Computability and Logic

Fifth Edition

GEORGE S. BOOLOS

JOHN P. BURGESS

Princeton University

RICHARD C. JEFFREY

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE UNIVERSITY PRESS
Cambridge, New York, Melbourne, Madrid, Cape Town, Singapore, São Paulo

Cambridge University Press
The Edinburgh Building, Cambridge CB2 8RU, UK
Published in the United States of America by Cambridge University Press, New York

www.cambridge.org
Information on this title: www.cambridge.org/9780521877527

First published in print format 2007

ISBN-13 978-0-511-36668-0 eBook (EBL)
ISBN-10 0-511-36668-X eBook (EBL)

ISBN-13 978-0-521-87752-7 hardback
ISBN-10 0-521-87752-0 hardback

ISBN-13 978-0-521-70146-4 paperback
ISBN-10 0-521-70146-5 paperback

10

A Précis of First-Order Logic: Semantics

This chapter continues the summary of background material on logic needed for later chapters. Section 10.1 studies the notions of truth *and* satisfaction, *and section 10.2 the so-called* metalogical *notions of validity, implication or consequence, and (un)satisfiability.*

10.1 Semantics

Let us now turn from the official definitions of syntactical notions in the preceding chapter to the official definitions of semantic notions. The task must be to introduce the same level of precision and rigor into the definition of truth of a sentence in or on or under an interpretation as we have introduced into the notion of sentence itself. The definition we present is a version or variant of the *Tarski definition* of what it is for a sentence F to be true in an interpretation $\mathcal{M}$, written $\mathcal{M} \models F$. (The double turnstile $\models$ may be pronounced 'makes true'.)

The first step is to define truth for atomic sentences. The official definition will be given first for the case where identity and function symbols are absent, then for the case where they are present. (If sentence letters were admitted, they would be atomic sentences, and specifying which of them are true and which not would be part of specifying an interpretation; but, as we have said, we generally are not going to admit them.) Where identity and function symbols are absent, so that every atomic sentence has the form $R(t_1, \ldots, t_n)$ for some nonlogical predicate R and constants t_i, the definition is straightforward:

(1a) $$\mathcal{M} \models R(t_1, \ldots, t_n) \quad \text{if and only if} \quad R^{\mathcal{M}}\big(t_1^{\mathcal{M}}, \ldots, t_n^{\mathcal{M}}\big).$$

The atomic sentence is true in the interpretation just in case the relation that the predicate is interpreted as denoting holds of the individuals that the constants are interpreted as denoting.

When identity is present, there is another kind of atomic sentence for which a definition of truth must be given:

(1b) $$\mathcal{M} \models =(t_1, t_2) \quad \text{if and only if} \quad t_1^{\mathcal{M}} = t_2^{\mathcal{M}}.$$

The atomic sentence is true in the interpretation just in case the individuals the constants are interpreted as denoting are the same.

When function symbols are present, we need a preliminary definition of the denotation $t^{\mathcal{M}}$ of a closed term t of a language L under an interpretation $\mathcal{M}$. Clauses (1a) and (1b) then apply, where the t_i may be any closed terms, and not just constants. For an atomic closed term, that is, for a constant c, specifying the denotation $c^{\mathcal{M}}$ of c is part of what is meant by specifying an interpretation. For more complex terms, we proceed as follows. If f is an n-place function symbol, then specifying the denotation $f^{\mathcal{M}}$ is again part of what is meant by specifying an interpretation. Suppose the denotations $t_1^{\mathcal{M}}, \ldots, t_n^{\mathcal{M}}$ of terms $t_1, \ldots, t_n$ have been defined. Then we define the denotation of the complex term $f(t_1, \ldots, t_n)$ to be the value of the function $f^{\mathcal{M}}$ that is the denotation of f applied to the individuals $t_1^{\mathcal{M}}, \ldots, t_n^{\mathcal{M}}$ that are the denotations of $t_1, \ldots, t_n$ as arguments:

(1c) $$(f(t_1, \ldots, t_n))^{\mathcal{M}} = f^{\mathcal{M}}\big(t_1^{\mathcal{M}}, \ldots, t_n^{\mathcal{M}}\big).$$

Since every term is built up from constants by applying function symbols a finite number of times, these specifications determine the denotation of every term.

So, for example, in the standard interpretation of the language of arithmetic, since $\mathbf{0}$ denotes the number zero and $'$ denotes the successor function, according to (1c) $\mathbf{0}'$ denotes the value obtained on applying the successor function to zero as argument, which is to say the number one, a fact we have anticipated in abbreviating $\mathbf{0}'$ as $\mathbf{1}$. Likewise, the denotation of $\mathbf{0}''$ is the value obtained on applying the successor function to the denotation of $\mathbf{0}'$, namely one, as argument, and this value is of course the number two, again a fact we have been anticipating in abbreviating $\mathbf{0}''$ as $\mathbf{2}$. Similarly, the denotation of $\mathbf{0}'''$ is three, as is, for instance, the denotation of $\mathbf{0}' + \mathbf{0}''$. No surprises here.

