A short introduction to first-order predicate logic

Dirk Schlimm

October 9, 2017

This handout covers most of the material of GEB, Ch. VIII 'Typographical Number Theory' (and more!), but presents it in a different order (and slightly different notation).

1 First-order predicate logic as a language

1.1 The alphabet and well-formed formulas

The **primitive symbols** of a language $\mathscr{L}$ of first-order logic:

1. An infinite sequence of *individual variables*: $\mathsf{x}_0, \mathsf{x}_1, \mathsf{x}_2, \ldots, \mathsf{x}_n, \ldots$
   [GEB *uses* $\mathtt{a}$, $\mathtt{b}$, $\mathtt{c}$, $\mathtt{d}$, $\mathtt{e}$, *and primes, such as* $\mathtt{a}'$, $\mathtt{a}''$.]
2. For each natural number n, a set of n*-ary function symbols*: $\mathsf{f}_0, \mathsf{f}_1, \mathsf{f}_2, \ldots$
   0-ary function symbols, which take no argument, are called *individual constants*.
3. For each natural number n, a set of n*-ary predicate symbols* (at least one of which must be non-empty): $\mathsf{P}_0, \mathsf{P}_1, \mathsf{P}_2, \ldots$
4. A first-order language *with equality* also contains the binary predicate symbol $=$.
5. The usual propositional *connectives*: $\wedge$, $\vee$, $\supset$, and $\sim$.
6. The *universal* and *existential quantifiers*, written as $\forall$ and $\exists$.
7. Finally, parentheses: (,). [GEB *uses* $\langle$ *and* $\rangle$.]

1.2 Logical and extra-logical symbols

The *logical* symbols of $\mathscr{L}$ are: variables (1.), connectives (5.), quantifiers (6.), and equality (4.).

The *extra-logical* symbols of $\mathscr{L}$ are the function (2.) and predicate (3.) symbols.
The meaning of the extra-logical symbols is determined by an *interpretation* (see Section 2).

Comparison: The languages of first-order vs. propositional logic.

	Propositional logic	First-order logic
Primitive symbols	Propositional variables	(Individual) Variables
		Predicate symbols
		Function symbols
	Connectives	Connectives
		Quantifiers
		(Equality)
Expressions (particular strings)		Terms
	Formulas (*wff*s)	Formulas (*wff*s)

1.3 Expressions: Terms

Recursive definition of **terms** in a language $\mathscr{L}$:

1. A single occurrence of a variable is a term.
2. If f is an n-ary function symbol, and $\mathsf{t}_1, \ldots, \mathsf{t}_n$ are terms, then the string $\mathsf{f}(\mathsf{t}_1, \ldots, \mathsf{t}_n)$ is a term. The t_i are called the *arguments* of the term.
   This includes the particular case where $n = 0$; a constant is also a term.
3. Nothing else is a term.

We will often use *infix* notation for binary (i. e., 2-ary) function symbols, and write $\mathsf{t}_1 \mathsf{f}\, \mathsf{t}_2$ instead of $\mathsf{f}(\mathsf{t}_1, \mathsf{t}_2)$; for example, we will write $\mathsf{x}_1 + \mathsf{x}_2$ for $+(\mathsf{x}_1, \mathsf{x}_2)$.

1.4 Expressions: Formulas

Recursive definition of **formulas** (also known as *well-formed formulas*) in a language $\mathscr{L}$:

1. If P is an n-ary predicate symbol, and $\mathsf{t}_1, \ldots, \mathsf{t}_n$ are terms, then the string $\mathsf{P}(\mathsf{t}_1, \ldots, \mathsf{t}_n)$ is a formula (*atomic formula*).
   The t_i are the *arguments* of the formula.
   If P is equality ('$=$'), the formula is called an *equation*, and is written as $(\mathsf{t}_1 = \mathsf{t}_2)$.
2. If A is a formula, so is $\sim\mathsf{A}$ (*negation formula*).
3. If A, B are formulas, so are $(\mathsf{A} \wedge \mathsf{B})$ (*conjunction*), $(\mathsf{A} \vee \mathsf{B})$ (*disjunction*), and $(\mathsf{A} \supset \mathsf{B})$ (*implication formula*, with *antecedent* A, and *consequent* B),
4. If x is a variable and A is a formula, then $(\forall \mathsf{x} : \mathsf{A})$ is a formula (*universal formula*; x is the *variable of quantification*, and A is the *scope*), and so is $(\exists \mathsf{x} : \mathsf{A})$ (*existential formula*).
   In *classical* logic, the quantifiers can be defined in terms of each other (just as we saw for the connectives): $\exists \mathsf{x} : \mathsf{A}(\mathsf{x}) \Leftrightarrow_{df} \sim\forall \mathsf{x} : \sim\mathsf{A}(\mathsf{x})$. As was the case in propositional logic, this seemingly innocuous convention commits us to the use of classical logic!
   In *intuitionistic* or *constructive* logic to assert '$\exists \mathsf{x} : \mathsf{A}(\mathsf{x})$' means to have a way of knowing *which* x is $\mathsf{A}(\mathsf{x})$, and this is *not* the same as asserting that there is none which is not $\mathsf{A}(\mathsf{x})$.
5. Nothing else is a formula.

As in propositional logic, parentheses will be omitted to reduce unnecessary clutter.

To look ahead a bit:

a) Terms are like *names*, consisting of variables and function symbols.
   In the standard interpretation (see below) they stand for *numbers*.
b) Formulas are like *statements*. They consist of predicate symbols, connectives, quantifiers.
   They can be *true* or *false*.

1.5 The language of first-order arithmetic

In the language of *first-order arithmetic* we have the *function symbols:* $\mathsf{0}$ (a 0-ary function, i. e., a constant), a unary function symbol S (successor), and the binary function symbols $+$ (addition) and $\times$ (multiplication). [*See* GEB*, pp. 213–214*]

Thus, for the language of arithmetic the **terms** are:

1. Every variable x_0, x_1, ... is a term.
2. $\mathsf{0}$ is a term. If t and u are terms, then so are St, $(\mathsf{t}+\mathsf{u})$, and $(\mathsf{t}\times\mathsf{u})$.
3. Nothing else is a term.

