HW 19 **Prof. Caldwell**
Due: 14 November 2006 **COSC 3015**

Exercise 0.1. Read the section in Bird on Rose trees (pgs 195 - 209)

Exercise 0.2. Write Haskell code to implement `mkStree'` using the `foldr` and the `insert` for Strees function. Note that, as we discussed in class, this will not necessarily make a balanced tree, but will create a tree with unique values.

```
mkStree' :: [a] -> Stree a
```

The following property should hold:

```
mkStree' l = mkStree (unique l)
```

Convince the grader that your code works.

Exercise 0.3. Write a Haskell program to sum up all of the values in a Rose tree.
The type specification is:

```
sumRose:: (Num a) => Rose a -> a
```

Exercise 0.4. Write flatten for Rose trees with the following specification:

1. The head of the resulting list is the root of the tree.
2. All of the values in the left most subtree appear first in the flattened list.

So for example:

```
flatten (Node 5 [Node 3 [], Node 7 [Node 8 [], Node 9 []], Node 4 []]) =
    [5,3,7,8,9,4]
```