1 Syntax of First-Order Logic

Definition 1 (Alphabet of First-Order Terms and Formulæ)
The alphabet of the language of first-order logic consists of the following symbols.

1. Constant Symbols: \(c, d, c_1, c_2, \ldots, d_1, d_2, \ldots\);
2. Function Symbols: \(f, g, h, f_1, f_2, \ldots, g_1, g_2, \ldots\);
3. Variables: \(x, y, z, x_1, x_2, \ldots, y_1, y_2, \ldots\);
4. Predicate (Relational) Symbols: \(P, Q, P_1, P_2, \ldots, Q_1, Q_2, \ldots\);
5. Logical Connectives: \(\neg, \land, \lor, \rightarrow\);
6. Quantifiers: \(\forall\) (read “for all” or “for each”) and \(\exists\) (read “there exists”); and
7. Punctuation: “(”, “), “,” and “.”.

Each predicate symbol \(P\) and each function symbol \(f\) is associated with a natural number called its \(arity\), written \(ar(P)\) and \(ar(f)\), respectively.

Predicate and function symbols with arity 1 (2, 3) are called \(unary\) (\(binary\), \(ternary\), respectively).

The constant, functional, and predicate symbols are called the \(non-log\al\) symbols (or \(parameters\).

Note: Predicate symbols with arity 0 are essentially propositional symbols, and function symbols with arity 0 are essentially constants (and thus we could omit constants in Definition 1).

Definition 2 (Terms)
Let \(CS\) be a set of constant symbols, \(FS\) a set of function symbols, and \(VS\) a set of variables. We define the \(set\ of\ terms\ \(TS\ inductively\ as\ follows.

1. \(CS \subseteq TS\),
2. \(VS \subseteq TS\), and
3. if \(f \in FS\) and \(t_1, \ldots, t_n \in TS\), then \(f(t_1, \ldots, t_n) \in TS\), where \(n = ar(f)\);

no other strings are terms.

Definition 3 (Well-Formed Formulae (WFFs))
Let \(PS\) be a set of predicate symbols, \(TS\) a set of terms, and \(VS\) a set of variables. We define the \(set\ of\ formulæ\ of\ first-order\ logic (WFF)\ inductively as follows.
1. If \( P \in \text{PS} \) and \( t_1, \ldots, t_n \in \text{TS} \), where \( n = \text{ar}(P) \), then \( P(t_1, \ldots, t_n) \in \text{WFF} \);
2. if \( \varphi \in \text{WFF} \), then \( (\neg \varphi) \in \text{WFF} \);
3. if \( \varphi, \psi \in \text{WFF} \), then \( (\varphi \star \psi) \in \text{WFF} \) for each \( \star \in \{\land, \lor, \rightarrow\} \);
4. if \( x \in \text{VS} \) and \( \varphi \in \text{WFF} \), then \( (\forall x.\varphi) \in \text{WFF} \) and \( (\exists x.\varphi) \in \text{WFF} \);

and no other strings are elements of \( \text{WFF} \).

In the following we assume that \( \varphi \lor \psi \) is a shorthand for \( (\neg \varphi) \rightarrow \psi \), \( \varphi \land \psi \) for \( \neg (\varphi \rightarrow (\neg \psi)) \), and \( (\exists x.\varphi) \) for \( \neg (\forall x. (\neg \varphi)) \).

**Definition 4 (Free and Bound Variables)**

Let \( \varphi \in \text{WFF} \). We define the set of free variables of \( \varphi \), denoted \( \text{FV}(\varphi) \), as follows.

1. If \( \varphi = P(t_1, \ldots, t_{\text{ar}(P)}) \), then \( \text{FV}(\varphi) = \{x \mid x \text{ appears in } t_i \text{ for some } 0 < i \leq \text{ar}(P)\} \);
2. if \( \varphi = (\neg \psi) \), then \( \text{FV}(\varphi) = \text{FV}(\psi) \);
3. if \( \varphi = (\psi \rightarrow \eta) \), then \( \text{FV}(\varphi) = \text{FV}(\psi) \cup \text{FV}(\eta) \); and
4. if \( \varphi = (\forall x.\psi) \), then \( \text{FV}(\varphi) = \text{FV}(\psi) \setminus \{x\} \).

Variables in the set \( \text{FV}(\varphi) \) are called free (in \( \varphi \)); other variables that occur in \( \varphi \) are called bound (in \( \varphi \)).

For a set of formulæ \( \Sigma \), we define \( \text{FV}(\Sigma) = \bigcup_{\varphi \in \Sigma} \text{FV}(\varphi) \).

**Definition 5 (Closed Formulae (Sentences))**

A first-order formula \( \varphi \in \text{WFF} \) is closed (or a sentence) iff \( \text{FV}(\varphi) = \emptyset \).