According to (1b), continuing the example, since the denotations of $\mathbf{0}'''$ or $\mathbf{3}$ and of $\mathbf{0}' + \mathbf{0}''$ or $\mathbf{1} + \mathbf{2}$ are the same, $\mathbf{0}''' = \mathbf{0}' + \mathbf{0}''$ or $\mathbf{3} = \mathbf{1} + \mathbf{2}$ is true, while by contrast $\mathbf{0}'' = \mathbf{0}' + \mathbf{0}''$ or $\mathbf{2} = \mathbf{1} + \mathbf{2}$ is false. Again no surprises. According to (1a), further continuing the example, since the denotation of $<$ is the strict less-than relation, and the denotations of $\mathbf{0}'''$ or $\mathbf{3}$ and of $\mathbf{0}' + \mathbf{0}''$ or $\mathbf{1} + \mathbf{2}$ are both three, the atomic sentence $\mathbf{0}''' < \mathbf{0}' + \mathbf{0}''$ or $\mathbf{3} < \mathbf{1} + \mathbf{2}$ is false, while by contrast $\mathbf{0}'' < \mathbf{0}' + \mathbf{0}''$ is true. Yet again, no surprises.

There is only one candidate for what the definition should be in each of the cases of negation and of the two junctions:

(2a)	$\mathcal{M} \models \sim F$	if and only if	not $\mathcal{M} \models F$
(2b)	$\mathcal{M} \models (F \,\&\, G)$	if and only if	$\mathcal{M} \models F$ and $\mathcal{M} \models G$
(2c)	$\mathcal{M} \models (F \vee G)$	if and only if	$\mathcal{M} \models F$ or $\mathcal{M} \models G$.

So, for example, in the standard interpretation of the language of arithmetic, since $\mathbf{0} = \mathbf{0}$ and $\mathbf{0} < \mathbf{0}'$ are true while $\mathbf{0} < \mathbf{0}$ is false, we have that $(\mathbf{0} = \mathbf{0} \vee \mathbf{0} < \mathbf{0}')$ is true, $(\mathbf{0} < \mathbf{0} \,\&\, \mathbf{0} = \mathbf{0})$ is false, $(\mathbf{0} < \mathbf{0} \,\&\, (\mathbf{0} = \mathbf{0} \vee \mathbf{0} < \mathbf{0}'))$ is false, and $((\mathbf{0} < \mathbf{0} \,\&\, \mathbf{0} = \mathbf{0}) \vee \mathbf{0} < \mathbf{0}')$ is true. Still no surprises.

One consequence of (2a)–(2c) worth mentioning is that $(F \,\&\, G)$ is true if and only if $\sim(\sim F \vee \sim G)$ is true, and $(F \vee G)$ is true if and only if $\sim(\sim F \,\&\, \sim G)$ is true. We could therefore if we wished drop one of the pair &, $\vee$ from the official language, and treat it as an unofficial abbreviation (for an expression involving $\sim$ and the other of the pair) on a par with $\rightarrow$ and $\leftrightarrow$.

The only slight subtlety in the business arises at the level of quantification. Here is a simple, tempting, and *wrong* approach to defining truth for the case of quantification, called the *substitutional* approach:

$$\begin{array}{lll} \mathcal{M} \models \forall x\, F(x) & \text{if and only if} & \text{for every closed term } t,\ \mathcal{M} \models F(t) \\ \mathcal{M} \models \exists x\, F(x) & \text{if and only if} & \text{for some closed term } t,\ \mathcal{M} \models F(t). \end{array}$$

In other words, under this definition a universal quantification is true if and only if every substitution instance is true, and an existential quantification is true if and only if some substitution instance is true. This definition in general produces results not in agreement with intuition, unless it happens that every individual in the domain of the interpretation is denoted by some term of the language. If the domain of the interpretation is enumerable, we could always *expand* the language to add more constants and extend the interpretation so that each individual in the domain is the denotation of one of them. But we cannot do this when the domain is nonenumerable. (At least we cannot do so while continuing to insist that a language is supposed to involve only a finite or enumerable set of symbols. Of course, to allow a 'language' with a nonenumerable set of symbols would involve a considerable stretching of the concept. We will briefly consider this extended concept of 'language' in a later chapter, but for the moment we set it aside.)

