and let $u : T_\Sigma(B) \to \mathfrak{B}$ be the valuation

$$u(r) \quad = \quad \prod_{[s]\neq[t]} u_{s,t}(r).$$

Since all components $\mathfrak{B}_{s,t}$ are models of $\mathbf{Th}\,\mathcal{D}$, so is their product $\mathfrak{B}$. By Theorem 3.36(ii), u factors through $T_\Sigma(B)/\mathbf{Th}\,\mathcal{D}$ as $[\ ] \circ v$, where $v : T_\Sigma(B)/\mathbf{Th}\,\mathcal{D} \to \mathfrak{B}$. Moreover, v is injective, since if $[s] \neq [t]$, then $v(s) \neq v(t)$ in at least one component, namely $\mathfrak{B}_{s,t}$. Thus $T_\Sigma(B)/\mathbf{Th}\,\mathcal{D}$ is isomorphic under v to a subalgebra of the product $\mathfrak{B}$. ■

Corollary 3.43 (Birkhoff): Let $\mathcal{D}$ be a class of Σ-algebras. The following are equivalent:

(i) $\mathcal{D}$ is a variety;

(ii) $\mathcal{D} = \mathbf{H}\,\mathbf{S}\,\mathbf{P}\,\mathcal{D}$;

(iii) $\mathcal{D} = \{\mathbf{H}, \mathbf{S}, \mathbf{P}\}^*\,\mathcal{D}$.

Proof That (ii) and (iii) are equivalent and imply (i) are immediate from Theorem 3.42. That (i) implies (ii) follows from Theorem 3.42 and the fact that for any set of formulas Φ, $\mathbf{Mod}\,\Phi = \mathbf{Mod}\,\mathbf{Th}\,\mathbf{Mod}\,\Phi$ (Exercise 3.21). ■

3.4 Predicate Logic

First-order predicate logic is the logic of predicates and quantification ($\forall$, $\exists$) over elements of a structure.

Syntax

Syntactically, we start with a countable signature as with equational logic, except that we include some *relation* or *predicate symbols* $p, q, r, \ldots$ in addition to the function symbols $f, g, \ldots$. A *signature* or *vocabulary* then consists of a set Σ of function and relation symbols, each with an associated *arity* (number of inputs). Function and relation symbols of arity 0, 1, 2, 3, and n are called *nullary, unary, binary, ternary*, and *n-ary*, respectively. Nullary elements are often called *constants*. One of the relation symbols may be the binary *equality symbol* $=$. In most applications, Σ is finite.

The language consists of:

- the function and relation symbols in Σ
- a countable set X of *individual variables* $x, y, \ldots$
- the propositional connectives $\rightarrow$ and $\mathbf{0}$
- the universal quantifier symbol $\forall$ ("for all")
- parentheses.

As in Section 3.2, the other propositional connectives $\vee$, $\wedge$, $\mathbf{1}$, $\neg$, and $\leftrightarrow$ can all be defined in terms of $\rightarrow$ and $\mathbf{0}$. Similarly, we will define below the existential quantifier $\exists$ ("there exists") in terms of $\forall$.

Terms $s, t, \ldots$ are exactly as in equational logic (see Section 3.3). A term is a *ground term* if it contains no variables.

Formulas $\varphi, \psi, \ldots$ are defined inductively. A formula is either

- an *atomic formula* $p(t_1, \ldots, t_n)$, where p is an n-ary relation symbol and $t_1, \ldots, t_n$ are terms; or
- $\varphi \rightarrow \psi$, $\mathbf{0}$, or $\forall x\ \varphi$, where φ and ψ are formulas and x is a variable.

Intuitively, in the formula $\forall x\ \varphi$, we think of φ as a property of an object x; then the formula $\forall x\ \varphi$ says that that property φ holds for all objects x.

The other propositional operators are defined from $\rightarrow$ and $\mathbf{0}$ as described in Section 3.2. The quantifier $\exists$ is defined as follows:

$$\exists x\ \varphi \quad \stackrel{\text{def}}{\iff} \quad \neg\forall x\ \neg\varphi. \tag{3.4.1}$$

Intuitively, in the formula $\exists x\ \varphi$, we again think of φ as a property of an object x; then the formula $\exists x\ \varphi$ says that that there exists an object x for which the property φ holds. The formal definition (3.4.1) asserts the idea that there exists an x for which φ is true if and only if it is not the case that for all x, φ is false.

As with propositional logic, we will assume a natural precedence of the operators and use parentheses where necessary to ensure that a formula can be read in one and only one way. The precedence of the propositional operators is the same as in Section 3.2. The quantifier $\forall$ binds more tightly than the propositional operators; thus $\forall x\ \varphi \rightarrow \psi$ should be parsed as $(\forall x\ \varphi) \rightarrow \psi$.

The family of languages we have just defined will be denoted collectively by $L_{\omega\omega}$. The two subscripts ω refer to the fact that we allow only finite (that is, $< \omega$) conjunctions and disjunctions and finitely many variables.