In the *standard interpretation* $\mathsf{SSS0}$ will be our formal representation of the number 3.

With regard to well-formed **formulas** in the language of first-order arithmetic we have:

1. If t and u are terms, then $\mathsf{t}=\mathsf{u}$ is an (*atomic*) formula.
2. If A and B are formulas, then so are $\sim\mathsf{A}$, $(\mathsf{A}\vee\mathsf{B})$, $(\mathsf{A}\wedge\mathsf{B})$, and $(\mathsf{A}\supset\mathsf{B})$.
3. If A and B are formulas and x is a variable, then $(\forall\mathsf{x}:\mathsf{A})$ and $(\exists\mathsf{x}:\mathsf{A})$ are formulas.
4. Nothing else is a formula.

Exercise 1: State five terms and five well-formed formulas in the language of first-order arithmetic.

2 Semantics

Goal: Give *meanings* to non-logical symbols to determine which formulas are *true.*

2.1 Structures

Since in the language of first-order logic we have formulas *and* terms, we cannot interpret the formulas directly with $\top$ and $\bot$, as we did in the propositional case. Instead, we need a *structure* that consists of

- a non-empty *domain* or *universe of discourse* over which the variables range, with
- some designated elements for the constants, and
- an n-ary *operation* on this domain for each n-ary function symbol, and
- an n-ary *relation* on the domain for each n-ary predicate symbol in our language.

Recall, in the language of *first-order arithmetic* we had the following extra-logical symbols: $\mathsf{0}$, S, $+$, and $\times$. Thus, any structure that interprets this language must contain one designated element, one unary operation, and two binary operations.

2.2 Interpretations

An *interpretation* consists of a *structure* together with a *mapping* from the non-logical symbols in the language to the elements of the structure.

The *standard interpretation* for the language of first-order arithmetic is based on the structure whose domain are the natural numbers (including the designated number zero), together with the unary successor function, and the binary addition and multiplication functions. Formally,

$$\langle \underbrace{\mathbb{N}}_{\textit{universe}} ; \underbrace{\text{zero}}_{\textit{designated element}} , \underbrace{\text{successor, addition, multiplication}}_{\textit{operations}} \rangle.$$

The mapping of the standard interpretation is (a function) σ from the language of first-order arithmetic to the natural number structure, defined as follows:[1]

Language:	$\mathsf{0}$	S	$+$	$\times$
Structure:	zero	successor	addition	multiplication
	0	s	$+$	$\times$

Thus, $\sigma(\mathsf{0})$ is 0, $\sigma(+)$ is $+$, etc. We will write this as: $\mathsf{0}^\sigma=0$, $+^\sigma=+$, etc.

The logical symbol '$=$' is always interpreted by the identity relation.

Notice that very different interpretations are also possible: For example, the domain could contain an additional element ω, or the domain could consist of only finitely many elements, with the successor, addition, and multiplication operations modified accordingly!

2.3 Truth in a structure (Tarski 1933)

Note that interpretations do not assign meanings to the individual variables. For this, we need valuations: A *valuation* is simply an interpretation that also assigns elements from the domain (i. e., individuals), to the variables. In the following, let σ be a valuation in a structure with universe U.

The following rules allow us to interpret complex terms (T1–T2) and formulas (F1–F4). As above, we write A^σ for the value that σ assigns to A (i. e., $\sigma(\mathsf{A})$). In the literature these rules are also referred to as Tarski's *Basic Semantic Definition.*

(T1) If x is a variable then x^σ is already defined (by the definition of a valuation).

(T2) If f is an n-ary function symbol and $\mathsf{t}_1,\dots,\mathsf{t}_n$ terms, then

$$\big(\mathsf{f}(\mathsf{t}_1\dots\mathsf{t}_n)\big)^\sigma=\mathsf{f}^\sigma(\mathsf{t}_1^\sigma,\dots,\mathsf{t}_n^\sigma).$$

(F1) If P is an (extra-logical) n-ary predicate symbol and $\mathsf{t}_1,\dots,\mathsf{t}_n$ terms, then

$$\big(\mathsf{P}(\mathsf{t}_1\dots\mathsf{t}_n)\big)^\sigma=\begin{cases}\top, & \text{if } \mathsf{P}^\sigma \text{ holds of } \mathsf{t}_1^\sigma,\dots,\mathsf{t}_n^\sigma,\\ \bot, & \text{otherwise.}\end{cases}$$

Here $\top$ and $\bot$ stand for 'true' and 'false', but they are not symbols of our language.

[1] To emphasize the distinction between symbols in the language (syntax) and operations in a structure (semantics), we follow GEB and use sans-serif symbols for the symbols in the language $\mathscr{L}$.

For the equality predicate, the above definition yields

$$(\mathsf{s}=\mathsf{t})^\sigma = \begin{cases} \top, & \text{if } \mathsf{s}^\sigma = \mathsf{t}^\sigma, \\ \bot, & \text{otherwise.}^{2} \end{cases}$$

(F2) $$(\sim \mathsf{A})^\sigma = \begin{cases} \top, & \text{if } \mathsf{A}^\sigma = \bot, \\ \bot, & \text{if } \mathsf{A}^\sigma = \top. \end{cases}$$

(F3) $$(\mathsf{A}\supset\mathsf{B})^\sigma = \begin{cases} \bot, & \text{if } \mathsf{A}^\sigma = \top \text{ and } \mathsf{B}^\sigma = \bot, \\ \top, & \text{otherwise, i.e., } \mathsf{A}^\sigma = \bot \text{ or } \mathsf{B}^\sigma = \top. \end{cases}$$

(F4) $$(\forall \mathsf{x} : \mathsf{A})^\sigma = \begin{cases} \top, & \text{if } \mathsf{A}^{\sigma(u/\mathsf{x})} = \top \text{ for every } u \in U,^{3} \\ \bot, & \text{otherwise, i.e., } \mathsf{A}^{\sigma(u/\mathsf{x})} = \bot \text{ for some } u \in U. \end{cases}$$

Thus, we see that σ maps *formulas* to truth values and *terms* to elements of the domain (see also the last remark in Section 1.4):

- (T1–T2) define a mapping from the set of terms to the universe U of σ, while
- (F1–F4) define a mapping from the set of formulas to $\{\top, \bot\}$, the truth values.