10.1 Example. Consider the language L^* of arithmetic and three different interpretations of it: first, the standard interpretation $\mathcal{N}^*$; second, the alternative interpretation $\mathcal{Q}$ we considered earlier, with domain the nonnegative rational numbers; third, the similar alternative interpretation $\mathcal{R}$ with domain the nonnegative real numbers. Now in fact the substitutional approach gives the intuitively correct results for $\mathcal{N}^*$ in all cases. Not so, however, for the other two interpretations. For, all closed terms in the language have the same denotation in all three interpretations, and from this it follows that all closed terms denote natural numbers. And from this it follows that $t + t = \mathbf{1}$ is false for all closed terms t, since there is no natural number that, added to itself, yields one. So on the substitutional approach, $\exists x(x + x = \mathbf{1})$ would come out false on all three interpretations. But intuitively 'there is something (in the domain) that added to itself yields one' is false only on the standard interpretation $\mathcal{N}^*$, and true on the rational and real interpretations $\mathcal{Q}$ and $\mathcal{R}$.

We could try to fix this by adding more constants to the language, so that there is one denoting each nonnegative rational number. If this were done, then on the rational and real interpretations, $\mathbf{1/2} + \mathbf{1/2} = \mathbf{1}$ would come out true, and hence $\exists x(x + x = \mathbf{1})$ would come out true using the substitutional approach, and this particular example of a problem with the substitutional approach would be fixed. Indeed, the substitutional approach would then give the intuitively correct results for $\mathcal{Q}$ in all cases. Not so, however, for $\mathcal{R}$. For, all terms in the language would denote rational numbers, and from this it would follow that $t \cdot t = \mathbf{2}$ is false for all terms t, since there is no rational number that, multiplied by itself, yields two. So on the substitutional approach, $\exists x(x \cdot x = \mathbf{2})$ would come out false. But intuitively, though 'there is something (in the domain) that multiplied by itself yields two' is false on the rational interpretation, it is true on the real interpretation. We could try to fix this by adding yet more terms to the language, but by Cantor's theorem there are too many real numbers to add a term for each of them while keeping the language enumerable.

The *right* definition for the case of quantification has to be a little more indirect. In defining when $\mathcal{M} \models \forall x F(x)$ we do not attempt to extend the given language L so as to provide constants for every individual in the domain of the interpretation at once. In general, that cannot be done without making the language nonenumerable. However, if we consider any particular individual in the domain, we *could* extend the language and interpretation to give *just it* a name, and what we do in defining when $\mathcal{M} \models \forall x F(x)$ is to consider *all possible* extensions of the language and interpretation by adding just one new constant and assigning it a denotation.

Let us say that in the interpretation $\mathcal{M}$ the individual m *satisfies* $F(x)$, and write $\mathcal{M} \models F[m]$, to mean 'if we considered the extended language $L \cup \{c\}$ obtained by adding a new constant c in to our given language L, and if among all the extensions of our given interpretation $\mathcal{M}$ to an interpretation of this extended language we considered the one $\mathcal{M}^c_m$ that assigns c the denotation m, then $F(c)$ would be true':

(3*) $$\mathcal{M} \models F[m] \quad \text{if and only if} \quad \mathcal{M}^c_m \models F(c).$$

(For definiteness, let us say the constant to be added should be the first constant not in L in our fixed enumeration of the stock of constants.)

For example, if $F(x)$ is $x \cdot x = \mathbf{2}$, then on the real interpretation of the language of arithmetic $\sqrt{2}$ satisfies $F(x)$, because if we extended the language by adding a constant c and extended the interpretation by taking c to denote $\sqrt{2}$, then $c \cdot c = \mathbf{2}$ would be true, because the real number denoted by c would be one that, multiplied by itself, yields two. This definition of satisfaction can be extended to formulas with more than one free variable. For instance, if $F(x, y, z)$ is $x \cdot y = z$, then $\sqrt{2}$, $\sqrt{3}$, $\sqrt{6}$ satisfy $F(x, y, z)$, because if we added c, d, e denoting them, $c \cdot d = e$ would be true.

Here, then, is the *right* definition, called the *objectual* approach:

(3a) $$\mathcal{M} \models \forall x F(x) \quad \text{if and only if} \quad \text{for every } m \text{ in the domain, } \mathcal{M} \models F[m]$$

(3b) $$\mathcal{M} \models \exists x F(x) \quad \text{if and only if} \quad \text{for some } m \text{ in the domain, } \mathcal{M} \models F[m].$$

So $\mathcal{R} \models \exists x F(x)$ under the above definitions, in agreement with intuition, even though there is no term t in the actual language such that $\mathcal{R} \models F(t)$, because $\mathcal{R} \models F[\sqrt{2}]$.

