T-79.3001 Logic in computer science: foundations **Spring 2007**
Exercise 7 ([NS, 1997], Predicate Logic, Chapters 1–4)
March 20–22, 2007

Solutions to demonstration problems

4. Formalize the following sentences using predicate logic:

a) There is a faulty gate.

b) This algorithm is the fastest.

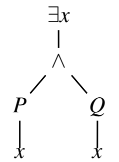
c) Each participant of this course has a workstation to use

d) Only one process can write in each file at a time

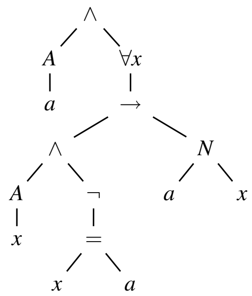
Draw the syntax trees for sentences a) and b).

Solution. All of the solutions use either universal or existential quantifiers or both. If we want to say that some property $\phi(x)$ holds for all those x that have also property $P(x)$, we formalize that with: $\forall x(P(x) \to \phi(x))$. If some property $\phi(x)$ holds for some x that also satisfies $P(x)$, we use $\exists x(P(x) \wedge \phi(x))$. In the solutios many predicates (e.g. $P(x)$ in the first case) are used to denote the type of the x.

a) $\exists x(P(x) \wedge V(x))$, when
$P(x) = x$ is a gate.
$V(x) = x$ is faulty.



b) $A(a) \wedge (\forall x(A(x) \wedge \neg(x = a) \to N(a,x))$, when
a = the algorithm in question
$A(x) = x$ is an algorithm
$N(x,y) = x$ is faster than y.



c) $\forall x(K(x) \to \exists y(T(y) \wedge R(x,y)))$, when
$K(x) = x$ is a participant of the course.
$T(x) = x$ is a workstation
$R(x,y) = x$ uses y.

d) $\forall x(T(x) \to \forall y \forall z(P(y) \wedge P(z) \wedge K(y,x) \wedge K(z,x) \to y = z))$, when
$P(x) = x$ is a process.
$T(x) = x$ is a file.
$K(x,y) = x$ writes in y.

The above solutions are not the only possible ones.

5. Remove unnecessary parenthesis so that the meaning of statement does not change.

a) $(\forall y((\exists x(P(x) \wedge Q(x))) \to L(y)))$

b) $((\exists x(\exists y(P(x,y) \vee Q(y,x)))) \leftrightarrow (\forall x(\neg K(f(x)))))$

c) $(\forall x(\forall y(A \wedge B)))$

Solution.

a) $\forall y(\exists x(P(x) \wedge Q(x)) \to L(x))$.

b) $\exists x \exists y(P(x,y) \vee Q(y,x)) \leftrightarrow \forall x \neg K(f(x))$

c) $\forall x \forall y(A \wedge B)$

6. What ground (variable-free) terms can you compose from a constant c, a unary function symbol f and a binary function symbol g?

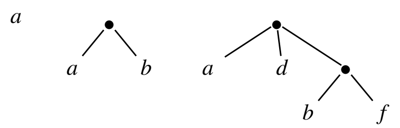
Solution. Using the constant c and function f we get the set of terms $\{c, f(c), f^2(c), f^3(c), \ldots\}$. More terms can be obtained using function g, arguments of g can be any pair from the previous set, for example $g(c,c)$

ja $g(f^3(c), f^{108}(c))$. Naturally these new terms can again be used as arguments for f and g, and we get, e.g, $f(g(f^5(c), f^{13}(c)))$ ja $g(g(c, f(c)), f^8(c))$. This process can be continued for arbitrarily many steps.

7. Represent arbitrary trees with function symbols using at most three constant or function symbols.

Solution. We represent trees as lists. Let constant e denote an empty list, and consider binary function $c \in \mathcal{F}_2$ (ensimmäinen argumentti listan ensimmäinen alkio ja toinen argumentti loput listasta), and unary function $l \in \mathcal{F}_1$ (lehtisolmu). Function $c(x,y)$ denotes a list: x is the first element in the list and y is the rest of the list. Function $l(x)$ denotes that x is a leaf node.

Consider the following trees:



The first of these is represented as $l(a)$, the second as $c(l(a), c(l(b), e))$ and the third as $c(l(a), c(l(d), c(c(l(b), c(l(f), e)), e)))$.

8. Show that if $\forall x \phi(x)$ is a sentence and t is a ground term, then $\phi(t)$ is a sentence.

Solution. A sentence is a formula with no free occurences of any variable. We know that $\forall x \phi(x)$ is a sentence. $\phi(t)$ means a formula in which each free occurence of x is replaced with t. Since t is ground, also $\phi(t)$ is a sentence.

9. Consider a domain $\mathbb{N}^2 = \{\langle x,y\rangle | x \in \mathbb{N}, y \in \mathbb{N}\}$. Choose interpretations for a constant c and a unary function symbol $f \in \mathcal{F}_1$ such that each element in the domain has an interpretation.