Notice also that (F4) is *non-effective*, i. e., it does not provide us with a *method* to find the truth value for $\forall \mathsf{x}\ \mathsf{A}$ if the universe of discourse is infinite!

Exercise 2: (a) Given a propositional formula, how can you *decide* (i. e., give a 'yes' or 'no' answer in a finite amount of time) whether it is a tautology or not? (b) How about whether it is a theorem? (c) Given a first-order formula and a structure, can you always decide whether the formula is true in the structure?

Here is an example of determining the truth value of a formula under an interpretation. To show that $\forall \mathsf{x} : 0 \neq \mathsf{x}$ is true in the standard interpretation, we have to argue that $(\forall \mathsf{x} : 0 \neq \mathsf{x})^\sigma$ is true. But, under which conditions is $(\forall \mathsf{x} : 0 \neq \mathsf{x})^\sigma$ true?

$$\begin{array}{lll}
 & (\forall \mathsf{x} : \sim(0 = \mathsf{Sx}))^\sigma \text{ is } \top & \\
\text{iff} & \sim(0 = \mathsf{Sx})^{\sigma(u/\mathsf{x})} \text{ is } \top \text{ for every } u \in \mathbb{N} & \text{(by F4)} \\
\text{iff} & (0 = \mathsf{Sx})^{\sigma(u/\mathsf{x})} \text{ is } \bot \text{ for every } u \in \mathbb{N} & \text{(by F2)} \\
\text{iff} & 0^{\sigma(u/\mathsf{x})} = (\mathsf{Sx})^{\sigma(u/\mathsf{x})} \text{ is } \bot \text{ for every } u \in \mathbb{N} & \text{(by F1)} \\
\text{iff} & 0 = \mathsf{S}^{\sigma(u/\mathsf{x})}\mathsf{x}^{\sigma(u/\mathsf{x})} \text{ is } \bot \text{ for every } u \in \mathbb{N} & \text{(by def. of } \sigma \text{ and T2)} \\
\text{iff} & 0 = su \text{ is } \bot \text{ for every } u \in \mathbb{N} & \text{(by def. of } \sigma) \\
\text{iff} & 0 = s0 \text{ is } \bot, \text{ and } 0 = s1 \text{ is } \bot, \text{ and } 0 = s2 \text{ is } \bot, \text{ and } 0 = s3 \text{ is } \bot, \text{ etc.} & \\
\text{iff} & 0 = 1 \text{ is } \bot, \text{ and } 0 = 2 \text{ is } \bot, \text{ and } 0 = 3 \text{ is } \bot, \text{ and } 0 = 4 \text{ is } \bot, \text{ etc.,} & \\
 & \text{which is obviously the case.} &
\end{array}$$

[2] Notice that the '$=$' in '$(\mathsf{s}=\mathsf{t})^\sigma$' is a predicate symbol in the language, but the '$=$' in '$\mathsf{s}^\sigma = \mathsf{t}^\sigma$' is the identity relation in the structure! The third '$=$' in this expression is a metalogical symbol used to talk about the relation between the language, the valuation, and the elements in the structure.

[3] The valuation $\sigma(u/\mathsf{x})$ is like the valuation σ, but where the variable x is mapped to the element $u \in U$, i. e., $\mathsf{x}^{\sigma(u/\mathsf{x})} = u$, regardless of what x^σ is. This is *not* the same as a substitution (a syntactic operation)!

2.4 Quantifiers

For the meaning of a formula with quantifiers, the order of the quantifiers matters! For example, consider the two formulas

(a) $\forall \mathsf{y} : \exists \mathsf{x} : \mathsf{M}(\mathsf{x},\mathsf{y})$, and
(b) $\exists \mathsf{x} : \forall \mathsf{y} : \mathsf{M}(\mathsf{x},\mathsf{y})$.

Let them be interpreted in a structure that has the set of human beings as its domain and where the predicate $\mathsf{M}(\mathsf{x},\mathsf{y})$ is interpreted by the relation 'x is the mother of y.' The meanings of the above two formulas then are:

a) 'For all human beings y, there is a mother x.' In this interpretation the formula is true, since everybody has a mother (not necessarily the same, of course).
b) 'There is a mother x of all human beings y.' In this interpretation the formula is false, since there is no human being who is everybody's mother.

Be sure to understand the difference between $\forall \mathsf{x}_1 : \exists \mathsf{x}_2 : (\mathsf{Sx}_1 = \mathsf{x}_2)$ and $\exists \mathsf{x}_1 : \forall \mathsf{x}_2 : (\mathsf{Sx}_1 = \mathsf{x}_2)$.

Exercise 3: Is it possible to find a structure and an interpretation of the above two formulas, such that (a) is false and (b) is true?

Exercise 4: Give two structures and interpretations (for which the domains are not numbers) in which $\forall \mathsf{x} : (\mathsf{x} = 0)$ is true and two in which it is false.

Exercise 5: If σ is an interpretation of first-order arithmetic in $\langle \mathbb{N}, 1;\ s,\ +,\ \times \rangle$, i. e., the structure whose domain are the natural numbers, and where $0^\sigma = 1$, $\mathsf{S}^\sigma = s$ (successor), $+^\sigma = +$, and $\times^\sigma = \times$, what are $(\mathsf{SS0} + \mathsf{SSS0} = \mathsf{SSSSS0})^\sigma$ and $(\forall \mathsf{x} : \sim(\mathsf{Sx} = 0))^\sigma$? How do we have to change the interpretation (in the same structure or a different one) so that the formulas become true?

2.5 Satisfaction, logical consequence, and models

Here are some important semantic concepts.

σ ***satisfies*** A**:** A is a formula and σ a valuation, such that $\mathsf{A}^\sigma = \top$.

σ ***satisfies*** Γ**:** Γ is a set of formulas and σ a valuation that satisfies every member of Γ.

