

Chapter 3

Predicate Logic

3.1 Quantifiers

In propositional logic we used large chunks of mathematical language, namely those parts that can have a truth value. Unfortunately this use of language is patently insufficient for mathematical practice. A simple argument, such as “all squares are positive, 9 is a square, therefore 9 is positive” cannot be dealt with. From the propositional point of view the above sentence is of the form $\varphi \wedge \psi \rightarrow \sigma$, and there is no reason why this sentence should be true, although we obviously accept it as true. The moral is that we have to extend the language, in such a way as to be able to discuss objects and relations. In particular we wish to introduce means to talk about *all* objects of the domain of discourse, e.g. we want to allow statements of the form “all even numbers are a sum of two odd primes”. Dually, we want a means of expressing “there exists an object such that ...”, e.g. in “there exists a real number whose square is 2”.

Experience has taught us that the basic mathematical statements are of the form “*a* has the property *P*” or “*a* and *b* are in the relation *R*”, etc. Examples are: “*n* is even”, “*f* is differentiable”, “ $3 = 5$ ”, “ $7 < 12$ ”, “*B* is between *A* and *C*”. Therefore we build our language from symbols for *properties*, *relations* and *objects*. Furthermore we add *variables* to range over objects (so called individual variables), and the usual logical connectives now including the *quantifiers* \forall and \exists (for “for all” and “there exists”).

We first give a few informal examples.

$\exists x P(x)$	there is an x with property P ,
$\forall y P(y)$	for all y P holds (all y have the property P),
$\forall x \exists y (x = 2y)$	for all x there is a y such that x is two times y ,
$\forall \varepsilon (\varepsilon > 0 \rightarrow \exists n (\frac{1}{n} < \varepsilon))$	for all positive ε there is an n such that $\frac{1}{n} < \varepsilon$,
$x < y \rightarrow \exists z (x < z \wedge z < y)$	if $x < y$, then there is a z such that $x < z$ and $z < y$,
$\forall x \exists y (x \cdot y = 1)$	for each x there exists an inverse y .

We know from elementary set theory that functions are a special kind of relations. It would, however, be in flagrant conflict with mathematical practice to avoid functions (or mappings). Moreover, it would be extremely cumbersome. So we will incorporate functions in our language.

Roughly speaking the language deals with two categories of syntactical entities: one for objects—the *terms*, and one for statements—the *formulas*. Examples of terms are: $17, x, (2 + 5) - 7, x^{3y+1}$.

What is the subject of predicate logic with a given language? Or, to put it differently, what are terms and formulas about? The answer is: formulas can express properties concerning a given set of relations and functions on a fixed domain of discourse. We have already met such situations in mathematics; we talked about *structures*, e.g. groups, rings, modules, ordered sets (see any algebra text). We will make structures our point of departure and we will get to the logic later.

In our logic we will speak about “all numbers” or “all elements”, but not about “all ideals” or “all subsets”, etc. Loosely speaking, our variables will vary over elements of a given universe (e.g. the $n \times n$ matrices over the reals), but not over properties or relations, or properties of properties, etc. For this reason the predicate logic of this book is called *first-order logic*, or also *elementary logic*. In everyday mathematics, e.g. analysis, one uses higher order logic. In a way it is a surprise that first-order logic can do so much for mathematics, as we will see. A short introduction to second-order logic will be presented in Chap. 5.

3.2 Structures

A group is a (non-empty) set equipped with two operations, a binary one and a unary one, and with a neutral element (satisfying certain laws). A partially ordered set is a set equipped with a binary relation (satisfying certain laws).

We generalize this as follows.

Definition 3.2.1 A structure is an ordered sequence $\langle A, R_1, \dots, R_n, F_1, \dots, F_m, \{c_i | i \in I\} \rangle$, where A is a non-empty set. R_1, \dots, R_n are relations on A , F_1, \dots, F_m are functions on A , and the c_i ($i \in I$) are elements of A (constants).

Warning The functions F_i are *total*, i.e. defined for all arguments; sometimes this calls for tricks, as with 0^{-1} (cf. p. 82).

Examples

$\langle \mathbb{R}, +, \cdot, ^{-1}, 0, 1 \rangle$ —the field of real numbers,
 $\langle \mathbb{N}, < \rangle$ —the ordered set of natural numbers.

We denote structures by Gothic capitals: $\mathfrak{A}, \mathfrak{B}, \mathfrak{C}, \mathfrak{D}, \dots$. The script letters are shown on p. 52.

If we overlook for a moment the special properties of the relations and operations (e.g. commutativity of addition on the reals), then what remains is the *type* of a structure, which is given by the number of relations, functions (or operations), and their respective arguments, plus the number (cardinality) of constants.

Definition 3.2.2 The *similarity type* of a structure $\mathfrak{A} = \langle A, R_1, \dots, R_n, F_1, \dots, F_m, \{c_i | i \in I\} \rangle$ is a sequence, $\langle r_1, \dots, r_n; a_1, \dots, a_m; \kappa \rangle$, where $R_i \subseteq A^{r_i}$, $F_j : A^{a_j} \rightarrow A$, $\kappa = |\{c_i | i \in I\}|$ (cardinality of I).

The two structures in our example have (similarity) type $\langle -; 2, 2, 1; 2 \rangle$ and $\langle 2; -, 0 \rangle$. The absence of relations, functions is indicated by $-$. There is no objection to extending the notion of structure to contain arbitrarily many relations or functions, but the most common structures have finite types (including finitely many constants).

It would, of course, have been better to use semicolons for our structures, i.e. $\langle A; R_1, \dots, R_n; F_1, \dots, F_m; c_i | i \in I \rangle$, but that would be too pedantic.

If $R \subseteq A$, then we call R a property (or *unary relation*), if $R \subseteq A^2$, then we call R a *binary relation*, if $R \subseteq A^n$, then we call R an *n-ary relation*.

The set A is called the *universe* of \mathfrak{A} .

Notation $A = |\mathfrak{A}|$. \mathfrak{A} is called (in)finite if its universe is (in)finite. We will mostly commit a slight abuse of language by writing down the constants instead of the set of constants; in the example of the field of real numbers we should have written: $\langle \mathbb{R}, +, \cdot, ^{-1}, \{0, 1\} \rangle$, but $\langle \mathbb{R}, +, \cdot, ^{-1}, 0, 1 \rangle$ is more traditional. Among the relations one finds in structures, there is a very special one: the *identity* (or *equality*) *relation*.

Since mathematical structures, as a rule, are equipped with the identity relation, we do not list the relation separately. It does, therefore, not occur in the similarity type. We henceforth assume all structures to possess an identity relation and we will explicitly mention any exceptions. For purely logical investigations it makes, of course, perfect sense to consider a logic without identity, but this book caters to readers from the mathematics or computer science community.

One also considers the “limiting cases” of relations and functions, i.e. 0-ary relations and functions. An 0-ary relation is a subset of A^\emptyset . Since $A^\emptyset = \{\emptyset\}$ there are two such relations: \emptyset and $\{\emptyset\}$ (considered as ordinals: 0 and 1). 0-ary relations can thus be seen as truth values, which makes them play the role of the interpretations of propositions. In practice 0-ary relations do not appear; e.g. they have no role to play in ordinary algebra. Most of the time the reader can joyfully forget about them, nonetheless we will allow them in our definition because they simplify certain considerations. A 0-ary function is a mapping from A^\emptyset into A , i.e. a mapping from $\{\emptyset\}$ into A . Since the mapping has a singleton as domain, we can identify it with its range.

In this way 0-ary functions can play the role of constants. The advantage of the procedure is, however, negligible in the present context, so we will keep our constants.

Exercises

1. Write down the similarity type for the following structures:
 - (i) $\langle \mathbb{Q}, <, 0 \rangle$
 - (ii) $\langle \mathbb{N}, +, \cdot, S, 0, 1, 2, 3, 4, \dots, n, \dots \rangle$, where $S(x) = x + 1$,
 - (iii) $\langle \mathcal{P}(\mathbb{N}), \subseteq, \cup, \cap, ^c, \emptyset \rangle$,
 - (iv) $\langle \mathbb{Z}/(5), +, \cdot, -, ^{-1}, 0, 1, 2, 3, 4 \rangle$,
 - (v) $\langle \{0, 1\}, \wedge, \vee, \rightarrow, \neg, 0, 1 \rangle$, where $\wedge, \vee, \rightarrow, \neg$ operate according to the ordinary truth tables,
 - (vi) $\langle \mathbb{R}, 1 \rangle$,
 - (vii) $\langle \mathbb{R} \rangle$,
 - (viii) $\langle \mathbb{R}, \mathbb{N}, <, T, ^2, | |, - \rangle$, where $T(a, b, c)$ is the relation “ b is between a and c ”, 2 is the square function, $-$ is the subtraction function and $| |$ the absolute value.
2. Give structures with type $\langle 1, 1; -, 3 \rangle, \langle 4; -, 0 \rangle$.

3.3 The Language of a Similarity Type

The considerations of this section are generalizations of those in Sect. 2.1. Since the arguments are rather similar, we will leave a number of details to the reader. For convenience we fix the similarity type in this section: $\langle r_1, \dots, r_n; a_1, \dots, a_m; \kappa \rangle$, where we assume $r_i \geq 0, a_j > 0$.

The alphabet consists of the following symbols:

1. Predicate symbols: P_1, \dots, P_n, \doteq
2. Function symbols: f_1, \dots, f_m
3. Constant symbols: \bar{c}_i for $i \in I$
4. Variables: x_0, x_1, x_2, \dots (countably many)
5. Connectives: $\vee, \wedge, \rightarrow, \neg, \leftrightarrow, \perp, \forall, \exists$
6. Auxiliary symbols: $(,)$,

\forall and \exists are called the *universal* and the *existential quantifier*. The curiously looking equality symbol has been chosen to avoid possible confusion. There are in fact a number of equality symbols in use: one to indicate the identity in the models, one to indicate the equality in the meta-language and the syntactic one introduced above. We will, however, practice the usual abuse of language, and use these distinctions only if it is really necessary. As a rule the reader will have no difficulty in recognizing the kind of identity involved.

Next we define the two syntactical categories.

Definition 3.3.1 TERM is the smallest set X with the properties

- (i) $\bar{c}_i \in X (i \in I)$ and $x_i \in X (i \in \mathbb{N})$,
- (ii) $t_1, \dots, t_{a_i} \in X \Rightarrow f_i(t_1, \dots, t_{a_i}) \in X$, for $1 \leq i \leq m$.

TERM is our *set of terms*.

Definition 3.3.2 FORM is the smallest set X with the properties:

- (i) $\perp \in X$; $P_i \in X$ if $r_i = 0$; $t_1, \dots, t_{r_i} \in \text{TERM} \Rightarrow P_i(t_1, \dots, t_{r_i}) \in X$; $t_1, t_2 \in \text{TERM} \Rightarrow t_1 = t_2 \in X$,
- (ii) $\varphi, \psi \in X \Rightarrow (\varphi \square \psi) \in X$, where $\square \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$,
- (iii) $\varphi \in X \Rightarrow (\neg \varphi) \in X$,
- (iv) $\varphi \in X \Rightarrow ((\forall x_i)\varphi), ((\exists x_i)\varphi) \in X$.

FORM is our *set of formulas*. We have introduced $t_1 = t_2$ separately, but we could have subsumed it under the first clause. If convenient, we will not treat equality separately. The formulas introduced in (i) are called *atoms*. We point out that (i) includes the case of 0-ary predicate symbols, conveniently called proposition symbols.

A proposition symbol is interpreted as a 0-ary relation, i.e. as 0 or 1 (cf. Definition 3.2.2). This is in accordance with the practice of propositional logic to interpret propositions as true or false. For our present purpose propositions are a luxury. In dealing with concrete mathematical situations (e.g. groups or posets) one has no reason to introduce propositions (things with a fixed truth value). However, propositions are convenient (and even important) in the context of Boolean-valued logic or Heyting-valued logic, and in syntactical considerations.

We will, however, allow a special proposition: \perp , the symbol for the false proposition (cf. Sect. 2.2).

The logical connectives have, what one could call “a domain of action”, e.g. in $\varphi \rightarrow \psi$ the connective \rightarrow yields the new formula $\varphi \rightarrow \psi$ from formulas φ and ψ , and so \rightarrow bears on φ, ψ and all their parts. For propositional connectives this is not terribly interesting, but for quantifiers (and variable-binding operators in general) it is. The notion goes by the name of *scope*. So in $((\forall x)\varphi)$ and $((\exists x)\varphi)$, φ is the *scope of the quantifier*. By locating the matching brackets one can easily effectively find the scope of a quantifier. If a variable, term or formula occurs in φ , we say that it is in the scope of the quantifier in $\forall x\varphi$ or $\exists x\varphi$.

Just as in the case of PROP, we have induction principles for TERM and FORM.

Lemma 3.3.3 *Let $A(t)$ be a property of terms. If $A(t)$ holds for t a variable or a constant, and if $A(t_1), A(t_2), \dots, A(t_n) \Rightarrow A(f(t_1, \dots, t_n))$, for all function symbols f , then $A(t)$ holds for all $t \in \text{TERM}$.*

Proof Cf. Theorem 2.1.3. □

Lemma 3.3.4 *Let $A(\varphi)$ be a property of formulas. If*

- (i) $A(\varphi)$ for atomic φ ,
- (ii) $A(\varphi), A(\psi) \Rightarrow A(\varphi \square \psi)$,
- (iii) $A(\varphi) \Rightarrow A(\neg \varphi)$,
- (iv) $A(\varphi) \Rightarrow A((\forall x_i)\varphi), A((\exists x_i)\varphi)$ for all i , then $A(\varphi)$ holds for all $\varphi \in \text{FORM}$.

Proof Cf. Theorem 2.1.3. □

We will straight away introduce a number of abbreviations. In the first place we adopt the bracket conventions of propositional logic. Furthermore we delete the outer brackets and the brackets round $\forall x$ and $\exists x$ whenever possible. We agree that quantifiers bind more strongly than binary connectives. Furthermore we join strings of quantifiers, e.g. $\forall x_1 x_2 \exists x_3 x_4 \varphi$ stands for $\forall x_1 \forall x_2 \exists x_3 \exists x_4 \varphi$. For better readability we will sometimes separate the quantifier and the formula by a dot: $\forall x \cdot \varphi$. We will also assume that n in $f(t_1, \dots, t_n)$, $P(t_1, \dots, t_n)$ always indicates the correct number of arguments. *A word of warning:* the use of $=$ might confuse a careless reader. The symbol “ $=$ ” is used in the language L , where it is a proper syntactic object. It occurs in formulas such as $x_0 = x_7$, but it also occurs in the meta-language, e.g. in the form $x = y$, which must be read “ x and y are one and the same variable”. However, the identity symbol in $x = y$ can just as well be the legitimate symbol from the alphabet, i.e. $x = y$ is a meta-atom, which can be converted into a proper atom by substituting genuine variable symbols for x and y . Sometimes \equiv is used for “syntactically identical”, as in “ x and y are the same variable”. We will opt for “ $=$ ” for the equality in structures (sets) and “ \doteq ” for the identity predicate symbol in the language. We will use \doteq a few times, but we prefer to stick to a simple “ $=$ ” trusting the alertness of the reader.

Example 3.3.5 Example of a language of type $\langle 2; 2, 1; 1 \rangle$.

predicate symbols: L, \doteq

function symbols: p, i

constant symbol: \bar{e}

Some terms: $t_1 := x_0$; $t_2 := p(x_1, x_2)$; $t_3 := p(\bar{e}, \bar{e})$; $t_4 := i(x_7)$; $t_5 := p(i(p(x_2, \bar{e})), i(x_1))$.

Some formulas:

$$\begin{array}{ll} \varphi_1 := x_0 \doteq x_2, & \varphi_4 := (x_0 \doteq x_1 \rightarrow x_1 \doteq x_0), \\ \varphi_2 := t_3 \doteq t_4, & \varphi_5 := (\forall x_0)(\forall x_1)(x_0 \doteq x_1 \rightarrow \neg L(v_0, x_1)) \\ \varphi_3 := L(i(x_5), \bar{e}), & \varphi_6 := (\forall x_0)(\exists x_1)(p(x_0, x_1) \doteq \bar{e}), \\ & \varphi_7 := (\exists x_1)(\neg x_1 \doteq \bar{e} \wedge p(x_1, x_1) \doteq \bar{e}). \end{array}$$

(We have chosen a suggestive notation; think of the language of ordered groups: L for “less than”, p, i for “product” and “inverse”.) Note that the order in which the various symbols are listed is important. In our example p has 2 arguments and i has 1.

In mathematics there are a number of *variable-binding operations*, such as summation, integration, abstraction. Consider, for example, integration, in $\int_0^1 \sin x \, dx$ the variable plays an unusual role for a variable. For x cannot “vary”; we cannot (without writing nonsense) substitute any number we like for x . In the integral the variable x is reduced to a tag. We say that the variable x is bound by the integration symbol. Analogously we distinguish in logic between *free* and *bound* variables.

A variable may occur in a formula more than once. It is quite often useful to look at a specific instance at a certain place in the string that makes up the formula. We call these *occurrences* of the variable, and we use expressions like “ x occurs in the subformula ψ of φ .” In general we consider occurrences of formulas, terms, quantifiers, and the like.

In defining various syntactical notions we again freely use the principle of *definition by recursion* (cf. Theorem 2.1.6). The justification is immediate: the value of a term (formula) is uniquely determined by the values of its parts. This allows us to determine the value of $H(t)$ for a mapping acting on terms, in finitely many steps.

Definition by Recursion on TERM Let $H_0 : Var \cup Const \rightarrow A$ (i.e. H_0 is defined on variables and constants), $H_i : A^{a_i} \rightarrow A$, then there is a unique mapping $H : TERM \rightarrow A$ such that

$$\begin{cases} H(t) = H_0(t) \text{ for } t \text{ a variable or a constant,} \\ H(f_i(t_1, \dots, t_{a_i})) = H_i(H(t_1), \dots, H(t_{a_i})). \end{cases}$$

Definition by Recursion on FORM Let

$$\begin{aligned} H_{at} : At &\rightarrow A \quad (\text{i.e. } H_{at} \text{ is defined on atoms}), \\ H_{\square} : A^2 &\rightarrow A \quad (\square \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}), \\ H_{\neg} : A &\rightarrow A, \\ H_{\forall} : A \times N &\rightarrow A, \\ H_{\exists} : A \times N &\rightarrow A. \end{aligned}$$

Then there is a unique mapping $H : FORM \rightarrow A$ such that

$$\begin{cases} H(\varphi) = H_{at}(\varphi) \quad \text{for atomic } \varphi, \\ H(\varphi \square \psi) = H_{\square}(H(\varphi), H(\psi)), \\ H(\neg \varphi) = H_{\neg}(H(\varphi)), \\ H(\forall x_i \varphi) = H_{\forall}(H(\varphi), i), \\ H(\exists x_i \varphi) = H_{\exists}(H(\varphi), i). \end{cases}$$

Definition 3.3.6 The set $FV(t)$ of free variables of t is defined by

- (i) $FV(x_i) := \{x_i\}$,
 $FV(\bar{c}_i) := \emptyset$
- (ii) $FV(f(t_1, \dots, t_n)) := FV(t_1) \cup \dots \cup FV(t_n)$.

