Preliminaries

The only prerequisite is a knowledge of naive set theory and familiarity with basic logic notation.

Terms. Terms are formed from variables and constants by applications of functions in the usual way. Closed terms do not have free variables. We use lower Greek letters τ, ρ, θ as syntactic variables ranging over terms. We will also use the symbol $\equiv$ as the syntactical identity over terms (and similarly over other syntactic domains).

Notational conventions for terms. We will write $\vec{x}$ in contexts like $f(\vec{x})$, where f is an n-ary function symbol, as an abbreviation for a sequence of n variables $x_1, \ldots, x_n$, i.e. we have $f(\vec{x}) \equiv f(x_1, \cdots, x_n)$. Generally, $f(\vec{\tau})$ will be an abbreviation for $f(\tau_1, \ldots, \tau_n)$, where $\vec{\tau}$ is the sequence $\tau_1, \ldots, \tau_n$ of terms. We will also write $f\,g(\vec{\tau})$ instead of $f(g(\vec{\tau}))$.

When we write $\tau[f; \vec{x}]$ we indicate that the term τ may apply the n-ary function symbol f and variables from among the m-variables $\vec{x}$. For an n-ary function symbol g and for an m-tuple of terms $\vec{\rho}$ we write $\tau[g; \vec{\rho}]$ for the term obtained from τ by the substitution of terms $\vec{\rho}$ for the corresponding variables of $\vec{x}$ as well as by the replacement of all applications $f(\vec{\tau})$ by applications $g(\vec{\tau})$.

Formulas. An atomic formula is either a predicate application or identity $\tau = \rho$. Formulas are formed from atomic formulas and propositional constants by applications of propositional connectives and quantifiers in the usual way:

$\top$ (true)	$\varphi \land \psi$ (conjunction)	$\varphi \leftrightarrow \psi$ (equivalence)
$\bot$ (falsehood)	$\varphi \lor \psi$ (disjunction)	$\forall x \varphi$ (universal quantifier)
$\neg\varphi$ (negation)	$\varphi \to \psi$ (implication)	$\exists x \varphi$ (existential quantifier).

Closed formulas (i.e. *sentences*) do not have free variables. We will use lower Greek letters φ, ψ as syntactic variables ranging over formulas.

Notational conventions for formulas. We let all binary propositional connectives in formulas group to the right. We assign the highest precedence to the quantifiers and negation. Next lower precedence has the conjunction and then the disjunction. The connectives of implication and equivalence have the lowest precedence.

By $\tau \neq \rho$ we designate the formula $\neg\tau = \rho$. We generalize some of the propositional connectives to for finite sequences. The *generalized conjunction* $\bigwedge_{i=1}^{n} \varphi_i$ stands for $\varphi_1 \wedge \cdots \wedge \varphi_n$ if $n \geq 1$ and for $\top$ if $n = 0$. We define the *generalized disjunction* $\bigvee_{i=1}^{n} \varphi_i$ similarly.

By $\forall \vec{x} \varphi$ and $\exists \vec{x} \varphi$ we designate the formulas $\forall x_1 \ldots \forall x_n \varphi$ and $\exists x_1 \ldots \exists x_n \varphi$, respectively. By $\forall \varphi$ we denote the *universal closure* of the formula φ.

Bounded quantifiers are formulas of the form $\forall x \leq \tau\, \varphi$ and $\exists x \leq \tau\, \varphi$, where the variable x is not free in τ. The bounded quantifiers abbreviate the formulas $\forall x(x \leq \tau \rightarrow \varphi)$ and $\exists x(x \leq \tau \wedge \varphi)$, respectively. *Strict* bounded quantifiers $\forall x < \tau\, \varphi$ and $\exists x < \tau\, \varphi$ are similar.

Similar conventions as those for terms will be adopted also for formulas. Only substitution requires a brief explanatation. Whenever we write $\varphi[\vec{\tau}]$ it is assumed that the bound variables of the formula $\varphi[\vec{x}]$ are first renamed so that they do not appear in the terms $\vec{\tau}$. Recall that a formula does not change its meaning if one of its bound variables is changed to another.

