Exam, TDDD08 Logic Programming

2016-01-04, 8:00 – 12:00
Person on duty (*jour*): Włodek Drabent, by phone

Means of assistance (hjälpmedel):
• A sheet of notes – 2 sided A5 or 1 sided A4. The notes should be signed in the same way as the exam sheets and returned together with the exam.
• English dictionary

You may answer in English or Swedish.

Grade limits: 3: 17 p, 4: 23 p, 5: 29 p (for total 35 p).

Remember to give explanations for all answers!
Unexplained answers may be granted 0 points.
For instance, when you write a program you should explain the relations defined by the predicates of the program, and the role of each clause.

GOOD LUCK!

---

1. Determine which of the following pairs of terms are unifiable, and provide a most general unifier (mgu) in case there is one.

   a) $p(\,Y, X, g(X), Y\,)$
   $p(\,g(V), f(V), g(f(b)), g(f(a))\,)$

   b) $p(\,X,\ Z,\ g(Y)\,)$
   $p(\,f(Y),\ g(X),\ Z\,)$

   c) $[f(X)|Y]$
   $[f(a), f(b)\,|\,f(c)]$

   d) $p(\,f(X), f(Y), X\,)$
   $p(\,Z, Z, g(V)\,)$

   (4 points)

2. A programmer compiled a program

   ```
   p :- q( t1 ).
   q( t2 ).
   ```

   where t_1 and t_2 are two non-unifiable terms (without any common variables). In spite of this the Prolog system answered "yes" for the goal `?-p`. Explain this. Give an example pair of such terms.

   What should be the result for this program and this query? (In other words: What are the SLD-derivations for this program and this query?) (3 points)

3. (a) Write a logic program defining a predicate *lsh*/2 (left shift) describing the left circular shift of a list. So it is true when its arguments are lists of the form $[t_1, \ldots, t_n]$ and $[t_2, \ldots, t_n, t_1]$ $(n > 0)$.

   Does your program make a useful Prolog program? (For instance, not looping for reasonable queries, and of reasonable efficiency.) Can it also be used to compute the right circular shift of a given list?

   (b) Write a logic program defining a predicate *lshdl*/2 describing the left circular shift relation for lists represented as difference lists. So it is true when its arguments are of the form $[t_1, \ldots, t_n|u]{-}u$ and $[t_2, \ldots, t_n, t_1|s]{-}s$ $(n > 0)$. Explain whether your program produces the left shift of a list in a number of steps independent from the length of the list.

   Can your program be used to compute the right circular shift of a given difference list? Can switching from lists to difference lists improve the efficiency of computing the right circular shift?

   Your programs should be definite clause programs, not using Prolog built-ins, negation, DCG, etc (4 points)

4. Assume that a tree (represented as a term) is either a constant *nil* representing a leaf, or a term $t(t_1, v, t_2)$ representing a node with two subtrees t_1, t_2 and containing a value v, which is a ground term. Assume that all possible values for the nodes are linearly ordered. Assume that a predicate *precedes*/2 is available, so that $precedes(v_1, v_2)$ holds when v_1 precedes v_2 in the ordering.

   Write a logic program defining a predicate *gv*/2 (greatest value) finding the greatest value in a given tree. (It should fail for the tree *nil*.)

   Hint: Introduce a predicate greatest/3 finding the greater value out of two given ones.
   The program should be a definite clause program, not using Prolog built-ins, negation, DCG, etc. Note that $precedes(v_1, v_2)$ implies $v_1 \neq v_2$. (4 points)

5. Consider the following definite program P:

$$
\begin{array}{l}
p(\,f(X), Y\,) \leftarrow q(\,X, g(Y)\,). \\
p(\,g(g(X)), f(Y)\,) \leftarrow p(X, Y). \\
q(\,a, g(g(X))\,). \\
r(X) \leftarrow r(X),\ p(\,g(g(a)), X\,).
\end{array}
$$

   Assume that the vocabulary $\mathcal{A}$ contains two constants a, b and two one-argument function symbols f, g. What is the Herbrand universe $\mathbf{U}_{\mathcal{A}}$ corresponding to $\mathcal{A}$?

   Find the least Herbrand model $\mathbf{M}_P$ of the program. Alternatively, find the set $PTR(P)$ of atomic logical consequences of the program.

Give two examples of ground atoms which are logical consequences of P and one which is not; the predicate symbol of the atoms should be p.

Give an example of non-ground atom with predicate symbol p which is a logical consequence of P, or explain that such atom does not exist. (5 points)

6. Choose a subset $P' \subseteq P$ of the previous program, and for P' and a query $r(X)$ construct two SLD-trees (using different selection rules) – one finite and one infinite. (2 points)

7. Consider the program P, which splits a list nondeterministically:

$$\begin{array}{l} s(\,[\,], [\,], [\,]\,). \\ s(\,[H|T], [H|U], V\,) \leftarrow s(T, U, V). \\ s(\,[H|T], U, [H|V]\,) \leftarrow s(T, U, V). \end{array}$$

The task is to prove that – in any answer $s(l, l_1, l_2)$ of the program – the three arguments of s are lists, each element of l is an element of l_1 or l_2, and each element of l_1 or l_2 is an element of l.

More formally: Prove that the program is correct w.r.t. the specification

$$S = \left\{ s(l, l_1, l_2) \in \mathbf{B}_\mathcal{A} \;\middle|\; \begin{array}{l} l = [t_1, \ldots, t_n],\ l_1 = [s_1, \ldots, s_k],\ l_2 = [u_1, \ldots, u_m], \\ \{t_1, \ldots, t_n\} = \{s_1, \ldots, s_k\} \cup \{u_1, \ldots, u_m\} \end{array} \right\}$$

(where $\mathbf{B}_\mathcal{A}$ is the Herbrand base). (4 points)

8. Translate the following DCG into a Prolog program (using a standard approach).

```
p( [a] ) --> [a].
p( [b] ) --> [b].
p( [a|T] ) --> [a], p( [a|T] ).
p( [b|T] ) --> [b], p( [b|T] ).
p( [a,b|T] ) --> [a], p( [b|T] ).
p( [b,a|T] ) --> [b], p( [a|T] ).
```

Show that $[a, b, b, a]$ is a member of the language of $p([a, b, a])$, by sketching a proof tree, or a successful SLD-derivation.

For which lists l the language of $p(l)$ is not empty? (3 points)

9. Consider the general program P:

$$\begin{array}{l} o(s(X)) \leftarrow \neg o(X). \\ p \leftarrow o(Y). \end{array}$$

Draw SLDNF-forests for queries p, $o(0)$, $o(s^3(0))$. Make it clear which leaves are floundered, which trees are finitely failed, which branches are successful, and what are their answers.

Construct the completion $comp(P)$ of the program. (You are not required to write the equality axioms.) Explain for which numbers i we have $comp(P) \models o(s^i(0))$, and for which ones $comp(P) \models \neg o(s^i(0))$. Compare this with the results of the SLDNF-forests for $o(s^i(0))$.

Explain whether $comp(P) \models p$, and compare this with the SLDNF-forest for p. (5 points)

10. Explain the notion(s) from a chosen item from the list below.

    Your explanation should be short but precise, and should show that you understand the notions. The chosen notions should not be explained in your sheet of notes.

    (a) Interpretation and Herbrand interpretation.

    (b) Logical consequence.

    (c) Answer (i.e. correct answer) of a definite clause program.

    (d) Specification and correctness of a program.

    (e) Successful SLDNF-derivation and its computed answer.

    (f) Negation as finite failure.

    (1 point)