## 2 Interpreations and Models

**Definition 6 (First-Order Interpretations (Structures))**

A first-order interpretation (or structure) \( I \) is a pair \( (D, (\cdot)^I) \) where

- \( D \) is a non-empty set, called the domain (or universe) and
- \( (\cdot)^I \) is an interpretation function that maps
  - constant symbols \( c \in \text{CS} \) to individuals \( (c)^I \in D \),
  - function symbols \( f \in \text{FS} \) to functions \( (f)^I : \text{ar}(f) \rightarrow D \), and
  - predicate symbols \( P \in \text{PS} \) to relations \( (P)^I \subseteq D^{\text{ar}(P)} \).

For a fixed selection \( L \) of constant, function, and predicate symbols, called the vocabulary (or the signature, or—slightly abusing terminology—the language), we define \( L \)-structures to be those interpretations restricted to symbols in \( L \).

**Definition 7 (Valuation)**

Let \( D \) be a domain and \( \text{VS} \) a set of variables. A valuation (or assignment) is a mapping \( \theta : \text{VS} \rightarrow D \).

For a valuation \( \theta \), a variable \( x \) and a term \( v \), the valuation \( \theta[x/v] \) is defined by

\[
\theta[x/v](y) = \begin{cases} 
  v & \text{if } x = y, \\
  \theta(y) & \text{otherwise.}
\end{cases}
\]
Definition 8 (Meaning of Terms)
Let $I$ be a first-order interpretation and $\theta$ a valuation. For a term $t \in TS$, we define the interpretation of $t$, denoted $(t)^{I,\theta}$, as follows.

1. $(c)^{I,\theta} = (c)^I$ for $t \in CS$ (i.e., $t$ is a constant),
2. $(x)^{I,\theta} = \theta(x)$ for $t \in VS$ (i.e., $t$ is a variable), and
3. $(f(t_1, \ldots, t_{ar(f)}))^{I,\theta} = (f)^I((t_1)^{I,\theta}, \ldots, (t_{ar(f)})^{I,\theta})$ otherwise (i.e., for $t$ a functional term).

Definition 9 (Satisfaction Relation)
The satisfaction relation $\models$ between a first-order interpretation $I$, a valuation $\theta$, and a formula $\varphi \in WFF$, written $I,\theta \models \varphi$, is defined as follows.

- $I,\theta \models P(t_1, \ldots, t_{ar(P)})$ iff $(t_1)^{I,\theta}, \ldots, (t_{ar(P)})^{I,\theta}) \in (P)^I$ for $P \in PS$;
- $I,\theta \models (\neg \varphi)$ iff $I,\theta \not\models \varphi$;
- $I,\theta \models (\varphi \rightarrow \psi)$ iff whenever $I,\theta \models \varphi$ then also $I,\theta \models \psi$.
- $I,\theta \models (\forall x. \varphi)$ iff $I,\theta[x/v] \models \varphi$ for all $v \in D$.

A pair $(I,\theta)$ such that $I,\theta \models \varphi$ is called a (pointed) model of $\varphi$. We define $\text{mod}(\varphi)$ to be the set of models of $\varphi$: $\text{mod}(\varphi) = \{(I,\theta) \mid I,\theta \models \varphi\}$.

Note: in many presentations the term model is used for an interpretation which makes a formula true (for all valuations); such a terminology, however, makes defining validity, satisfiability, and logical implication cumbersome. The two uses coincide for sentences—a particular case of the following lemma.

Lemma 10 (Relevance)
Let $L$ be the set of all non-logical symbols in $\varphi \in WFF$, and let

1. $I_1$ and $I_2$ be two interpretations such that $I_1(s) = I_2(s)$ for all $s \in L$ and
2. $\theta_1$ and $\theta_2$ be two valuations such that $\theta_1(x) = \theta_2(x)$ for all $x \in FV(\varphi)$.

Then $I_1,\theta_1 \models \varphi$ if and only if $I_2,\theta_2 \models \varphi$.

The above lemma allows us to consider only $L$-structures for an appropriately chosen set $L$ of non-logical parameters.

Definition 11 (Satisfiability and Validity)
A modal formula $\varphi$ is

- valid iff $I,\theta \models \varphi$ for all interpretations $I$ and all valuations $\theta$ (i.e., true in all models),
- satisfiable iff $I,\theta \models \varphi$ for some interpretation $I$ and some valuation $\theta$ (i.e., has a model), and
- unsatisfiable otherwise.

Definitions of logical implication ($\Sigma \models \varphi$) and equivalence and their properties are now the same as for propositional logic.
3 Hilbert Proof System

Definition 12 (Substitution)
A (syntactic) substitution of a term \( t \) for a variable \( x \), written \( (\cdot \ )_t^x \) : \( \text{WFF} \rightarrow \text{WFF} \), is a mapping of terms to terms and formulæ to formulæ, customarily written as a post-fix operator (i.e., \( \varphi_t^x \) stands for applying the substitution \( (\cdot \ )_t^x \) to \( \varphi \)). It is defined as follows.

