Timothy Hoang

5/12/19

CMPT435-111

Assignment 13

**Date Assigned: 05/06/2019**

**Due: Midnight 05/12/2019 on iLearn**

**Please read turn-in checklist at the end of this document before you start doing exercises.**

**Section 1: Pen-and-paper Exercises**

1. Given a BST and a positive integer k, find the k\_th smallest element in the BST.

For example, in the following BST, if k = 3, then output should be 10, and if k = 5, then output should be 14.



Assume the tree is balanced, and the tree height is O(log n). Design an O(log n) algorithm to solve this problem.

1. describe the idea behind your algorithm in English (2 points);

All leaf nodes have attribute 1

All root nodes have all the attributes in it added together

There exists k-1 nodes <= root

(ii) provide pseudocode (5 points);

nth(root ,k)

If left.size == k-1

RETURN ROOT

Else if left.size > k-1

nth(root.left, k)

Else (left.size < k-1)

nth(root.right, k-left.size -1)

(iii) analyze its running time (3 points).

Assuming a balanced tree, it cuts off half of tree after every iteration if root size != k.

Olog(n)

**Note: We will discuss this problem in class.**

**Note: Full credit (10 points) will be awarded for an algorithm in O(log n) time. Algorithms slower will be scored out of 5 points.**

**Section 2: Java Implementation**

1. Implement the problem 1 in Java.

Note:

Find a file called BST.java in assignment 13 folder.

Complete the method of ReturnKthSmallestElement ().

Test your method in the main method provided following the comments.

**TURN-IN CHECKLIST:**

1. **Answers to Section 1 (.doc/.txt/.pdf), and to Section 2 (all your source Code (.java files)). Remember to include your name, the date, and the course number in comments near the beginning of your code/report.**
2. **Create a folder and name it 'FirstName\_LastName\_assignment\_10'. In the newly created folder copy and paste your files (.doc/.txt/.java files). Then compress the folder, and push it to iLearn.**