150015-Semester II - 5781

Data Structures I

**Homework Assignment #7**

**Question 1**

Given: An AVL tree ( a balanced BST)

1. Write a function that accepts p, a pointer to a node in the AVL tree and performs on it a RR rotation. p is the node that breaks the balance, that is, the node is the root of the sub tree that you are to perform the rotation. The function should return a pointer to the new root of the updated sub tree ( either p or a different node)
2. Write a function as above that performs a RL rotation.

**Question 2**

1. Build an AVL tree from the following values. Draw the tree after each insertion and write which rotation you used when a rotation was needed.

27, 15, 6, 40, 2, 7, 1, 10, 11, 8.

1. Delete from the above tree the following values. Draw the tree after each deletion and write which rotation you used when a rotation was needed.

10, 6, 40, 27

**Question 3**

Write a linear algorithim (that is, that runs in Θ(n)) that builds an AVL tree from an array sorted in non-descending order.

**Question 4**

1. What is the maximum number of possible nodes for an AVL tree of height h? Explain.
2. Given a BST, *T*, can you make it an AVL by using rotations only? Explain.