Remark To avoid messy notation we will usually drop the indices and tacitly assume that the number of arguments is correct. The reader can easily provide the correct details, should he wish to do so.

Definition 3.3.7 The set $FV(\varphi)$ of free variables of φ is defined by

- (i) $FV(P(t_1, \dots, t_p)) := FV(t_1) \cup \dots \cup FV(t_p)$,
 $FV(t_1 = t_2) := FV(t_1) \cup FV(t_2)$, $FV(\perp) = FV(P) := \emptyset$ for P a proposition symbol,

- (ii) $FV(\varphi \square \psi) := FV(\varphi) \cup FV(\psi),$
 $FV(\neg \varphi) := FV(\varphi),$
- (iii) $FV(\forall x_i \varphi) := FV(\exists x_i \varphi) := FV(\varphi) - \{x_i\}.$

Definition 3.3.8 t or φ is called *closed* if $FV(t) = \emptyset$, resp. $FV(\varphi) = \emptyset$. A closed formula is also called a *sentence*. A formula without quantifiers is called *open*. $TERM_c$ denotes the set of closed terms; $SENT$ denotes the set of sentences.

It is left to the reader to define the set $BV(\varphi)$ of *bound variables* of φ .

Continuation of Example 3.3.5

$$FV(t_2) = \{x_1, x_2\}; FV(t_3) = \emptyset; FV(\varphi_2) = FV(t_3) \cup FV(t_4) = \{x_7\};$$

$$FV(\varphi_7) = \emptyset; BV(\varphi_4) = \emptyset; BV(\varphi_6) = \{x_0, x_1\}. \varphi_5, \varphi_6, \varphi_7 \text{ are sentences.}$$

Warning $FV(\varphi) \cap BV(\varphi)$ need not be empty; in other words, the same variable may occur free *and* bound. To handle such situations one considers free (resp. bound) *occurrences* of variables. When necessary we will make informal use of occurrences of variables; see also p. 59.

Example $\forall x_1(x_1 = x_2) \rightarrow P(x_1)$ contains x_1 both free and bound, for the occurrence of x_1 in $P(x_1)$ is not within the scope of the quantifier.

In predicate calculus we have substitution operators for terms and for formulas.

Definition 3.3.9 Let s and t be terms, then $s[t/x]$ is defined by:

- (i) $y[t/x] := \begin{cases} y & \text{if } y \not\equiv x \\ t & \text{if } y \equiv x \end{cases}$
 $c[t/x] := c$
- (ii) $f(t_1, \dots, t_p)[t/x] := f(t_1[t/x], \dots, t_p[t/x]).$

Note that in the clause (i) $y \equiv x$ means “ x and y are the same variables”.

Definition 3.3.10 $\varphi[t/x]$ is defined by:

- (i) $\perp[t/x] := \perp,$
 $P[t/x] := P$ for propositions P ,
 $P(t_1, \dots, t_p)[t/x] := P(t_1[t/x], \dots, t_p[t/x]),$
 $(t_1 = t_2)[t/x] := t_1[t/x] = t_2[t/x],$
- (ii) $(\varphi \square \psi)[t/x] := \varphi[t/x] \square \psi[t/x],$
 $(\neg \varphi)[t/x] := \neg \varphi[t/x]$

$$\begin{aligned}
\text{(iii)} \quad (\forall y\varphi)[t/x] &:= \begin{cases} \forall y\varphi[t/x] & \text{if } x \not\equiv y \\ \forall y\varphi & \text{if } x \equiv y \end{cases} \\
(\exists y\varphi)[t/x] &:= \begin{cases} \exists y\varphi[t/x] & \text{if } x \not\equiv y \\ \exists y\varphi & \text{if } x \equiv y \end{cases}
\end{aligned}$$

Substitution of formulas is defined as in the case of propositions; for convenience we use “\$” as a symbol for the propositional symbol (0-ary predicate symbol) which acts as a “place holder”.

Definition 3.3.11 $\sigma[\varphi/\$]$ is defined by:

$$\begin{aligned}
\text{(i)} \quad \sigma[\varphi/\$] &:= \begin{cases} \sigma & \text{if } \sigma \not\equiv \$ \\ \varphi & \text{if } \sigma \equiv \$ \end{cases} \quad \text{for atomic } \sigma, \\
\text{(ii)} \quad (\sigma_1 \Box \sigma_2)[\varphi/\$] &:= \sigma_1[\varphi/\$] \Box \sigma_2[\varphi/\$] \\
(\neg \sigma_1)[\varphi/\$] &:= \neg \sigma_1[\varphi/\$] \\
(\forall y\sigma)[\varphi/\$] &:= \forall y. \sigma[\varphi/\$] \\
(\exists y\sigma)[\varphi/\$] &:= \exists y. \sigma[\varphi/\$].
\end{aligned}$$

Continuation of Example 3.3.5

$$\begin{aligned}
t_4[t_2/x_1] &= i(x_7); & t_4[t_2/x_7] &= i(p(x_1, x_2)); \\
t_5[x_2/x_1] &= p(i(p(x_2, \bar{e}), i(x_2))), \\
\varphi_1[t_3/x_0] &= p(\bar{e}, \bar{e}) \doteq x_2; & \varphi_5[t_3/x_0] &= \varphi_5.
\end{aligned}$$

We will sometimes make *simultaneous substitutions*, the definition is a slight modification of Definitions 3.3.9, 3.3.10 and 3.3.11. The reader is asked to write down the formal definitions. We denote the result of a simultaneous substitution of t_1, \dots, t_n for y_1, \dots, y_n in t by $t[t_1, \dots, t_n/y_1, \dots, y_n]$ (similarly for φ).

Note that a simultaneous substitution is not the same as its corresponding repeated substitution.

Example $(x_0 \doteq x_1)[x_1, x_0/x_0, x_1] = (x_1 \doteq x_0)$, but $((x_0 \doteq x_1)[x_1/x_0])[x_0/x_1] = (x_1 \doteq x_1)[x_0/x_1] = (x_0 \doteq x_0)$.

The quantifier clause in Definition 3.3.10 forbids substitution for bound variables. There is, however, one more case we want to forbid: a substitution, in which some variable after the substitution becomes bound. We will give an example of such a substitution; the reason why we forbid it is that it can change the truth value in an absurd way. At this moment we do not have a truth definition, so the argument is purely heuristic.

Example $\exists x(y < x)[x/y] = \exists x(x < x)$.

Note that the right-hand side is false in an ordered structure, whereas $\exists x(y < x)$ may very well be true. We make our restriction precise.

Definition 3.3.12 t is free for x in φ if

- (i) φ is atomic,
- (ii) $\varphi := \varphi_1 \square \varphi_2$ (or $\varphi := \neg \varphi_1$) and t is free for x in φ_1 and φ_2 (resp. φ_1),
- (iii) $\varphi := \exists y \psi$ (or $\varphi := \forall y \psi$) and if $x \in FV(\varphi)$, then $y \notin FV(t)$ and t is free for x in ψ .

Examples

1. x_2 is free for x_0 in $\exists x_3 P(x_0, x_3)$,
2. $f(x_0, x_1)$ is not free for x_0 in $\exists x_1 P(x_0, x_3)$,
3. x_5 is free for x_1 in $P(x_1, x_3) \rightarrow \exists x_1 Q(x_1, x_2)$.

Note that the use of “ t is free for x in φ ” comes down to the fact that the (free) variables of t are not going to be bound after substitution in φ .

Lemma 3.3.13 t is free for x in $\varphi \Leftrightarrow$ the variables of t in $\varphi[t/x]$ are not bound by a quantifier.

Proof Induction on φ .

- For atomic φ the lemma is evident.
- $\varphi = \varphi_1 \square \varphi_2$. t is free for x in $\varphi \stackrel{\text{def.}}{\Leftrightarrow} t$ is free for x in φ_1 and t is free for x in $\varphi_2 \stackrel{\text{i.h.}}{\Leftrightarrow}$ the variables of t in $\varphi_1[t/x]$ are not bound by a quantifier and the variables of t in $\varphi_2[t/x]$ are not bound by a quantifier \Leftrightarrow the variables of t in $(\varphi_1 \square \varphi_2)[t/x]$ are not bound by a quantifier.
- $\varphi = \neg \varphi_1$, similar.
- $\varphi = \exists y \psi$. It suffices to consider the case $x \in FV(\varphi)$. t is free for x in $\varphi \stackrel{\text{def.}}{\Leftrightarrow} y \notin FV(t)$ and t is free for x in $\psi \stackrel{\text{i.h.}}{\Leftrightarrow}$ the variables of t are not in the scope of $\exists y$ and the variables of t in $\psi[t/x]$ are not bound by (another) quantifier \Leftrightarrow the variables of t in $\varphi[t/x]$ are not bound by a quantifier. \square

There is an analogous definition and lemma for the substitution of formulas.

Definition 3.3.14 φ is free for $\$$ in σ if:

- (i) σ is atomic,
- (ii) $\sigma := \sigma_1 \square \sigma_2$ (or $\neg \sigma_1$) and φ is free for $\$$ in σ_1 and in σ_2 (or in σ_1),
- (iii) $\sigma := \exists y \tau$ (or $\forall y \tau$) and if $\$$ occurs in σ then $y \notin FV(\varphi)$ and φ is free for $\$$ in τ .

Lemma 3.3.15 φ is free for $\$$ in $\sigma \Leftrightarrow$ the free variables of φ are in $\sigma[\varphi/\$]$ not bound by a quantifier.

Proof As for Lemma 3.3.13. \square

From now on we tacitly suppose that all our substitutions are “free for”.

For convenience we introduce an informal notation that simplifies reading and writing.

Notation In order to simplify the substitution notation and to conform to an ancient suggestive tradition we will write down (meta-) expressions like $\varphi(x, y, z)$, $\psi(x, x)$, etc. This neither means that the listed variables occur free nor that no other ones occur free. It is merely a convenient way to handle substitution informally: $\varphi(t)$ is the result of replacing x by t in $\varphi(x)$; $\varphi(t)$ is called a *substitution instance* of $\varphi(x)$.

We use the languages introduced above to describe structures, or classes of structures of a given type. The predicate symbols, function symbols and constant symbols act as names for various relations, operations and constants. In describing a structure it is a great help to be able to refer to all elements of $|\mathfrak{A}|$ individually, i.e. to have *names* for all elements (if only as an auxiliary device). Therefore we introduce the following.

Definition 3.3.16 The *extended language*, $L(\mathfrak{A})$, of \mathfrak{A} is obtained from the language L , of the type of \mathfrak{A} , by adding constant symbols for all elements of $|\mathfrak{A}|$. We denote the constant symbol, belonging to $a \in |\mathfrak{A}|$, by \bar{a} .

Example Consider the language L of groups; then $L(\mathfrak{A})$, for \mathfrak{A} the additive group of integers, has (extra) constant symbols $\bar{0}, \bar{1}, \bar{2}, \dots, \bar{-1}, \bar{-2}, \bar{-3}, \dots$. Observe that in this way 0 gets two names: the old one and one of the new ones. This is no problem, why shouldn't something have more than one name?

Exercises

1. Write down an alphabet for the languages of the types given in Exercise 1 of Sect. 3.2.
2. Write down five terms of the language belonging to Exercise 1 (iii), (viii). Write down two atomic formulas of the language belonging to Exercise 1 (vii) and two closed atoms for Exercise 1 (iii), (vi).
3. Write down an alphabet for languages of types $\langle 3; 1, 1, 2; 0 \rangle$, $\langle -, 2; 0 \rangle$ and $\langle 1; -, 3 \rangle$.
4. Check which terms are free in the following cases, and carry out the substitution:

- | | |
|---|--|
| (a) x for x in $x = x$, | (f) $x + w$ for z in $\forall w(x + z = \bar{0})$, |
| (b) y for x in $x = x$, | (g) $x + y$ for z in $\forall w(x + z = \bar{0}) \wedge$ |
| (c) $x + y$ for y in $z = \bar{0}$, | $\exists y(z = x)$, |
| (d) $\bar{0} + y$ for y in $\exists x(y = x)$, | (h) $x + y$ for z in $\forall u(u = v) \rightarrow$ |
| (e) $x + y$ for z in $\exists w(w + x = \bar{0})$, | $\forall z(z = y)$. |

3.4 Semantics

The art of interpreting (mathematical) statements presupposes a strict separation between “language” and the mathematical “universe” of entities. The objects of language are symbols, or strings of symbols, the entities of mathematics are numbers, sets, functions, triangles, etc. It is a matter for the philosophy of mathematics to reflect on the universe of mathematics; here we will simply accept it as given to us. Our requirements concerning the mathematical universe are, at present, fairly modest. For example, ordinary set theory will do very well for us. Likewise our desiderata with respect to language are modest. We just suppose that there is an unlimited supply of symbols.

The idea behind the semantics of predicate logic is very simple. Following Tarski, we assume that a statement σ is true in a structure, if σ actually is the case (the sentence “Snow is white” is true if snow actually is white). A mathematical example: “ $\overline{2} + \overline{2} = \overline{4}$ ” is true in the structure of natural numbers (with addition) if $2 + 2 = 4$ (i.e. if addition of the *numbers* 2 and 2 yields the *number* 4). Interpretation is the art of relating syntactic objects (strings of symbols) and states of affairs “in reality”.

We will start by giving an example of an interpretation in a simple case. We consider the structure $\mathfrak{A} = (\mathbb{Z}, <, +, -, 0)$, i.e. the ordered group of integers.

The language has in its alphabet:

predicate symbols: \doteq, L
 function symbols: P, M
 constant symbol: $\overline{0}$

$L(\mathfrak{A})$ has, in addition to all that, constant symbols \overline{m} for all $m \in \mathbb{Z}$. We first interpret the closed terms of $L(\mathfrak{A})$; the interpretation $t^{\mathfrak{A}}$ of a term t is an element of \mathbb{Z} .

t	$t^{\mathfrak{A}}$
\overline{m}	m
$P(t_1, t_2)$	$t_1^{\mathfrak{A}} + t_2^{\mathfrak{A}}$
$M(t)$	$-t^{\mathfrak{A}}$

Roughly speaking, we interpret \overline{m} as “its number”, P as *plus*, M as *minus*. Note that we interpret only closed terms. This stands to reason, how should one assign a definite integer to x ?

Next we interpret *sentences* of $L(\mathfrak{A})$ by assigning one of the truth values 0 or 1. As far as the propositional connectives are concerned, we follow the semantics for propositional logic.

$$\begin{aligned}
 v(\perp) &= 0, \\
 v(t \doteq s) &= \begin{cases} 1 & \text{if } t^{\mathfrak{A}} = s^{\mathfrak{A}} \\ 0 & \text{else,} \end{cases} \\
 v(L(t, s)) &= \begin{cases} 1 & \text{if } t^{\mathfrak{A}} < s^{\mathfrak{A}} \\ 0 & \text{else} \end{cases}
 \end{aligned}$$

$$\begin{aligned}
v(\varphi \Box \psi) & \quad \text{as in Definition 2.2.1} \\
v(\neg \varphi) & \\
v(\forall x \varphi) &= \min\{v(\varphi[\bar{n}/x]) \mid n \in \mathbb{Z}\} \\
v(\exists x \sigma) &= \max\{v(\sigma[\bar{n}/x]) \mid n \in \mathbb{Z}\}
\end{aligned}$$

A few remarks are in order.

1. In fact we have defined a function v by recursion on φ .
2. The valuation of a universally quantified formula is obtained by taking the minimum of all valuations of the individual instances, i.e. the value is 1 (true) iff all instances have the value 1. In this respect \forall is a generalization of \wedge . Likewise \exists is a generalization of \vee .
3. v is uniquely determined by \mathfrak{A} , hence $v_{\mathfrak{A}}$ would be a more appropriate notation. For convenience we will, however, stick to just v .
4. As in the semantics of propositional logic, we will write $\llbracket \varphi \rrbracket_{\mathfrak{A}}$ for $v_{\mathfrak{A}}(\varphi)$, and when no confusion arises we will drop the subscript \mathfrak{A} .
5. It would be tempting to make our notation really uniform by writing $\llbracket t \rrbracket_{\mathfrak{A}}$ for $t^{\mathfrak{A}}$. We will, however, keep both notations and use whichever is the most readable. The superscript notation has the drawback that it requires more brackets, but the $\llbracket \rrbracket$ -notation does not improve readability.

Examples

1. $(P(P(\bar{2}, \bar{3}), M(\bar{7})))^{\mathfrak{A}} = P(\bar{2}, \bar{3})^{\mathfrak{A}} + M(\bar{7})^{\mathfrak{A}} = (\bar{2}^{\mathfrak{A}} + \bar{3}^{\mathfrak{A}}) + (-\bar{7}^{\mathfrak{A}}) = 2 + 3 + (-7) = -2$,
2. $\llbracket \bar{2} \doteq \bar{-1} \rrbracket = 0$, since $2 \neq -1$,
3. $\llbracket \bar{0} \doteq \bar{1} \rightarrow L(\bar{25}, \bar{10}) \rrbracket = 1$, since $\llbracket \bar{0} = \bar{1} \rrbracket = 0$ and $\llbracket L(\bar{25}, \bar{10}) \rrbracket = 0$; by the interpretation of the implication the value is 1,
4. $\llbracket \forall x \exists y (L(x, y)) \rrbracket = \min_n (\max_m \llbracket L(\bar{n}, \bar{m}) \rrbracket)$.
 $\llbracket L(\bar{n}, \bar{m}) \rrbracket = 1$ for $m > n$, so for fixed n , $\max_m \llbracket L(\bar{n}, \bar{m}) \rrbracket = 1$, and hence $\min_n \max_m \llbracket L(\bar{n}, \bar{m}) \rrbracket = 1$.

Let us now present a definition of interpretation for the general case. Consider $\mathfrak{A} = \langle A, R_1, \dots, R_n, F_1, \dots, F_m, \{c_i \mid i \in I\} \rangle$ of a given similarity type $\langle r_1, \dots, r_n; a_1, \dots, a_m; |I| \rangle$.

The corresponding language has predicate symbols $\bar{R}_1, \dots, \bar{R}_n$, function symbols $\bar{F}_1, \dots, \bar{F}_m$ and constant symbols \bar{c}_i . $L(\mathfrak{A})$, moreover, has constant symbols \bar{a} for all $a \in |A|$.

