.

Proof. It is sufficient to find a theory with a finite number of axioms which is not decidable. Let us consider the theory TT $\cup$ AX($\mathscr{M}_K$), where $L(\mathscr{M}_K) = K$; if this theory were decidable, then also K would be decidable (K is representable in this theory); but this is absurd because we know that $\overline{K}$ is not decidable.

A Σ theory Φ is *complete* if $\varphi \in \Phi$ or $\neg\varphi \in \Phi$ for any $\varphi \in F_\Sigma$.

Any countable satisfiable Σ theory Φ can be always extended to a satisfiable complete Σ theory. The existence of this extension, called *Lindenbaum Completion,* is due to the fact that, for any Σ sentence φ, one of the following two theories must be satisfiable: $\Phi \cup \{\neg\varphi\}$ or $\Phi \cup \{\varphi\}$ otherwise $\Phi \vDash \varphi$ and $\Phi \vDash \neg\varphi$, and therefore Φ would be unsatisfiable. We then enumerate all the Σ sentences and consider a denumerable chain $\bigcup_{n\in\omega} \Phi_n$, where $\Phi_0 = \Phi$, and $\Phi_{n+1} = \Phi_n \cup \{\varphi_n\}$ or $\Phi_{n+1} = \Phi_n \cup \{\neg\varphi_n\}$, depending on which of the two theories is satisfiable. By compactness, the resulting completion is satisfiable, and it is obviously a complete theory.

For any Σ model $\mathscr{M}$ the theory TH($\mathscr{M}$) constituted by all Σ sentences that hold in $\mathscr{M}$ is of course a complete theory. A theory is *axiomatizable* when it is an axiomatic theory with a recursively enumerable set of axioms. Given the *computable* nature of the logical calculi, an axiomatizable theory is recursively enumerable. In fact, due to the finiteness of predicate logic, when the axioms AX are recursively enumerated, we can recursively enumerate all closed AX tableaux, that is, all the theorems of the theory.

A very important property of axiomatizable and complete Σ theories is their decidability. In fact we can generate all Σ sentences that are theorems of a complete axiomatizable theory Φ. Given a Σ sentence φ, if it is generated we know that $\varphi \in \Phi$, if $\neg\varphi$ is generated we know that $\varphi \notin \Phi$; by the completeness of Φ, one of these two alternatives must happen, and therefore Φ is decidable.

We say that a Σ theory Φ is *Gödelian* if any recursively enumerable language included in T_Σ can be represented within Φ. Of course, a Gödelian theory cannot be decidable.

The theory TT is Gödelian. It can be shown that TH($\mathscr{AR}$), PA, RR, and SS are Gödelian.

No theory can exist that is axiomatizable, complete, and Gödelian. In fact, if a theory Φ is axiomatizable and complete, it is also decidable; therefore it cannot represent recursively enumerable sets, that is, it cannot be Gödelian. As a simple consequence of incompatibility among axiomatizability, completeness, and *Gödelianity,* we get these famous incompleteness results:

Proposition 10. The theories TT, PA, RR, and SS are incomplete.

Proposition 11. The theory TH($\mathscr{AR}$) is not axiomatizable.

An axiomatizable theory Φ in the signature Σ_{AR} of $\mathscr{AR}$ is *arithmetically sound* when its theorems are true in the model $\mathscr{AR}$.

Proposition 12 (Gödel's First Incompleteness Theorem). For any axiomatizable arithmetically sound Σ_{AR} theory Φ there exists a Σ_{AR} sentence that is true in $\mathscr{AR}$ but is not a theorem of Φ.

Gödel's epoch-making paper of 1931 appears also in Refs. 3 and in 4; a general study of *incompleteness* proofs is developed in Ref. 17. The celebrated *Gödel's Second Incompleteness Theorem,* in its abstract form (17), is related to axiomatic systems $\mathscr{S}$ that are self-referential inasmuch as they own a *provability predicate* P such that $\mathscr{S} \vDash \varphi \Rightarrow \mathscr{S} \vDash P(\overline{\varphi})$ (where $\overline{\varphi}$ is a term uniquely associated with the sentence φ); moreover, $\mathscr{S} \vDash P(\overline{\varphi}) \to P(\overline{P(\overline{\varphi})})$, and $\mathscr{S} \vDash P(\overline{\varphi \to \psi}) \to (P(\overline{\varphi}) \to P(\overline{\psi}))$. In this case, under very reasonable hypotheses, fulfilled by PA, RR, SS, or TT, such theories cannot prove their own consistency; that is, they cannot deduce, for some sentence φ, the sentence $\neg P(\overline{\varphi \wedge \neg\varphi})$.

BIBLIOGRAPHY

1. A. Church, *Introduction to Mathematical Logic I,* Princeton, NJ: Princeton Univ. Press, 1956.

2. G. H. Moore, Beyond first-order logic: The historical interplay between mathematical logic and axiomatic set theory, *Hist. Philos. Logic,* **1**: 95–137, 1980.
3. M. Davis, *The Undecidable,* New York: Raven Press, 1965.
4. J. van Heijenoort, *From Frege to Gödels, A Source Book in Mathematical Logic, 1879–1931,* Cambridge, MA: Harvard Univ. Press, 1967.
5. J. van Leeuwen, *Handbook of Theoretical Computer Science,* Amsterdam: Elsevier, 2 vols., 1990.
6. D. M. Gabbay and F. Guenthner (eds.), *Handbook of Philosophical Logic,* 4 vols., Dordrecht: Reidel, 1983.
7. S. Abramsky, D. M. Gabbay, and T. S. Maibaum (eds.), *Handbook of Logic in Computer Science,* 4 vols., Oxford: Clarendon Press, 1992–1995.
8. J. van Benthem and A. ter Meulen (eds.), *Handbook of Logic and Language,* Amsterdam: Elsevier Science, 1997.
9. J. Y. Girard, Y. Lafont, and L. Regnier (eds.), *Advances in Linear Logic,* London Math. Soc. Lect. Note Ser. 222, Cambridge, MA: Cambridge Univ. Press, 1995.
10. M. Fitting, *First-Order Logic and Automated Theorem Proving,* New York: Springer-Verlag, 1996.

10a. E. Börger, *Computability, Complexity, Logic,* Amsterdam: North-Holland, 1989.

11. D. M. Gabbay, C. G. Hogger, and J. A. Robinson (eds.), *Handbook of Logic in Artificial Intelligence and in Logic Programming,* 4 vols., Oxford: Clarendon Press, 1993–1995.
12. J. Barwise (ed.), *Handbook of Mathematical Logic,* Amsterdam: North-Holland, 1977.
13. J. L. Bell and M. Machover, *A Course in Mathematical Logic,* Amsterdam: North-Holland, 1977.
14. J. N. Crossley et al., *What Is Mathematical Logic?,* London: Oxford Univ. Press, 1972.
15. D. Van Dalen, *Logic and Structure,* Berlin: Springer-Verlag, 1981.
16. W. Hodges, Elementary Predicate Logic, in D. Gabbay and F. Guenthner (eds.), *Handbook of Philosophical Logic,* vol. I: *Elements of Classical Logic,* Dordrecht: Reidel, 1983, pp. 1–131.
17. C. Smoryński, *Logic Number Theory,* Berlin: Springer-Verlag, 1991.
18. R. M. Smullyan, *Gödels's Incompleteness Theorems,* New York: Oxford Univ. Press, 1992.
19. R. M. Smullyan, *First-Order Logic,* New York: Dover Publications, Inc., 1968.

VINCENZO MANCA
University of Pisa