One immediate implication of the above definitions worth mentioning is that $\forall x F$ turns out to be true just in case $\sim\exists x \sim F$ is true, and $\exists x F$ turns out to be true just in case $\sim\forall x \sim F$ is true, so it would be possible to drop one of the pair $\forall$, $\exists$ from the official language, and treat it as an unofficial abbreviation.

The method of proof by induction on complexity can be used to prove semantic as well as syntactic results. The following result can serve as a warm-up for more substantial proofs later, and provides an occasion to review the definition of truth clause by clause.

10.2 Proposition (Extensionality lemma).

(a) Whether a sentence A is true depends only on the domain and denotations of the nonlogical symbols in A.
(b) Whether a formula $F(x)$ is satisfied by an element m of the domain depends only on the domain, the denotations of the nonlogical symbols in F, and the element m.
(c) Whether a sentence $F(t)$ is true depends only on the domain, the denotations of the nonlogical symbols in $F(x)$, and the denotation of the closed term t.

Here (a), for instance, means that the truth value of A does not depend on what the nonlogical symbols in A *themselves* are, but only on what their *denotations* are, and does not depend on the denotations of nonlogical symbols *not* in A. (So a more formal statement would be: If we start with a sentence A and interpretation I, and change A to B by changing zero or more nonlogical symbols to others of the same kind, and change I to $\mathcal{J}$, then the truth value of B in $\mathcal{J}$ will be the same as the truth value of A in I provided $\mathcal{J}$ has the same domain as I, $\mathcal{J}$ assigns each unchanged nonlogical symbol the same denotation I did, and whenever a nonlogical symbol S is changed to T, then $\mathcal{J}$ assigns to T the same denotation I assigned to S. The proof, as will be seen, is hardly longer than this formal statement!)

Proof: In proving (a) we consider first the case where function symbols are absent, so the only closed terms are constants, and proceed by induction on complexity. By the atomic clause in the definition of truth, the truth value of an atomic sentence depends only on the denotation of the predicate in it (which in the case of the identity predicate cannot be changed) and the denotations of the constants in it. For a negation $\sim B$, assuming as induction hypothesis that (a) holds for B, then (a) holds for $\sim B$ as well, since by the negation clause in the definition of truth, the truth value of $\sim B$ depends only on the truth value of B. The cases of disjunction and conjunction are similar.

For a universal quantification $\forall x\, B(x)$, assuming as induction hypothesis that (a) holds for sentences of form $B(c)$, then (b) holds for $B(x)$, for the following reason. By the definition of satisfaction, whether m satisfies $B(x)$ depends on the truth value of $B(c)$ where c is a constant not in $B(x)$ that is assigned denotation m. [For definiteness, we specified which constant was to be used, but the assumption of (a) for sentences of form $B(c)$ implies that it does not matter what constant is used, so long as it is assigned denotation m.] By the induction hypothesis, the truth value of $B(c)$ depends only on the domain and the denotations of the nonlogical symbols in $B(c)$, which is to say, the denotations of the nonlogical symbols in $B(x)$ and the element m that is the denotation of the nonlogical symbol c, just as asserted by (b) for $B(x)$. This preliminary observation made, (a) for $\forall x\, B(x)$ follows at once, since by the universal quantification clause in the definition of truth, the truth value of $\forall x\, B(x)$ depends only on the domain and which of its elements satisfy $B(x)$. The case of existential quantification is the same.

If function symbols are present, we must as a preliminary establish by induction on complexity of terms that the denotation of a term depends only on the denotations of the nonlogical symbols occurring in it. This is trivial in the case of a constant. If it is true for terms $t_1, \ldots, t_n$, then it is true for the term $f(t_1, \ldots, t_n)$, since the definition of denotation of term mentions only the denotation of the nonlogical symbol f and the denotations of the terms $t_1, \ldots, t_n$. This preliminary observation made, (a) for atomic sentences follows, since by the atomic clause in the definition of truth, the truth value of an atomic sentence depends only on the denotation of its predicate and the denotations of its terms. The nonatomic cases in the proof require no change.

We have proved (b) in the course of proving (a). Having (b), the proof of (c) reduces to showing that whether a sentence $F(t)$ is true depends only on whether the element m denoted by t satisfies $F(x)$, which by the definition of satisfaction is to say, on whether $F(c)$ is true, where c is a constant having the same denotation m as t. The proof that $F(c)$ and $F(t)$ have the same truth value if c and t have the same denotation is relegated to the problems at the end of the chapter.

