Muhammad Atif, Ms. Asma Sattar and Muhammad Haris

DS 302 Assignment#03 for All Sections

Last Date:11/11/2018 11:55 pm

Question no 1:

Implement the following functions for Binary Search Tree.

a. Function to insert a node in BST.
b. Function to search a specific node in BST.
c. Function to delete a specific node in BST.
d. Function to find the number of nodes at a specific height (given by user).
e. Function should return siblings of the input passed as a parameter.
   // Int sibling(root, input)
f. Function should display all the ancestors of the node containing the input passed as parameter.
   //Void ancestors(root, data)
g. Function to find if the Binary Tree is BST or not.
h. Function to find if the BST is Complete binary tree or not.
i. Function to count leaf nodes in a BST.

Question no 2:

Write a c++ code that merge three BST, consider root as the largest element node in all the tree's roots.

Then write a function to mirror the resultant BST.

Question no 3:

Make a program to implement AVL trees. The program would consist of the following parts:

a. A function to insert a new node in the AVL tree.
b. Single rotation functions (both LL and RR cases)
c. Double rotation functions (both LR and RL cases), without the inefficiency of doing two single rotations.
d. A function to search and delete a specific node from the AVL Tree.
e. A function to traverse/displays all the contents of the AVL tree.

Question no 4:

Level Order Traversal

Level order traversal processes the nodes level by level. It first processes the root, and then its children, then its grandchildren, and so on. Unlike the other traversal methods, a recursive version does not exist.

A traversal algorithm is similar to the non-recursive preorder traversal algorithm. The only difference is that a stack is replaced with a FIFO queue.

a. Write a function for level order traversal.
b. Implement pre-order, post-order and in-order traversals in an iterative fashion using stacks and queues classes objects implemented in last assignment.