York University- Department of Computer Science and Engineering

SC/CSE 3401 3.00 – Functional and Logic Programming

Assignment 3

Notes:
1- This assignment is due on Monday, July 19 at 2:30pm.
2- To do this assignment, you can use what we covered in Lambda calculus plus Chap. 1-4 of Wilensky.
3- Please check the website regularly for updates.

1) (5 marks) What is the order of evaluation of arguments in Common LISP? Are the arguments evaluated from left to right, or right to left? Under what circumstances does the order make a difference? Write two expressions (one for each of above methods) to test LISP’s strategy. [refer to exercise 7 of Chapter 1- Wilensky]

2) (10 marks) In a certain application, we want to be able to call a function dist that can find the Euclidean distance between two points for which the coordinates are given as a list of numbers. In addition, we want to be able to call this function with one (or two) nil argument(s) indicating that we want to find the distance to a reference point, the coordinates of which are saved in a global variable ref. See the following for examples:

> (dist '(1 2) '(4 -2))
5
> (setq ref '(0 1))
(0 1)
> (dist '(0 2) nil)
1
> (dist nil nil)
0

Write the function definition for dist. The efficiency and readability of your code is important. Do not use setq in the function definition.

3) (6 marks) Assume you have evaluated the following in LISP:

(setq v1 '((a b) (x d)))
(setq v2 '((a b) . (x d)))
(setq v3 (append (car v1) (car (cdr v1)) ))
Use car and cdr to return X when applied to

(a) V1  
(b) V2  
(c) V3

4) (8 marks) Do Exercise 3 on page 12 of Selinger’s lecture notes (both parts a and b). In addition, for part b, (iii) show the term calculation.

5) (8 marks) Assume that true (T) is defined as \(\lambda xy.x\) and false (F) is defined as \(\lambda xy.y\). Prove that the function \(\lambda pq.ppq\) can implement OR(\(\lor\)) in logic. (Hint: Show evaluation of OR for all four cases of the truth table of OR).

6) (8 marks) Do the following \(\lambda\)-terms have a \(\beta\)-normal form? If yes, how many? Show reduction steps.  
   Note: In \(\lambda\)-calculus, instead of the infix operators such as \(x + y\), the prefix operator is used e.g. \((+ x y)\).
   (a) \((\lambda x.* x 4)5\)  
   (b) \((\lambda fx.f(fx))(\lambda y.+ y 3)4\)  
   (c) \((\lambda x.xx)(\lambda y.yy)\)  
   (d) \((\lambda xy.xy)y\)

7) (10 marks) [ref: A short introduction to the Lambda Calculus- by A. Jung] Let S be the term \(\lambda xyz.(xz)(yz)\) and K the term \(\lambda xy.x\).
   (a) Reduce SKK to \(\beta\)-normal form. (Hint: To keep things manageable, keep the abbreviations S and K as long as you can; replace them with their corresponding lambda terms only when needed.)
   (b) Find the set of free variables of SKK. Show steps. Is your answer in accordance with part (a)?