CSE 326 DATA STRUCTURES
HOMEWORK 4

Due: **Friday, Feb 12, 2010** at the beginning of class. Please put your quiz section in addition to your name at the top of your homework.

Problem 1. AVL Trees

Show the result of inserting 3, 8, 9, 4, 5, 6, 1, 2, then 7 into an initially empty AVL tree. Be sure to show the tree after each insertion, although showing intermediate steps (e.g. both parts of a double rotation) will help when assigning partial credit. Be sure to indicate which tree is the final result of each insertion.

Problem 2. Verifying AVL Trees

Design a linear time algorithm that verifies that the height information in an AVL tree is correctly maintained and that the balance property is in order. You must verify that the balance condition is correct, *and also that the height value has been correctly maintained*. You do not need to balance trees that are found to be unbalanced. Make your answer as short and clean as possible. In your answer, use Assert() statments and assume a tree structure like the following:

```
class Node {
    int data;
    int height;
    Node left, right;
}
```

Problem 3. Splay Trees

You only need to show the final result for each find, but intermediate steps will help when assigning partial credit. Be sure to indicate which tree is the final result of each operation.

(a) Weiss problem 4.27

(b) Weiss problem 4.28

Problem 4. Properties of Splay Trees

(a) Prove or disprove ("splay trees retain perfection"): Suppose that the splay tree T is a perfect tree and we perform a find operation on *any* key in the tree. The resulting tree will also be a perfect tree.

(b) Given a splay tree S with N keys, specify a sequence of M find operations that is guaranteed (after all M finds are completed) to produce a list tree. You may assume that the N keys contain no duplicates. Prove that your approach works for all splay trees.