Definition 3.4.1 An interpretation of the closed terms of $L(\mathfrak{A})$ in \mathfrak{A} is a mapping $(\cdot)^{\mathfrak{A}} : \text{TERM}_c \rightarrow |A|$ satisfying:

- (i) $\bar{c}_i^{\mathfrak{A}} = c_i$ ($= \llbracket \bar{c}_i \rrbracket_{\mathfrak{A}}$)
 $\bar{a}^{\mathfrak{A}} = a$, ($= \llbracket \bar{a} \rrbracket_{\mathfrak{A}}$)

$$(ii) \quad (\overline{F}_i(t_1, \dots, t_p))^{\mathfrak{A}} = F_i(t_1^{\mathfrak{A}}, \dots, t_p^{\mathfrak{A}}), \quad (= \llbracket F_i(t_1, \dots, t_p) \rrbracket_{\mathfrak{A}})$$

$$\text{where } p = a_i \quad \quad \quad = \overline{F}_i(\llbracket t_1 \rrbracket_{\mathfrak{A}}, \dots, \llbracket t_p \rrbracket_{\mathfrak{A}})$$

There is also a *valuation notation* using Scott brackets; we have indicated in the above definition how these brackets are to be used. The following definition is exclusively in terms of valuations.

Definition 3.4.2 An interpretation of the sentences φ of $L(\mathfrak{A})$ in \mathfrak{A} is a mapping $\llbracket \cdot \rrbracket_{\mathfrak{A}} : SENT \rightarrow \{0, 1\}$, satisfying:

- (i) $\llbracket \perp \rrbracket_{\mathfrak{A}} := 0$,
 $\llbracket \overline{R} \rrbracket_{\mathfrak{A}} := R$ (i.e. 0 or 1).
- (ii) $\llbracket \overline{R}_i(t_1, \dots, t_p) \rrbracket_{\mathfrak{A}} := \begin{cases} 1 & \text{if } \langle t_1^{\mathfrak{A}}, \dots, t_p^{\mathfrak{A}} \rangle \in R_i, \\ 0 & \text{else.} \end{cases}$ where $p = r_i$,
 $\llbracket t_1 = t_2 \rrbracket_{\mathfrak{A}} := \begin{cases} 1 & \text{if } t_1^{\mathfrak{A}} = t_2^{\mathfrak{A}} \\ 0 & \text{else.} \end{cases}$
- (iii) $\llbracket \varphi \wedge \psi \rrbracket_{\mathfrak{A}} := \min(\llbracket \varphi \rrbracket_{\mathfrak{A}}, \llbracket \psi \rrbracket_{\mathfrak{A}})$,
 $\llbracket \varphi \vee \psi \rrbracket_{\mathfrak{A}} := \max(\llbracket \varphi \rrbracket_{\mathfrak{A}}, \llbracket \psi \rrbracket_{\mathfrak{A}})$,
 $\llbracket \varphi \rightarrow \psi \rrbracket_{\mathfrak{A}} := \max(1 - \llbracket \varphi \rrbracket_{\mathfrak{A}}, \llbracket \psi \rrbracket_{\mathfrak{A}})$,
 $\llbracket \varphi \leftrightarrow \psi \rrbracket_{\mathfrak{A}} := 1 - |\llbracket \varphi \rrbracket_{\mathfrak{A}} - \llbracket \psi \rrbracket_{\mathfrak{A}}|$,
 $\llbracket \neg \varphi \rrbracket_{\mathfrak{A}} := 1 - \llbracket \varphi \rrbracket_{\mathfrak{A}}$.
- (iv) $\llbracket \forall x \varphi \rrbracket_{\mathfrak{A}} := \min\{\llbracket \varphi[\overline{a}/x] \rrbracket_{\mathfrak{A}} \mid a \in |\mathfrak{A}|\}$,
 $\llbracket \exists x \varphi \rrbracket_{\mathfrak{A}} := \max\{\llbracket \varphi[\overline{a}/x] \rrbracket_{\mathfrak{A}} \mid a \in |\mathfrak{A}|\}$.

Convention: from now on we will assume that all structures and languages have the appropriate similarity type, so that we don't have to specify the types all the time.

In predicate logic there is a popular and convenient alternative for the valuation notation:

$\mathfrak{A} \models \varphi$ stands for $\llbracket \varphi \rrbracket_{\mathfrak{A}} = 1$. We say that “ φ is true, valid, in \mathfrak{A} ” if $\mathfrak{A} \models \varphi$. The relation \models is called the *satisfaction relation*.

Note that the same notation is available in propositional logic—there the role of \mathfrak{A} is taken by the valuation, so one could very well write $v \models \varphi$ for $\llbracket \varphi \rrbracket_v = 1$.

So far we have only defined truth for sentences of $L(\mathfrak{A})$. In order to extend \models to arbitrary formulas we introduce a new notation.

Definition 3.4.3 Let $FV(\varphi) = \{z_1, \dots, z_k\}$, then $Cl(\varphi) := \forall z_1 \dots z_k \varphi$ is the *universal closure* of φ (we assume the order of variables z_i to be fixed in some way).

Definition 3.4.4

- (i) $\mathfrak{A} \models \varphi$ iff $\mathfrak{A} \models Cl(\varphi)$,
- (ii) $\models \varphi$ iff $\mathfrak{A} \models \varphi$ for all \mathfrak{A} (of the appropriate type),
- (iii) $\mathfrak{A} \models \Gamma$ iff $\mathfrak{A} \models \psi$ for all $\psi \in \Gamma$,
- (iv) $\Gamma \models \varphi$ iff $(\mathfrak{A} \models \Gamma \Rightarrow \mathfrak{A} \models \varphi)$, where $\Gamma \cup \{\varphi\}$ consists of sentences.

If $\mathfrak{A} \models \sigma$, we call \mathfrak{A} a *model* of σ . In general: if $\mathfrak{A} \models \Gamma$, we call \mathfrak{A} a *model* of Γ . We say that φ is *true* if $\models \varphi$, φ is a *semantic consequence* of Γ if $\Gamma \models \varphi$, i.e. φ holds in each model of Γ . Note that this is all a straightforward generalization of Definition 2.2.4.

If φ is a formula with free variables, say $FV(\varphi) = \{z_1, \dots, z_k\}$, then we say that φ is *satisfied* by $a_1, \dots, a_k \in |\mathfrak{A}|$ if $\mathfrak{A} \models \varphi[\bar{a}_1, \dots, \bar{a}_k/z_1, \dots, z_k]$, φ is called *satisfiable in \mathfrak{A}* if there are a_1, \dots, a_k such that φ is satisfied by a_1, \dots, a_k and φ is called *satisfiable* if it is satisfiable in some \mathfrak{A} . Note that φ is satisfiable in \mathfrak{A} iff $\mathfrak{A} \models \exists z_1 \dots z_k \varphi$.

The properties of the satisfaction relation are in understandable and convenient correspondence with the intuitive meaning of the connectives.

Lemma 3.4.5 *If we restrict ourselves to sentences, then*

- (i) $\mathfrak{A} \models \varphi \wedge \psi \Leftrightarrow \mathfrak{A} \models \varphi$ and $\mathfrak{A} \models \psi$,
- (ii) $\mathfrak{A} \models \varphi \vee \psi \Leftrightarrow \mathfrak{A} \models \varphi$ or $\mathfrak{A} \models \psi$,
- (iii) $\mathfrak{A} \models \neg \varphi \Leftrightarrow \mathfrak{A} \not\models \varphi$,
- (iv) $\mathfrak{A} \models \varphi \rightarrow \psi \Leftrightarrow (\mathfrak{A} \models \varphi \Rightarrow \mathfrak{A} \models \psi)$,
- (v) $\mathfrak{A} \models \varphi \leftrightarrow \psi \Leftrightarrow (\mathfrak{A} \models \varphi \Leftrightarrow \mathfrak{A} \models \psi)$,
- (vi) $\mathfrak{A} \models \forall x \varphi \Leftrightarrow \mathfrak{A} \models \varphi[\bar{a}/x]$, for all $a \in |\mathfrak{A}|$,
- (vii) $\mathfrak{A} \models \exists x \varphi \Leftrightarrow \mathfrak{A} \models \varphi[\bar{a}/x]$, for some $a \in |\mathfrak{A}|$.

Proof Immediate from Definition 3.4.2. We will do two cases.

- (iv) $\mathfrak{A} \models \varphi \rightarrow \psi \Leftrightarrow \llbracket \varphi \rightarrow \psi \rrbracket_{\mathfrak{A}} = \max(1 - \llbracket \varphi \rrbracket_{\mathfrak{A}}, \llbracket \psi \rrbracket_{\mathfrak{A}}) = 1$. Suppose $\mathfrak{A} \models \varphi$, i.e. $\llbracket \varphi \rrbracket_{\mathfrak{A}} = 1$, then clearly $\llbracket \psi \rrbracket_{\mathfrak{A}} = 1$, or $\mathfrak{A} \models \psi$.
Conversely, let $\mathfrak{A} \models \varphi \Rightarrow \mathfrak{A} \models \psi$, and suppose $\mathfrak{A} \not\models \varphi \rightarrow \psi$, then $\llbracket \varphi \rightarrow \psi \rrbracket_{\mathfrak{A}} = \max(1 - \llbracket \varphi \rrbracket_{\mathfrak{A}}, \llbracket \psi \rrbracket_{\mathfrak{A}}) = 0$. Hence $\llbracket \psi \rrbracket_{\mathfrak{A}} = 0$ and $\llbracket \varphi \rrbracket_{\mathfrak{A}} = 1$. Contradiction.
- (vii) $\mathfrak{A} \models \exists x \varphi(x) \Leftrightarrow \max\{\llbracket \varphi(\bar{a}) \rrbracket_{\mathfrak{A}} \mid a \in |\mathfrak{A}|\} = 1 \Leftrightarrow$ there is an $a \in |\mathfrak{A}|$ such that $\llbracket \varphi(\bar{a}) \rrbracket_{\mathfrak{A}} = 1 \Leftrightarrow$ there is an $a \in |\mathfrak{A}|$ such that $\mathfrak{A} \models \varphi(\bar{a})$. \square

Lemma 3.4.5 tells us that the interpretation of sentences in \mathfrak{A} runs parallel to the construction of the sentences by means of the connectives. In other words, we replace the connectives by their analogues in the meta-language and interpret the atoms by checking the relations in the structure.

For example, consider our example of the ordered additive group of integers: $\mathfrak{A} \models \neg \forall x \exists y (x \doteq P(y, y)) \Leftrightarrow$ It is not the case that for each number n there exists an m such that $n = 2m \Leftrightarrow$ not every number can be halved in \mathfrak{A} . This clearly is correct, take for instance $n = 1$.

Let us reflect for a moment on the valuation of proposition symbols; an 0-ary relation is a subset of $A^\emptyset = \{\emptyset\}$, i.e. it is \emptyset or $\{\emptyset\}$ and these are, considered as ordinals, 0 or 1. So $\llbracket P \rrbracket_{\mathfrak{A}} = P$, and P is a truth value. This makes our definition perfectly reasonable. Indeed, without aiming for a systematic treatment, we may observe that formulas correspond to subsets of A^k , where k is the number of free variables. For example, let $FV(\varphi) = \{z_1, \dots, z_k\}$, then we could put $\llbracket \varphi \rrbracket_{\mathfrak{A}} = \{\langle a_1, \dots, a_k \rangle \mid \mathfrak{A} \models \varphi(\bar{a}_1, \dots, \bar{a}_k)\} (= \{\langle a_1, \dots, a_n \rangle \mid \llbracket \varphi(\bar{a}_1, \dots, \bar{a}_k) \rrbracket_{\mathfrak{A}} = 1\})$, thus stretching the meaning of $\llbracket \varphi \rrbracket_{\mathfrak{A}}$ a bit. It is immediately clear that applying quantifiers to φ reduces the “dimension”. For example, $\llbracket \exists x P(x, y) \rrbracket_{\mathfrak{A}} = \{a \mid \mathfrak{A} \models P(\bar{b}, \bar{a}) \text{ for some } b\}$, which is the projection of $\llbracket P(x, y) \rrbracket_{\mathfrak{A}}$ onto the y -axis.

Exercises

1. Let $\mathfrak{N} = \langle N, +, \cdot, S, 0 \rangle$, and L a language of type $\langle -, 2, 2, 1; 1 \rangle$.
 - (i) Give two distinct terms t in L such that $t^{\mathfrak{N}} = 5$.
 - (ii) Show that for each natural number $n \in N$ there is a term t such that $t^{\mathfrak{N}} = n$.
 - (iii) Show that for each $n \in N$ there are infinitely many terms t such that $t^{\mathfrak{N}} = n$.
2. Let \mathfrak{A} be the structure of Exercise 1 (v) of Sect. 3.2. Evaluate $((\bar{1} \rightarrow \bar{0}) \rightarrow \neg \bar{0}) \wedge (\neg \bar{0} \rightarrow (\bar{1} \rightarrow \bar{0}))^{\mathfrak{A}}, (\bar{1} \rightarrow \neg(\neg \bar{0} \vee \bar{1}))^{\mathfrak{A}}$.
3. Let \mathfrak{A} be the structure of Exercise 1 (viii) of Sect. 3.2. Evaluate $(|(\sqrt{3})^2 - \overline{-5}|)^{\mathfrak{A}}, (\bar{1} - (|(\overline{-2})| - (\bar{5} - \overline{-2})))^{\mathfrak{A}}$.
4. Which cases of Lemma 3.4.5 remain correct if we consider formulas in general?
5. For sentences σ we have $\mathfrak{A} \models \sigma$ or $\mathfrak{A} \models \neg \sigma$. Show that this does not hold for φ with $FV(\varphi) \neq \emptyset$. Show that not even for sentences $\models \sigma$ or $\models \neg \sigma$ holds in general.
6. Show for closed terms t and formulas φ (in $L(\mathfrak{A})$):

$$\mathfrak{A} \models t = \llbracket t \rrbracket_{\mathfrak{A}},$$

$$\mathfrak{A} \models \varphi(t) \leftrightarrow \varphi(\llbracket t \rrbracket_{\mathfrak{A}}).$$
 (We will also obtain this as a corollary to the Substitution Theorem, Corollary 3.5.9.)
7. Show that $\mathfrak{A} \models \varphi \Rightarrow \mathfrak{A} \models \psi$ for all \mathfrak{A} , implies $\models \varphi \Rightarrow \models \psi$, but not vice versa.

3.5 Simple Properties of Predicate Logic

Our definition of validity (truth) was a straightforward extension of the valuation definition of propositional logic. As a consequence formulas which are instances of tautologies are true in all structures \mathfrak{A} (Exercise 1). So we can copy many results from Sects. 2.2 and 2.3. We will use these results with a simple reference to propositional logic.

The specific properties concerning quantifiers will be treated in this section. First we consider the generalizations of De Morgan’s laws.

Theorem 3.5.1

- (i) $\models \neg \forall x \varphi \leftrightarrow \exists x \neg \varphi,$
- (ii) $\models \neg \exists x \varphi \leftrightarrow \forall x \neg \varphi,$

- (iii) $\models \forall x \varphi \leftrightarrow \neg \exists x \neg \varphi$,
- (iv) $\models \exists x \varphi \leftrightarrow \neg \forall x \neg \varphi$.

Proof If there are no free variables involved, then the above equivalences are almost trivial. We will do one general case.

- (i) Let $FV(\forall x \varphi) = \{z_1, \dots, z_k\}$, then we must show $\mathfrak{A} \models \forall z_1 \dots z_k (\neg \forall x \varphi(x, z_1, \dots, z_k) \leftrightarrow \exists x \neg \varphi(x, z_1, \dots, z_k))$, for all \mathfrak{A} .

So we have to show $\mathfrak{A} \models \neg \forall x \varphi(x, \bar{a}_1, \dots, \bar{a}_k) \leftrightarrow \exists x \neg \varphi(x, \bar{a}_1, \dots, \bar{a}_k)$ for arbitrary $a_1, \dots, a_k \in |\mathfrak{A}|$. We apply the properties of \models as listed in Lemma 3.4.5:
 $\mathfrak{A} \models \neg \forall x \varphi(x, \bar{a}_1, \dots, \bar{a}_k) \leftrightarrow \mathfrak{A} \not\models \forall x \varphi(x, \bar{a}_1, \dots, \bar{a}_k) \leftrightarrow$ not for all $b \in |\mathfrak{A}|$ $\mathfrak{A} \models \varphi(\bar{b}, \bar{a}_1, \dots, \bar{a}_k) \leftrightarrow$ there is a $b \in |\mathfrak{A}|$ such that $\mathfrak{A} \models \neg \varphi(\bar{b}, \bar{a}_1, \dots, \bar{a}_k) \leftrightarrow \mathfrak{A} \models \exists x \neg \varphi(x, \bar{a}_1, \dots, \bar{a}_k)$.

- (ii) is similarly dealt with,
- (iii) can be obtained from (i), (ii),
- (iv) can be obtained from (i), (ii). □

The order of quantifiers of the same sort is irrelevant, and quantification over a variable that does not occur can be deleted.

Theorem 3.5.2

- (i) $\models \forall x \forall y \varphi \leftrightarrow \forall y \forall x \varphi$,
- (ii) $\models \exists x \exists y \varphi \leftrightarrow \exists y \exists x \varphi$,
- (iii) $\models \forall x \varphi \leftrightarrow \varphi$ if $x \notin FV(\varphi)$,
- (iv) $\models \exists x \varphi \leftrightarrow \varphi$ if $x \notin FV(\varphi)$.

Proof Left to the reader. □

We have already observed that \forall and \exists are, in a way, generalizations of \wedge and \vee . Therefore it is not surprising that \forall (resp. \exists) distributes over \wedge (resp. \vee). \forall (and \exists) distributes over \vee (resp. \wedge) only if a certain condition is met.

Theorem 3.5.3

- (i) $\models \forall x (\varphi \wedge \psi) \leftrightarrow \forall x \varphi \wedge \forall x \psi$,
- (ii) $\models \exists x (\varphi \vee \psi) \leftrightarrow \exists x \varphi \vee \exists x \psi$,
- (iii) $\models \forall x (\varphi(x) \vee \psi) \leftrightarrow \forall x \varphi(x) \vee \psi$ if $x \notin FV(\psi)$,
- (iv) $\models \exists x (\varphi(x) \wedge \psi) \leftrightarrow \exists x \varphi(x) \wedge \psi$ if $x \notin FV(\psi)$.

Proof (i) and (ii) are immediate.

