# Syntax of predicate logic: variables and terms

Variables are expressions of the form $x_i$ mit $i = 1, 2, 3 \ldots$.
Predicate symbols are expressions of the form $P_i^k$, where $i = 1, 2, 3 \ldots$ and $k = 0, 1, 2 \ldots$.
Function symbols are expressions of the form $f_i^k$, where $i = 1, 2, 3 \ldots$ und $k = 0, 1, 2 \ldots$.
We call $i$ the (identification) index and $k$ die arity of the symbol.
Terms are inductively defined as follows:

(1) Variables are terms.

(2) Function symbols of arity $0$ are terms.

(3) If $f$ is a function symbol with arity $k \geq 1$ and $t_1, \ldots, t_k$ are terms then $f(t_1, \ldots, t_k)$ is a term.

Function symbols of arity $0$ are called constants.

# Syntax of predicate logic: formulas

Formulas (of predicate logic) are inductively defined as follows:.

(1) Predicate symbols of arity $0$ are formulas.

(2) If $P$ is a predicate symbol of arity $k \geq 1$ and $t_1, \ldots, t_k$ are terms then $P(t_1, \ldots, t_k)$ is a formula.

(3) If $F$ is a formula, then $\neg F$ is also a formula.

(4) If $F$ and $G$ are formulas, then $(F \wedge G)$ and $(F \vee G)$ are also formulas.

(5) If $x$ is a variable and $F$ is a formula, then $\exists x\ F$ and $\forall x\ F$ are also formulas. The symbols $\exists$ and $\forall$ are called the existential and the universal quantifier, respectively.

Formulas of the form $P$ for some predicate symbol of arity $0$ or of the form $P(t_1, \ldots, t_k)$ are called atomic formulas. The syntax tree and the subformulas of a formula are defined as usual.

# Free and bounded variables, closed formulas

A variable $x$ occurs in a formula $F$ if it appears in some term of $F$.

An occurrence of a variable in a formula is either free or bounded.

An occurrence of $x$ in $F$ is bounded if it belongs to some subformula of $F$ of the form $\exists x G$ or $\forall x G$; the smallest such subformula is the scope of the occurrence. Otherwise the occurrence is free.

A formula without any free occurrence of any variable is closed.

The matrix of a formula $F$ is the formula obtained by removing from $F$ every occurrence of the quantifiers $\exists$ and $\forall$, together with the (occurrence of a) variable following them. The matrix of $F$ is denoted by $F^*$.

# Exercise

NF: non-formula  F: formula, but not closed  C: closed

| | NF | F | C |
|---|---|---|---|
| $\forall x\ P(a)$ | | | |
| $\forall x \exists y\ (Q(x,y) \vee R(x,y))$ | | | |
| $\forall x\ Q(x,x) \rightarrow \exists x\ Q(x,y)$ | | | |
| $\forall x\ P(x) \vee \forall x\ Q(x,x)$ | | | |
| $\forall x\ (P(y) \wedge \forall y\ P(x))$ | | | |
| $P(x) \rightarrow \exists x\ Q(x, P(x))$ | | | |
| $\forall f\ \exists x\ P(f(x))$ | | | |

NF: non-formula     F: formula, but not closed     C: closed

| | NF | F | C |
|---|---|---|---|
| $\forall x\ (\neg\forall y\ Q(x,y) \wedge R(x,y))$ | | | |
| $\exists z\ (Q(z,x) \vee R(y,z)) \rightarrow \exists y\ (R(x,y) \wedge Q(x,z))$ | | | |
| $\exists x\ (\neg P(x) \vee P(f(a)))$ | | | |
| $P(x) \rightarrow \exists x\ P(x)$ | | | |
| $\exists x \forall y\ ((P(y) \rightarrow Q(x,y)) \vee \neg P(x))$ | | | |
| $\exists x \forall x\ Q(x,x)$ | | | |

# Semantics of predicate logic: structures

A structure is a pair $\mathcal{A} = (U_{\mathcal{A}}, I_{\mathcal{A}})$, where $U_{\mathcal{A}}$ is an arbitrary, nonempty set called the ground set or universe of $\mathcal{A}$, and $I_{\mathcal{A}}$ is a partial function that maps

- predicate symbols of arity $k \geq 1$ to predicates over $U_{\mathcal{A}}$ of arity $k$ (i.e., to functions of type $U_{\mathcal{A}}^k \to \{0, 1\}$ or, equivalently, to subsets of $U_{\mathcal{A}}^k$),
- predicate symbols of arity $0$ to either $0$ or $1$
- function symbols of arity $k \geq 1$ to functions over $U_{\mathcal{A}}$ of arity $k$ (i.e., to functions of type $U_{\mathcal{A}}^k \to U_{\mathcal{A}}$),
- constants $f$ of arity $0$ to elements of the universe $U_{\mathcal{A}}$, and
- variables $x$ to elements of the universe $U_{\mathcal{A}}$.