It is also extensionality (specifically, part (c) of Proposition 10.2) that justifies our earlier passing remarks to the effect that the substitutional approach to defining quantification does work *when every element of the domain is the denotation of some closed term.* If for some closed term t the sentence $B(t)$ is true, then letting m be the denotation of t, it follows by extensionality that m satisfies $B(x)$, and hence $\exists x\, B(x)$ is true; and conversely, if $\exists x\, B(x)$ is true, then some m satisfies $B(x)$, and *assuming that every element of the domain is the denotation of some closed term*, then some term t denotes m, and by extensionality, $B(t)$ is true. Thus under the indicated assumption, $\exists x B(x)$ is true if and only if for some term t, $B(t)$ is true, and similarly $\forall x\, B(x)$ is true if and only if for every term t, $B(t)$ is true.

Similarly, if every element of the domain is the denotation of a closed term *of some special kind* then $\exists x\, B(x)$ is true if and only if $B(t)$ is true for some closed term t that is *of that special kind*. In particular, for the standard interpretation $\mathcal{N}^*$ of the language of arithmetic L^*, where every element of the domain is the denotation of one of the terms $\mathbf{0}$, $\mathbf{1}$, $\mathbf{2}$, $\ldots$, we have

$$\mathcal{N}^* \models \forall x\, F(x) \quad \text{if and only if} \quad \text{for every natural number } m, \mathcal{N}^* \models F(\mathbf{m})$$
$$\mathcal{N}^* \models \exists x\, F(x) \quad \text{if and only if} \quad \text{for some natural number } m, \mathcal{N}^* \models F(\mathbf{m})$$

where $\mathbf{m}$ is the numeral for the number m (that is, the term consisting of the cipher $\mathbf{0}$ followed by m copies of the accent $'$).

10.2 Metalogical Notions

Now that rigorous definitions of formula and sentence, and of satisfaction and truth, have been given, we can proceed to the definitions of the main notions of logical theory. A set of sentences Γ *implies* or has as a *consequence* the sentence D if there is no interpretation that makes every sentence in Γ true, but makes D false. This is the same as saying that every interpretation that makes every sentence in Γ true makes D true. (Or almost the same. Actually, if D contains a nonlogical symbol not in Γ, an interpretation might make Γ true but assign no denotation to this symbol and therefore no truth value to D. But in such a case, however the denotation is extended to assign a denotation to any such symbols and therewith a truth value to D, Γ will still be true by the extensionality lemma, so D cannot be false and must be true. To avoid fuss over such points, in future we tacitly understand 'every interpretation' to mean 'every interpretation that assigns denotations to all the nonlogical symbols in whatever sentences we are considering'.) We use 'makes every sentence in Γ true' and 'makes Γ true' interchangeably, and likewise 'the sentences in the set Γ imply D' and 'Γ implies D'. When Γ contains but a single sentence C (in symbols, when $\Gamma = \{C\}$), we use 'Γ implies D' and 'C implies D' interchangeably. Let us give a few examples. There are more in the problems at the end of the chapter (and many, many, many more in introductory textbooks).

10.3 Example. Some implication principles

(a) $\sim\sim B$ implies B.

(b) B implies $(B \vee C)$ and C implies $(B \vee C)$.

(c) $\sim(B \vee C)$ implies $\sim B$ and $\sim C$.

(d) $B(t)$ implies $\exists x B(x)$.

(e) $\sim\exists x B(x)$ implies $\sim B(t)$.

(f) $s = t$ and $B(s)$ imply $B(t)$.

Proofs: For (a), by the negation clause in the definition of truth, in any interpretation, if $\sim\sim B$ is true, then $\sim B$ must be false, and B must be true. For (b), by the disjunction clause in the definition of truth, in any interpretation, if B is true, then $(B \vee C)$ is true; similarly for C. For (c), by what we have just shown, any interpretation that does *not* make $(B \vee C)$ true can*not* make B true; hence any intepretation that makes $\sim(B \vee C)$ true makes $\sim B$ true; and similarly for $\sim C$. For (d), in any interpretation, by the extensionality lemma $B(t)$ is true if and only if the element m of the domain that is denoted by t satisfies $B(x)$, in which case $\exists x B(x)$ is true. As for (e), it follows from what we have just shown much as (c) follows from (b). For (f), by the identity clause in the definition of truth, in any interpretation, if $s = t$ is true, then s and t denote the same element of the domain. Then by the extensionality lemma $B(s)$ is true if and only if $B(t)$ is true.

There are two more important notions to go with implication or consequence. A sentence D is *valid* if no interpretation makes D false. In this case, *a fortiori* no interpretation makes Γ true and D false; Γ implies D for *any* Γ. Conversely, if every Γ implies D, then since for every interpretation there is a set of sentences Γ it makes true, no interpretation can make D false, and D is valid. A set of sentences Γ is *unsatisfiable* if no interpretation makes Γ true (and is *satisfiable* if some interpretation does). In this case, *a fortiori* no interpretation makes Γ true and D false, so Γ implies D for any D. Conversely, if Γ implies every D, then since for every interpretation there is a sentence it makes false, there can be no interpretation making Γ true, and Γ is unsatisfiable.