A **is *logically true* (or *logically valid*), written** $\models \mathsf{A}$**:**
The formula A is satisfied by *every* valuation.

A **is *logical consequence* of** Γ**, written** $\Gamma \models \mathsf{A}$**:**
The formula A is satisfied by every valuation that satisfies Γ. In other words, every valuation that makes every formula in Γ true, also makes A true.

Γ **is *unsatisfiable*:** There is no valuation that satisfies each formula in Γ at the same time.

Model: A *model* of a set of formulas Γ is an interpretation in a structure, such that all formulas in Γ are true in the structure.

For example, let $\Gamma = \{ \forall \mathsf{x} : \exists \mathsf{y} : (\mathsf{Sx} = \mathsf{y}) \}$. Then, the standard interpretation (in the natural number structure) is a model for Γ.
However, also the following structure is a model for Γ, where the universe of discourse is $\{ a \}$ and S is interpreted as the identity function, so that the successor of a is a.

Exercise 6: Verify that the formula in Γ (in the example above) is true in the second model mentioned in the example.

Exercise 7: State a universal and an existential formula, both of which are logically true.

2.6 (Meta-) Proofs using semantic notions

Using the above rules (T1–T2, F1–F4) we can now prove that certain properties hold of *logical consequences.*

Theorem: If $\Gamma, \mathsf{A} \models \mathsf{B}$, then $\Gamma \models \mathsf{A} \supset \mathsf{B}$.

Proof: Let σ be any valuation, such that $\Gamma^\sigma = \top$.

Assume: $\Gamma, \mathsf{A} \models \mathsf{B}$, i. e., if $\mathsf{A}^\sigma = \top$, then $\mathsf{B}^\sigma = \top$.

To show: $\Gamma \models \mathsf{A} \supset \mathsf{B}$, i. e., $(\mathsf{A} \supset \mathsf{B})^\sigma = \top$.

According to (F3), $(\mathsf{A} \supset \mathsf{B})^\sigma = \top$ only if $\mathsf{A}^\sigma = \bot$ or $\mathsf{B}^\sigma = \top$.

But, this is guaranteed by our assumption regarding σ! □

Exercise 8: Prove the converse of the above theorem, i. e., if $\Gamma \models \mathsf{A} \supset \mathsf{B}$ then $\Gamma, \mathsf{A} \models \mathsf{B}$.

3 Substitutions in first-order logic

There is a certain intricacy that one has to take care of when dealing with the syntax of first-order logic, namely *substitutions.*

First, let us distinguish between two different uses of individual variables.

3.1 Bound occurrences of variables

A variable x is *bound* if it falls inside the scope (recall, A is the scope of $\forall \mathsf{x} : \mathsf{A}$) of a quantifier that has x as its variable of quantification.

Exercise 9: Is x_1 bound in: $P(\mathsf{x}_1, \mathsf{x}_2)$, $\forall \mathsf{x}_1 : P(\mathsf{x}_1, \mathsf{x}_2)$, $\exists \mathsf{x}_2 : P(\mathsf{x}_1, \mathsf{x}_2)$? What about x_2 ?

3.2 Free occurrences of variables

We could just define *free* variables as those variables in a formula that are not bound. However, we can also defined them explicitly as follows.

Recursive defintion of the **free** variables in a formula α:

1. If α is an atomic formula (i. e., a predicate): All occurrences of x in α are free.
2. If α is $\sim\mathsf{A}$: x is free in α if it is free in A.
3. If α is $(\mathsf{A} \supset \mathsf{B})$, or $(\mathsf{A} \wedge \mathsf{B})$, or $(\mathsf{A} \wedge \mathsf{B})$: x is free in α, if it is free in A or in B.
4. If α is $(\forall \mathsf{x} : \mathsf{A})$ or $(\exists \mathsf{x} : \mathsf{A})$: All occurrences of x in A are *bound*. If y is not the variable of quantification, then y is free in α, if it is free in A.

To highlight that a *wff* A contains a free variable x, we write $\mathsf{A}(\mathsf{x})$.

3.3 Closed terms and sentences

- A term is *closed* if it contains no variables. Otherwise, it is *open.*
- A *sentence* is a formula without free variables.

Exercise 10: State five terms that are closed and five that are not. State five sentences in first-order arithmetic and five formulas that are not sentences. Say whether these sentences are true in the natural number structure or not.

3.4 Substitution

When we substitute a term t for a variable x we have to be careful not to change the meaning of the formula.

Given a formula A and a variable x, we say that

a term t is *free for an occurrence of* x *in* A

if that occurrence of x is free in A and does not lie within the scope of any quantifier $\forall \mathsf{y}$ or $\exists \mathsf{y}$, where y is a variable that appears in t. That is, by substituting t for that occurrence of x, no new variable becomes bound.

If t is free for all free occurrences of x in A, we can *substitute* t for x in A, i. e., replace all free occurrences of x in A simultaneously by t. In this case we write $\mathsf{A}(\mathsf{t}/\mathsf{x})$, or simply $\mathsf{A}(\mathsf{t})$, if x is the only free variable in A.

[*Compare the notion of* 't *is free for* x *in* A' *with the 'Restriction' for the 'Rule of Specification' in* GEB, *p. 217.*]

Exercise 11: Let t be $((\mathsf{x}_1 + 0') \times \mathsf{y})$. Is t free for x in A, if A is:
a) $\forall \mathsf{y} : \sim(\mathsf{x} = \mathsf{y})$, b) $\forall \mathsf{x} : \sim(\mathsf{x} = \mathsf{y})$, c) $\forall \mathsf{z} : \sim(\mathsf{x} = \mathsf{z})$, d) $\forall \mathsf{y} : \exists \mathsf{x} : (\mathsf{x} = \mathsf{y})$,
e) $\forall \mathsf{x}_1 : \sim(\mathsf{x} = \mathsf{y})$? For a)–e): If t is free for x in A, what is $\mathsf{A}(\mathsf{t}/\mathsf{x})$?