In other words:

- The domain of $I_{\mathcal{A}}$ is a subset of $\{P_i^k, f_i^k, x_i \mid i = 1, 2, 3, \ldots, \; k = 0, 1, 2., \ldots\}$.
- The image of $I_{\mathcal{A}}$ is a subset of the set of all predicates and functions over $U_{\mathcal{A}}$ and elements of $U_{\mathcal{A}}$.

We abbreviate $I_{\mathcal{A}}(P)$ to $P^{\mathcal{A}}$, $I_{\mathcal{A}}(f)$ to $f^{\mathcal{A}}$, and$I_{\mathcal{A}}(x)$ to $x^{\mathcal{A}}$.

Let $F$ be a formula and let $\mathcal{A} = (U_{\mathcal{A}}, I_{\mathcal{A}})$ be a structure.
$\mathcal{A}$ is suitable for $F$ if all predicate and function symbols occurring in $F$ and all variables occurring free in $F$ belong to the domain of $I_{\mathcal{A}}$.

# Evaluation of a formula in a structure

Let $F$ be a formula and let $\mathcal{A}$ be a structure suitable for $F$. For every term $t$ that can be constructed from variables and function symbols that appear in $F$, we define the value of $t$ in the structure $\mathcal{A}$, denoted by $\mathcal{A}(t)$. The definition is inductive:

(1) If $t = x$ for some variable $x$), then $\mathcal{A}(t) = x^{\mathcal{A}}$.

(2) If $t = f(t_1, \ldots, t_k)$ for some function symbol $f$ of arity $k$ and terms $t_1, \ldots, t_k$, then $\mathcal{A}(t) = f^{\mathcal{A}}(\mathcal{A}(t_1), \ldots, \mathcal{A}(t_k))$.

(3) If $t = a$ for some constant $a$, then $\mathcal{A}(t) = a^{\mathcal{A}}$.

Analogously, we define inductively the (truth-)value of a formula $F$ in the structure $\mathcal{A}$, denoted by $\mathcal{A}(F)$:

- If $F = P(t_1, \dots, t_k)$ for some predicate symbol $P$ of arity $k$ and terms $t_1, \dots, t_k$ then

$$\mathcal{A}(F) = \begin{cases} 1 & \text{if } (\mathcal{A}(t_1), \dots, \mathcal{A}(t_k)) \in P^{\mathcal{A}} \\ 0 & \text{otherwise} \end{cases}$$

- If $F = \neg G$ for some formula $G$ then

$$\mathcal{A}(F) = \begin{cases} 1 & \text{if } \mathcal{A}(G) = 0 \\ 0 & \text{otherwise} \end{cases}$$

- If $F = (G \wedge H)$ for some formulas $G$ and $H$ then

$$\mathcal{A}(F) = \begin{cases} 1 & \text{if } \mathcal{A}(G) = 1 \text{ and } \mathcal{A}(H) = 1 \\ 0 & \text{otherwise} \end{cases}$$

- If $F = (G \vee H)$ for some formulas $G$ and $H$ then

$$\mathcal{A}(F) = \begin{cases} 1 & \text{if } \mathcal{A}(G) = 1 \text{ or } \mathcal{A}(H) = 1 \\ 0 & \text{otherwise} \end{cases}$$

- If $F = \forall x\ G$ for some formula $G$ and variable $x$ then

$$\mathcal{A}(F) = \begin{cases} 1 & \text{if for every } d \in U_{\mathcal{A}}:\ \mathcal{A}_{[x/d]}(G) = 1 \\ 0 & \text{otherwise} \end{cases}$$

- If $F = \exists x\ G$ for some formula $G$ and variable $x$ then

$$\mathcal{A}(F) = \begin{cases} 1 & \text{if there exists } d \in U_{\mathcal{A}} \text{ such that: } \mathcal{A}_{[x/d]}(G) = 1 \\ 0 & \text{otherwise} \end{cases}$$

where $\mathcal{A}_{[x/d]}$ denotes the structure $\mathcal{A}'$ that coincides with $\mathcal{A}$ everywhere, but (possibly) in the definition of $x^{\mathcal{A}'}$: it holds $x^{\mathcal{A}'} = d$, whether $x$ belongs to the domain of $I_{\mathcal{A}}$ or not.