Notions such as consequence, unsatisfiability, and validity are often called 'metalogical' in contrast to the notions of negation, conjunction, disjunction, and universal and existential quantification, which are simply called 'logical'. Terminology aside, the difference is that there are symbols $\sim$, &, $\vee$, $\forall$, $\exists$ in our formal language (the 'object language') for negation and the rest, whereas words like 'consequence' only appear in the unformalized prose, the mathematical English, in which we talk *about* the formal language (the 'metalanguage').

Just as for implication or consequence, so for validity and for unsatisfiability and satisfiability, there are innumerable little principles that follow directly from the definitions. For instance: if a set is satisfiable, then so is every subset (since an interpretation making every sentence in the set true will make every sentence in the subset true); no set containing both a sentence and its negation is satisfiable (since no interpretation makes them both true); and so on. The plain assertions of Example 10.3 can each be elaborated into fancier versions about validity and (un)satisfiability, as we next illustrate in the case of 10.3(a).

10.4 Example. Variations on a theme

(a) $\sim\sim B$ implies B.

(b) If Γ implies $\sim\sim B$, then Γ implies B.

(c) If B implies D, then $\sim\sim B$ implies D.

(d) If $\Gamma \cup \{B\}$ implies D, then $\Gamma \cup \{\sim\sim B\}$ implies D.

(e) If $\sim\sim B$ is valid, then B is valid.

(f) If $\Gamma \cup \{B\}$ is unsatisfiable, then $\Gamma \cup \{\sim\sim B\}$ is unsatisfiable.

(g) If $\Gamma \cup \{\sim\sim B\}$ is satisfiable, then $\Gamma \cup \{B\}$ is satisfiable.

Proof: (a) is a restatement of 10.3(a). For (b), we are given that every interpretation that makes Γ true makes $\sim\sim B$ true, and want to show that any interpretation that makes Γ true makes B true. But this is immediate from (a), which says that any interpretation that makes $\sim\sim B$ true makes B true. For (c), we are given that any interpretation that makes B true makes D true, and want to show that any interpretation that makes $\sim\sim B$ true makes D true. But again, this is immediate from the fact that any interpretation that makes $\sim\sim B$ true makes B true. In (d), $\Gamma \cup \{B\}$ denotes the result of adding B to Γ. The proof in this case is a combination of the proofs of (b) and (c). For (e), we are given that every interpretation makes $\sim\sim B$ true, and want to show that every interpretation makes B true, while for (f), we are given that no interpretation makes Γ and B true, and want to show that no interpretation makes Γ and $\sim\sim B$ true. But again both are immediate from (a), that is, from the fact that every interpretation that makes $\sim\sim B$ true makes B true. Finally, (g) is immediate from (f).

We could play the same game with any of 10.3(b)–10.3(f). Some results exist only in the fancy versions, so to speak.

10.5 Example. Some satisfiability principles

(a) If $\Gamma \cup \{(A \vee B)\}$ is satisfiable, then either $\Gamma \cup \{A\}$ is satisfiable, or $\Gamma \cup \{B\}$ is satisfiable.

(b) If $\Gamma \cup \{\exists x B(x)\}$ is satisfiable, then for any constant c not occurring in Γ or $\exists x B(x)$, $\Gamma \cup \{B(c)\}$ is satisfiable.

(c) If Γ is satisfiable, then $\Gamma \cup \{t = t\}$ is satisfiable.

Proof: For (a), we are given that some interpretation makes Γ and $A \vee B$ true, and want to show that some interpretation makes Γ and A true, or some makes Γ and B true. In fact, the *same* interpretation that makes Γ and $A \vee B$ true either makes A true or makes B true, by the disjunction clause in the definition of truth. For (b), we are given that some interpretation makes Γ and $\exists x B(x)$ true, and want to show that some interpretation makes Γ and $B(c)$ true, assuming c does not occur in Γ or $\exists x B(x)$. Well, since $\exists x B(x)$ is true, some element m of the domain satisfies $B(x)$. And since c does not occur in Γ or $\exists x B(x)$, we can change the interpretation to make m the denotation of c, without changing the denotations of any nonlogical symbols in Γ or $\exists x B(x)$, and so by extensionality not changing their truth values. But then Γ is still true, and since m satisfies $B(x)$, $B(c)$ is also true. For (c), we are given that some interpretation makes Γ true and want to show that some interpretation makes Γ and $t = t$ true. But *any* interpretation makes $t = t$ true, so long as it assigns a denotation to each nonlogical symbol in t, and if our given interpretation does not, it at least assigns a denotation to every nonlogical symbol in t that occurs in Γ, and if we extend it to assign denotations to any other nonlogical symbols in t, by extensionality Γ will still be true, and now $t = t$ will be true also.