4 The first-order predicate calculus

4.1 Axiomatic calculus

For the sake of simplicity, let us just consider the connectives $\sim$ and $\supset$ and the universal quantifier $\forall$ as primitives, and the other symbols as being defined in terms of them. (Note, this is possible in the case of classical logic.)

- **Logical axioms.** For all variables x and y, terms t, and all well-formed formulae A and B (we write $A(x)$ to indicate that x is a free variable in A; if we write A, it may or may not contain free variables):

  CP. All axiom schemata of propositional logic.

  L1. (*Substitution*) $\big(\forall x : A(x)\big) \supset A(t)$
  where $A(t)$ arises by substituting t for all the occurrences of x in A for which t is free for x (i. e., $A(t) = A(x)(t/x)$).
  Note: Since t is any term, and since variables are terms, t could also be the variable x itself! Thus, $\big(\forall x : A(x)\big) \supset A(x)$ is also an instance of this axiom.

  L2. ($\forall$*-Distribution*) $\big(\forall x : (A \supset B)\big) \supset (A \supset \forall x : B)$
  if A contains no free occurrence of x.

- **Axioms for equality.**

  E1. $(x = x)$

  E2. $(x = y) \supset \big(A(x) \supset A(y)\big)$
  where $A(y)$ arises from $A(x)$ by replacing some, but not necessarily all, free occurrences of x by y, and y is free for the occurrences of x which it replaces.

- **Inference rules.**

  Modus ponens: From A and $A \supset B$ infer B.

  Generalization: From A infer $\forall x : A$.

  Substitution rule: Any *wff* can be substituted for the metalogical variables A and B (of course, the same metalogical variable must be substituted by the same *wff*).

4.2 Axiomatic proofs

The following theorems are proved in detail in Epstein and Carnielli, *Logic and Computability* (2008), pp. 182–184.

1. **$\exists$-Introduction**

   (a) If t is free for x in $A(x)$, then $\vdash A(t) \supset \exists x : A(x)$.

   (b) If $\vdash A \supset B$, then $\vdash (\exists x : A) \supset B$, whenever x is not free in B.

2. **Properties of =**

   (a) For any term t, $\vdash t = t$.

   (b) $\vdash (t = u) \supset \big(A(t) \supset A(u)\big)$
   for any formula $A(z)$ where both t and u are free for z in A, and $A(t)$ arises from $A(z)$ by replacing all occurrences of z by t, and $A(u)$ by replacing all occurrences of z by u.

   (c) $\vdash t = u \supset u = t$.

   (d) $\vdash t = u \supset (u = v \supset t = v)$.

Here are some more examples of axiomatic proofs.

4.2.1 $\forall x_1 : A(x_1) \vdash \exists x_1 : A(x_1)$

1.	$\forall x_1 : A(x_1)$	Assumption
2.	$\forall x_1 : A(x_1) \supset A(t)$	Axiom L1
3.	$A(t)$	Modus Ponens on lines 1 and 2 (1, 2 MP, for short)
4.	$A(t) \supset \exists x_1 : A(x_1)$	If t is free for x in A (Theorem 1a, § 4.2)
5.	$\exists x_1 : A(x_1)$	MP 3, 4

4.2.2 $\forall x_1 : \forall x_2 : A(x_1, x_2) \vdash \forall x_2 : \forall x_1 : A(x_1, x_2)$

1. $\forall x_1 : \forall x_2 : A(x_1, x_2)$
2. $\forall x_1 : \forall x_2 : A(x_1, x_2) \supset \forall x_2 : A(x_1, x_2)$
3. $\forall x_2 : A(x_1, x_2)$
4. $\forall x_2 : A(x_1, x_2) \supset A(x_1, x_2)$
5. $A(x_1, x_2)$
6. $\forall x_1 : A(x_1, x_2)$
7. $\forall x_2 : \forall x_1 : A(x_1, x_2)$

Exercise 12: Justify all steps in the derivation 4.2.2.

4.2.3 $\forall x : B,\ \sim\exists x : C\ \vdash\ \forall x_1 : \sim(B(x) \supset C(x))$

1.	$\forall x : B(x)$	Assumption
2.	$\sim\exists x : C(x)$	Assumption
3.	$\sim\sim\forall x : \sim C(x)$	2 Definition of $\exists$
4.	$\forall x : \sim C(x)$	3 Propositional Logic
5.	$\forall x : B(x) \supset B(x)$	Axiom L1
6.	$B(x)$	MP 1, 5
7.	$(\forall x : \sim C(x)) \supset \sim C(x)$	Axiom L1
8.	$\sim C(x)$	MP 4, 7
9.	$B(x) \wedge \sim C(x)$	6, 8 Propositional Logic
10.	$(B(x) \wedge \sim C(x)) \supset \sim(B(x) \supset C(x))$	Tautology in Propositional Logic
11.	$\sim(B(x) \supset C(x))$	MP 9, 10
12.	$\forall x : \sim(B(x) \supset C(x))$	11 Generalization

4.3 Natural deduction calculus

In addition to the inference rules of propositional logic, we also have introduction and elimination rules for first-order logic. Here a and x are variables, and t is a term:

$$\boxed{\dfrac{\mathsf{A}\,(\mathsf{a}/\mathsf{x})}{\forall \mathsf{x}\ \mathsf{A}}\ \forall\mathit{Intro}} \qquad \boxed{\dfrac{\forall \mathsf{x}\ \mathsf{A}}{\mathsf{A}\,(\mathsf{t}/\mathsf{x})}\ \forall\mathit{Elim}}$$

$$\boxed{\dfrac{\mathsf{A}\,(\mathsf{t}/\mathsf{x})}{\exists \mathsf{x}\ \mathsf{A}}\ \exists\mathit{Intro}} \qquad \boxed{\dfrac{\exists \mathsf{x}\ \mathsf{A} \quad \begin{matrix}[\,\mathsf{A}\,(\mathsf{a}/\mathsf{x})\,]\\ \vdots \\ \mathsf{C}\end{matrix}}{\mathsf{C}}\ \exists\mathit{Elim}}$$

Important restrictions:

$\forall$*Intro* and $\exists$*Elim*: The (free) variable a (called an 'Eigenvariable', sometimes also a 'parameter') must be free for x in A.

