Assignment # 4 – Functional Programming
Assigned: November 6, 2014
Due: November 17, 2014 (11:59 pm)

1. Suppose we represent a lambda term by the ML concrete type

datatype term = V **of** string | L **of** string * term | A **of** term * term

Given the above type, a lambda term ((λf.λx.(f x) a) b) would be represented by the ML term

A(A(L("f", L("x", A(V("f"), V("x")))), V("a")), V("b"))

a. Write an ML function show: term → string which produces a string representation of the ML term. For the above ML term, the output string should be: ((Lf.Lx.(f x) a) b). In other words, the output should conform to the proper syntax of a lambda term, except that L is used instead of λ in the output

b. Write an ML function alpha: term * term → boolean which returns true or false indicating whether the two input lambda terms are alpha-equivalent.

Create one file called, lambda.sml, containing the datatype term and your function definitions for show and alpha.

2. Consider the **datatype** 'a gametree = node **of** 'a * 'a gamtree **list** for an infinite game tree discussed in Lecture 20. The strength assessment function for a game position is of the form

fun assess = minimax ∘ treemap(strength) ∘ prune(5) ∘ game

where

assess: position → int
minimax: int gametree → int
strength: position gametree → int
treemap: (position gametree → int) → position gametree → int gametree
prune: int → position gametree → position gametree
game: position → position gametree

Note that 'position' is the type for a typical position (or configuration) of a game. Define treemap and prune using ML notation. Since ML does not support lazy evaluation, you are not required to execute this definition in ML.

Create one file called gametree.txt containing the definitions of treemap and prune.

Note: This assignment may be done by a team of two students. Create a directory containing lambda.sml and gametree.txt. Zip the directory and name it as *Asst4_<UBIT Id1>_<UBIT Id1>.zip* if done by two students; otherwise, name it as *Asst4_<UBIT Id>.zip* if the assignment is done solo. Submit online the zip file. Unix/Linux users may submit a tar file instead of a zip file.

End of Assignment 4