Solution. The pairs in $\mathbb{N}^2$ can be placed in an array as follows:

$$\begin{array}{ccccc}
\langle 0,0\rangle & \langle 0,1\rangle & \langle 0,2\rangle & \langle 0,3\rangle & \cdots \\
\langle 1,0\rangle & \langle 1,1\rangle & \langle 1,2\rangle & \langle 1,3\rangle & \cdots \\
\langle 2,0\rangle & \langle 2,1\rangle & \langle 2,2\rangle & \langle 2,3\rangle & \cdots \\
\vdots & \vdots & \vdots & \vdots & \ddots
\end{array}$$

The idea is the same as when showing that there are equally many elements in $\mathbb{N}^2$ and in $\mathbb{N}$, i.e., a bijective mapping from $\mathbb{N}$ to pairs is defined as $f(0) = \langle 0,0\rangle$ and working along diagonals for larger values, for instance, $f(1) = \langle 0,1\rangle$, $f(2) = \langle 1,0\rangle$ etc.

Now, we choose the interpretations as follows: $c^{\mathcal{S}} = \langle 0,0\rangle$, and

$$\begin{array}{rclrcl}
f(c)^{\mathcal{S}} & = & \langle 0,1\rangle & f(f(c))^{\mathcal{S}} & = & \langle 1,0\rangle \\
f^3(c)^{\mathcal{S}} & = & \langle 0,2\rangle & f^4(c)^{\mathcal{S}} & = & \langle 1,1\rangle \\
\vdots & & & \vdots & &
\end{array}$$

Thus $f^{\mathcal{S}}$ is

$$\begin{aligned}
f^{\mathcal{S}} : \langle x,y\rangle &\to \langle x',y'\rangle \\
x' &= g(x)(y+1) + (1-g(x))(x-1) \\
y' &= (1-g(x))(y+1)
\end{aligned}$$

where $g(x)$ is

$$g(x) = \begin{cases} 1, & \text{if } x = 0. \\ 0, & \text{otherwise.} \end{cases}$$

10. A graph is a set S of nodes and a set K of edges between the nodes ($K \subseteq S \times S$). The nodes s and s' of the graph are adjacent, if they are connected with an edge ($\langle s,s'\rangle \in K$). Let C be a set of colors. The problem of *node coloring* is to find a color in C for each node of the graph so that each node has a unique color and two adjacent nodes have different colors.

a) Formalize the node coloring problem using predicate logic.

b) Give a model for your formalization.

c) Give a structure that doesn't satisfy your formalization.

a) In the graphs we are particularly interested in edges, which we will denote by predicate $K(x,y)$ (there is a edge from node x to node y in the graph). There are several possible ways to denote the colors.

(i) We can fix the set of the colors and represent them as predicates. If there are n different colors in set C, we define predicates $C_1(x), \ldots, C_n(n)$. A predicate $C_i(x)$ means that the node x is of the color C_i. The problem description demands that each node has a unique color and that if there is a edge between two nodes the nodes have different colors.

The first condition can be stated with a set of statements of the form:

$$\forall x(C_i(x) \leftrightarrow \neg C_1(x) \wedge \cdots \wedge \neg C_{i-1}(x) \wedge \neg C_{i+1} \wedge \cdots \wedge \neg C_n(x))$$

where $i = 1, \ldots, n$ (notice that $\neg C_i(x)$ is not in the conjunction of the right side).

The second condition is formalized for each $C_i(x)$ as follows:

$$\forall x \forall y (K(x,y) \to (C_i(x) \to \neg C_i(y))).$$

(ii) The second possibility is to leave the definition of the colors open and use a predicate $V(x,y)$ (the node x is of the color y).

Now the uniqueness of node colors can be expressed as:

$$\forall x \forall y \forall z (V(x,y) \wedge V(x,z) \to y = z).$$

Informally, if a node x has both colors y and x, then the colors y and z must, in fact, be the same color.

The second condition can be expressed with:

$$\forall x \forall y \forall z (K(x,y) \to (V(x,z) \to \neg V(y,z)).$$

(iii) The third possibility is to define a function symbol v. Now $v(x)$ means the color of the node x. Because the value of a function is by definition unique, only the second condition has to be formalized:

$$\forall x \forall y (K(x,y) \to \neg(v(x) = v(y))).$$

b) Let's construct a model for the case (i) when $n = 2$. We will define a structure $\mathcal{S}$, where the universume is $U = \{a_1, a_2\}$ (two nodes). The interpretation of predicate K is $K^{\mathcal{S}} = \{\langle a_1, a_2 \rangle, \langle a_2, a_1 \rangle\}$ (there is a edge from node a_1 to a_2 and from a_2 to a_1).

The interpretation of the colors C_1 and C_2 are $C_1^{\mathcal{S}} = \{a_1\}$ and $C_2^{\mathcal{S}} = \{a_2\}$.