Example 3.44: The first-order language of number theory is suitable for expressing properties of the natural numbers $\mathbb{N}$. The signature consists of binary function symbols $+$ and $\cdot$ (written in infix), constants 0 and 1, and binary relation symbol $=$ (also written in infix). A typical term is $(x+1)\cdot y$ and a typical atomic formula is $x+y=z$. The formula

$$\forall x\ \exists y\ (x \leq y \wedge \forall z\ (z \mid y \rightarrow (z=1 \vee z=y)))$$

expresses the statement that there are infinitely many primes. Here $s \leq t$ is an abbreviation for $\exists w\ s+w=t$ and $s \mid t$ (read "s divides t") is an abbreviation for $\exists w\ s\cdot w=t$.

Scope, Bound and Free Variables

Let Q be either $\forall$ or $\exists$. If $\mathsf{Q}x\ \varphi$ occurs as a subformula of some formula ψ, then that occurrence of φ in ψ is called the *scope* of that occurrence of $\mathsf{Q}x$ in ψ. An occurrence of a variable y in ψ that occurs in a term is a *free occurrence of y in ψ* if it is not in the scope of any quantifier $\mathsf{Q}y$ with the same variable y. If $\mathsf{Q}y\ \varphi$ occurs as a subformula of ψ and y occurs free in φ, then that occurrence of y is said to be *bound to* that occurrence of $\mathsf{Q}y$. Thus an occurrence of y in ψ is bound to the $\mathsf{Q}y$ with smallest scope containing that occurrence of y, if such a $\mathsf{Q}y$ exists; otherwise it is free.

We say that a term t is *free for y in φ* if no free occurrence of y in φ occurs in the scope of a quantifier $\mathsf{Q}x$, where x occurs in t. This condition says that it is safe to substitute t for free occurrences of y in φ without fear of some variable x of t being inadvertently captured by a quantifier.

EXAMPLE 3.45: In the formula

$$\exists x\ ((\forall y\ \exists x\ q(x,y)) \wedge p(x,y,z)),$$

the scope of the first $\exists x$ is $(\forall y\ \exists x\ q(x,y)) \wedge p(x,y,z)$, the scope of the $\forall y$ is $\exists x\ q(x,y)$, and the scope of the second $\exists x$ is $q(x,y)$. The occurrence of x in $q(x,y)$ is bound to the second $\exists x$. The x in $p(x,y,z)$ occurs free in the subformula $(\forall y\ \exists x\ q(x,y)) \wedge p(x,y,z)$ but is bound to the first $\exists x$. The occurrence of y in $q(x,y)$ is bound to the $\forall y$, but the occurrence of y in $p(x,y,z)$ is free. The only occurrence of z in the formula is a free occurrence. The term $f(x)$ is not free for either y or z in the formula, because substitution of $f(x)$ for y or z would result in the capture of x by the first $\exists x$.

Note that the adjectives "free" and "bound" apply not to variables but to *occurrences* of variables in a formula. A formula may have free and bound occurrences of the same variable. For example, the variable y in the formula of Example 3.45 has one free and one bound occurrence. Note also that occurrences of variables in quantifiers—occurrences of the form $\forall y$ and $\exists y$—do not figure in the definition of free and bound.

A variable is called a *free variable* of a formula φ if it has a free occurrence in φ. The notation $\varphi[x_1/t_1, \ldots, x_n/t_n]$ or $\varphi[x_i/t_i \mid 1 \leq i \leq n]$ denotes the formula φ with all free occurrences of x_i replaced with t_i, $1 \leq i \leq n$. The substitution is done for all variables simultaneously. Note that $\varphi[x/s, y/t]$ can differ from $\varphi[x/s][y/t]$ if s has an occurrence of y. Although notationally similar, the substitution operator $[x/t]$ should not be confused with the function-patching operator defined in Section 1.3.

We occasionally write $\varphi(x_1, \ldots, x_n)$ to indicate that all free variables of φ are among $x_1, \ldots, x_n$. The variables $x_1, \ldots, x_n$ need not all appear in $\varphi(x_1, \ldots, x_n)$, however. When $\varphi = \varphi(x_1, \ldots, x_n)$, we sometimes write $\varphi(t_1, \ldots, t_n)$ instead of $\varphi[x_1/t_1, \ldots, x_n/t_n]$.

A formula is a *closed formula* or *sentence* if it contains no free variables. The *universal closure* of a formula φ is the sentence obtained by preceding φ with enough universal quantifiers $\forall x$ to bind all the free variables of φ.