$\forall$ *Intro*: a may not occur free in $\mathsf{A(x)}$, nor in any hypothesis on which the derivation depends, i. e., in any uncancelled hypothesis in the derivation of $\mathsf{A(a/x)}$.

$\exists$ *Elim*: the (free) variable a may not occur free in $\mathsf{A(x)}$, nor in C, nor in any hypothesis on which the derivation depends (i. e., uncancelled hypotheses in the derivations of $\exists \mathsf{x} : \mathsf{A}$ and C).

$\forall$ *Elim* and $\exists$ *Intro*: The term t must be free for x in A.

In $\forall$ *Elim* and $\exists$ *Intro* we allow substitutions of terms for *some* occurrences of the quantified variable! This allows the following deduction:

$$\dfrac{\dfrac{\forall \mathsf{x} : (\mathsf{x} = \mathsf{x})}{\mathsf{x} = \mathsf{x}}\ \forall\mathit{Elim}}{\exists \mathsf{y} : (\mathsf{x} = \mathsf{y})}\ \exists\mathit{Intro}$$

4.4 Natural deduction proofs

4.4.1 $\forall \mathsf{x} : \mathsf{A(x)} \vdash \exists \mathsf{x} : \mathsf{A(x)}$

$$\dfrac{\dfrac{\forall \mathsf{x} : \mathsf{A(x)}}{\mathsf{A(t)}}\ \forall\mathit{Elim}}{\exists \mathsf{x} : \mathsf{A(x)}}\ \exists\mathit{Intro}$$

Compare with the proof in Section 4.2.1.

4.4.2 $\forall \mathsf{x}_1 : \forall \mathsf{x}_2 : \mathsf{A}(\mathsf{x}_1, \mathsf{x}_2) \vdash \forall \mathsf{x}_2 : \forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1, \mathsf{x}_2)$

$$\dfrac{\dfrac{\dfrac{\dfrac{\forall \mathsf{x}_1 : \forall \mathsf{x}_2 : \mathsf{A}(\mathsf{x}_1, \mathsf{x}_2)}{\forall \mathsf{x}_2 : \mathsf{A}(\mathsf{y}_1, \mathsf{x}_2)}\ \forall\mathit{Elim}}{\mathsf{A}(\mathsf{y}_1, \mathsf{y}_2)}\ \forall\mathit{Elim}}{\forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1, \mathsf{y}_2)}\ \forall\mathit{Intro}\ (\mathsf{y}_1 \text{ is free for } \mathsf{x}_1 \text{ in } \mathsf{A})}{\forall \mathsf{x}_2 : \forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1, \mathsf{x}_2)}\ \forall\mathit{Intro}\ (\mathsf{y}_2 \text{ is free for } \mathsf{x}_2 \text{ in } \mathsf{A})$$

Compare with the proof in Section 4.2.2

Exercise 13: Which of the following derivations for $\forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1) \vdash \forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1)$ are correct, and which are not? Explain.

$$\text{a)}\ \dfrac{\dfrac{\forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1)}{\mathsf{A}(0)}}{\forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1)} \qquad \text{b)}\ \dfrac{\dfrac{\forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1)}{\mathsf{A}(\mathsf{x}_1)}}{\forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1)} \qquad \text{c)}\ \dfrac{\dfrac{\forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1)}{\mathsf{A}(\mathsf{x}_2)}}{\forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1)}$$

4.4.3 $\vdash \forall \mathsf{x}(\mathsf{A(x)} \supset \mathsf{B}) \supset (\exists \mathsf{x} : \mathsf{A(x)} \supset \mathsf{B})$, where x is not free in B.

$$\dfrac{\dfrac{\dfrac{[\exists \mathsf{x} : \mathsf{A(x)}]_2 \qquad \dfrac{\dfrac{[\forall \mathsf{x} : (\mathsf{A(x)} \supset \mathsf{B})]_3}{\mathsf{A(x)} \supset \mathsf{B}}\ \forall\mathit{Elim} \qquad [\mathsf{A(x)}]_1}{\mathsf{B}}\ {\supset}\mathit{Elim}}{\mathsf{B}}\ \exists\mathit{Elim}_1\ (\mathsf{x} \text{ is not free in } \mathsf{B})}{\exists \mathsf{x} : \mathsf{A(x)} \supset \mathsf{B}}\ {\supset}\mathit{Intro}_2}{\forall \mathsf{x} : (\mathsf{A(x)} \supset \mathsf{B}) \supset (\exists \mathsf{x} : \mathsf{A(x)} \supset \mathsf{B})}\ {\supset}\mathit{Intro}_3$$

4.5 Natural deduction rules for equality

If we have a language with equality, we add the following rules (*van Dalen 1994, p. 99*).

$$\boxed{\dfrac{}{\mathsf{x} = \mathsf{x}}\ \mathit{Eq}_1} \qquad \boxed{\dfrac{\mathsf{x} = \mathsf{y}}{\mathsf{y} = \mathsf{x}}\ \mathit{Eq}_2} \qquad \boxed{\dfrac{\mathsf{x} = \mathsf{y} \quad \mathsf{y} = \mathsf{z}}{\mathsf{x} = \mathsf{z}}\ \mathit{Eq}_3}$$

$$\boxed{\dfrac{\mathsf{x}_1 = \mathsf{y}_1, \ldots, \mathsf{x}_n = \mathsf{y}_n}{\mathsf{t}(\mathsf{x}_1, \ldots, \mathsf{x}_n) = \mathsf{t}(\mathsf{y}_1, \ldots, \mathsf{y}_n)}\ \mathit{Eq}_4} \qquad \boxed{\dfrac{\mathsf{x}_1 = \mathsf{y}_1, \ldots, \mathsf{x}_n = \mathsf{y}_n \quad \mathsf{A}(\mathsf{x}_1, \ldots, \mathsf{x}_n)}{\mathsf{A}(\mathsf{y}_1, \ldots, \mathsf{y}_n)}\ \mathit{Eq}_5}$$

The $\mathsf{y}_1, \ldots, \mathsf{y}_n$ must be free for $\mathsf{x}_1, \ldots, \mathsf{x}_n$ in the formula A. We also allow substitution of the variables y_i ($i \leq n$) for *some* and not necessarily all occurrences of the variable x_i.

