|  |  |
| --- | --- |
| **WEEK-8** | **TREES** |
| **LAB A** | **26 sep-1st oct** |
| 1. Complete AVL tree questions of last lab. 2. A binary tree is a rooted tree where each node contains at most two children. Each child can beidentified as either a left or right child.   Write functions to:  (a) Delete all nodes at a given height.  (b) Display the subtree starting with a given node/ element value.  (c) Delete the subtree beginning from a given node/ element value.   1. Binary Search Trees: A binary search tree labels every node in a binary tree with a single key such that for any node x, aswell as nodes within the left subtree of x have keys <= x and all nodes in the right subtree of x havekey's >= x.   Write functions to:  (a) NumberofNodes //Returns the number of nodes stored in the tree  (b) WeightofTree // Returns the weight of the tree (i.e., sum of all node depths)  (c) AvgSearchCost //Returns the average cost of a search in the tree (i.e., weight/size)  (d) WhetherAVL // Returns whether the tree has the AVL property, i.e., whether it is balanced  or not.  (e) Given a BST, transform it into greater sum tree where each node contains sum of all nodes  greater than that node. | |
| **LAB B** | **Lab test 1** |
|  | **26 sep-1st oct** |