# Model, validity, satisfiability

We write $\mathcal{A} \models F$ to denote that the structure $\mathcal{A}$ is suitable for the formula $F$ and $\mathcal{A}(F) = 1$ holds. We say that $F$ holds in $\mathcal{A}$ or that $\mathcal{A}$ is a model of $F$.

If every structure suitable for $F$ is a model of $F$, then we write $\models F$ and say that $F$ ist valid.

If $F$ has at least one model then we say that $F$ is satisfiable.

# Exercise

V: valid S: satisfiable, but not valid U: unsatisfiable

| | V | S | U |
|---|---|---|---|
| $\forall x\ P(a)$ | | | |
| $\exists x\ (\neg P(x) \vee P(a))$ | | | |
| $P(a) \rightarrow \exists x\ P(x)$ | | | |
| $P(x) \rightarrow \exists x\ P(x)$ | | | |
| $\forall x\ P(x) \rightarrow \exists x\ P(x)$ | | | |
| $\forall x\ P(x) \wedge \neg \forall y\ P(y)$ | | | |

V: valid S: satisfiable, but not valid U: unsatisfiable

| | V | S | U |
|---|---|---|---|
| $\forall x\ (P(x,x) \rightarrow \exists x \forall y\ P(x,y))$ | | | |
| $\forall x \forall y\ (x = y \rightarrow f(x) = f(y))$ | | | |
| $\forall x \forall y\ (f(x) = f(y) \rightarrow x = y)$ | | | |
| $\exists x \exists y \exists z\ (f(x) = y \land f(x) = z \land y \neq z)$ | | | |

# Consequence and equivalence

A formula $G$ is a consequence of the formulas $F_1, \ldots, F_k$ if every structure suitable for $F_1, \ldots, F_k$ and for $G$ that is model of $\{F_1, \ldots, F_k\}$ is also model of $G$.

We write $F_1, \ldots, F_k \models G$ to denote that $G$ is a consequence of $F_1, \ldots, F_k$.

Two formulas $F$ and $G$ are (semantically) equivalent if every structure $\mathcal{A}$ suitable for both $F$ and $G$ satisfies $\mathcal{A}(F) = \mathcal{A}(G)$. We then write $F \equiv G$.

# Exercise

(1) $\forall x \; P(x) \vee \forall x \; Q(x, x)$

(2) $\forall x \; (P(x) \vee Q(x, x))$

(3) $\forall x \; (\forall z P(z) \vee \forall y \; Q(x, y))$

| | Y | N |
|---|---|---|
| $1 \models 2$ | | |
| $2 \models 3$ | | |
| $3 \models 1$ | | |

# Exercise

(1) $\exists y \forall x \; P(x, y)$

(2) $\forall x \exists y \; P(x, y)$

| | Y | N |
|---|---|---|
| $1 \models 2$ | | |
| $2 \models 1$ | | |

# Exercise

| | Y | N |
|---|---|---|
| $\forall x \forall y\ F \equiv \forall y \forall x\ F$ | | |
| $\forall x \exists y\ F \equiv \exists x \forall y\ F$ | | |
| $\exists x \exists y\ F \equiv \exists y \exists x\ F$ | | |
| $\forall x\ F \vee \forall x\ G \equiv \forall x\ (F \vee G)$ | | |
| $\forall x\ F \wedge \forall x\ G \equiv \forall x\ (F \wedge G)$ | | |
| $\exists x\ F \vee \exists x\ G \equiv \exists x\ (F \vee G)$ | | |
| $\exists x\ F \wedge \exists x\ G \equiv \exists x\ (F \wedge G)$ | | |

# Predicate logic with equality

Predicate logic

$+$

distinguished predicate symbol "=" of arity 2.

Semantics : a structure $\mathcal{A}$ of predicate logic with equality always maps the predicate symbol = to the identity relation:

$$\mathcal{A}(=) \;=\; \{(d,d) \mid d \in U_{\mathcal{A}}\}\,.$$

# Formalizing statements

A statement in natural language is formalized as a formula $F$ and a structure $\mathcal{A}$. The formalizer claims that the statement is true iff $F$ is true in some adequate structure extending $\mathcal{A}$.

$\mathcal{A}$ fixes the meaning of the predicates that are taken as known. $F$ may contain definitions of predicates or functions (see example in the next slides).

The names of the predicate symbols are chosen to suggest their meaning in the structure. The structure is often omitted, because it is assumed to be known (danger!).