- (iii) Let $FV(\forall x (\varphi(x) \vee \psi)) = \{z_1, \dots, z_k\}$. We must show that $\mathfrak{A} \models \forall z_1 \dots z_k [\forall x (\varphi(x) \vee \psi) \leftrightarrow \forall x \varphi(x) \vee \psi]$ for all \mathfrak{A} , so we show, using Lemma 3.4.5, that $\mathfrak{A} \models \forall x [\varphi(x, \bar{a}_1, \dots, \bar{a}_k) \vee \psi(\bar{a}_1, \dots, \bar{a}_k)] \leftrightarrow \mathfrak{A} \models \forall x \varphi(x, \bar{a}_1, \dots, \bar{a}_k) \vee \psi(\bar{a}_1, \dots, \bar{a}_k)$ for all \mathfrak{A} and all $a_1, \dots, a_k \in |\mathfrak{A}|$.

Note that in the course of the argument a_1, \dots, a_k remain fixed, so in the future we will no longer write them down.

\Leftarrow : $\mathcal{A} \models \forall x \varphi(x, \text{---}) \vee \psi(\text{---}) \Leftrightarrow \mathcal{A} \models \forall x \varphi(x, \text{---})$ or $\mathcal{A} \models \psi(\text{---})$
 $\Leftrightarrow \mathcal{A} \models \varphi(\bar{b}, \text{---})$ for all b or $\mathcal{A} \models \psi(\text{---})$.

If $\mathcal{A} \models \psi(\text{---})$, then also $\mathcal{A} \models \varphi(\bar{b}, \text{---}) \vee \psi(\text{---})$ for all b , and so
 $\mathcal{A} \models \forall x [\varphi(x, \text{---}) \vee \psi(\text{---})]$. If for all b $\mathcal{A} \models \varphi(\bar{b}, \text{---})$ then

$\mathcal{A} \models \varphi(\bar{b}, \text{---}) \vee \psi(\text{---})$ for all b , so $\mathcal{A} \models \forall x (\varphi(x, \text{---}) \vee \psi(\text{---}))$.

In both cases we get the desired result.

\Rightarrow : We know that for each $b \in |\mathcal{A}|$ $\mathcal{A} \models \varphi(\bar{b}, \text{---}) \vee \psi(\text{---})$.

If $\mathcal{A} \models \psi(\text{---})$, then also $\mathcal{A} \models \forall x \varphi(x, \text{---}) \vee \psi(\text{---})$, so we are done.

If $\mathcal{A} \not\models \psi(\text{---})$ then necessarily $\mathcal{A} \models \varphi(\bar{b}, \text{---})$ for all b , so

$\mathcal{A} \models \forall x \varphi(x, \text{---})$ and hence $\mathcal{A} \models \forall x \varphi(x, \text{---}) \vee \psi(\text{---})$.

(iv) is similar. □

In the proof above we have demonstrated a technique for dealing with the extra free variables z_1, \dots, z_k , that do not play an actual role. One chooses an arbitrary string of elements a_1, \dots, a_k to substitute for the z_i 's and keeps them fixed during the proof. So in the future we will mostly ignore the extra variables.

WARNING

$\forall x (\varphi(x) \vee \psi(x)) \rightarrow \forall x \varphi(x) \vee \forall x \psi(x)$, and
 $\exists x \varphi(x) \wedge \exists x \psi(x) \rightarrow \exists x (\varphi(x) \wedge \psi(x))$ are *not* true.

One of the Cinderella tasks in logic is the bookkeeping of substitution, keeping track of things in iterated substitution, etc. We will provide a number of useful lemmas; none of them is difficult—it is a mere matter of clerical labor.

A word of advice to the reader: none of these syntactical facts is hard to prove, nor is there a great deal to be learned from the proofs (unless one is after very specific goals, such as complexity of certain predicates); the best procedure is to give the proofs directly and only to look at the proofs in the book in case of emergency.

Lemma 3.5.4

- (i) Let x and y be distinct variables such that $x \notin FV(r)$, then $(t[s/x])[r/y] = (t[r/y])[s[r/y]/x]$,
- (ii) Let x and y be distinct variables such that $x \notin FV(s)$ and let t and s be free for x and y in φ , then $(\varphi[t/x])[s/y] = (\varphi[s/y])[t[s/y]/x]$,
- (iii) Let ψ be free for $\$$ in φ , and let t be free for x in φ and ψ , then $(\varphi[\psi/\$])[t/x] = (\varphi[t/x])[\psi[t/x]/\$]$,
- (iv) Let φ, ψ be free for $\$, \2 in σ , let ψ be free for $\$, \2 in φ , and let $\$, \1 not occur in ψ , then $(\sigma[\varphi/\$1])[\psi/\$2] = (\sigma[\psi/\$2])[\varphi[\psi/\$2]/\$1]$.

Proof (i) Induction on t .

- $t = c$, trivial.
- $t = x$. Then $t[s/x] = s$ and $(t[s/x])[r/y] = s[r/y]$; $(t[r/y])[s[r/y]/x] = x[s[r/y]/x] = s[r/y]$.

- $t = y$. Then $(t[s/x])[r/y] = y[r/y] = r$ and $(t[r/y])[s[r/y]/x] = r[s[r/y]/x] = r$, since $x \notin FV(r)$.
- $t = z$, where $z \neq x, y$, trivial.
- $t = f(t_1, \dots, t_n)$. Then $(t[s/x])[r/y] = (f(t_1[s/x], \dots))[r/y] = f((t_1[s/x])[r/y], \dots) \stackrel{\text{i.h.}}{=} f((t_1[r/y])[s[r/y]/x], \dots) = f(t_1[r/y], \dots)[s[r/y]/x] = (t[r/y])[s[r/y]/x]$.¹

(ii) Induction on φ . Left to the reader.

(iii) Induction on φ .

- $\varphi = \perp$ or P distinct from $\$$. Trivial.
- $\varphi = \$$. Then $(\$[\psi/\$])[t/x] = \psi[t/x]$ and $(\$[t/x])[\psi[t/x]/\$] = \$[\psi[t/x]/\$] = \psi[t/x]$.
- $\varphi = \varphi_1 \Box \varphi_2, \neg \varphi_1$. Trivial.
- $\varphi = \forall y \varphi_1$. Then $(\forall y \cdot \varphi_1[\psi/\$])[t/x] = (\forall y \cdot \varphi_1[\psi/\$])[t/x] = \forall y \cdot ((\varphi_1[\psi/\$])[t/x]) \stackrel{\text{i.h.}}{=} \forall y ((\varphi_1[t/x])[\psi[t/x]/\$]) = ((\forall y \varphi_1)[t/x])[\psi[t/x]/\$]$. $\varphi = \exists y \varphi_1$. Idem.

(iv) Induction on σ . Left to the reader. \square

We immediately get the following corollary.

Corollary 3.5.5

- (i) If $z \notin FV(t)$, then $t[\bar{a}/x] = (t[z/x])[\bar{a}/z]$,
- (ii) If $z \notin FV(\varphi)$ and z is free for x in φ , then $\varphi[\bar{a}/x] = (\varphi[z/x])[\bar{a}/z]$.

It is possible to pull out quantifiers from a formula. The trick is well known in analysis: the bound variable in an integral may be changed, e.g. $\int x dx + \int \sin y dy = \int x dx + \int \sin x dx = \int (x + \sin x) dx$. In predicate logic we have a similar phenomenon.

Theorem 3.5.6 (Change of Bound Variables) *If x, y are free for z in φ and $x, y \notin FV(\varphi)$, then $\models \exists x \varphi[x/z] \leftrightarrow \exists y \varphi[y/z], \models \forall x \varphi[x/z] \leftrightarrow \forall y \varphi[y/z]$.*

Proof It suffices to consider φ with $FV(\varphi) \subseteq \{z\}$. We have to show $\mathfrak{A} \models \exists x \varphi[x/z] \leftrightarrow \mathfrak{A} \models \exists y \varphi[y/z]$ for any \mathfrak{A} . $\mathfrak{A} \models \exists x \varphi[x/z] \leftrightarrow \mathfrak{A} \models (\varphi[x/z])[\bar{a}/x]$ for some $a \leftrightarrow \mathfrak{A} \models \varphi[\bar{a}/z]$ for some $a \leftrightarrow \mathfrak{A} \models (\varphi[y/z])[\bar{a}/y]$ for some $a \leftrightarrow \mathfrak{A} \models \exists y \varphi[y/z]$.

The universal quantifier is handled completely similarly. \square

The upshot of this theorem is that one can always replace a bound variable by a “fresh” one, i.e. one that did not occur in the formula. Now one easily concludes the following.

Corollary 3.5.7 *Every formula is equivalent to one in which no variable occurs both free and bound.*

¹“i.h.” indicates the use of the induction hypothesis.

We now can pull out quantifiers: $\forall x\varphi(x) \vee \forall x\psi(x) \leftrightarrow \forall x\varphi(x) \vee \forall y\psi(y)$ and $\forall x\varphi(x) \vee \forall y\psi(y) \leftrightarrow \forall xy(\varphi(x) \vee \psi(y))$, for a suitable y .

In order to handle predicate logic in an algebraic way we need the technique of substituting equivalents for equivalents.

Theorem 3.5.8 (Substitution Theorem)

- (i) $\models t_1 = t_2 \rightarrow s[t_1/x] = s[t_2/x]$,
- (ii) $\models t_1 = t_2 \rightarrow (\varphi[t_1/x] \leftrightarrow \varphi[t_2/x])$,
- (iii) $\models (\varphi \leftrightarrow \psi) \rightarrow (\sigma[\varphi/\$] \leftrightarrow \sigma[\psi/\$])$.

Proof It is no restriction to assume that the terms and formulas are closed. We tacitly assume that the substitutions satisfy the “free for” conditions.

- (i) Let $\mathfrak{A} \models t_1 = t_2$, i.e. $t_1^{\mathfrak{A}} = t_2^{\mathfrak{A}}$. Now use induction on s .

- s is a constant or a variable. Trivial.
- $s = \overline{F}(s_1, \dots, s_k)$. Then $s[t_i/x] = \overline{F}(s_1[t_i/x], \dots)$ and $(s[t_i/x])^{\mathfrak{A}} = F((s_1[t_i/x])^{\mathfrak{A}}, \dots)$. Induction hypothesis: $(s_j[t_1/x])^{\mathfrak{A}} = (s_j[t_2/x])^{\mathfrak{A}}$, $1 \leq j \leq k$. So $(s[t_1/x])^{\mathfrak{A}} = F((s_1[t_1/x])^{\mathfrak{A}}, \dots) = F((s_1[t_2/x])^{\mathfrak{A}}, \dots) = (s[t_2/x])^{\mathfrak{A}}$. Hence $\mathfrak{A} \models s[t_1/x] = s[t_2/x]$.

- (ii) Let $\mathfrak{A} \models t_1 = t_2$, so $t_1^{\mathfrak{A}} = t_2^{\mathfrak{A}}$. We show $\mathfrak{A} \models \varphi[t_1/x] \Leftrightarrow \mathfrak{A} \models \varphi[t_2/x]$ by induction on φ .

- φ is atomic. The case of a propositional symbol (including \perp) is trivial. So consider $\varphi = \overline{P}(s_1, \dots, s_k)$. $\mathfrak{A} \models \overline{P}(s_1, \dots, s_k)[t_1/x] \Leftrightarrow \mathfrak{A} \models \overline{P}(s_1[t_1/x], \dots) \Leftrightarrow \langle (s_1[t_1/x])^{\mathfrak{A}}, \dots, (s_k[t_1/x])^{\mathfrak{A}} \rangle \in P$. By (i) $(s_j[t_1/x])^{\mathfrak{A}} = (s_j[t_2/x])^{\mathfrak{A}}$, $j = 1, \dots, k$.

So we get $\langle (s_1[t_1/x])^{\mathfrak{A}}, \dots \rangle \in P \Leftrightarrow \dots \Leftrightarrow \mathfrak{A} \models \overline{P}(s_1, \dots)[t_2/x]$.

- $\varphi = \varphi_1 \vee \varphi_2$, $\varphi_1 \wedge \varphi_2$, $\varphi_1 \rightarrow \varphi_2$, $\neg\varphi_1$. We consider the disjunction: $\mathfrak{A} \models (\varphi_1 \vee \varphi_2)[t_1/x] \Leftrightarrow \mathfrak{A} \models \varphi_1[t_1/x] \text{ or } \mathfrak{A} \models \varphi_2[t_1/x] \stackrel{\text{i.h.}}{\Leftrightarrow} \mathfrak{A} \models \varphi_1[t_2/x] \text{ or } \mathfrak{A} \models \varphi_2[t_2/x] \Leftrightarrow \mathfrak{A} \models (\varphi_1 \vee \varphi_2)[t_2/x]$.

The remaining connectives are treated similarly.

- $\varphi = \exists y\psi$, $\varphi = \forall y\psi$.

We consider the existential quantifier. $\mathfrak{A} \models (\exists y\psi)[t_1/x] \Leftrightarrow \mathfrak{A} \models \exists y(\psi[t_1/x]) \Leftrightarrow \mathfrak{A} \models \psi[t_1/x][\overline{a}/y]$ for some a .

By Lemma 3.5.4 $\mathfrak{A} \models \psi[t_1/x][\overline{a}/y] \Leftrightarrow \mathfrak{A} \models (\psi[\overline{a}/y])[t_1[\overline{a}/y]/x]$. Apply the induction hypothesis to $\psi[\overline{a}/y]$ and the terms $t_1[\overline{a}/y]$, $t_2[\overline{a}/y]$. Observe that t_1 and t_2 are closed, so $t_1[\overline{a}/y] = t_1$ and $t_2 = t_2[\overline{a}/y]$. We get $\mathfrak{A} \models \psi[t_2/x][\overline{a}/y]$, and hence $\mathfrak{A} \models \exists y\psi[t_2/x]$. The other implication is similar, and so is the case of the universal quantifier.

- (iii) Let $\mathfrak{A} \models \varphi \Leftrightarrow \mathfrak{A} \models \psi$. We show $\mathfrak{A} \models \sigma[\varphi/\$] \Leftrightarrow \mathfrak{A} \models \sigma[\psi/\$]$ by induction on σ .

- σ is atomic. Both cases $\sigma = \$$ and $\sigma \neq \$$ are trivial.
- $\sigma = \sigma_1 \Box \sigma_2$ (or $\neg\sigma_1$). Left to the reader.

- $\sigma = \forall x \cdot \tau$. Observe that φ and ψ are closed, but even if they were not then x could not occur free in φ, ψ .
 $\mathfrak{A} \models (\forall x \cdot \tau)[\varphi/\$] \Leftrightarrow \mathfrak{A} \models \forall x(\tau[\varphi/\$])$. Pick an $a \in |\mathfrak{A}|$, then $\mathfrak{A} \models (\tau[\varphi/\$])[\bar{a}/x] \stackrel{3.5.4}{\Leftrightarrow} \mathfrak{A} \models (\tau[\bar{a}/x])[\varphi[\bar{a}/x]/\$] \Leftrightarrow \mathfrak{A} \models (\tau[\bar{a}/x])[\varphi/\$] \stackrel{i.h.}{\Leftrightarrow} \mathfrak{A} \models \tau[\bar{a}/x][\psi/\$] \Leftrightarrow \mathfrak{A} \models \tau[\bar{a}/x][\psi[\bar{a}/x]/\$] \Leftrightarrow \mathfrak{A} \models (\tau[\psi/\$])[\bar{a}/x]$. Hence $\mathfrak{A} \models \sigma[\varphi/\$] \Leftrightarrow \mathfrak{A} \models \sigma[\psi/\$]$. The existential quantifier is treated similarly. \square

Observe that in the above proof we have applied induction to “ $\sigma[\varphi/\$]$ for all φ ”, because the substitution formula changed during the quantifier case.

Note that also the σ changed, so properly speaking we are applying induction to the rank (or we have to formulate the induction principle 3.3.4 a bit more liberally).

Corollary 3.5.9

- (i) $\llbracket s[t/x] \rrbracket = \llbracket s \rrbracket \llbracket t \rrbracket / x$,
- (ii) $\llbracket \varphi[t/x] \rrbracket = \llbracket \varphi \rrbracket \llbracket t \rrbracket / x$.

Proof We apply the Substitution Theorem. Consider an arbitrary \mathfrak{A} . Note that $\llbracket \llbracket t \rrbracket \rrbracket = \llbracket t \rrbracket$ (by definition), so $\mathfrak{A} \models \llbracket t \rrbracket = t$. Now (i) and (ii) follow immediately. \square

In a more relaxed notation, we can write (i) and (ii) as $\llbracket s(t) \rrbracket = \llbracket s(\llbracket t \rrbracket) \rrbracket$, or $\mathfrak{A} \models s(t) = s(\llbracket t \rrbracket)$ and $\llbracket \varphi(t) \rrbracket = \llbracket \varphi(\llbracket t \rrbracket) \rrbracket$, or $\mathfrak{A} \models \varphi(t) \leftrightarrow \varphi(\llbracket t \rrbracket)$.

Observe that $\llbracket t \rrbracket (= \llbracket t \rrbracket_{\mathfrak{A}})$ is just another way to write $t^{\mathfrak{A}}$.

Proofs involving detailed analysis of substitution are rather dreary but, unfortunately, unavoidable. The reader may simplify the above and other proofs by supposing the formulas involved to be closed. There is no real loss in generality, since we only introduce a number of constants from $L(\mathfrak{A})$ and check that the result is valid for all choices of constants.

Now we really can manipulate formulas in an algebraic way. Again, write $\varphi \approx \psi$ for $\models \varphi \leftrightarrow \psi$.

Examples

1. $\forall x \varphi(x) \rightarrow \psi \approx \neg \forall x \varphi(x) \vee \psi \approx \exists x (\neg \varphi(x)) \vee \psi \approx \exists x (\neg \varphi(x) \vee \psi) \approx \exists x (\varphi(x) \rightarrow \psi)$, where $x \notin FV(\psi)$.
2. $\forall x \varphi(x) \rightarrow \exists x \varphi(x) \approx \neg \forall x \varphi(x) \vee \exists x \varphi(x) \approx \exists x (\neg \varphi(x) \vee \varphi(x))$. The formula in the scope of the quantifier is true (already by propositional logic), so the formula itself is true.

Definition 3.5.10 A formula φ is in *prenex (normal) form* if φ consists of a (possibly empty) string of quantifiers followed by an open (i.e. quantifier free) formula. We also say that φ is a prenex formula.

Examples $\exists x \forall y \exists z \exists v (x = z \vee y = z \rightarrow v < y)$, $\forall x \forall y \exists z (P(x, y) \wedge Q(y, x) \rightarrow P(z, z))$.

By pulling out quantifiers we can reduce each formula to a formula in prenex form.

Theorem 3.5.11 *For each φ there is a prenex formula ψ such that $\models \varphi \leftrightarrow \psi$.*

Proof First eliminate \rightarrow and \leftrightarrow . Use induction on the resulting formula φ' .