We now check that sentences

$$\forall x (C_1(x) \leftrightarrow \neg C_2(x))$$

$$\forall x \forall y (K(x,y) \to (C_1(x) \to \neg C_1(y))$$

and

$$\forall x \forall y (K(x,y) \to (C_2(x) \to \neg C_2(y))$$

are true in the structure. $\mathcal{S}$ (that is, $\mathcal{S}$ is a model for the sentences). The first of the sentences is equivalent to

$$\forall x (C_2(x) \leftrightarrow \neg C_1(x)),$$

which also belongs to the set of sentences when $n = 2$.

Now

$$\mathcal{S} \models \forall x (C_1(x) \leftrightarrow \neg C_2(x))$$

if and only if

$$\mathcal{S}[x \mapsto a_1] \models (C_1(x) \leftrightarrow \neg C_2(x)) \qquad \text{and} \qquad \mathcal{S}[x \mapsto a_2] \models (C_1(x) \leftrightarrow \neg C_2(x))$$

Since $a_1 \in C_1^{\mathcal{S}}$, we have $\mathcal{S}[x \mapsto a_1] \models C_1(x)$. Also, since $a_1 \notin C_2^{\mathcal{S}}$, it holds $\mathcal{S}[x \mapsto a_1] \not\models C_2(x)$. Thus

$$\mathcal{S}[x \mapsto a_1] \models (C_1(x) \leftrightarrow \neg C_2(x))$$

Similarly we show $\mathcal{S}[x \mapsto a_2] \models (C_1(x) \leftrightarrow \neg C_2(x))$, and $\mathcal{S} \models \forall x (C_1(x) \leftrightarrow \neg C_2(x))$ follows.

Now $\mathcal{S} \models \forall x \forall y (K(x,y) \to (C_1(x) \to \neg C_1(y))$ if and only if

$$K(x,y) \to (C_1(x) \to \neg C_1(y))$$

is true in

$$\begin{array}{lll} \mathcal{S}[x \mapsto a_1, y \mapsto a_1], & & \mathcal{S}[x \mapsto a_1, y \mapsto a_2], \\ \mathcal{S}[x \mapsto a_2, y \mapsto a_1] & \text{ja} & \mathcal{S}[x \mapsto a_2, y \mapsto a_2]. \end{array}$$

Because pairs $\langle a_1, a_1 \rangle$ and $\langle a_2, a_2 \rangle$ don't belong to $K^{\mathcal{S}}$, atomic sentence $K(x,y)$ is false in the first and the last case, and then $K(x,y) \to (C_1(x) \to \neg C_1(y))$ is true in these cases. Since pair $\langle a_1, a_2 \rangle$ belongs to $K^{\mathcal{S}}$, $\mathcal{S}[x \mapsto a_1, y \mapsto a_2] \models K(x,y)$ and the proposition is true for $\mathcal{S}[x \mapsto a_1, y \mapsto a_2]$ if and only if $\mathcal{S}[x \mapsto a_1, y \mapsto a_2] \models C_1(x) \to \neg C_1(y)$. This holds, because $a_1 \in C_1^{\mathcal{S}}$ and $a_2 \notin C_2^{\mathcal{S}}$, and therefore $\mathcal{S}[x \mapsto a_1, y \mapsto a_2] \models C_1(x)$ and $\mathcal{S}[x \mapsto a_1, y \mapsto a_2] \models \neg C_2(y)$. The proposition is also true in the third case. Difference to the previous one is that implication $C_1(x) \to \neg C_1(y)$ is true in $\mathcal{S}$, because $\mathcal{S}[x \mapsto a_2, y \mapsto a_1] \not\models C_1(x)$. Thus $\mathcal{S}$ is a model for $\forall x \forall y (K(x,y) \to (C_1(x) \to \neg C_1(y))$.

Because the sentences are symmetrical, $\mathcal{S}$ is also a model for sentence

$$\forall x \forall y (K(x,y) \to (C_2(x) \to \neg C_2(y)).$$

The models will be more complex, if the colors are implemented according to formalizations (ii) or (iii).

c) We will define a structure $\mathcal{S}$ when $n = 2$, where the set of sentences is not satisfiable. We will choose as the universume $U = \{a\}$ (there is only one node) and the interpretation of predicate $K^{\mathcal{S}} = \{\langle a, a \rangle\}$. Now

$$\forall x (C_1(x) \leftrightarrow \neg C_2(x))$$

is not satisfied in structure $\mathcal{S}$, if

$$\mathcal{S}[x \mapsto a] \not\models C_1(x) \leftrightarrow \neg C_2(x).$$

So we can construct the interpretations of the color predicates as follows:

$$\mathcal{S}[x \mapsto a] \models C_1(x) \qquad \text{and} \qquad \mathcal{S}[x \mapsto a] \models C_2(x)$$

choosing

$$C_1^{\mathcal{S}} = C_2^{\mathcal{S}} = \{a\}$$

Now $\mathcal{S}$ cannot be a model for the set of sentences.