Rheinische Friedrich-Wilhelms-Universität Bonn
Institut für Informatik III
Prof. Dr. A. B. Cremers

Advanced Logic Programming
Summer Semester 2012
Dr. Günter Kniesel

# Assignment 2

Due: Sunday, 29.04.2012, 23:59:59 via SVN

For help, contact alp-staff@lists.iai.uni-bonn.de (staff only) or alp-course@lists.iai.uni-bonn.de (staff and participants).

Start working on the exercises early enough so that you can contact the staff in time in case of problems. Don't expect them to be available during weekends!

## Task 1. *Function-Terms = Structured data* (8 Points)

Recall the representation of a binary tree by a term and the predicate that defines the structure of terms that represent binary trees (slides 2-21 and 2-22). For conciseness ,we now use the functor t/3 for "tree" and the minus symbol – for an empty tree. So t(-,1,-) is the tree that contains just the node with the value 1 and two empty subtrees.

a) (1 Points) Adapt the predicate from the lecture so that it checks the above representation. Use it to test whether the following term is a legal tree:
t(t(t(-,d,-), b, t(-,e,-)) , a, t(-,c,t(t(-,g,-), f,-)) ).

b) (1 Points) Draw the tree represented by the term from a)

c) (4 Points) We say that a binary tree is symmetric if you can draw a vertical line through the root node and then the structure of the right subtree is the mirror image of the structure of the left subtree. We are only interested in the tree structure, not in the contents of the nodes.
Implement a predicate symmetric/1 to check whether a binary tree is symmetric. Hint: Write first a predicate mirrors/2 to check whether one tree mirrors the structure of another tree.

d) (2 Points) Write a predicate size(Tree,Size) that succeeds whenever Size is the number of non-empty elements in Tree. Hint: Recall that arithmetic is done using the is/2 predicate. See the online SWI-Prolog manual for details.

## Task 2. *Declarative semantics* (6 Points)

```
extends(a, b).
extends(c, d).
extends(d, e).
subtype(X,Y) :- extends(X,Y).
subtype(X,Y) :- extends(X,Z), subtype(Z,Y).
```

For the above program write down

e) (2 Points) its translation to first order logic (quantified implications).
f) (1 Points) its Herbrand universe.
g) (2 Points) its Herbrand base.
h) (2 Points) its Herbrand model (its logical consequences).

Tip: See lecture slides 2-54 to 2-68 (numbers refer to the slide set of April 11, 2012).

## Task 3. *Declarative semantics* (4 Points)

```
       extends(class(a), class(b)).
       extends(class(c), class(d)).
       extends(class(d), class(e)).
       subtype(X,Y) :- extends(X,Y).
       subtype(X,Y) :- extends(X,Z), subtype(Z,Y).
```

For the above, slightly modified, program write down

a) (1 Points) its Herbrand universe.
b) (2 Points) its Herbrand model .
c) What difficulty did you encounter in point a)? Make a general statement about the effect of function symbols on the Herbrand universe.
d) How does the effect described in c) affect the Herbrand model?

## Task 4. *Declarative semantics* (2 Points)

Consider the Herbrand universe and Herbrand model of the following program and compare them to your findings in Task 3, point c) and d).

```
       natural(0).
       natural( s(X) ) :- natural(X).
```