For atomic φ' the theorem is trivial. If $\varphi' = \varphi_1 \vee \varphi_2$ and φ_1, φ_2 are equivalent to prenex ψ_1, ψ_2 then

$$\begin{aligned}\psi_1 &= (Q_1 y_1) \cdots (Q_n y_n) \psi^1, \\ \psi_2 &= (Q'_1 z_1) \cdots (Q'_m z_m) \psi^2,\end{aligned}$$

where Q_i, Q'_j are quantifiers and ψ^1, ψ^2 open. By Theorem 3.5.6 we can choose all bound variables distinct, taking care that no variable is both free and bound. Applying Theorem 3.5.3 we find

$$\models \varphi' \leftrightarrow (Q_1 y_1) \cdots (Q_n y_n) (Q'_1 z_1) \cdots (Q'_m z_m) (\psi^1 \vee \psi^2),$$

so we are done.

The remaining cases are left to the reader. □

In ordinary mathematics it is usually taken for granted that the benevolent reader can guess the intentions of the author, not only the explicit ones, but also the ones that are tacitly handed down generations of mathematicians. Take for example the definition of convergence of a sequence: $\forall \varepsilon > 0 \exists n \forall m (|a_n - a_{n+m}| < \varepsilon)$. In order to make sense out of this expression one has to add: the variables n, m range over natural numbers. Unfortunately our syntax does not allow for variables of different sorts. So how do we incorporate expressions of the above kind? The answer is simple: we add predicates of the desired sort and indicate inside the formula the “nature” of the variable.

Example Let $\mathfrak{A} = \langle R, Q, < \rangle$ be the structure of the reals with the set of rational numbers singled out, provided with the natural order. The sentence $\sigma := \forall xy (x < y \rightarrow \exists z (Q(z) \wedge x < z \wedge z < y))$ can be interpreted in $\mathfrak{A} : \mathfrak{A} \models \sigma$, and it tells us that the rationals are dense in the reals (in the natural ordering). We find this mode of expression, however, rather cumbersome. Therefore we introduce the notion of *relativized quantifiers*. Since it does not matter whether we express informally “ x is rational” by $x \in Q$ or $Q(x)$, we will suit ourselves and at any time choose the notation which is most convenient. We use $(\exists x \in Q)$ and $(\forall x \in Q)$ as informal notation for “there exists an x in Q ” and “for all x in Q ”. Now we can write σ as $\forall xy (x < y \rightarrow \exists z \in Q (x < z \wedge z < y))$. Note that we do *not* write $(\forall xy \in R)(\text{---})$, since: (1) there is no relation R in \mathfrak{A} , (2) variables automatically range over $|\mathfrak{A}| = R$.

Let us now define the relativization of a quantifier properly.

Definition 3.5.12 If P is a unary predicate symbol, then $(\forall x \in P)\varphi := \forall x(P(x) \rightarrow \varphi)$, $(\exists x \in P)\varphi := (\exists x)(P(x) \wedge \varphi)$.

This notation has the intended meaning, as appears from $\mathfrak{A} \models (\forall x \in P)\varphi \Leftrightarrow$ for all $a \in P^{\mathfrak{A}}$ $\mathfrak{A} \models \varphi[\bar{a}/x]$, $\mathfrak{A} \models (\exists x \in P)\varphi \Leftrightarrow$ there exists an $a \in P^{\mathfrak{A}}$ such that $\mathfrak{A} \models \varphi[\bar{a}/x]$. The proof is immediate.

We will often use informal notation, such as $(\forall x > 0)$ or $(\exists y \neq 1)$, which can be cast into the above form. The meaning of such notation will always be evident. One can restrict *all* quantifiers to the same set (predicate), which amounts to passing to a restricted universe (cf. Exercise 11).

It is a common observation that by strengthening a part of a conjunction (disjunction) the whole formula is strengthened, but that by strengthening φ in $\neg\varphi$ the whole formula is weakened. This phenomenon has a syntactic origin, and we will introduce a bit of terminology to handle it smoothly. We inductively define that a subformula occurrence φ is positive (negative) in σ .

Definition 3.5.13 Sub^+ and Sub^- are defined simultaneously by

$$\begin{aligned} Sub^+(\varphi) &= \{\varphi\} \\ Sub^-(\varphi) &= \emptyset \quad \text{for atomic } \varphi \\ Sub^+(\varphi_1 \square \varphi_2) &= Sub^+(\varphi_1) \cup Sub^+(\varphi_2) \cup \{\varphi_1 \square \varphi_2\} \\ Sub^-(\varphi_1 \square \varphi_2) &= Sub^-(\varphi_1) \cup Sub^-(\varphi_2) \quad \text{for } \square \in \{\wedge, \vee\} \\ Sub^+(\varphi_1 \rightarrow \varphi_2) &= Sub^-(\varphi_1) \cup Sub^+(\varphi_2) \cup \{\varphi_1 \rightarrow \varphi_2\} \\ Sub^-(\varphi_1 \rightarrow \varphi_2) &= Sub^+(\varphi_1) \cup Sub^-(\varphi_2) \\ Sub^+(Qx.\varphi) &= Sub^+(\varphi) \cup \{Qx.\varphi\} \\ Sub^-(Qx.\varphi) &= Sub^-(\varphi) \quad \text{for } Q \in \{\forall, \exists\}. \end{aligned}$$

If $\varphi \in Sub^+(\psi)$, then we say that φ *occurs positively* in ψ (similarly for negative occurrences).

We could have restricted ourselves to \wedge , \rightarrow and \forall , but it does not ask much extra space to handle the other connectives.

The following theorem makes the basic intuition clear: if a positive part of a formula increases in truth value then the formula increases in truth value (better: does not decrease in truth value). We express this role of positive and negative subformulas as follows.

Theorem 3.5.14 Let φ (ψ) not occur negatively (not positively) in σ , then:

- (i) $\llbracket \varphi_1 \rrbracket \leq \llbracket \varphi_2 \rrbracket \Rightarrow \llbracket \sigma[\varphi_1/\varphi] \rrbracket \leq \llbracket \sigma[\varphi_2/\varphi] \rrbracket$,
- (ii) $\llbracket \psi_1 \rrbracket \leq \llbracket \psi_2 \rrbracket \Rightarrow \llbracket \sigma[\psi_1/\psi] \rrbracket \geq \llbracket \sigma[\psi_2/\psi] \rrbracket$,
- (iii) $\mathfrak{A} \models (\varphi_1 \rightarrow \varphi_2) \rightarrow (\sigma[\varphi_1/\varphi] \rightarrow \sigma[\varphi_2/\varphi])$,
- (iv) $\mathfrak{A} \models (\psi_1 \rightarrow \psi_2) \rightarrow (\sigma[\psi_2/\psi] \rightarrow \sigma[\psi_1/\psi])$.

Proof Induction on σ . □

Exercises

1. Show that all propositional tautologies are true in all structures (of the right similarity type).
2. Let $x \notin FV(\psi)$. Show
 - (i) $\models (\forall x\varphi \rightarrow \psi) \leftrightarrow \exists x(\varphi \rightarrow \psi)$,
 - (ii) $\models (\exists x\varphi \rightarrow \psi) \leftrightarrow \forall x(\varphi \rightarrow \psi)$,
 - (iii) $\models (\psi \rightarrow \exists x\varphi) \leftrightarrow \exists x(\psi \rightarrow \varphi)$,
 - (iv) $\models (\psi \rightarrow \forall x\varphi) \leftrightarrow \forall x(\psi \rightarrow \varphi)$.
3. Show that the condition on $FV(\psi)$ in Exercise 2 is necessary.
4. Show $\not\models \forall x\exists y\varphi \leftrightarrow \exists y\forall x\varphi$.
5. Show $\models \varphi \Rightarrow \models \forall x\varphi$ and $\models \exists x\varphi$.
6. Show $\not\models \exists x\varphi \rightarrow \forall x\varphi$.
7. Show $\not\models \exists x\varphi \wedge \exists x\psi \rightarrow \exists x(\varphi \wedge \psi)$.
8. Show that the condition on x, y in Theorem 3.5.6 is necessary.
9. Show
 - (i) $\models \forall x(\varphi \rightarrow \psi) \rightarrow (\forall x\varphi \rightarrow \forall x\psi)$;
 - (ii) $\models (\exists x\varphi \rightarrow \exists x\psi) \rightarrow \exists x(\varphi \rightarrow \psi)$;
 - (iii) $\models \forall x(\varphi \leftrightarrow \psi) \rightarrow (\forall x\varphi \leftrightarrow \forall x\psi)$;
 - (iv) $\models (\forall x\varphi \rightarrow \exists x\psi) \leftrightarrow \exists x(\varphi \rightarrow \psi)$;
 - (v) $\models (\exists x\varphi \rightarrow \forall x\psi) \rightarrow \forall x(\varphi \rightarrow \psi)$.
10. Show that the converses of Exercises 9 (i)–(iii) and (v) do not hold.
11. Let L have a unary predicate P . Define the relativization σ^P of σ by

$$\begin{aligned}
 \sigma^P &:= \sigma \quad \text{for atomic } \varphi, \\
 (\varphi \Box \psi)^P &:= \varphi^P \Box \psi^P, \\
 (\neg \varphi)^P &:= \neg \varphi^P, \\
 (\forall x\varphi)^P &:= \forall x(P(x) \rightarrow \varphi^P), \\
 (\exists x\varphi)^P &:= \exists x(P(x) \wedge \varphi^P).
 \end{aligned}$$

Let \mathfrak{A} be a structure without functions and constants. Consider the structure \mathfrak{B} with universe $P^{\mathfrak{A}}$ and relations which are restrictions of the relations of \mathfrak{A} , where $P^{\mathfrak{A}} \neq \emptyset$. Show $\mathfrak{A} \models \sigma^P \Leftrightarrow \mathfrak{B} \models \sigma$ for sentences σ . Why are only relations allowed in \mathfrak{A} ?

12. Let S be a binary predicate symbol. Show $\models \neg \exists y \forall x (S(y, x) \leftrightarrow \neg S(x, x))$. (Think of “ y shaves x ” and recall Russell’s barber paradox.)
13. (i) Show that the “free for” conditions cannot be dropped from Theorem 3.5.8.
 (ii) Show $\models t = s \Rightarrow \models \varphi[t/x] \leftrightarrow \varphi[s/x]$.
 (iii) Show $\models \varphi \leftrightarrow \psi \Rightarrow \models \sigma[\varphi/\$] \leftrightarrow \sigma[\psi/\$]$.
14. Find prenex normal forms for
 - (a) $\neg((\neg \forall x\varphi(x) \vee \forall x\psi(x)) \wedge (\exists x\sigma(x) \rightarrow \forall x\tau(x)))$,
 - (b) $\forall x\varphi(x) \leftrightarrow \exists x\psi(x)$,
 - (c) $\neg(\exists x\varphi(x, y) \wedge (\forall y\psi(y) \rightarrow \varphi(x, y)) \rightarrow \exists x\forall y\sigma(x, y))$,
 - (d) $((\forall x\varphi(x) \rightarrow \exists y\psi(x, y)) \rightarrow \psi(x, x)) \rightarrow \exists x\forall y\sigma(x, y)$.
15. Show $\models \exists x(\varphi(x) \rightarrow \forall y\varphi(y))$. (It is instructive to think of $\varphi(x)$ as “ x drinks”.)

3.6 Identity

We have limited ourselves in this book to the consideration of structures with identity (see Definition 3.4.2), and hence of languages with identity. Therefore we classified “=” as a logical symbol, rather than a mathematical one. We can, however, not treat = as just some binary predicate, since identity satisfies a number of characteristic axioms, listed below.

- $I_1 \quad \forall x(x = x),$
- $I_2 \quad \forall xy(x = y \rightarrow y = x),$
- $I_3 \quad \forall xyz(x = y \wedge y = z \rightarrow x = z),$
- $I_4 \quad \forall x_1 \dots x_n y_1 \dots y_n (\bigwedge_{i \leq n} x_i = y_i \rightarrow t(x_1, \dots, x_n) = t(y_1, \dots, y_n)),$
 $\forall x_1 \dots x_n y_1 \dots y_n (\bigwedge_{i \leq n} x_i = y_i \rightarrow (\varphi(x_1, \dots, x_n) \rightarrow \varphi(y_1, \dots, y_n))).$

One simply checks that I_1, I_2, I_3 are true in all structures \mathfrak{A} . For I_4 , observe that we can suppose the formulas to be closed. Otherwise we add quantifiers for the remaining variables and add dummy identities, e.g. $\forall z_1 \dots z_k x_1 \dots x_n y_1 \dots y_n (\bigwedge_{i \leq n} x_i = y_i \wedge \bigwedge_{i \leq k} z_i = z_i \rightarrow t(x_1, \dots, x_n) = t(y_1, \dots, y_n))$. Now $(t(\bar{a}_1, \dots, \bar{a}_n))^{\mathfrak{A}}$ defines a function $t^{\mathfrak{A}}$ on $|\mathfrak{A}|^n$, obtained from the given functions of \mathfrak{A} by various substitutions, hence $a_i = b_i (i \leq n) \Rightarrow (t(\bar{a}_1, \dots, \bar{a}_n))^{\mathfrak{A}} = (t(\bar{b}_1, \dots, \bar{b}_n))^{\mathfrak{A}}$. This establishes the first part of I_4 .

The second part is proved by induction on φ (using the first part): e.g. consider the universal quantifier case and let $a_i = b_i$ for all $i \leq n$. $\mathfrak{A} \models \forall u \varphi(u, \bar{a}_1, \dots, \bar{a}_n) \Leftrightarrow \mathfrak{A} \models \varphi(\bar{c}, \bar{a}_1, \dots, \bar{a}_n)$ for all $c \stackrel{\text{i.h.}}{\Leftrightarrow} \mathfrak{A} \models \varphi(\bar{c}, \bar{b}_1, \dots, \bar{b}_n)$ for all $c \Leftrightarrow \mathfrak{A} \models \forall u \varphi(u, \bar{b}_1, \dots, \bar{b}_n)$. So $\mathfrak{A} \models (\bigwedge_{i \leq n} \bar{a}_i = \bar{b}_i) \Rightarrow \mathfrak{A} \models \forall u \varphi(u, \bar{a}_1, \dots, \bar{a}_n) \rightarrow \forall u \varphi(u, \bar{b}_1, \dots, \bar{b}_n)$. This holds for all $a_1, \dots, a_n, b_1, \dots, b_n$, hence $\mathfrak{A} \models \forall x_1, \dots, x_n y_1 \dots y_n (\bigwedge_{i \leq n} x_i = y_i \rightarrow (\forall u \varphi(u, x_1, \dots, x_n) \rightarrow \forall u \varphi(u, y_1, \dots, y_n)))$.

Note that φ (respectively t) in I_4 can be any formula (respectively term), so I_4 stands for infinitely many axioms. We call such an “instant axiom” an *axiom schema*.

The first three axioms state that identity is an equivalence relation. I_4 states that identity is a congruence with respect to all (definable) relations.

It is important to realize that from the axioms alone, we cannot determine the precise nature of the interpreting relation. We explicitly adopt the convention that “=” will always be interpreted by real equality. Inequality, $\neg x = y$, is abbreviated as $x \neq y$.

Exercises

1. Show $\models \forall x \exists y (x = y)$.
2. Show $\models \forall x (\varphi(x) \leftrightarrow \exists y (x = y \wedge \varphi(y)))$ and $\models \forall x (\varphi(x) \leftrightarrow \forall y (x = y \rightarrow \varphi(y)))$, where y does not occur in $\varphi(x)$.
3. Show that $\models \varphi(t) \leftrightarrow \forall x (x = t \rightarrow \varphi(x))$ if $x \notin FV(t)$.
4. Show that the conditions in Exercises 2 and 3 are necessary.
5. Consider $\sigma_1 = \forall x (x \sim x)$, $\sigma_2 = \forall xy (x \sim y \rightarrow y \sim x)$, $\sigma_3 = \forall xyz (x \sim y \wedge y \sim z \rightarrow x \sim z)$. Show that if $\mathfrak{A} \models \sigma_1 \wedge \sigma_2 \wedge \sigma_3$, where $\mathfrak{A} = \langle A, R \rangle$, then R is an equivalence relation. N.B. $x \sim y$ is a suggestive notation for the atom $\bar{R}(x, y)$.

6. Let $\sigma_4 = \forall xyz(x \sim y \wedge x \sim z \rightarrow y \sim z)$. Show that $\sigma_1, \sigma_4 \models \sigma_2 \wedge \sigma_3$.
7. Consider the schema $\sigma_5 : x \sim y \rightarrow (\varphi[x/z] \rightarrow \varphi[y/z])$. Show that $\sigma_1, \sigma_5 \models \sigma_2 \wedge \sigma_3$. N.B. If σ is a schema, then $\Delta \cup \{\sigma\} \models \varphi$ stands for $\Delta \cup \Sigma \models \varphi$, where Σ consists of all instances of σ .
8. Derive the term version of I_4 from the formula version.

3.7 Examples

We will consider languages for some familiar kinds of structures. Since all languages are built in the same way, we shall not list the logical symbols. All structures are supposed to satisfy the identity axioms $I_1 - I_4$.

For a refinement see Lemma 3.10.2.

1. *The language of identity.* Type: $\langle -; -; 0 \rangle$.

Alphabet.

Predicate symbol: =

The structures of this type are of the form $\mathfrak{A} = \langle A \rangle$, and satisfy I_1, I_2, I_3 . (In this language I_4 follows from I_1, I_2, I_3 , cf. Sect. 3.10 Exercise 5.)

In an identity structure there is so little “structure”, that all one can virtually do is look for the number of elements (cardinality). There are sentences λ_n and μ_n saying that there are at least (or at most) n elements (Exercise 3, Sect. 4.1)

$$\lambda_n := \exists y_1 \dots y_n \bigwedge_{i \neq j} y_i \neq y_j \quad (n > 1),$$

$$\mu_n := \forall y_0 \dots y_n \bigvee_{i \neq j} y_i = y_j \quad (n > 0).$$

So $\mathfrak{A} \models \lambda_n \wedge \mu_n$ iff $|\mathfrak{A}|$ has exactly n elements. Since universes are not empty $\models \exists x(x = x)$ always holds.

We can also formulate “*there exists a unique x such that ...*”.

Definition 3.7.1 $\exists! x \varphi(x) := \exists x(\varphi(x) \wedge \forall y(\varphi(y) \rightarrow x = y))$, where y does not occur in $\varphi(x)$.

Note that $\exists! x \varphi(x)$ is an (informal) abbreviation.

2. *The language of partial order.* Type: $\langle 2; -; 0 \rangle$.

Alphabet.

