CSE 326: Data Structures
Assignment #3
April 21, 1999
Due: Wednesday, April 28

Reading Assignment: Weiss, Chapter 4, plus handout on splay trees.

Problems:

1. Weiss, p. 171, problem 4.6.
2. Show the AVL tree that results from inserting the keys 186, 39, 991, 336, 778, 66, 564, 154, 538 and 645 into an initially empty tree. Show the intermediate AVL trees after each insertion.
3. Weiss, p. 173, problem 4.27. Show the splay tree after each access.
4. Weiss, p. 173, problem 4.28.
5. Describe how to implement nonlazy deletion in an AVL tree in $O(\log n)$ time, where n is the number of nodes in the tree. Give both a high-level description and pseudocode for your deletion algorithm. You may use the single rotation subroutines as black boxes. (e.g., `rotateWithLeftChild` and `rotateWithRightChild` from the book).
6. AVL trees are often used to implement Set ADTs. Consider the abstract operation `Union`(S, T), one of the operations that might be useful in an application involving sets. `Union`(S, T) is the operation which, given two sets S and T, returns the set $S \cup T$, that is, the set consisting of every x that is a member of either set S or set T or both. Describe an implementation of `Union`(S, T) where S, T and the result $S \cup T$ are represented as AVL trees, that runs in time $O(|S| + |T|)$. Hint: Show that, given a set of n sorted keys, a perfectly balanced binary search tree (i.e., all levels in the tree are full except for perhaps the bottom-most level) can be constructed in time $O(n)$.
7. **Extra Credit:** Consider the special case of `Union`(S, T), where every key in S is less than every key in T. (This is the `Concat`(S, T) operation we discussed in class.) Show that if S and T are implemented as AVL trees, then `Concat`(S, T) can be performed in time $O(\log |S| + \log |T|)$.
8. Weiss, p. 175, problem 4.46, part a. You can assume that the trees are binary and that there are no duplicate values.
9. **Extra Credit:** Weiss, p. 175, problem 4.46, part b. Prove that the running time of your algorithm is linear in the worst case.