There is one more important metalogical notion: two sentences are *equivalent over an interpretation* $\mathcal{M}$ if they have the same truth value. Two formulas $F(x)$ and $G(x)$ are equivalent over $\mathcal{M}$ if, taking a constant c occurring in neither, the sentences $F(c)$ and $G(c)$ are equivalent over every interpretation $\mathcal{M}_m^c$ obtained by extending $\mathcal{M}$ to provide some denotation m for c. Two sentences are (*logically*) *equivalent* if they are equivalent over all interpretations. Two formulas $F(x)$ and $G(x)$ are (*logically*) equivalent if, taking a constant c occurring in neither, the sentences $F(c)$ and $G(c)$ are (logically) equivalent. A little thought shows that formulas are (logically) equivalent if they are equivalent over every interpretation. The definitions may be extended to formulas with more than one free variable. We leave the development of the basic properties of equivalence entirely to the problems.

Before closing this chapter and bringing on those problems, a remark will be in order. The method of induction on complexity we have used in this chapter and the preceding to prove such unexciting results as the parenthesis and extensionality lemmas will eventually be used to prove some less obvious and more interesting results. Much of the interest of such results about formal languages depends on their being applicable to ordinary language. We have been concerned here mainly with how to read sentences of our formal language in ordinary language, and much less with writing sentences of ordinary language in our formal language, so we need to say a word about the latter topic.

In later chapters of this book there will be many examples of writing assertions from *number theory*, the branch of mathematics concerned with the natural numbers, as first-order sentences in the language of arithmetic. But the full scope of what can be done with first-order languages will not be apparent from these examples, or this book, alone. Works on set theory give examples of writing assertions from other branches of mathematics as first-order sentences in a *language of set theory*, and make it plausible that in virtually *all* branches of mathematics, what we want to say can be said in a first-order language. Works on logic at the introductory level contain a wealth of examples of how to say what we want to say in a first-order language from outside mathematics (as in our genealogical examples).

But this cannot *always* be done outside of mathematics, and some of our results *do not* apply unrestrictedly to ordinary language. A case in point is unique readability. In ordinary language, ambiguous sentences of the type 'A and B or C' are perfectly possible. Of course, though *possible*, they are not *desirable*: the sentence ought to be rewritten to indicate whether 'A, and either B or C' or 'Either A and B, or C' is meant. A more serious case in point is extensionality. In ordinary language it is *not* always the case that one expression can be changed to another denoting the same thing without altering truth values. To give the classic example, Sir Walter Scott was the author of the historical novel *Waverley*, but there was a time when this fact was unknown, since the work was originally published anonymously. At that time, 'It is known that Scott is Scott' was as always true, but 'It is known that the author of *Waverley* is Scott' was false, even though 'Scott' and 'the author of *Waverly*' had the same denotation.

To put the matter another way, writing s for 'Scott' and t for 'the author of *Waverley*', and writing $A(x)$ for 'x is Scott' and $\Box$ for 'it is known that', what we have just said is that $s = t$ and $\Box A(s)$ may be true without $\Box A(t)$ being true, in contrast to one of our examples above, according to which, in our formal languages, $s = t$ and $B(s)$ always imply $B(t)$. There is no contradiction with our example, of course, since our formal languages do not contain any operator like $\Box$; but for precisely this reason, *not* everything that can be expressed in ordinary language can be expressed in our formal languages. There is a separate branch of logic, called *modal logic*, devoted to operators like $\Box$, and we are eventually going to get a peek at a corner of this branch of logic, though only in the last chapter of the book.

Problems

10.1 Complete the proof of the extensionality lemma (Proposition 10.2) by showing that if c is a constant and t a closed term having the same denotation, then substituting t for c in a sentence does not change the truth value of the sentence.

10.2 Show that $\exists y \forall x R(x, y)$ implies $\forall x \exists y R(x, y)$.

10.3 Show that $\forall x \exists y F(x, y)$ does not imply $\exists y \forall x F(x, y)$.

10.4 Show that:

(a) If the sentence E is implied by the set of sentences Δ and every sentence D in Δ is implied by the set of sentences Γ, then E is implied by Γ.

(b) If the sentence E is implied by the set of sentences $\Gamma \cup \Delta$ and every sentence D in Δ is implied by the set of sentences Γ, then E is implied by Γ.