Predicate symbols: =, \leq

Abbreviations

$$\begin{aligned} x \neq y &:= \neg x = y, & x < y &:= x \leq y \wedge x \neq y, \\ x > y &:= y < x, & x \geq y &:= y \leq x, \\ x \leq y \leq z &:= x \leq y \wedge y \leq z. \end{aligned}$$

Definition 3.7.2 \mathfrak{A} is a partially ordered set (poset) if \mathfrak{A} is a model of

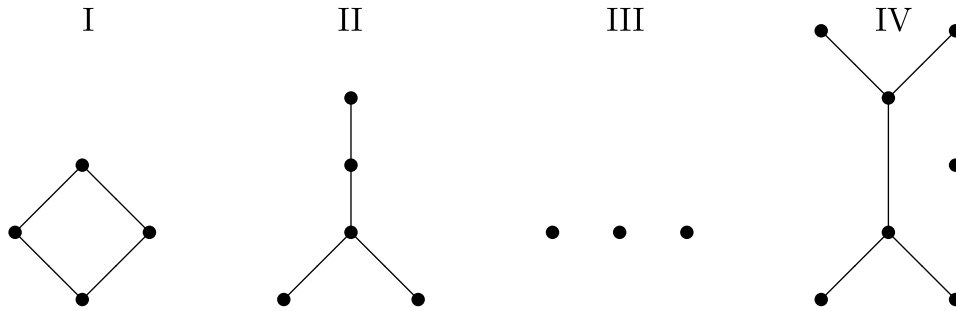
$$\begin{aligned} \forall xyz(x \leq y \leq z \rightarrow x \leq z), \\ \forall xy(x \leq y \leq x \leftrightarrow x = y). \end{aligned}$$

The notation may be misleading, since one usually introduces the relation \leq (e.g. on the reals) as a disjunction: $x < y$ or $x = y$. In our alphabet the relation is primitive; another symbol might have been preferable, but we chose to observe the tradition. Note that the relation is reflexive: $x \leq x$.

Partially ordered sets are very basic in mathematics, appearing in many guises. It is often convenient to visualize posets by means of diagrams, where $a \leq b$ is represented as equal or above (respectively to the right). One of the traditions in logic is to keep objects and their names apart. Thus we speak of function symbols which are interpreted as functions, etc. However, in practice this is a bit cumbersome. We prefer to use the same notation for the syntactic objects and their interpretations, e.g. if $\mathfrak{R} = \langle \mathbb{R}, \leq \rangle$ is the partially ordered set of reals, then $\mathfrak{R} \models \forall x \exists y(x \leq y)$, whereas it should be something like $\forall x \exists y(x \overline{\leq} y)$ to distinguish the symbol from the relation.

The “ \leq ” in \mathfrak{R} stands for the actual relation and the “ \leq ” in the sentence stands for the predicate symbol. The reader is urged to distinguish symbols in their various guises.

We show some diagrams of posets.



From the diagrams we can easily read off a number of properties. For example, $\mathfrak{A}_1 \models \exists x \forall y(x \leq y)$ (\mathfrak{A}_i is the structure with the diagram of Fig. I), i.e. \mathfrak{A}_1 has a least element (a *minimum*). $\mathfrak{A}_3 \models \forall x \neg \exists y(x < y)$, i.e. in \mathfrak{A}_3 no element is strictly less than another element.

Definition 3.7.3

- (i) \mathfrak{A} is a (*linearly* or *totally*) *ordered set* if it is a poset and $\mathfrak{A} \models \forall xy(x \leq y \vee y \leq x)$ (each two elements are comparable).
- (ii) \mathfrak{A} is *densely ordered* if $\mathfrak{A} \models \forall xy(x < y \rightarrow \exists z(x < z \wedge z < y))$ (between any two elements there is a third one).

It is a moderately amusing exercise to find sentences that distinguish between structures and vice versa. For example, we can distinguish \mathfrak{A}_3 and \mathfrak{A}_4 (from the

diagram above) as follows: in \mathfrak{A}_4 there is precisely *one* element that is incomparable with all other elements, in \mathfrak{A}_3 there are more such elements. Put $\sigma(x) := \forall y(y \neq x \rightarrow \neg y \leq x \wedge \neg x \leq y)$. Then $\mathfrak{A}_4 \models \forall xy(\sigma(x) \wedge \sigma(y) \rightarrow x = y)$, but $\mathfrak{A}_3 \models \neg \forall xy(\sigma(x) \wedge \sigma(y) \rightarrow x = y)$.

3. *The language of groups.* Type: $\langle -, 2, 1; 1 \rangle$.

Alphabet.

Predicate symbol: $=$

Function symbols: $\cdot, ^{-1}$

Constant symbol: e

Notation In order to conform with practice we write $t \cdot s$ and t^{-1} instead of $\cdot(t, s)$ and $^{-1}(t)$.

Definition 3.7.4 \mathfrak{A} is a *group* if it is a model of

$$\forall xyz((x \cdot y) \cdot z = x \cdot (y \cdot z)),$$

$$\forall x(x \cdot e = x \wedge e \cdot x = x),$$

$$\forall x(x \cdot x^{-1} = e \wedge x^{-1} \cdot x = e).$$

When convenient, we will write ts for $t \cdot s$; we will adopt the bracket conventions from algebra. A group \mathfrak{A} is *commutative* or *abelian* if $\mathfrak{A} \models \forall xy(xy = yx)$.

Commutative groups are often described in the language of *additive* groups, which have the following alphabet:

Predicate symbol: $=$

Function symbols: $+, -$

Constant symbol: 0

4. *The language of plane projective geometry.* Type: $\langle 2; -, 0 \rangle$

The structures one considers are projective planes, which are usually taken to consist of *points* and *lines* with an *incidence relation*. In this approach the type would be $\langle 1, 1, 2; -, 0 \rangle$. We can, however, use a more simple type, since a point can be defined as something that is incident with a line, and a line as something for which we can find a point which is incident with it. Of course this requires a non-symmetric incidence relation.

We will now list the axioms, which deviate somewhat from the traditional set. It is a simple exercise to show that the system is equivalent to the standard sets.

Alphabet.

Predicate symbols: $I, =$

We introduce the following abbreviations:

$$\Pi(x) := \exists y(x I y), \quad \Lambda(y) := \exists x(x I y).$$

Definition 3.7.5 \mathfrak{A} is a *projective plane* if it satisfies

$$\begin{aligned}\gamma_0 &: \forall x (\Pi(x) \leftrightarrow \neg \Lambda(x)), \\ \gamma_1 &: \forall xy (\Pi(x) \wedge \Pi(y) \rightarrow \exists z (xIz \wedge yIz)), \\ \gamma_2 &: \forall uv (\Lambda(u) \wedge \Lambda(v) \rightarrow \exists x (xIu \wedge xIv)), \\ \gamma_3 &: \forall xyuv (xIu \wedge yIu \wedge xIv \wedge yIv \rightarrow x = y \vee u = v), \\ \gamma_4 &: \exists x_0 x_1 x_2 x_3 u_0 u_1 u_2 u_3 \left(\bigwedge_{i=0}^3 x_i I u_i \wedge \bigwedge_{j=i-1 \pmod 3} x_i I u_j \wedge \bigwedge_{\substack{j \neq i-1 \pmod 3 \\ i \neq j}} \neg x_i I u_j \right).\end{aligned}$$

γ_0 tells us that in a projective plane everything is either a point or a line, γ_1 and γ_2 tell us that “any two lines intersect in a point” and “any two points can be joined by a line”, by γ_3 this point (or line) is unique if the given lines (or points) are distinct. Finally γ_4 makes projective planes non-trivial, in the sense that there are enough points and lines.

$\Pi^{\mathfrak{A}} = \{a \in |\mathfrak{A}| \mid \mathfrak{A} \models \Pi(\bar{a})\}$ and $\Lambda^{\mathfrak{A}} = \{b \in |\mathfrak{A}| \mid \mathfrak{A} \models \Lambda(\bar{b})\}$ are the sets of *points* and *lines* of \mathfrak{A} ; $I^{\mathfrak{A}}$ is the *incidence relation* on \mathfrak{A} .

The above formalization is rather awkward. One usually employs a two-sorted formalism, with P, Q, R, \dots varying over points and ℓ, m, n, \dots varying over lines. The first axiom is then suppressed by convention. The remaining axioms become

$$\begin{aligned}\gamma'_1 &: \forall P Q \exists \ell (PI\ell \wedge QI\ell), \\ \gamma'_2 &: \forall \ell m \exists P (PI\ell \wedge PIm), \\ \gamma'_3 &: \forall P Q \ell m (PI\ell \wedge QI\ell \wedge PIm \wedge QIm \rightarrow P = Q \vee \ell = m), \\ \gamma'_4 &: \exists P_0 P_1 P_2 P_3 \ell_0 \ell_1 \ell_2 \ell_3 \left(\bigwedge_{i=0}^3 P_i I \ell_i \wedge \bigwedge_{j=i-1 \pmod 3} P_i I \ell_j \wedge \bigwedge_{\substack{j \neq i-1 \pmod 3 \\ i \neq j}} \neg P_i I \ell_j \right).\end{aligned}$$

The translation from one language to the other presents no difficulty. The above axioms are different from the ones usually given in the course in projective geometry. We have chosen these particular axioms because they are easy to formulate and also because the *duality principle* follows immediately (cf. Sect. 3.10, Exercise 8). The fourth axiom is an existence axiom, it merely says that certain things exist; it can be paraphrased differently: there are four points no three of which are collinear (i.e. on a line). Such an existence axiom is merely a precaution to make sure that trivial models are excluded. In this particular case, one would not do much geometry if there were only one triangle!

5. *The language of rings with unity.* Type: $\langle -; 2, 2, 1; 2 \rangle$

Alphabet.

Predicate symbol: =

Function symbols: +, ·, −

Constant symbols: 0, 1

Definition 3.7.6 \mathfrak{A} is a *ring* (with unity) if it is a model of

$$\begin{aligned} &\forall xyz((x + y) + z = x + (y + z)), \\ &\forall xy(x + y = y + x), \\ &\forall xyz((xy)z = x(yz)), \\ &\forall xyz(x(y + z) = xy + xz), \\ &\forall xyz((x + y)z = xz + yz), \\ &\forall x(x + 0 = x), \\ &\forall x(x + (-x) = 0), \\ &\forall x(1 \cdot x = x \wedge x \cdot 1 = x), 0 \neq 1. \end{aligned}$$

A ring \mathfrak{A} is *commutative* if $\mathfrak{A} \models \forall xy(xy = yx)$.

A ring \mathfrak{A} is a *division ring* if $\mathfrak{A} \models \forall x(x \neq 0 \rightarrow \exists y(xy = 1))$.

A commutative division ring is called a *field*.

Actually it is more convenient to have an inverse function symbol available in the language of fields, which therefore has type $\langle -; 2, 2, 1, 1; 2 \rangle$.

Therefore we add to the above list the sentences $\forall x(x \neq 0 \rightarrow x \cdot x^{-1} = 1 \wedge x^{-1} \cdot x = 1)$ and $0^{-1} = 1$.

Note that we must somehow “fix the value of 0^{-1} ”, the reason will appear in Sect. 3.10, Exercise 2.

6. *The language of arithmetic.* Type $\langle -; 2, 2, 1; 1 \rangle$.

Alphabet.

Predicate symbol: =

Function symbols: +, ·, S

Constant symbol: 0

(S stands for the successor function $n \mapsto n + 1$).

Historically, the language of arithmetic was introduced by Peano with the intention to describe the natural numbers with plus, times and successor up to an isomorphism. This is in contrast to, e.g. the theory of groups, in which one tries to capture a large class of non-isomorphic structures. It has turned out, however, that Peano’s axioms characterize a large class of structures, which we will call (lacking a current term) *Peano structures*. Whenever confusion threatens we will use the official notation for the zero symbol: $\bar{0}$, but mostly we will trust the good sense of the reader.

Definition 3.7.7 A *Peano structure* \mathfrak{A} is a model of

$$\begin{aligned} &\forall x(0 \neq S(x)), \\ &\forall xy(S(x) = S(y) \rightarrow x = y), \\ &\forall x(x + 0 = x), \end{aligned}$$

$$\begin{aligned}
& \forall xy(x + S(y) = S(x + y)), \\
& \forall x(x \cdot 0 = 0), \\
& \forall xy(x \cdot S(y) = x \cdot y + x), \\
& \varphi(0) \wedge \forall x(\varphi(x) \rightarrow \varphi(S(x))) \rightarrow \forall x\varphi(x).
\end{aligned}$$

The last axiom schema is called the *induction schema* or the *principle of mathematical induction*.

It will prove convenient to introduce some notation. We define:

$$\begin{aligned}
\bar{1} &:= S(\bar{0}), & \bar{2} &:= S(\bar{1}), & \text{and in general } \overline{n+1} &:= S(\bar{n}), \\
x < y &:= \exists z(x + Sz = y), \\
x \leq y &:= x < y \vee x = y.
\end{aligned}$$

There is one particular Peano structure which is the intended model of arithmetic, namely the structure of the ordinary natural numbers, with the ordinary addition, multiplication and successor (e.g. the finite ordinals in set theory). We call this Peano structure the *standard model* \mathfrak{N} , and the ordinary natural numbers are called the *standard numbers*.

One easily checks that $\bar{n}^{\mathfrak{N}} = n$ and $\mathfrak{N} \models \bar{n} < \bar{m} \Leftrightarrow n < m$: by definition of interpretation we have $\bar{0}^{\mathfrak{N}} = 0$. Assume $\bar{n}^{\mathfrak{N}} = n$, then $\overline{n+1}^{\mathfrak{N}} = (S(\bar{n}))^{\mathfrak{N}} = \bar{n}^{\mathfrak{N}} + 1 = n + 1$. We now apply mathematical induction in the meta-language, and obtain $\bar{n}^{\mathfrak{N}} = n$ for all n . For the second claim see Exercise 13. In \mathfrak{N} we can define all kinds of sets, relations and numbers. To be precise we say that a k -ary relation R in \mathfrak{N} is defined by φ if $\langle a_1, \dots, a_k \rangle \in R \Leftrightarrow \mathfrak{N} \models \varphi(\bar{a}_1, \dots, \bar{a}_k)$. An element $a \in |\mathfrak{N}|$ is defined in \mathfrak{N} by φ if $\mathfrak{N} \models \varphi(\bar{b}) \Leftrightarrow b = a$, or $\mathfrak{N} \models \forall x(\varphi(x) \Leftrightarrow x = \bar{a})$.

Examples

- (a) The set of even numbers is defined by $E(x) := \exists y(x = y + y)$.
- (b) The divisibility relation is defined by $x|y := \exists z(xz = y)$.
- (c) The set of prime numbers is defined by $P(x) := \forall yz(x = yz \rightarrow y = 1 \vee z = 1) \wedge x \neq 1$.

We say that we have introduced predicates E , $|$ and P by (explicit) definition.

7. The language of graphs

We usually think of graphs as geometric figures consisting of vertices and edges connecting certain of the vertices. A suitable language for the theory of graphs is obtained by introducing a predicate R which expresses the fact that two vertices are connected by an edge. Hence, we don't need variables or constants for edges.

Alphabet.

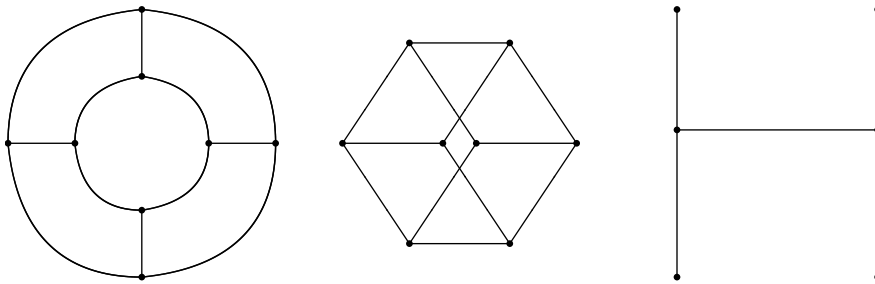
Predicate symbols: $R, =$

Definition 3.7.8 A *graph* is a structure $\mathfrak{A} = \langle A, R \rangle$ satisfying the following axioms:

$$\begin{aligned} \forall xy(R(x, y) \rightarrow R(y, x)) \\ \forall x \neg R(x, x). \end{aligned}$$

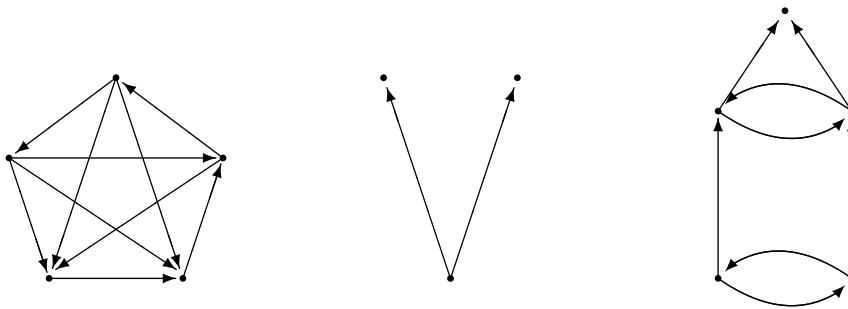
This definition is in accordance with the geometric tradition. There are elements, called vertices, of which some are connected by edges. Note that two vertices are connected by at most one edge. Furthermore there is no (need for an) edge from a vertex to itself. This is geometrically inspired; however, from the point of view of the numerous applications of graphs it appears that more liberal notions are required.

Examples



We can also consider graphs in which the edges are directed. A *directed graph* $\mathfrak{A} = \langle A, R \rangle$ satisfies only $\forall x \neg R(x, x)$.

Examples



If we drop the condition of irreflexivity then a “graph” is just a set with a binary relation. We can generalize the notion even further, so that more edges may connect a pair of vertices.

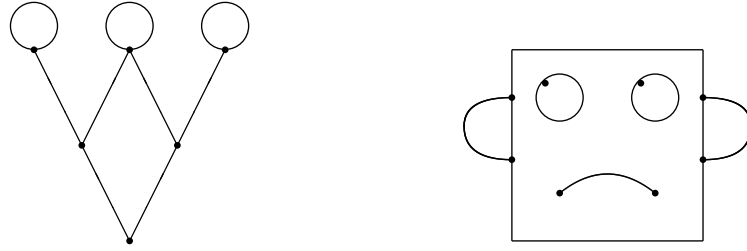
In order to treat those generalized graphs we consider a language with two unary predicates V , E and one ternary predicate C . Think of $V(x)$ as “ x is a vertex”, $E(x)$

as “ x is an edge”, and $C(x, z, y)$ as “ z connects x and y ”. A directed *multigraph* is a structure $\langle A, V, E, C \rangle$ satisfying the following axioms:

$$\begin{aligned} \forall x (V(x) \leftrightarrow \neg E(x)), \\ \forall xyz (C(x, z, y) \rightarrow V(x) \wedge V(y) \wedge E(z)). \end{aligned}$$

The edges can be seen as arrows. By adding the symmetry condition, $\forall xyz (C(x, z, y) \rightarrow C(y, z, x))$ one obtains plain multigraphs.