We consider the following example

There are infinitely many prime numbers

# Formalization I

If the meaning of "prime" and "'greater-than" are assumed to be known, then we can take

Formula $F_1$: $\forall x \exists y \ (Pri(y) \wedge Gt(y, x))$

Structure $\mathcal{A}_1$:

$$\begin{array}{rcl} U_{\mathcal{A}_1} & = & \mathbb{N} \\ Pri^{\mathcal{A}_1} & = & \{n \in \mathbb{N} \mid n \text{ is prime}\} \\ Gt^{\mathcal{A}_1} & = & \{(n, m) \in \mathbb{N} \mid n > m\} \end{array}$$

What if the meaning of "prime" is not clear to everybody?

# Formalization II

If the meaning of "divides" is known , then we can take

Formula $F_2$: $F_1 \wedge \forall x \, (Pri(x) \leftrightarrow \forall y \; Div(y,x) \rightarrow (y = x \vee y = one))$

Structure $\mathcal{A}_2$:

$$
\begin{aligned}
U_{\mathcal{A}_2} &= \mathbb{N} \\
Gt^{\mathcal{A}_2} &= \{(n,m) \in \mathbb{N} \mid n > m\} \\
Div^{\mathcal{A}_2} &= \{(n,m) \in \mathbb{N} \mid n \text{ divides } m\} \\
one^{\mathcal{A}_2} &= 1
\end{aligned}
$$

$\mathcal{A}_2$ does not interpret $Pri$, but an structure that extends $\mathcal{A}_2$ and interprets $Pri$ can only satisfy $F_2$ if it assigns to $Pri$ "the right meaning".

What if the meaning of "divides" is not clear to everybody?

# Formalization III

If the meaning of "product" is known , then we can take

Formula $F_3$: $F_2 \wedge \forall x \forall y \; (Div(x,y) \leftrightarrow \exists z \; prod(x,z) = y)$

Structure $\mathcal{A}_3$:

$$
\begin{aligned}
U_{\mathcal{A}_3} &= \mathbb{N} \\
Gt^{\mathcal{A}_3} &= \{(n,m) \in \mathbb{N} \mid n > m\} \\
one^{\mathcal{A}_3} &= 1 \\
prod^{\mathcal{A}_3}(n,m) &= n \cdot m
\end{aligned}
$$

What if the meaning of "product" is not clear to everybody?

# Formalization IV

If the meaning of "sum", "successor", "one" and "zero" is known, then we can take

Formula $F_4$: $F_3 \wedge F_4' \wedge F_4''$

$$
\begin{aligned}
F_4' &= \forall x \; prod(x, zero) = zero \\
F_4'' &= \forall x \forall y \; prod(x, succ(y)) = sum(prod(x, y), y)
\end{aligned}
$$

Structure $\mathcal{A}_4$:

$$
\begin{aligned}
U_{\mathcal{A}_4} &= \mathbb{N} \\
Gt^{\mathcal{A}_4} &= \{(n, m) \in \mathbb{N} \mid n > m\} \\
one^{\mathcal{A}_4} &= 1 \qquad zero^{\mathcal{A}_4} = 0 \\
sum^{\mathcal{A}_4}(n, m) &= n + m \\
succ^{\mathcal{A}_4}(n) &= n + 1
\end{aligned}
$$

What if the meaning of "sum" is not clear to everybody?

# Formalization V

We can take

Formula $F_5$: $F_4 \wedge F_5' \wedge F_5''$

$$
\begin{aligned}
F_5' &= \forall x \; sum(x, zero) = x \\
F_5'' &= \forall x \forall y \; sum(x, succ(y)) = succ(sum(x, y))
\end{aligned}
$$

Structure $\mathcal{A}_5$:

$$
\begin{aligned}
U_{\mathcal{A}_5} &= \mathbb{N} \\
Gt^{\mathcal{A}_5} &= \{(n, m) \in \mathbb{N} \mid n > m\} \\
one^{\mathcal{A}_5} &= 1 \qquad zero^{\mathcal{A}_5} = 0 \\
succ^{\mathcal{A}_5}(n) &= n + 1
\end{aligned}
$$

What if the meaning of ‘greater than” and “one” is not clear to everybody?

# Formalization VI

We can take

Formula $F_6$: $F_5 \wedge F_6' \wedge F_6''$

$$\begin{aligned} F_6' &= succ(zero) = one \\ F_6'' &= \forall x \forall y \; (Gt(x, y) \leftrightarrow \exists z \; (sum(y, z) = x \wedge \neg(z = zero))) \end{aligned}$$

Structure $\mathcal{A}_6$:

$$\begin{aligned} U_{\mathcal{A}_6} &= \mathbb{N} \\ zero^{\mathcal{A}_6} &= 0 \\ succ^{\mathcal{A}_6}(n) &= n + 1 \end{aligned}$$