10.5 Let $\varnothing$ be the empty set of sentences, and let $\perp$ be any sentence that is not true on any interpretation. Show that:

(a) A sentence D is valid if and only if D is a consequence of $\varnothing$.

(b) A set of sentences Γ is unsatisfiable if and only if $\perp$ is a consequence of Γ.

10.6 Show that:

(a) $\{C_1, \ldots, C_m\}$ is unsatisfiable if and only if $\sim C_1 \vee \cdots \vee \sim C_m$ is valid.

(b) D is a consequence of $\{C_1, \ldots, C_m\}$ if and only if $\sim C_1 \vee \cdots \vee \sim C_m \vee D$ is valid.

(c) D is a consequence of $\{C_1, \ldots, C_m\}$ if and only if $\{C_1, \ldots, C_m, \sim D\}$ is unsatisfiable.

(d) D is valid if and only if $\sim D$ is unsatisfiable.

10.7 Show that $B(t)$ and $\exists x(x = t \,\&\, B(x))$ are logically equivalent.

10.8 Show that:

(a) $(B \,\&\, C)$ implies B and implies C.

(b) $\sim B$ implies $\sim(B \,\&\, C)$, and $\sim C$ implies $\sim(B \,\&\, C)$.

(c) $\forall x\, B(x)$ implies $B(t)$.

(d) $\sim B(t)$ implies $\sim\forall x\, B(x)$.

10.9 Show that:

(a) If $\Gamma \cup \{\sim(B \,\&\, C)\}$ is satisfiable, then either $\Gamma \cup \{\sim B\}$ is satisfiable or $\Gamma \cup \{\sim C\}$ is satisfiable.

(b) If $\Gamma \cup \{\sim\forall x\, B(x)\}$ is satisfiable, then for any constant c not occurring in Γ or $\forall x\, B(x)$, $\Gamma \cup \{\sim B(c)\}$ is satisfiable.

10.10 Show that the following hold for equivalence over any interpretation (and hence for logical equivalence), for any sentences (and hence for any formulas):

(a) F is equivalent to F.

(b) If F is equivalent to G, then G is equivalent to F.

(c) If F is equivalent to G and G is equivalent to H, then F is equivalent to H.

(d) If F and G are equivalent, then $\sim F$ and $\sim G$ are equivalent.

(e) If F_1 and G_1 are equivalent, and F_2 and G_2 are equivalent, then $F_1 \,\&\, F_2$ and $G_1 \,\&\, G_2$ are equivalent, and similarly for $\vee$.

(f) If c does not occur in $F(x)$ or $G(x)$, and $F(c)$ and $G(c)$ are equivalent, then $\forall x\, F(x)$ and $\forall x\, G(x)$ are equivalent, and similarly for $\exists$.

10.11 (*Substitution of equivalents.*) Show that the following hold for equivalence over any interpretation (and hence for logical equivalence):

(a) If sentence G results from sentence F on replacing each occurrence of an atomic sentence A by an equivalent sentence B, then F and G are equivalent.

(b) Show that the same holds for an atomic formula A and an equivalent formula B (provided, to avoid complications, that no variable occurring in A occurs bound in B or F).

(c) Show that the same holds even when A is not atomic.

10.12 Show that $F(x)$ is (logically) equivalent to $G(x)$ if and only if $\forall x(F(x) \leftrightarrow G(x))$ is valid.

10.13 (*Relettering bound variables.*) Show that:

(a) If F is a formula and y a variable not occurring free in F, then F is (logically) equivalent to a formula in which y does not occur at all. The same applies to any number of variables $y_1, \ldots, y_n$.

(b) Every formula is logically equivalent to a formula having no subformulas in which the same variable occurs both free and bound.

10.14 Show that the following pairs are equivalent:

(a) $\forall x F(x) \,\&\, \forall y G(y)$ and $\forall u(F(u) \,\&\, G(u))$.

(b) $\forall x F(x) \vee \forall y G(y)$ and $\forall u \forall v(F(u) \vee G(v))$.

(c) $\exists x F(x) \,\&\, \exists y G(y)$ and $\exists u \exists v(F(u) \,\&\, G(v))$.

(d) $\exists x F(x) \vee \exists y G(y)$ and $\exists u(F(u) \vee G(u))$.

[In (a), it is to be understood that u may be a variable not occurring free in $\forall x F(x)$ or $\forall y G(y)$; in particular, if x and y are the same variable, u may be that same variable. In (b) it is to be understood that u and v may be any distinct variables not occurring free in $\forall x F(x) \vee \forall y G(y)$; in particular, if x does not occur in free in $\forall y G(y)$ and y does not occur free in $\forall x F(x)$, then u may be x and y may be v. Analogously for (d) and (c).]