4.6 GEB calculus

[*Note:* GEB *uses 'u' for '*x*' and 'x' for '*A*'; see pp. 217–218.*]

Rule of Specification: Suppose x is a variable which occurs inside the string A. If the string $\forall \mathsf{x} : \mathsf{A}$ is a theorem, then so is A, and so are any strings made from A by replacing x, wherever it occurs, by one and the same term t.

(*Restriction:* The term which replaces x must not contain any variable that is quantified in A.) [*See* GEB*, p. 220 for a justification*].

Rule of Generalization: Suppose A is a theorem in which x, a variable, occurs free. Then $\forall \mathsf{x} : \mathsf{A}$ is a theorem.

(*Restriction:* No generalization is allowed in a fantasy on any variable which appeared free in the fantasy's premise.) [*See* GEB*, p. 220 for a justification.*]

Rule of Interchange: Suppose x is a variable. Then the strings $\forall \mathsf{x} : \sim$ and $\sim\exists \mathsf{x}$ are interchangeable anywhere inside any theorem.

Rule of Existence: Suppose a term (which may contain variables as long as they are free) appears once, or multiply, in a theorem. Then any (or several, or all) of the appearances of the term may be replaced by a variable which otherwise does not occur in the theorem, and the corresponding existential quantifier must be placed in front.

Exercise 14: Carefully compare these rules with the inference rule of the Natural Deduction calculus in Section 4.5.

In the following, $\mathsf{s}, \mathsf{t}, \mathsf{u}$ are terms. [*See* GEB*, p. 219.*]

Rules of Equality:

Symmetry: If $\mathsf{t} = \mathsf{s}$ is a theorem, then so is $\mathsf{s} = \mathsf{t}$.

Transitivity: If $\mathsf{t} = \mathsf{s}$ and $\mathsf{s} = \mathsf{u}$ are theorems, then so is $\mathsf{t} = \mathsf{u}$.

Rules of Successorship:

Add S: If $\mathsf{t} = \mathsf{u}$ is a theorem, then $\mathsf{St} = \mathsf{Su}$ is a theorem.

Drop S: If $\mathsf{St} = \mathsf{Su}$ is a theorem, then $\mathsf{t} = \mathsf{u}$ is a theorem.

4.7 Equivalence of axiomatic, Natural Deduction, and GEB calculus

As in the case of propositional logic, it is possible to show that these calculi are equivalent.

Exercise 15: Think about how one would go about showing this equivalence.

5 Soundness and completeness for first-order logic

Just as for propositional logic, one can also show for first-order logic, for any set Γ of formulas:

Soundness: If $\Gamma \vdash \mathsf{A}$, then $\Gamma \models \mathsf{A}$.

Completeness: If $\Gamma \models \mathsf{A}$, then $\Gamma \vdash \mathsf{A}$. (Gödel, 1930)

In both theorems '$\vdash$' can stand for deduction in the axiomatic calculus, Natural Deduction calculus, or the GEB-system.

6 Axiom systems for arithmetic

So far, we have only seen the axioms for first-order predicate *logic*. However, to formalize any particular *theory*, we need to introduce additional axioms that involve the primitives in our language.

Clearly, from logic alone you cannot derive any propositions about *geometry*, but from logic together with Euclid's axioms you can.

In the rest of this course we will study theories of *arithmetic*, and therefore we will need some axioms about the natural numbers.

Here is a version of the axioms of Dedekind-Peano Arithmetic (DPA): [*See* GEB*, p. 216.*]
[*With the* Drop S *rule (p. 219) Axiom 2 become superfluous.*]

1. $\forall \mathsf{x} : \sim \mathsf{Sx} = 0$
2. $\forall \mathsf{x} : \forall y : \mathsf{Sx} = \mathsf{Sy} \supset \mathsf{x} = \mathsf{y}$
3. $\forall \mathsf{x} : (\mathsf{x} + 0) = \mathsf{x}$
4. $\forall \mathsf{x} : \forall \mathsf{y} : (\mathsf{x} + \mathsf{Sy}) = \mathsf{S}(\mathsf{x} + \mathsf{y})$
5. $\forall \mathsf{x} : (\mathsf{x} \cdot 0) = 0$
6. $\forall \mathsf{x} : \forall \mathsf{y} : (\mathsf{x} \cdot \mathsf{Sy}) = ((\mathsf{x} \cdot \mathsf{y}) + \mathsf{x})$

6. *Axiom schema of Induction*: $\mathsf{A}(0) \supset \big(\forall \mathsf{x} : \big(\mathsf{A}(\mathsf{x}) \supset \mathsf{A}(\mathsf{Sx})\big) \supset \forall \mathsf{x} : \mathsf{A}(\mathsf{x})\big)$

   Hofstadter presents this as the *Rule of Induction* (GEB, p. 224): Suppose x is a variable, and $\mathsf{A}(\mathsf{x})$ ($X\{u\}$ in GEB) is a well-formed formula in which x occurs free.

   If both $\underbrace{\forall \mathsf{x} : (\mathsf{A}(\mathsf{x}) \supset \mathsf{A}(\mathsf{Sx}/\mathsf{x}))}_{\text{Induction step}}$ and $\underbrace{\mathsf{A}(0/\mathsf{x})}_{\text{Base case}}$ are theorems, then $\forall \mathsf{x} : \mathsf{A}(\mathsf{x})$ is also a theorem.

Without Axiom 6, the system is ω-incomplete:

A system is ω-incomplete if all the strings in a *pyramidal family* are theorems, but the universally quantified *summarizing string* is not a theorem.

Example: Pyramidal family

$$\begin{aligned} (0{+}0) &= 0 \\ (0{+}\mathsf{S}0) &= \mathsf{S}0 \\ (0{+}\mathsf{SS}0) &= \mathsf{SS}0 \\ (0{+}\mathsf{SSS}0) &= \mathsf{SSS}0 \\ &\text{etc.} \end{aligned}$$

with summarizing string $\forall \mathsf{x} : (0 + \mathsf{x}) = \mathsf{x}$.

