Homework 11: Logic Programming and Prolog

This homework is due Thursday, May 11. Each weekday that the homework is late you lose 20% of the possible points. (Note that this is larger than usual.) One point from the total for each of the problems will be awarded for how well your answer is presented (i.e., logic and style, which tend to be the same thing in Prolog programs).

In these exercises you should avoid the use of cut (i.e, `!`) and `not`. Mail the programs to `matthews`, as usual.

1. (7 points) Do exercise 13.6 on page 490 of the text. Note that satisfaction of the goals in question should be with respect to the facts and rules in Figure 13.1. You may either do this exercise by hand or use Prolog, but in either case write down the solution on paper.

2. The following are some warm-up problems in Prolog.

   (a) (3 points) Define a relation `find_last` such that `find_last(X,[a,b,c])` binds X to `c`.

   (b) (5 points) Choose a representation for directed graphs and write a relation `connected` that tests whether two nodes are connected.

3. Consider representing the natural numbers with terms as follows. The term `0` represents the number 0, `s(0)` represents 1, `s(s(0))` represents 2, and so on.

   (a) (7 points) Define a 3-ary predicate `prod` such that `prod(X,Y,Z)` succeeds if `Z` is the product of `Z` and `Y`. (Hint: see page 493 of the text.)

   (b) (2 points) Show how to use your predicate `prod` to divide integers by writing a question (in Prolog) that will bind `X` to the result of dividing 12 by 4.

   (c) (10 points) Write a 3-ary Prolog relation `generate` that successively generates the $k-i+1$ numbers (represented as above) in the closed interval from i to k. For example, your relation should have the following behavior:

```
| ?- generate(0,s(s(s(0))),R).
R = 0 ;
R = s(0);
R = s(s(0));
R = s(s(s(0)));
no
```

4. Consider the following logic program for manipulating queues.

```
qdelete(q(X,Y), X, Y).

qinsert(Y, q(W,X1), W, Y).
qinsert(Y, q(W,X1), W, X1, Y).
```

   A queue with elements `a`, `b`, and `c` such that `a` is at the head and `c` at the rear of the queue is represented by the term `q(a,q(b,q(c,X)))`. For each of the following, give a goal statement that will:

(a) (3 points) remove the head of the queue `q(a,q(b,q(c,X)))`,

(b) (3 points) insert the element `d` at the rear of the queue `q(a,q(b,q(c,X)))`,

(c) (3 points) insert the queue `q(d,q(e,X1))` at the end of the queue `q(a,X)`.

5. (20 points) An engineering department must remove 5 cubic meters of garbage per day, but there are only two means: a digging machine and manual workers. It is known that the machine removes 4 cubic meters per day and one worker removes 1 cubic meter per day. Write a Prolog program that will find all the different ways of accomplishing this goal.

   (Hint: you may use standard arithmetic predicates on numbers, such as `>` and `=<`. Also note that `X is Y+4` binds `X` to `Y` + 4, provided `Y` is already bound.)

6. (20 points) In this problem you will modify a meta-circular interpreter for a subset of Prolog without variables. (That is, it only handles clauses of the form `p:-q1,q2,...,qn`.) A depth-first interpreter is found in the file `/users/class/cs342/public/depthfst.pl` on Zaphod. Make a copy of this program and modify it to use breadth-first search. Call the top-level predicate `bfprove`.

   Test your breadth-first interperter by seeing that it finds a solution to the following problem:

```
p :- p, q.
p.
q.
```

7. (15 points) Do exercise 13-22 on page 504 of the text.

8. (10 points) Do exercise 13-32 on page 514 of the text.

9. (20 points) Do exercise 13-45 on page 526 of the text.

10. (20 points) Do exercise 13-47 on page 529 of the text.