Semantics

A *relational structure* over signature Σ is a structure $\mathfrak{A} = (A, \mathfrak{m}_{\mathfrak{A}})$ where A is a nonempty set, called the *carrier* or *domain* of $\mathfrak{A}$, and $\mathfrak{m}_{\mathfrak{A}}$ is a function assigning an n-ary function $f^{\mathfrak{A}} : A^n \to A$ to each n-ary function symbol $f \in \Sigma$ and an n-ary relation $p^{\mathfrak{A}} \subseteq A^n$ to each n-ary relation symbol $p \in \Sigma$. As with equational logic, nullary functions are considered elements of A; thus constant symbols $c \in \Sigma$ are interpreted as elements $c^{\mathfrak{A}} \in A$.

As in equational logic, we define a *valuation* to be a Σ-homomorphism $u : T_\Sigma(X) \to \mathfrak{A}$. A valuation u is uniquely determined by its values on the variables X.

Given a valuation u, we define $u[x/a]$ to be the new valuation obtained from u by changing the value of x to a and leaving the values of the other variables intact; thus

$$\begin{aligned} u[x/a](y) &\stackrel{\text{def}}{=} u(y), \quad y \neq x, \\ u[x/a](x) &\stackrel{\text{def}}{=} a. \end{aligned}$$

This is the same as in equational logic.

The *satisfaction relation* $\vDash$ is defined inductively as follows:

$$\begin{aligned}\mathfrak{A}, u \vDash p(t_1, \ldots, t_n) &\stackrel{\text{def}}{\iff} p^{\mathfrak{A}}(u(t_1), \ldots, u(t_n))\\ \mathfrak{A}, u \vDash \varphi \to \psi &\stackrel{\text{def}}{\iff} (\mathfrak{A}, u \vDash \varphi \Longrightarrow \mathfrak{A}, u \vDash \psi)\\ \mathfrak{A}, u \vDash \forall x\ \varphi &\stackrel{\text{def}}{\iff} \text{for all } a \in A,\ \mathfrak{A}, u[x/a] \vDash \varphi.\end{aligned}$$

It follows that

$$\begin{aligned}\mathfrak{A}, u \vDash \varphi \vee \psi &\iff \mathfrak{A}, u \vDash \varphi \text{ or } \mathfrak{A}, u \vDash \psi\\ \mathfrak{A}, u \vDash \varphi \wedge \psi &\iff \mathfrak{A}, u \vDash \varphi \text{ and } \mathfrak{A}, u \vDash \psi\\ \mathfrak{A}, u \vDash \neg\varphi &\iff \mathfrak{A}, u \nvDash \varphi;\ \text{that is, if it is } \textit{not} \text{ the case that } \mathfrak{A}, u \vDash \varphi\\ \mathfrak{A}, u \vDash \exists x\ \varphi &\iff \text{there exists an } a \in A \text{ such that } \mathfrak{A}, u[x/a] \vDash \varphi.\end{aligned}$$

Also, $\mathfrak{A}, u \nvDash \mathbf{0}$ and $\mathfrak{A}, u \vDash \mathbf{1}$.

If $\mathfrak{A}, u \vDash \varphi$, we say that φ is *true in* $\mathfrak{A}$ *under valuation* u, or that $\mathfrak{A}, u$ is a *model* of φ, or that $\mathfrak{A}, u$ *satisfies* φ. If Φ is a set of formulas, we write $\mathfrak{A}, u \vDash \Phi$ if $\mathfrak{A}, u \vDash \varphi$ for all $\varphi \in \Phi$ and say that $\mathfrak{A}, u$ *satisfies* Φ. If φ is true in all models of Φ, we write $\Phi \vDash \varphi$ and say that φ is a *logical consequence*[3] of Φ. If $\varnothing \vDash \varphi$, we write $\vDash \varphi$ and say that φ is *valid*.

It can be shown that if φ is a sentence, then $\vDash$ does not depend on the valuation u; that is, if $\mathfrak{A}, u \vDash \varphi$ for some u, then $\mathfrak{A}, u \vDash \varphi$ for all u (Exercise 3.29). In this case, we omit the u and just write $\mathfrak{A} \vDash \varphi$. If Φ is a set of sentences, then $\mathfrak{A} \vDash \Phi$ means that $\mathfrak{A} \vDash \varphi$ for all $\varphi \in \Phi$.

Two formulas φ, ψ are said to be *logically equivalent* if $\vDash \varphi \leftrightarrow \psi$.

The following lemma establishes a relationship between the function-patching operator $[x/a]$ on valuations and the substitution operator $[x/t]$ on terms and formulas.

LEMMA 3.46:

(i) For any valuation u and terms $s, t \in T_\Sigma(X)$,

$$u[x/u(t)](s) \quad = \quad u(s[x/t]).$$

(ii) If t is free for x in φ, then

$$\mathfrak{A}, u[x/u(t)] \vDash \varphi \iff \mathfrak{A}, u \vDash \varphi[x/t].$$

3 This notion of logical consequence is slightly different from the one used in equational logic (Section 3.3). There, the free variables of formulas were assumed to be implicitly universally quantified. We abandon that assumption here because we have explicit quantification.