Exam, TDDD08 Logic Programming

2019-01-09, 14:00 – 18:00, U1, U2, U3

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) $p(X, f(a), Y)$
   $p(Y, f(X), f(Z))$

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

   c) $[X, Y|X]$
   $[Z, a, b, c]$

   d) $p(X, f(a), f(Y))$
   $p(Y, f(X), Z)$

   (4 points)

2. Give a simple example of a program and a query for which the results given by Prolog are different from those described by the theory (SLD-resolution). The difference should be finite failure in one case and success the other one (under the same selection rule). (2 points)

3. Write a definite clause program which compares two lists; the comparison means checking whether the corresponding elements of the list are in a given relation p. Assume that a program fragment is given that defines the relation p/2 and a relation nonp/2, so that for any ground terms t_1, t_2 either p(t_1, t_2) or nonp(t_1, t_2) holds.

(Continued on the next page)

Your program should have answers of the form

$$\texttt{co}(\,[s_1,\ldots,s_m],[t_1,\ldots,t_n],[u_1,\ldots,u_k]\,),$$

where $k = \max(m,n)$, and the elements u_i (for $i = 1,\ldots,k$) are

$$u_i = \begin{cases} \text{p} & \text{when } 1 \le i \le \min(m,n) \text{ and } \text{p}(s_i,t_i) \text{ holds} \\ \text{np} & \text{when } 1 \le i \le \min(m,n) \text{ and } \text{nonp}(s_i,t_i) \text{ holds} \\ 1 & \text{when } n < i \le m \\ 2 & \text{when } m < i \le n \end{cases}$$

Prolog built-ins and Prolog constructs like negation and Prolog if-then-else (-> ;) are forbidden.
Note that $1 \le i \le \min(m,n)$ means that both s_i, t_i exist, and $n < i \le m$ or $m < i \le n$ means that only one of s_i, t_i exists.
The program should work as a Prolog program for queries of the form $\texttt{co}(s,t,u)$, where s,t are ground lists.

(3 points)

4. Write a program defining the following predicates.

   (a) nl/1 describing the lists of numbers.
   (So [77, 3], [] are such lists, and [a, 3] is not.)

   (b) us/1 describing the unsorted lists of numbers (neither sorted in increasing, nor in decreasing order).
   For instance, [1, 2, 2, 4], and [3, 2, 1] are sorted, and [1, 2, 1] is unsorted.

   (c) snd/2 finding the second smallest number in a given list longer than one.
   Example answers: snd([3, 2, 1, 4], 2), snd([3, 1, 3, 1], 1).

   Use the Prolog built-in predicate number/1, which recognizes that its argument is a number. Use the built-in predicates </2, =</2 to compare numbers. Otherwise, your program should be a definite clause logic program; you should not use other Prolog built-in predicates, this includes sorting, negation, and Prolog if-then-else (-> ;).

   In problem 4c, snd may fail or produce a run-time error when applied to an argument of a wrong kind (not a list of numbers).

   For a full score the program should avoid obvious inefficiencies, like unnecessary multiple traversals of a list.
   Hint: Begin with a simple, possibly inefficient program.

   (5 points)

5. Consider the following definite clause program P:

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

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

(b) Is the Herbrand interpretation

$$\{ p(g(t)),\ p(f(t)),\ q(g(f(t)), f(t)),\ r(f(t)),\ r(g(t)) \mid t \in \mathbf{U}_{\mathcal{A}} \}$$

a model of the program?

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

(d) Give an example of a ground atom which is a logical consequence of P, and is not an instance of a unary clause of the program.

(e) Give an example of a ground atom which is not a logical consequence of P; the predicate symbol of the atom should be p.

(f) Give an example of a non-ground atom which is a logical consequence of P, and is not an instance of any unary clause of the program.

(6 points)

6. For a chosen query Q and a chosen subset $P_6 \subseteq P$ of the previous program, construct two SLD-trees – one finite and one infinite. (2 points)

7. Consider the program SUFFIX

$$s(L, L).$$
$$s(L, [H|T]) \leftarrow s(L, T).$$

(a) Is the program correct with respect to the specification

$$S_1 = \{ s([t_j, \ldots, t_n], [t_1, \ldots, t_n]) \in \mathbf{B}_{\mathcal{A}} \mid n \geq 0,\ 0 < j \leq n+1 \} \ ?$$

A brief explanation is sufficient here.
$\mathbf{B}_{\mathcal{A}}$ is the Herbrand base. Expression $[t_i, \ldots, t_k]$ stands for the empty list $[\,]$ when $i > k$.

(b) Let $||t||$ denote the number of (the occurrences of) one-argument function symbols in a term t. So for instance $||\,[\mathtt{a}, \mathtt{b}, \mathtt{c}]\,|| = 0$, and $||\,[\mathtt{a}, \mathtt{f}(\mathtt{b}), \mathtt{g}(\mathtt{h}(\mathtt{c}), \mathtt{f}(\mathtt{d}))]\,|| = 3$. Note that $||\,[t_1, \ldots, t_n]\,|| = \Sigma_{i=1}^{n} ||t_i||$
Using a standard method, prove that the program is correct with respect to the specification

$$S = \{ s(t, u) \in \mathbf{B}_{\mathcal{A}} \mid \ ||t|| \leq ||u|| \}.$$

(1+3 points)

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

```
p(0) --> [].
p(s(X)) --> [b], p(X), [b].
p(s(X)) --> [a], p(X).
```

Show that $[b, a, b]$ is a member of the language of $p(s^2(0))$ by sketching a proof tree, a successful SLD-derivation, or a derivation of the DCG. (Choose one of three possibilities.)

For which terms u the language of $p(u)$ is not empty? (3 points)

9. Consider the following general program P_9:

$$p(X) \leftarrow \neg q(X).$$
$$q(s(Y)) \leftarrow p(Y).$$
$$q(a).$$

(a) Draw SLDNF-forests for queries $q(X)$ and $p(s^i(b))$ for $i = 1, 2$. Make it clear which trees are finitely failed, which leaves are floundered, which branches are successful derivations, and what are their answers.

(b) Construct the completion $comp(P_9)$ of the program (except for the equality axioms CET). Explain whether $q(a)$ is a logical consequence of $comp(P_9)$; the same for $q(s(b))$. Does the SLDNF-forest for $q(X)$ provide the corresponding results? (5 points)

10. Choose one case from the list below, and explain the notion(s).

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. Herbrand interpretation.

(b) Renaming substitution (include an example).

(c) Completeness of SLD-resolution.

(d) Answer of a definite clause program (called also a correct answer).

(e) Incompleteness symptom, incompleteness diagnosis.

(f) Constraint predicate. Solution of a constraint

(g) Negation as finite failure.

(1 point)

Version 1.1