Examples



Remark The nomenclature in graph theory is not very uniform. We have chosen our formal framework such that it lends itself to treatment in first-order logic.

For the purpose of describing multigraphs a two-sorted language (cf. geometry) is well suited. The reformulation is left to the reader.

Exercises

1. Consider the language of partial order. Define predicates for (a) x is the *maximum*; (b) x is *maximal*; (c) there is no element between x and y ; (d) x is an *immediate successor* (respectively *predecessor*) of y ; (e) z is the *infimum* of x and y .
2. Give a sentence σ such that $\mathfrak{A}_2 \models \sigma$ and $\mathfrak{A}_4 \models \neg\sigma$ (for \mathfrak{A}_i associated to the diagrams of p. 79).
3. Let $\mathfrak{A}_1 = \langle \mathbb{N}, \leq \rangle$ and $\mathfrak{A}_2 = \langle \mathbb{Z}, \leq \rangle$ be the ordered sets of natural, respectively integer, numbers. Give a sentence σ such that $\mathfrak{A}_1 \models \sigma$ and $\mathfrak{A}_2 \models \neg\sigma$. Do the same for \mathfrak{A}_2 and $\mathfrak{B} = \langle \mathbb{Q}, \leq \rangle$ (the ordered set of rationals). N.B. σ is in the language of posets; in particular, you may not add extra constants, function symbols, etc., defined abbreviations are of course harmless.
4. Let $\sigma = \exists x \forall y (x \leq y \vee y \leq x)$. Find posets \mathfrak{A} and \mathfrak{B} such that $\mathfrak{A} \models \sigma$ and $\mathfrak{B} \models \neg\sigma$.
5. Do the same for $\sigma = \forall xy \exists z [(x \leq z \wedge y \leq z) \vee (z \leq x \wedge z \leq y)]$.
6. Using the language of identity structures give an (infinite) set Γ such that \mathfrak{A} is a model of Γ iff \mathfrak{A} is infinite.
7. Consider the language of groups. Define the properties: (a) x is idempotent; (b) x belongs to the center.

8. Let \mathfrak{A} be a ring, give a sentence σ such that $\mathfrak{A} \models \sigma \Leftrightarrow \mathfrak{A}$ is an integral domain (has no divisors of zero).
9. Give a formula $\sigma(x)$ in the language of rings such that $\mathfrak{A} \models \sigma(\bar{a}) \Leftrightarrow$ the principal ideal (a) is prime (in \mathfrak{A}).
10. Define in the language of arithmetic: (a) x and y are relatively prime; (b) x is the smallest prime greater than y ; (c) x is the greatest number with $2x < y$.
11. $\sigma := \forall x_1 \dots x_n \exists y_1 \dots y_m \varphi$ and $\tau := \exists y_1 \dots y_m \psi$ are sentences in a language without identity, function symbols and constants, where φ and ψ are quantifier free. Show: $\models \sigma \Leftrightarrow \sigma$ holds in all structures with n elements. $\models \tau \Leftrightarrow \tau$ holds in all structures with 1 element.
12. *Monadic predicate calculus* has only unary predicate symbols (no identity). Consider $\mathfrak{A} = \langle A, R_1, \dots, R_n \rangle$ where all R_i are sets. Define $a \sim b := a \in R_i \Leftrightarrow b \in R_i$ for all $i \leq n$. Show that \sim is an equivalence relation and that \sim has at most 2^n equivalence classes. The equivalence class of a is denoted by $[a]$. Define $B = A/\sim$ and $[a] \in S_i \Leftrightarrow a \in R_i$, $\mathfrak{B} = \langle B, S_1, \dots, S_n \rangle$. Show $\mathfrak{A} \models \sigma \Leftrightarrow \mathfrak{B} \models \sigma$ for all σ in the corresponding language. For such σ show $\models \sigma \Leftrightarrow \mathfrak{A} \models \sigma$ for all \mathfrak{A} with at most 2^n elements. Using this fact, outline a decision procedure for truth in monadic predicate calculus.
13. Let \mathfrak{N} be the standard model of arithmetic. Show $\mathfrak{N} \models \bar{n} < \bar{m} \Leftrightarrow n < m$.
14. Let $\mathfrak{A} = \langle \mathbb{N}, < \rangle$ and $\mathfrak{B} = \langle \mathbb{N}, \Delta \rangle$, where $n \Delta m$ iff (i) $n < m$ and n, m both even or both odd, or (ii) if n is even and m odd. Give a sentence σ such that $\mathfrak{A} \models \sigma$ and $\mathfrak{B} \models \neg \sigma$.
15. If $\langle A, R \rangle$ is a projective plane, then $\langle A, \check{R} \rangle$ is also a projective plane (*the dual plane*), where \check{R} is the converse of the relation R . Formulated in the two-sorted language: if $\langle A_P, A_L, I \rangle$ is a projective plane, then so is $\langle A_L, A_P, \check{I} \rangle$.

3.8 Natural Deduction

We extend the system of Sect. 2.5 to predicate logic. For reasons similar to the ones mentioned in Sect. 2.5 we consider a language with connectives $\wedge, \rightarrow, \perp$ and \forall . The existential quantifier is left out, but will be considered later.

We adopt all the rules of propositional logic and we add

$$\forall I \frac{\varphi(x)}{\forall x \varphi(x)} \quad \forall E \frac{\forall x \varphi(x)}{\varphi(t)}$$

where in $\forall I$ the variable x may not occur free in any hypothesis on which $\varphi(x)$ depends, i.e. an uncanceled hypothesis in the derivation of $\varphi(x)$. In $\forall E$ we, of course, require t to be free for x .

$\forall I$ has the following intuitive explanation: if an arbitrary object x has the property φ , then every object has the property φ . The problem is that none of the objects we know in mathematics can be considered “arbitrary”. So instead of looking for the “arbitrary object” in the real world (as far as mathematics is concerned), let us try

to find a syntactic criterion. Consider a variable x (or a constant) in a derivation, are there reasonable grounds for calling x “arbitrary”? Here is a plausible suggestion: in the context of the derivations we shall call x arbitrary if nothing has been assumed concerning x . In more technical terms, x is arbitrary at its particular occurrence in a derivation if the part of the derivation above it contains no hypotheses containing x free.

We will demonstrate the necessity of the above restrictions, keeping in mind that the system at least has to be *sound*, i.e. that derivable statements should be true.

Restriction on $\forall I$:

$$\frac{\frac{\frac{[x = 0]}{\forall x(x = 0)}}{x = 0 \rightarrow \forall x(x = 0)}}{\forall x(x = 0 \rightarrow \forall x(x = 0))} \\ \frac{}{0 = 0 \rightarrow \forall x(x = 0)}$$

The \forall introduction at the first step was illegal.

So $\vdash 0 = 0 \rightarrow \forall x(x = 0)$, but clearly $\not\models 0 = 0 \rightarrow \forall x(x = 0)$ (take any structure containing more than just 0).

Restriction on $\forall E$:

$$\frac{\frac{[\forall x \neg \forall y(x = y)]}{\neg \forall y(y = y)}}{\forall x \neg \forall y(x = y) \rightarrow \neg \forall y(y = y)}$$

The \forall elimination at the first step was illegal.

Note that y is not free for x in $\neg \forall y(x = y)$. The derived sentence is clearly not true in structures with at least two elements.

We now give some examples of derivations. We assume that the reader has enough experience by now in canceling hypotheses, so we will no longer indicate the cancellations by encircled numbers.

$\frac{\frac{\frac{[\forall x \forall y \varphi(x, y)]}{\forall y \varphi(x, y)} \forall E}{\varphi(x, y)} \forall E}{\forall x \varphi(x, y)} \forall I \\ \frac{}{\forall y \forall x(\varphi(x, y))} \forall I \quad \rightarrow I$	$\frac{[\forall x(\varphi(x) \wedge \psi(x))]}{\varphi(x) \wedge \psi(x)} \\ \frac{}{\varphi(x)} \\ \frac{}{\forall x \varphi(x)}$	$\frac{[\forall x(\varphi(x) \wedge \psi(x))]}{\varphi(x) \wedge \psi(x)} \\ \frac{}{\psi(x)} \\ \frac{}{\forall x \psi(x)}$
$\forall x \forall y \varphi(x, y) \rightarrow \forall y \forall x \varphi(x, y)$	$\frac{}{\forall x \varphi(x) \wedge \forall x \psi(x)} \\ \forall x(\varphi \wedge \psi) \rightarrow \forall x \varphi \wedge \forall x \psi$	

Let $x \notin FV(\varphi)$

$$\begin{array}{c}
\frac{[\forall x(\varphi \rightarrow \psi(x))]}{\varphi \rightarrow \psi(x)} \forall E \quad \frac{[\varphi]}{\psi(x)} \rightarrow E \\
\frac{\psi(x)}{\forall x \psi(x)} \forall I \quad \frac{[\varphi]}{\forall x \varphi} \forall I \quad \frac{[\forall x \varphi]}{\varphi} \forall E \\
\frac{\forall x \psi(x)}{\varphi \rightarrow \forall x \psi(x)} \rightarrow I \quad \frac{}{\varphi \leftrightarrow \forall x \varphi} \\
\hline
\forall x(\varphi \rightarrow \psi(x)) \rightarrow (\varphi \rightarrow \forall x \psi(x))
\end{array}$$

In the right-hand derivation $\forall I$ is allowed, since $x \notin FV(\varphi)$, and $\forall E$ is applicable.

Note that $\forall I$ in the bottom left derivation is allowed because $x \notin FV(\varphi)$, for at that stage φ is still (part of) a hypothesis.

The reader will have grasped the technique behind the quantifier rules: reduce a $\forall x \varphi$ to φ and reintroduce \forall later, if necessary. Intuitively, one makes the following step: to show “for all $x \dots x \dots$ ” it suffices to show “ $\dots x \dots$ ” for an arbitrary x . The latter statement is easier to handle. Without going into fine philosophical distinctions, we note that the distinction “for all $x \dots x \dots$ ” – “for an arbitrary $x \dots x \dots$ ” is embodied in our system by means of the distinction. “quantified statement” – “free variable statement”.

The reader will also have observed that under a reasonable derivation strategy, roughly speaking, elimination precedes introduction. There is a sound explanation for this phenomenon, its proper treatment belongs to *proof theory*, where *normal derivations* (derivations without superfluous steps) are considered. See Chap. 7. For the moment the reader may accept the above mentioned fact as a convenient rule of thumb.

We can formulate the derivability properties of the universal quantifier in terms of the relation \vdash :

$$\begin{aligned}
\Gamma \vdash \varphi(x) &\Rightarrow \Gamma \vdash \forall x \varphi(x) \quad \text{if } x \notin FV(\psi) \text{ for all } \psi \in \Gamma \\
\Gamma \vdash \forall x \varphi(x) &\Rightarrow \Gamma \vdash \varphi(t) \quad \text{if } t \text{ is free for } x \text{ in } \varphi.
\end{aligned}$$

The above implications follow directly from $(\forall I)$ and $(\forall E)$.

Our next goal is the correctness of the system of natural deduction for predicate logic. We first extend the definition of \models .

Definition 3.8.1 Let Γ be a set of formulas and let $\{x_{i_1}, x_{i_2}, \dots\} = \bigcup \{FV(\psi) \mid \psi \in \Gamma \cup \{\sigma\}\}$. If \mathbf{a} is a sequence (a_1, a_2, \dots) of elements (repetitions allowed) of $|\mathfrak{A}|$, then $\Gamma(\mathbf{a})$ is obtained from Γ by replacing simultaneously in all formulas of Γ the x_{i_j} by \bar{a}_j ($j \geq 1$) (for $\Gamma = \{\psi\}$ we write $\psi(\mathbf{a})$). We now define

- (i) $\mathfrak{A} \models \Gamma(\mathbf{a})$ if $\mathfrak{A} \models \psi$ for all $\psi \in \Gamma(\mathbf{a})$,
- (ii) $\Gamma \models \sigma$ if $\mathfrak{A} \models \Gamma(\mathbf{a}) \Rightarrow \mathfrak{A} \models \sigma(\mathbf{a})$ for all \mathfrak{A}, \mathbf{a} .

In case only sentences are involved, the definition can be simplified:

$$\begin{aligned}
\Gamma \models \sigma &\text{ if } \mathfrak{A} \models \Gamma \Rightarrow \mathfrak{A} \models \sigma \text{ for all } \mathfrak{A}. \\
\text{If } \Gamma = \emptyset, &\text{ we write } \models \sigma.
\end{aligned}$$

We can paraphrase this definition as: $\Gamma \models \sigma$, if for all structures \mathfrak{A} and all choices of \mathbf{a} , $\sigma(\mathbf{a})$ is true in \mathfrak{A} if all hypotheses of $\Gamma(\mathbf{a})$ are true in \mathfrak{A} .

Now we can formulate the following lemma.

Lemma 3.8.2 (Soundness) $\Gamma \vdash \sigma \Rightarrow \Gamma \models \sigma$.

Proof By definition of $\Gamma \vdash \sigma$ it suffices to show that for each derivation \mathcal{D} with hypothesis set Γ and conclusion σ $\Gamma \models \sigma$. We use induction on \mathcal{D} (cf. Lemma 2.5.1 and Exercise 2).

Since we have cast our definition of satisfaction in terms of valuations, which evidently contains the propositional logic as a special case, we can copy the cases of (1) the one-element derivation, and (2) the derivations with a propositional rule as the last step, from Lemma 2.6.1 (please check this claim).

So we have to treat derivations with $(\forall I)$ or $(\forall E)$ as the final step.

$$\begin{array}{ll}
 (\forall I) & \begin{array}{l} \mathcal{D} \quad \mathcal{D} \text{ has its hypotheses in } \Gamma \text{ and } x \text{ is not free in } \Gamma. \\ \hline \varphi(x) \quad \text{Induction hypothesis: } \Gamma \models \varphi(x), \text{ i.e. } \mathfrak{A} \models \Gamma(\mathbf{a}) \Rightarrow \\ \forall x \varphi(x) \quad \mathfrak{A} \models (\varphi(x))(\mathbf{a}) \text{ for all } \mathfrak{A} \text{ and all } \mathbf{a}. \end{array}
 \end{array}$$

It is no restriction to suppose that x is the first of the free variables involved (why?). So we can substitute \bar{a}_1 for x in φ . Put $\mathbf{a} = (a_1, \mathbf{a}')$. Now we have:

for all a_1 and $\mathbf{a}' = (a_2, \dots)$ $\mathfrak{A} \models \Gamma(\mathbf{a}') \Rightarrow \mathfrak{A} \models \varphi(\bar{a}_1)(\mathbf{a}')$, so
 for all \mathbf{a}' $\mathfrak{A} \models \Gamma(\mathbf{a}') \Rightarrow \mathfrak{A} \models (\varphi(\bar{a}_1))(\mathbf{a}')$ for all a_1 , so
 for all \mathbf{a}' $\mathfrak{A} \models \Gamma(\mathbf{a}') \Rightarrow \mathfrak{A} \models (\forall x \varphi(x))(\mathbf{a}')$.

This shows $\Gamma \models \forall x \varphi(x)$. (Note that in this proof we used $\forall x(\sigma \rightarrow \tau(x)) \rightarrow (\sigma \rightarrow \forall x \tau(x))$, where $x \notin FV(\sigma)$, in the meta-language. Of course we may use sound principles on the meta-level.)

$$\begin{array}{ll}
 (\forall E) & \begin{array}{l} \mathcal{D} \quad \text{Induction hypothesis: } \Gamma \models \forall x \varphi(x), \\ \hline \forall x \varphi(x) \quad \text{i.e. } \mathfrak{A} \models \Gamma(\mathbf{a}) \Rightarrow \mathfrak{A} \models (\forall x \varphi(x))(\mathbf{a}), \\ \varphi(t) \quad \text{for all } \mathbf{a} \text{ and } \mathfrak{A}. \end{array}
 \end{array}$$

So let $\mathfrak{A} \models \Gamma(\mathbf{a})$, then $\mathfrak{A} \models \varphi(\bar{b})(\mathbf{a})$ for all $b \in |\mathfrak{A}|$. In particular we may take $t[\bar{\mathbf{a}}/\mathbf{z}]$ for \bar{b} , where we slightly abuse the notation; since there are finitely many variables z_1, \dots, z_n , we only need finitely many of the a_i 's, and therefore we consider it an ordinary simultaneous substitution.

$\mathfrak{A} \models (\varphi[\mathbf{a}/\mathbf{z}])[t[\mathbf{a}/\mathbf{z}]/x]$, hence by Lemma 3.5.4, $\mathfrak{A} \models (\varphi[t/x])(\mathbf{a}/\mathbf{z})$, or $\mathfrak{A} \models (\varphi(t))(\mathbf{a})$. \square

Having established the soundness of our system, we can easily get non-derivability results.

Examples

1. $\not\vdash \forall x \exists y \varphi \rightarrow \exists y \forall x \varphi$.

Take $\mathfrak{A} = \langle \{0, 1\}, \{\langle 0, 1 \rangle, \langle 1, 0 \rangle\} \rangle$ (type $\langle 2; -; 0 \rangle$) and consider $\varphi := P(x, y)$, the predicate interpreted in \mathfrak{A} .

- $\mathfrak{A} \models \forall x \exists y P(x, y)$, since for 0 we have $\langle 0, 1 \rangle \in P$ and for 1 we have $\langle 1, 0 \rangle \in P$.
 But, $\mathfrak{A} \not\models \exists y \forall x P(x, y)$, since for 0 we have $\langle 0, 0 \rangle \notin P$ and for 1 we have $\langle 1, 1 \rangle \notin P$.
2. $\forall x \varphi(x, x), \forall xy(\varphi(x, y) \rightarrow \varphi(y, x)) \not\models \forall xyz(\varphi(x, y) \wedge \varphi(y, z) \rightarrow \varphi(x, z))$.
 Consider $\mathfrak{B} = \langle \mathbb{R}, P \rangle$ with $P = \{\langle a, b \rangle \mid |a - b| \leq 1\}$.

Although variables and constants are basically different, they share some properties. Both constants and free variables may be introduced in derivations through $\forall E$, but only free variables can be subjected to $\forall I$,—that is free variables can disappear in derivations by other than propositional means. It follows that a variable can take the place of a constant in a derivation but in general not vice versa. We make this precise as follows.

