York University- Department of Computer Science and Engineering

SC/CSE 3401 3.00 – Functional and Logic Programming

Assignment 1

---

1) This assignment is due on **May 26, 2010**

2) Please provide your first name, last name and student number on the first page of your assignment.

3) Review policy on academic honesty. The submitted assignment must be each individual's own work.

4) NO LATE ASSIGNMENTS!

---

1) (5 marks) Consider the following formulae:

$$A: \quad p \to q \to r$$
$$B: \quad p \to r \to q$$
$$C: \quad p \wedge \neg(r \to q)$$

Using truth tables, show which of the following sets are satisfiable and why?

a) {A, B}

b) {B, C}

c) {A, B, C}

2) (5 marks) Consider the following formula in the sets domain:

$$(\forall x)(\forall y)(\forall z)((\{a\} \bigcup x \subseteq y \bigcup x) \wedge (y \bigcup x \subseteq z \bigcup x) \to (\{a\} \bigcup x \subset z \bigcup x))$$

(a) List all the terms in the above formula that are not object variables or object constants.

(b) List all atomic formulas in the above formula.

(c) Is the formula semantically true? If yes, say why and if no, provide a counter example.

3) (6 marks) Convert the following formula in propositional logic to logic programming clauses. Indicate which ones are Horn clauses. Clearly show ALL steps.

$$((p \to q) \vee r) \equiv (q \wedge s)$$

4) (8 marks) Convert the following formula in predicate logic to logic programming clauses. Indicate which ones are Horn clauses. Clearly show ALL steps (m and n are predicates):

$$\big((\exists x)(\forall y)m(x,y)\big) \rightarrow \big((\forall x)(\exists y)n(y,x)\big)$$

5) (5 marks) (a) Using what we covered about arithmetic in Prolog, write a simple predicate **convert(Pounds, Kilos)** that can convert weight in pounds to kilograms. (b) Write a query to get your weight in kilograms given your weight in pounds. (c) Write a query to get your weight in pounds given your weight in Kilos. This query will not return an answer to you, why?

6) (14 marks) Consider the following program:

```
C0:   findS(X, s(L1, L2)):- findL(X, L1).
C1:   findS(X,s(L1,L2)):- findL(X, L2).
C2:   findL(X, [X|L]).
C3:   findL(X, [_|L]):- findL(X,L).
```

Draw the complete search tree for this query (assuming the user keeps asking for more answers):

```
G0:   :- findS(X, s([a,b], [1])).
```

Label all branches. Under each leaf node, mention outputs by Prolog and why backtracking occurs.

7) (12 marks) Write a Prolog program, by the following design specifications, which given a list of numbers returns a list of remainders of division by 8. For example:

:- rem8([8, 9, 16, 15], [0,1,0,7]). will return true and

:- rem8([1,20,10,11], L). will return L=[1,4,2,3].

(a) (3 marks) Write a recursive code.

(b) (4 marks) Write an iterative code with an accumulator. (It will return the list in the reverse order, for example the second example above will return L=[3,2,4,1])

(c) (5 marks) Write the above code with difference lists, similar to the parts example in class.