Natural numbers. If we do not state explicitly n-ary functions and predicates are over the domain of natural numbers

$$\mathrm{N} = \{0, 1, 2, 3, 4, 5, \ldots\}.$$

We implicitly assume that we have $n \geq 1$; this means that our functions and predicates have always non-zero arity. Furthermore, n-ary functions are always *total*, i.e. with the domain the whole cartesian product N^n.

Natural numbers are closed under the operations of addition $x + y$ and multiplication $x \times y$ (written xy for short) but not under subtraction $x - y$ and division $\frac{x}{y}$. For instance, we have $3 - 5 = -2 < 0$ and $1 < \frac{5}{3} < 2$.

Instead of subtraction we will use *modified subtraction* $x \dot{-} y$ which is over natural numbers and it is defined by

$$x \dot{-} y = \begin{cases} x - y & \text{if } x \geq y, \\ 0 & \text{otherwise.} \end{cases}$$

The modified subtraction has the following basic properties:

$$y \leq x \rightarrow x = y + (x \dot{-} y) \qquad x \leq y \rightarrow x \dot{-} y = 0.$$

Note that we then have $5 \dot{-} 3 = 2$ and $3 \dot{-} 5 = 0$.

Instead of division we use *euclidean division*. Recall that for every natural numbers x and $y \neq 0$ there exist unique natural numbers q and $r < y$ such that $x = qy + r$ holds. The numbers q and r are called respectively the *quotient* and the *remainder* of the euclidean division of x by y. We denote by $x \div y$ the binary *integer division* function and by $x \bmod y$ the binary *remainder function* yielding respectively the quotient and remainder of the euclidean division of the number x by y. The functions are defined to satisfy:

$$x \div y = \begin{cases} q & \text{if } y \neq 0 \text{ and } x = qy + r \text{ for some } r < y, \\ 0 & \text{otherwise.} \end{cases}$$

$$x \bmod y = \begin{cases} r & \text{if } y \neq 0 \text{ and } x = qy + r \text{ for some } q \text{ such that } r < y, \\ 0 & \text{otherwise.} \end{cases}$$

The functions have the following basic properties:

$$x \div 0 = x \bmod 0 = 0$$
$$y \neq 0 \to x = (x \div y)y + x \bmod y \wedge x \bmod y < y.$$

For instance, we have $5 \div 3 = 1$ and $5 \bmod 3 = 2$.

The binary exponentiation function x^y has a following recursive definition:

$$x^0 = 1$$
$$x^{y+1} = xx^y.$$

Note that we have

$$x^y = 0 \leftrightarrow x = 0 \wedge y \neq 0 \qquad x^y = 1 \leftrightarrow x = 1 \vee y = 0.$$

The binary *divisibility* predicate $x \mid y$ holds if the number x divides the number y, i.e. if we have $y = xz$ for some number z.

Characteristic functions of predicates. Let R be an n-ary predicate. We denote by R_* its characteristic function which is an n-ary function such that

$$R_*(\vec{x}) = \begin{cases} 1 & \text{if } R(\vec{x}), \\ 0 & \text{if not } R(\vec{x}). \end{cases}$$

Note that the value 1 of the characteristic function means truth, while the value 0 means falsehood.

We denote by $x =_* y$, $x \neq_* y$, $x \leq_* y$ and $x <_* y$ the characteristic functions of the binary predicates $x = y$, $x \neq y$, $x \leq y$ and $x < y$, respectively.

References and meta-logical notation. Chapters are divided into sections, and these into consecutively numbered paragraphs such as definitions, theorems and remarks. Thus 2.3.4 is the 4th paragraph of the 3rd section of the 2nd chapter. When a reference is made to a numbered equation within the same paragraph, both chapter and section numbers are omitted.

The word "iff" abreviates "if and only if"; "s.t." abbreviates "such that"; "IH" abreviates "induction hypothesis" and "IHs" is the plural form of "IH". The symbol $\Rightarrow$ denotes the word "implies", while the symbol $\Leftrightarrow$ means "implies and is implied by". Finally note that the conclusion of a proof is usually indicated by the symbol $\square$.