Theorem 3.8.3 *Let x be a variable not occurring in Γ or φ .*

- (i) $\Gamma \vdash \varphi \Rightarrow \Gamma[x/c] \vdash \varphi[x/c]$.
 (ii) *If c does not occur in Γ , then $\Gamma \vdash \varphi(c) \Rightarrow \Gamma \vdash \forall x \varphi(x)$.*

Proof (ii) follows immediately from (i) by $\forall I$. (i) Induction on the derivation of $\Gamma \vdash \varphi$. Left to the reader. \square

Observe that the result is rather obvious, changing c to x is just as harmless as coloring c red—the derivation remains intact.

Exercises

- Show:
 - $\vdash \forall x(\varphi(x) \rightarrow \psi(x)) \rightarrow (\forall x \varphi(x) \rightarrow \forall x \psi(x))$,
 - $\vdash \forall x \varphi(x) \rightarrow \neg \forall x \neg \varphi(x)$,
 - $\vdash \forall x \varphi(x) \rightarrow \forall z \varphi(z)$ if z does not occur in $\varphi(x)$,
 - $\vdash \forall x \forall y \varphi(x, y) \rightarrow \forall y \forall x \varphi(x, y)$,
 - $\vdash \forall x \forall y \varphi(x, y) \rightarrow \forall x \varphi(x, x)$,
 - $\vdash \forall x(\varphi(x) \wedge \psi(x)) \leftrightarrow \forall x \varphi(x) \wedge \forall x \psi(x)$,
 - $\vdash \forall x(\varphi \rightarrow \psi(x)) \leftrightarrow (\varphi \rightarrow \forall x \psi(x))$, where $x \notin FV(\varphi)$.
- Extend the definition of derivation to the present system (cf. Definition 2.4.1).
- Show $(s(t)[\bar{a}/x])^{\mathfrak{A}} = (s((t[\bar{a}/x])^{\mathfrak{A}})[\bar{a}/x])^{\mathfrak{A}}$.
- Show the inverse implications of Theorem 3.8.3.
- Assign to each atom $P(t_1, \dots, t_n)$ a proposition symbol, denoted by P . Now define a translation \dagger from the language of predicate logic into the language of propositional logic by

$$\begin{aligned}
 (P(t_1, \dots, t_n))^{\dagger} &:= P \quad \text{and} \quad \perp^{\dagger} := \perp, \\
 (\varphi \Box \psi)^{\dagger} &:= \varphi^{\dagger} \Box \psi^{\dagger}, \\
 (\neg \varphi)^{\dagger} &:= \neg \varphi^{\dagger}, \\
 (\forall x \varphi)^{\dagger} &:= \varphi^{\dagger}.
 \end{aligned}$$

Show $\Gamma \vdash \varphi \Rightarrow \Gamma^\dagger \vdash^\dagger \varphi^\dagger$, where \vdash^\dagger stands for “derivable without using $(\forall I)$ or $(\forall E)$ ” (does the converse hold?)

Conclude the consistency of predicate logic.

Show that predicate logic is conservative over propositional logic (cf. Definition 4.1.5).

3.9 Adding the Existential Quantifier

Let us introduce $\exists x\varphi$ as an abbreviation for $\neg\forall x\neg\varphi$ (Theorem 3.5.1 tells us that there is a good reason for doing so). We can prove the following:

Lemma 3.9.1

- (i) $\varphi(t) \vdash \exists x\varphi(x)$ (t free for x in φ)
- (ii) $\Gamma, \varphi(x) \vdash \psi \Rightarrow \Gamma, \exists x\varphi(x) \vdash \psi$ if x is not free in ψ or any formula of Γ .

Proof (i)

$$\frac{\frac{[\forall x\neg\varphi(x)]}{\neg\varphi(t)} \forall E \quad \varphi(t)}{\perp} \rightarrow E$$

$$\frac{\perp}{\neg\forall x\neg\varphi(x)} \rightarrow I$$

so $\varphi(t) \vdash \exists x\varphi(x)$

(ii)

$$\frac{\frac{\frac{[\varphi(x)]}{\mathcal{D}} \quad \psi}{[\neg\psi]} \rightarrow E \quad \perp}{\neg\varphi(x)} \rightarrow I$$

$$\frac{\neg\forall x\neg\varphi(x) \quad \frac{\neg\varphi(x)}{\forall x\neg\varphi(x)} \forall I}{\perp} \rightarrow E$$

$$\frac{\perp}{\psi} \text{ RAA}$$

□

Explanation The subderivation top left is the given one; its hypotheses are in $\Gamma \cup \{\varphi(x)\}$ (only $\varphi(x)$ is shown). Since $\varphi(x)$ (that is, all occurrences of it) is canceled and x does not occur free in Γ or ψ , we may apply $\forall I$. From the derivation we conclude that $\Gamma, \exists x\varphi(x) \vdash \psi$.

We can compress the last derivation into an elimination rule for \exists :

$$\frac{\begin{array}{c} [\varphi[y/x]] \\ \vdots \\ \exists x \varphi(x) \quad \psi \end{array}}{\psi} \exists E$$

with the conditions: y is not free in ψ , or in a hypothesis of the subderivation of ψ , other than $\varphi(x)$.

This is easily seen to be correct since we can always fill in the missing details, as shown in the preceding derivation.

By (i) we also have an introduction rule: $\frac{\varphi(t)}{\exists x \varphi(x)} \exists I$ for t free for x in φ .

Examples of derivations

$$\begin{array}{c} \frac{[\forall x(\varphi(x) \rightarrow \psi)]^3}{\varphi(x) \rightarrow \psi} \forall E \quad \frac{[\varphi(x)]^1}{\varphi(x)} \rightarrow E \\ \frac{[\exists x \varphi(x)]^2}{\psi} \exists E_1 \quad x \notin FV(\psi) \\ \frac{\psi}{\exists x \varphi(x) \rightarrow \psi} \rightarrow I_2 \\ \frac{\exists x \varphi(x) \rightarrow \psi}{\forall x(\varphi(x) \rightarrow \psi) \rightarrow (\exists x \varphi(x) \rightarrow \psi)} \rightarrow I_3 \\ \frac{[\varphi(x)]^1}{\exists x \varphi(x)} \quad \frac{[\psi(x)]^1}{\exists x \psi(x)} \\ \frac{[\varphi(x) \vee \psi(x)]^2}{\exists x \varphi(x) \vee \exists x \psi(x)} \vee E_1 \\ \frac{[\exists x(\varphi(x) \vee \psi(x))]^3}{\exists x \varphi(x) \vee \exists x \psi(x)} \exists E_2 \\ \frac{\exists x \varphi(x) \vee \exists x \psi(x)}{\exists x(\varphi(x) \vee \psi(x)) \rightarrow \exists x \varphi(x) \vee \exists x \psi(x)} \rightarrow I_3 \end{array}$$

We will also sketch the alternative approach, that of enriching the language.

Theorem 3.9.2 *Consider predicate logic with the full language and rules for all connectives, then $\vdash \exists x \varphi(x) \leftrightarrow \neg \forall x \neg \varphi(x)$.*

Proof Compare Theorem 2.6.3. □

It is time now to state the rules for \forall and \exists with more precision. We want to allow substitution of terms for some occurrences of the quantified variable in ($\forall E$) and ($\exists E$). The following example motivates this.

$$\frac{\forall x(x = x)}{x = x} \forall E \\ \frac{x = x}{\exists y(x = y)} \exists I$$

The result would not be derivable if we could only make substitutions for *all* occurrences at the same time. Yet, the result is evidently true.

The proper formulation of the rules now is:

$$\begin{array}{c}
 \forall I \frac{\varphi}{\forall x \varphi} \qquad \forall E \frac{\forall x \varphi}{\varphi[t/x]} \\
 \qquad \qquad \qquad [\varphi] \\
 \qquad \qquad \qquad \cdot \\
 \qquad \qquad \qquad \cdot \\
 \exists I \frac{\varphi[t/x]}{\exists x \varphi} \qquad \exists E \frac{\exists x \varphi \quad \psi}{\psi}
 \end{array}$$

with the appropriate restrictions.

Exercises

1. $\vdash \exists x(\varphi(x) \wedge \psi) \leftrightarrow \exists x \varphi(x) \wedge \psi$ if $x \notin FV(\psi)$,
2. $\vdash \forall x(\varphi(x) \vee \psi) \leftrightarrow \forall x \varphi(x) \vee \psi$ if $x \notin FV(\psi)$,
3. $\vdash \forall x \varphi(x) \leftrightarrow \neg \exists x \neg \varphi(x)$,
4. $\vdash \neg \forall x \varphi(x) \leftrightarrow \exists x \neg \varphi(x)$,
5. $\vdash \neg \exists x \varphi(x) \leftrightarrow \forall x \neg \varphi(x)$,
6. $\vdash \exists x(\varphi(x) \rightarrow \psi) \leftrightarrow (\forall x \varphi(x) \rightarrow \psi)$ if $x \notin FV(\psi)$,
7. $\vdash \exists x(\varphi \rightarrow \psi(x)) \leftrightarrow (\varphi \rightarrow \exists x \psi(x))$ if $x \notin FV(\varphi)$,
8. $\vdash \exists x \exists y \varphi \leftrightarrow \exists y \exists x \varphi$,
9. $\vdash \exists x \varphi \leftrightarrow \varphi$ if $x \notin FV(\varphi)$.

3.10 Natural Deduction and Identity

We will give rules, corresponding to the axioms $I_1 - I_4$ of Sect. 3.6.

$$\begin{array}{c}
 \frac{}{x = x} \text{RI}_1 \\
 \frac{x = y}{y = x} \text{RI}_2 \\
 \frac{x = y \quad y = z}{x = z} \text{RI}_3 \\
 \frac{x_1 = y_1, \dots, x_n = y_n}{t(x_1, \dots, x_n) = t(y_1, \dots, y_n)} \text{RI}_4 \\
 \frac{x_1 = y_1, \dots, x_n = y_n \quad \varphi(x_1, \dots, x_n)}{\varphi(y_1, \dots, y_n)} \text{RI}_4
 \end{array}$$

where y_1, \dots, y_n are free for x_1, \dots, x_n in φ . Note that we want to allow substitution of the variable y_i ($i \leq n$) for *some* and not necessarily all occurrences of the variable x_i . We can express this by formulating RI_4 in the precise terms of the simultaneous substitution operator:

$$\begin{array}{c}
x_1 = y_1, \dots, x_n = y_n \\
\hline
t[x_1, \dots, x_n/z_1, \dots, z_n] = t[y_1, \dots, y_n/z_1, \dots, z_n] \\
x_1 = y_1, \dots, x_n = y_n \quad \varphi[x_1, \dots, x_n/z_1, \dots, z_n] \\
\hline
\varphi[y_1, \dots, y_n/z_1, \dots, z_n]
\end{array}$$

Example

$$\begin{array}{c}
x = y \quad x^2 + y^2 > 12x \\
\hline
2y^2 > 12x \\
x = y \quad x^2 + y^2 > 12x \\
\hline
x^2 + y^2 > 12y \\
x = y \quad x^2 + y^2 > 12x \\
\hline
2y^2 > 12y
\end{array}$$

The above are three legitimate applications of RI_4 having three different conclusions.

The rule RI_1 has no hypotheses, which may seem surprising, but which certainly is not forbidden.

The rules RI_4 have many hypotheses; as a consequence the derivation trees can look a bit more complicated. Of course one can get all the benefits from RI_4 by a restricted rule, allowing only one substitution at the time.

Lemma 3.10.1 $\vdash I_i$ for $i = 1, 2, 3, 4$.

Proof Immediate. □

We can weaken the rules RI_4 slightly by considering only the simplest terms and formulas.

Lemma 3.10.2 Let L be of type $\langle r_1, \dots, r_n; a_1, \dots, a_m; k \rangle$. If the rules

$$\begin{array}{c}
\frac{x_1 = y_1, \dots, x_{r_i} = y_{r_i} \quad P_1(x_1, \dots, x_{r_i})}{P_1(y_1, \dots, y_{r_i})} \quad \text{for all } i \leq n \\
\text{and} \\
\frac{x_1 = y_1, \dots, x_{a_j} = y_{a_j}}{f_j(x_1, \dots, x_{a_j}) = f_j(y_1, \dots, y_{a_j})} \quad \text{for all } j \leq m
\end{array}$$

are given, then the rules RI_4 are derivable.

Proof We consider a special case. Let L have one binary predicate symbol and one unary function symbol.

- (i) We show $x = y \vdash t(x) = t(y)$ by induction on t .
 (a) $t(x)$ is a variable or a constant. Immediate.
 (b) $t(x) = f(s(x))$. Induction hypothesis: $x = y \vdash s(x) = s(y)$

$$\begin{array}{c}
 \frac{[x = y]}{f(x) = f(y)} \\
 \frac{\frac{\frac{f(x) = f(y)}{\forall xy(x = y \rightarrow f(x) = f(y))} \forall I \ 2 \times}{s(x) = s(y) \rightarrow f(s(x)) = f(s(y))} \forall I \ 2 \times \quad \frac{x = y}{\mathcal{D}}}{f(s(x)) = f(s(y))} \mathcal{D}
 \end{array}$$

This shows $x = y \vdash f(s(x)) = f(s(y))$.

- (ii) We show $\vec{x} = \vec{y}, \varphi(\vec{x}) \vdash \varphi(\vec{y})$
 (a) φ is atomic, then $\varphi = P(t, s)$. t and s may (in this example) contain at most one variable each. So it suffices to consider $x_1 = y_1, x_2 = y_2, P(t(x_1, x_2), s(x_1, x_2)) \vdash P(t(y_1, y_2), s(y_1, y_2))$ (i.e. $P(t[x_1, x_2/z_1, z_2], \dots)$).
 Now we get, by applying $\rightarrow E$ twice, from

$$\begin{array}{c}
 \frac{[x_1 = y_1] \quad [x_2 = y_2] \quad [P(x_1, x_2)]}{P(y_1, y_2)} \\
 \frac{\frac{\frac{P(y_1, y_2)}{x_1 = y_1 \rightarrow (x_2 = y_2 \rightarrow (P(x_1, x_2) = P(y_1, y_2)))} \rightarrow I \ 3 \times}{\forall x_1 x_2 y_1 y_2 (x_1 = y_1 \rightarrow (x_2 = y_2 \rightarrow (P(x_1, x_2) = P(y_1, y_2))))} \forall I}{s(x_1, x_2) = s(y_1, y_2) \rightarrow (t(x_1, x_2) = t(y_1, y_2) \rightarrow (P(s_x, t_x) = P(s_y, t_y)))} \forall E
 \end{array}$$

and the following two instances of (i)

$$\begin{array}{ccc}
 x_1 = y_1 & x_2 = y_2 & x_1 = y_1 \quad x_2 = y_2 \\
 \mathcal{D} & \text{and} & \mathcal{D}' \\
 s(x_1, x_2) = s(y_1, y_2) & & t(x_1, x_2) = t(y_1, y_2),
 \end{array}$$

the required result, $(P(s_x, t_x) = P(s_y, t_y))$.

So

$$x_1 = y_1, x_2 = y_2 \vdash P(s_x, t_x) \rightarrow P(s_y, t_y)$$

where

$$\begin{array}{ll}
 s_x = s(x_1, x_2), & s_y = s(y_1, y_2) \\
 t_x = t(x_1, x_2), & t_y = t(y_1, y_2).
 \end{array}$$

- (b) $\varphi = \sigma \rightarrow \tau$.

Induction hypotheses:

$$\begin{array}{l}
 \vec{x} = \vec{y}, \sigma(\vec{y}) \vdash \sigma(\vec{x}) \\
 \vec{x} = \vec{y}, \tau(\vec{x}) \vdash \tau(\vec{y})
 \end{array}$$

$$\begin{array}{c}
\vec{x} = \vec{y} \quad [\sigma(\vec{y})] \\
\mathcal{D} \\
\frac{\sigma(\vec{x}) \rightarrow \tau(\vec{x}) \quad \sigma(\vec{x})}{\tau(\vec{x})} \quad \vec{x} = \vec{y} \\
\mathcal{D}' \\
\frac{\tau(\vec{y})}{\sigma(\vec{y}) \rightarrow \tau(\vec{y})}
\end{array}$$

So $\vec{x} = \vec{y}, \sigma(\vec{x}) \rightarrow \tau(\vec{x}) \vdash \sigma(\vec{y}) \rightarrow \tau(\vec{y})$.

(c) $\varphi = \sigma \wedge \tau$, left to the reader.

(d) $\varphi = \forall z \psi(z, \vec{x})$

Induction hypothesis: $\vec{x} = \vec{y}, \psi(z, \vec{x}) \vdash \psi(z, \vec{y})$

$$\begin{array}{c}
\frac{\forall z \psi(z, \vec{x})}{\psi(z, \vec{x})} \quad \vec{x} = \vec{y} \\
\mathcal{D} \\
\frac{\psi(z, \vec{y})}{\forall z \psi(z, \vec{y})}
\end{array}$$

So $\vec{x} = \vec{y}, \forall z \psi(z, \vec{x}) \vdash \forall z \psi(z, \vec{y})$.

This establishes, by induction, the general rule. □

Exercises

1. Show that $\forall x(x = x), \forall xyz(x = y \wedge z = y \rightarrow x = z) \vdash I_2 \wedge I_3$ (using predicate logic only).
2. Show $\vdash \exists x(t = x)$ for any term t . Explain why all functions in a structure are total (i.e. defined for all arguments); what is 0^{-1} ?
3. Show $\vdash \forall z(z = x \rightarrow z = y) \rightarrow x = y$.
4. Show $\vdash \forall xyz(x \neq y \rightarrow x \neq z \vee y \neq z)$.
5. Show that in the language of identity $I_1, I_2, I_3 \vdash I_4$.
6. Show $\forall x(x = a \vee x = b \vee x = c) \vdash \forall x \varphi(x) \leftrightarrow (\varphi(a) \wedge \varphi(b) \wedge \varphi(c))$, where a, b, c , are constants.
7. Show
 - (i) $\forall xy(f(x) = f(y) \rightarrow x = y), \forall xy(g(x) = g(y) \rightarrow x = y) \vdash \forall xy(f(g(x)) = f(g(y)) \rightarrow x = y)$,
 - (ii) $\forall y \exists x(f(x) = y), \forall y \exists x(g(x) = y) \vdash \forall y \exists x(f(g(x)) = y)$.
 Which properties are expressed by this exercise?
8. Prove the following *duality principle* for projective geometry (cf. Definition 3.7.5): If $\Gamma \vdash \varphi$ then also $\Gamma \vdash \varphi^d$, where Γ is the set of axioms of projective geometry and φ^d is obtained from φ by replacing each atom xIy by yIx . (Hint: check the effect of the translation d on the derivation of φ from Γ .)