7 Proofs in first-order Dedekind-Peano-Arithmetic

Let Π be the set of axioms for DPA from Section 6.

Prove: $\Pi \vdash \forall \mathsf{x}_1 : (\mathsf{0} \cdot \mathsf{x}_1 = \mathsf{0})$**.** For the sake of abbreviation, let $\mathsf{A}(\mathsf{x}_1)$ be '$\mathsf{0} \cdot \mathsf{x}_1 = \mathsf{0}$'.

(1)	$\Pi \vdash$	$\mathsf{A}(\mathsf{0}) \supset [\forall \mathsf{x}_1 : \big(\mathsf{A}(\mathsf{x}_1) \supset \mathsf{A}(\mathsf{S}\mathsf{x}_1)\big) \supset \forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1)]$	Induction axiom 6.
(2)	$\Pi \vdash$	$\forall \mathsf{x}_1 : (\mathsf{x}_1 \cdot \mathsf{0} = \mathsf{0})$	Axiom 4
(3)	$\Pi \vdash$	$\mathsf{0} \cdot \mathsf{0} = \mathsf{0}$	Specification on (2) $(\mathsf{0}/\mathsf{x}_1)$
(4)	$\Pi \vdash$	$\mathsf{A}(\mathsf{0})$	Definition of A
(5)	$\Pi \vdash$	$\forall \mathsf{x}_1 : \big(\mathsf{A}(\mathsf{x}_1) \supset \mathsf{A}(\mathsf{S}\mathsf{x}_1)\big) \supset \forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1)$	Modus Ponens on (1) and (4)
(6)	$\Pi, \mathsf{A}(\mathsf{x}_1) \vdash$	$\mathsf{A}(\mathsf{x}_1)$	Clearly
(7)	$\Pi, \mathsf{A}(\mathsf{x}_1) \vdash$	$\mathsf{0} \cdot \mathsf{x}_1 = \mathsf{0}$	Definition of A
(8)	$\Pi \vdash$	$\forall \mathsf{x}_1 : \forall \mathsf{x}_2 : (\mathsf{x}_1 \cdot \mathsf{S}\mathsf{x}_2 = (\mathsf{x}_1 \cdot \mathsf{x}_2) + \mathsf{x}_1)$	Axiom 5
(9)	$\Pi \vdash$	$\mathsf{0} \cdot \mathsf{S}\mathsf{x}_1 = \mathsf{0} \cdot \mathsf{x}_1 + \mathsf{0}$	Specification on (8) $(\mathsf{0}/\mathsf{x}_1, \mathsf{x}_1/\mathsf{x}_2)$
(10)	$\Pi, \mathsf{A}(\mathsf{x}_1) \vdash$	$\mathsf{0} \cdot \mathsf{S}\mathsf{x}_1 = \mathsf{0} + \mathsf{0}$	Rules for $=$ on (7) and (9)
(11)	$\Pi, \vdash$	$\forall \mathsf{x}_1 : (\mathsf{x}_1 + \mathsf{0} = \mathsf{x}_1)$	Axiom 2
(12)	$\Pi, \vdash$	$\mathsf{0} + \mathsf{0} = \mathsf{0}$	Specification on (11) $(\mathsf{0}/\mathsf{x}_1)$
(13)	$\Pi, \mathsf{A}(\mathsf{x}_1) \vdash$	$\mathsf{0} \cdot \mathsf{S}\mathsf{x}_1 = \mathsf{0}$	Rules for $=$ on (10) and (12)
(14)	$\Pi, \mathsf{A}(\mathsf{x}_1) \vdash$	$\mathsf{A}(\mathsf{S}\mathsf{x}_1)$	Definition of A
(15)	$\Pi \vdash$	$\mathsf{A}(\mathsf{x}_1) \supset \mathsf{A}(\mathsf{S}\mathsf{x}_1)$	Deduction Theorem (Prop. Logic)
(14)	$\Pi \vdash$	$\forall \mathsf{x}_1 : \big(\mathsf{A}(\mathsf{x}_1) \supset \mathsf{A}(\mathsf{S}\mathsf{x}_1)\big)$	Generalization on (15) (x_1 not free in Π)
(15)	$\Pi \vdash$	$\forall \mathsf{x}_1 : \mathsf{A}(\mathsf{x}_1)$	Modus Ponens on (5) and (14)
(16)	$\Pi \vdash$	$\forall \mathsf{x}_1 : (\mathsf{0} \cdot \mathsf{x}_1 = \mathsf{0})$	Definition of A $\square$

Notice, how lines (2)–(5) correspond to the *base case*, lines (6)–(14) to the *induction step*, and lines (15)–(16) to the *conclusion* of an argument by weak induction!

Additional problems:

1. $\Pi \vdash \forall \mathsf{x}_1 : (\mathsf{S0} \cdot \mathsf{x}_1 = \mathsf{x}_1)$.
2. $\Pi \vdash \forall \mathsf{x}_1 : \forall \mathsf{x}_2 : \forall \mathsf{x}_3 : \big((\mathsf{x}_1 + \mathsf{x}_2) + \mathsf{x}_3 = \mathsf{x}_1 + (\mathsf{x}_2 + \mathsf{x}_3)\big)$.

   Suggestion: Use the induction schema on x_3 and let x_1 and x_2 be arbitrary terms, so you can use the *Rule of Specification* at the end after you've applied the induction schema. Thus, your $\mathsf{A}(\mathsf{x}_3)$ will be "$(\mathsf{x}_1 + \mathsf{x}_2) + \mathsf{x}_3 = \mathsf{x}_1 + (\mathsf{x}_2 + \mathsf{x}_3)$."
3. $\Pi \vdash \forall \mathsf{x}_1 : \forall \mathsf{x}_2 : (\mathsf{x}_1 + \mathsf{x}_2 = \mathsf{x}_2 + \mathsf{x}_1)$. [*See* GEB, *pp .225–227.*]