Formale Methoden der Softwaretechnik
Formal methods of software engineering

Till Mossakowski, Christoph Lüth

SoSe 2011

Quantifiers: Motivating examples

$$\frac{\begin{array}{l}\forall x\ Cube(x)\ (\text{``All objects are cubes.''})\\ \forall x\ (Cube(x) \rightarrow Large(x))\ (\text{``All cubes are large.''})\end{array}}{\forall x\ Large(x)\ (\text{``All objects are large.''})}$$

$$\exists x\ Cube(x)$$

"There exists a cube."

$$\exists x\ (Cube(x) \wedge Large(x))$$

"There exists a large cube."

First-order signatures

A first-order signature consists of

- a set of predicate symbols with arities, like $Smaller^{(2)}$, $Dodec^{(1)}$, $Between^{(3)}$, $\leq^{(2)}$, including propositional symbols (nullary predicate symbols), like $A^{(0)}$, $B^{(0)}$, $C^{(0)}$, (written uppercase)
- its names or constants for individuals, like a, b, c, (written lowercase)
- its function symbols with arities, like $f^{(1)}, +^{(2)}, \times^{(2)}$.

Usually, arities are omitted.
In the book "language, proof and logic", the terminology "language" is used. "Signature" is more precise, since it exactly describes the ingredients that are needed to generate a (first-order) language.

Terms

$t ::= a$	constant
$t ::= x$	variable
$\mid f^{(n)}(t_1, \ldots, t_n)$	application of function symbols to terms

Usually, arities are omitted.
Variables are: t, u, v, w, x, y, z, possibly with subscripts.

Well-formed formulas

$$
\begin{array}{rll}
F ::= & P^{(n)}(t_1, \ldots, t_n) & \text{application of predicate symbols} \\
| & t_1 = t_2 & \text{equalities} \\
| & \bot & \text{contradiction} \\
| & \neg F & \text{negation} \\
| & (F_1 \wedge \ldots \wedge F_n) & \text{conjunction} \\
| & (F_1 \wedge \ldots \vee F_n) & \text{disjunction} \\
| & (F_1 \rightarrow F_2) & \text{implication} \\
| & (F_1 \leftrightarrow F_2) & \text{equivalence} \\
| & \forall \nu F & \text{universal quantification} \\
| & \exists \nu F & \text{existential quantification}
\end{array}
$$

The variable ν is said to be bound in $\forall \nu F$ and $\exists \nu F$.

Parentheses

The outermost parenthese of a well-formed formula can be omitted:

$$Cube(x) \land Small(x)$$

In general, parentheses are important to determine the scope of a quantifier (see next slide).

Free and bound variables

An occurrence of a variable in a formula that is not bound is said to be free.

$\exists y\ \mathit{LeftOf}(x, y)$	x is free, y is bound
$(\mathit{Cube}(x) \wedge \mathit{Small}(x)) \rightarrow \exists y\ \mathit{LeftOf}(x, y)$	x is free, y is bound
$\exists x\ (\mathit{Cube}(x) \wedge \mathit{Small}(x))$	Both occurrences of x are bound
$\exists x\ \mathit{Cube}(x) \wedge \mathit{Small}(x)$	The first occurrence of x is bound, the second one is free

Sentences

A sentence is a well-formed formula without free variables.

$$\bot \qquad\qquad A \wedge B$$

$$Cube(a) \vee Tet(b)$$

$$\forall x\ (Cube(x) \to Large(x))$$

$$\forall x\ ((Cube(x) \wedge Small(x)) \to \exists y\ LeftOf(x, y))$$

Semantics of quantification

- We need to fix some domain of discourse.
- $\forall x\ S(x)$ is true iff for every object in the domain of discourse with name n, $S(n)$ is true.
- $\exists x\ S(x)$ is true iff for some object in the domain of discourse with name n, $S(n)$ is true.
- Not all objects need to have names — hence we assume that for objects, names $n_1, n_2, \ldots$ can be invented "on the fly".

The game rules

Form	Your commitment	Player to move	Goal
$P \lor Q$	TRUE FALSE	you Tarski's World	Choose one of P, Q that is true.
$P \land Q$	TRUE FALSE	Tarski's World you	Choose one of P, Q that is false.
$\exists x\, P(x)$	TRUE FALSE	you Tarski's World	Choose some ***b*** that satisfies the wff $P(x)$.
$\forall x\, P(x)$	TRUE FALSE	Tarski's World you	Choose some ***b*** that does not satisfy $P(x)$.

The four Aristotelian forms

All P's are Q's.	$\forall x(P(x) \to Q(x))$
Some P's are Q's.	$\exists x(P(x) \land Q(x))$
No P's are Q's.	$\forall x(P(x) \to \neg Q(x))$
Some P's are not Q's.	$\exists x(P(x) \land \neg Q(x))$

Note:
$\forall x(P(x) \to Q(x))$ does not imply that there are some $P's$.
$\exists x(P(x) \land Q(x))$ does not imply that not all $P's$ are $Q's$.