Computer Science 131, Spring 2002
Assignment 9: λ-Calculus Implementation (Part 1)

Out: Wednesday, April 3

Due: Monday, April 8, at the beginning of class

Your code should be placed in a file named `assign9.sml` and handed in using `cs131submit`. Recall that commands like

```
Compiler.Control.Print.printDepth := 1000;
Compiler.Control.Print.printLength := 1000;
```

(or some other large number) may be necessary to make SML/NJ print out entire values, instead of abbreviating deeply-nested parts with the character `#`.

1 Abstract Syntax (20%)

Define a type `varname` whose members correspond in some way to the infinite set of variables. Use this to define a type named `lam` that can be used to represent expressions in the untyped λ-calculus.

2 Capture-Free Substitution (70%)

Define a function `subst : lam * varname * lam -> lam` such that `subst(exp1,vn,exp2)` returns the result of replacing the variable represented by `vn` by `exp2` where it occurs *free* in `exp1`.

Because we need capture-free substitution, bound variables in the original term may have to be renamed, which makes this problem slightly less trivial than it first appears.

3 α-Equivalence

Define the function `aequiv : lam*lam -> bool` which determines whether two functions are the same except for the names of bound variables. Be sure that your code returns `false` when comparing $\lambda x.y$ and $\lambda y.y$ and vice-versa.