Data Structures & Algorithms 2
Tutorial 4
Trees

OBJECTIVES

- Understand the basics for different types of Trees (Binary, AVL ..)

Exercise 1

- Show the result of inserting 3, 1, 4, 6, 9, 2, 5, 7 into an initially empty binary search tree.
- Show the result of deleting the root.

Exercise 2

Show that the maximum number of nodes in a binary tree of height h is $2^{h+1} - 1$

Exercise 3

Two binary trees are similar if they are both empty or both nonempty and have similar left and right sub trees. Write a function to decide whether two binary trees are similar. What is the running time of your function?

Exercise 4

Write a function to generate a perfectly balanced binary search tree of height h with keys 1 through $2^{h+1} - 1$. What is the running time of your function?

Exercise 5

Show the result of inserting 2, 1, 4, 5, 9, 3, 6, 7 into an initially empty AVL tree.

Exercise 6

Write a routine to list out the nodes of a binary tree in level-order. List the root, then nodes at depth 1, followed by nodes at depth 2, and so on. You must do this in linear time. Prove your time bound.

Exercise 7

Suppose we want to add the operation findKth to our repertoire. The operation findKth(k) returns the kth smallest item in the tree. Assume all items are distinct. Explain how to modify the binary search tree to support this operation in O(log N) average time, without sacrificing the time bounds of any other operation