1. For a term \( t_1 \), \( (t_1)_t^x \) is \( t_1 \) with each occurrence of the variable \( x \) replaced by the term \( t \).
2. For \( \varphi = P(t_1, \ldots, t_{\text{ar}(P)}) \), \( (\varphi)_t^x = P((t_1)_t^x, \ldots, (t_{\text{ar}(P)})_t^x) \).
3. For \( \varphi = (\neg \psi) \), \( (\varphi)_t^x = (\neg(\psi)_t^x) \);
4. For \( \varphi = (\psi \rightarrow \eta) \), \( (\varphi)_t^x = ((\psi)_t^x \rightarrow (\eta)_t^x) \), and
5. for \( \varphi = (\forall y.\psi) \), there are two cases:
   - if \( x \) is \( y \), then \( (\varphi)_t^x = \varphi = (\forall y.\psi) \), and
   - otherwise, then \( (\varphi)_t^x = (\forall z.(\psi)_{y/z}^x) \), where \( z \) is any variable that is not free in \( t \) or in \( \varphi \).

Note: in the last case above, the additional substitution \( (\cdot \ )_t^y \) (i.e., renaming the variable \( y \) to \( z \) in \( \psi \)) is needed in order to avoid an accidental capture of a variable by the quantifier (i.e., capture of any \( y \) that is possibly free in \( t \)).

Lemma 13 (Substitution)
Let \( I \) be an interpretation, \( \theta \) a valuation, \( t \) a term, \( x \) a variable. Then \( I, \theta \models \varphi_t^x \) if and only if \( I, \theta[ x/(t,\theta)] \models \varphi \) for all \( \varphi \in \text{WFF} \).

Definition 14 (Hibert System)
The First-Order Hilbert System is a deduction system for first-order logic defined by the tuples generated by the following schemes:

Ax1 \( \langle \forall^*(\varphi \rightarrow (\psi \rightarrow \varphi)) \rangle \);
Ax2 \( \langle \forall^*((\varphi \rightarrow (\psi \rightarrow \eta)) \rightarrow ((\varphi \rightarrow \psi) \rightarrow (\varphi \rightarrow \eta))) \rangle \);
Ax3 \( \langle \forall^*((\neg \varphi \rightarrow \neg \psi) \rightarrow (\psi \rightarrow \varphi)) \rangle \);
Ax4 \( \langle \forall^*((\forall x.\varphi) \rightarrow \psi) \rightarrow ((\forall x.\varphi) \rightarrow (\forall x.\psi)) \rangle \);
Ax5 \( \langle \forall^*(\forall x.\varphi) \rightarrow \varphi_t^x \rangle \) for \( t \in \text{TS} \) a term;
Ax6 \( \langle \forall^*(\varphi \rightarrow \forall x.\varphi) \rangle \) for \( x \notin \text{FV}(\varphi) \); and
MP \( \langle \varphi, (\varphi \rightarrow \psi), \psi \rangle \).

where \( \forall^* \) is a finite sequence of universal quantifiers (e.g., \( \forall x_1.\forall y.\forall x \)).

Theorem 15
Hilbert system is sound (\( \Sigma \vdash \varphi \) then \( \Sigma \models \varphi \)) and complete (\( \Sigma \models \varphi \) then \( \Sigma \vdash \varphi \)).

Lemma 16 (Generalization)
Let \( \Sigma \vdash \varphi \) and \( x \notin \text{FV}(\Sigma) \). Then \( \Sigma \vdash \forall x.\varphi \).

4 Equality

Definition 17 (Axioms of Equality)
Let \( \approx \) be a binary predicate symbol (written in infix). We define the First-Order Axioms of Equality as follows:
EqId \langle \forall x. (x \approx x) \rangle;
EqCong \langle \forall x. \forall y. (x \approx y) \rightarrow (\varphi \forall z \rightarrow \varphi \forall y) \rangle;

Theorem 18 (Gödel 1930)
Hilbert system with (axiomatized) equality is sound (\Sigma \vdash \varphi \text{ then } \Sigma \models \varphi) and complete (\Sigma \models \varphi \text{ then } \Sigma \vdash \varphi) with respect to first-order logic with (true) equality.

5 Definability

Definition 19 (Definability in an Interpretation)
Let \( I = (D, (\cdot)^I) \) be a first-order interpretation and \( \varphi \) a first-order formula. A set \( S \) of \( k \)-tuples over \( D \), \( S \subseteq D^k \), is defined by the formula \( \varphi \) if \( S = \{ (\theta(x_1), \ldots, \theta(x_k)) \mid I, \theta \models \varphi \} \).

A set \( S \) is definable in first-order logic if it is defined by some first-order formula \( \varphi \).

Definition 20 (Definability of a Set of Interpretations)
Let \( \Sigma \) be a set of first-order sentences and \( \mathcal{K} \) a set of interpretations. We say that \( \Sigma \) defines \( \mathcal{K} \) if

\[
I \in \mathcal{K} \text{ if and only if } I \models \Sigma.
\]

A set \( \mathcal{K} \) is (strongly) definable if it is defined by a (finite) set of first-order formulæ \( \Sigma \).

Theorem 21 (Compactness)
A set \( \Sigma \) is consistent if and only if every finite \( \Sigma_0 \subseteq \Sigma \) is consistent.

Corollary 22
The class of interpretations with finite domain is not definable in first-order logic.