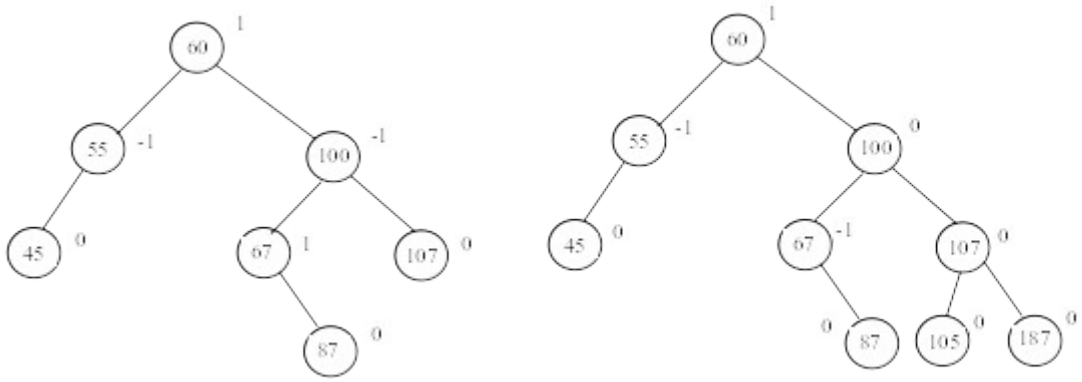
Chapter 20 AVL Trees

1. AVL trees are well-balanced. In an AVL tree, the difference between the heights of two subtrees for every node is 0 or 1.
The *balance factor* of a node is the height of its right subtree minus the height of its left subtree. A node is said to be *balanced* if its balance factor is -1, 0, or 1. A node is said to be *left-heavy* if its balance factor is -1. A node is said to be *right-heavy* if its balance factor is +1.
2.



3. If a node is not balanced after an insertion or deletion operation, you need to rebalance it. The process of rebalancing a node is called a *rotation*. There are four possible rotations: LL, LR, RR, and RL.

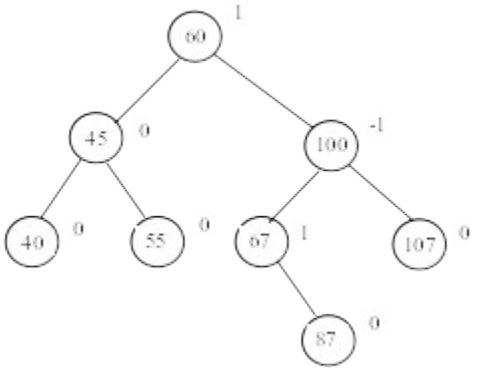
An *LL imbalance* occurs at a node A such that A has a balance factor -2 and a left child B with a balance factor -1 or 0. This type of imbalance can be fixed by performing a single right rotation at A.

4. AVLTreeNode inherits from TreeNode. The height is a new data field defined in AVLTreeNode. The data fields in TreeNode are left and right, pointing to the left and right subtree.

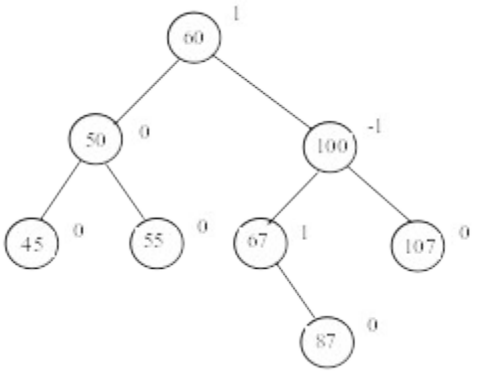
5. True

6. True

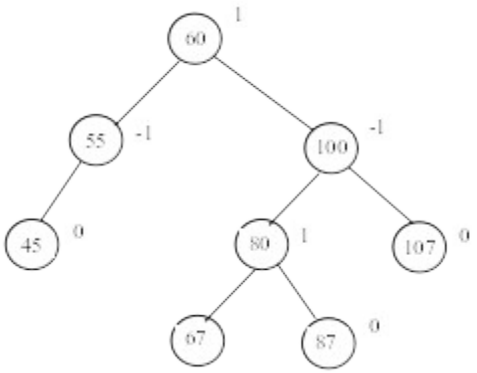
7. After inserting 40, node 55 is unbalanced, perform LL rotation. The resulting tree is



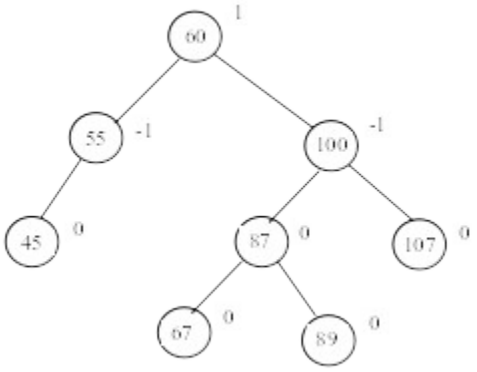
8. After inserting 50, node 55 is unbalanced, perform LR rotation. The resulting tree is



9. After inserting 80, node 67 is unbalanced, perform RL rotation. The resulting tree is



10. After inserting 89, node 67 is unbalanced, perform RR rotation. The resulting tree is



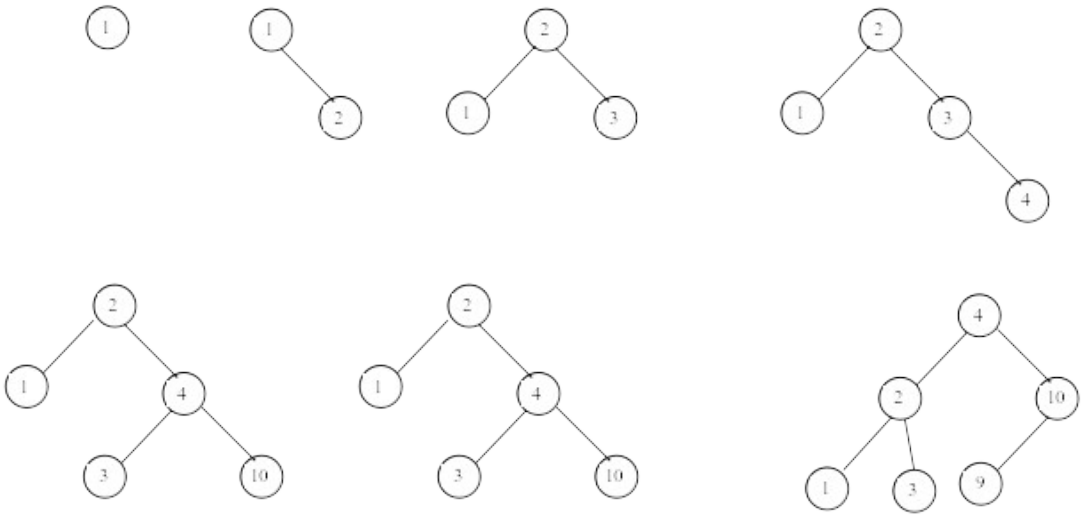
Question 11 should be changed to “How is the createNewNode method in the AVLTree class is invoked”?

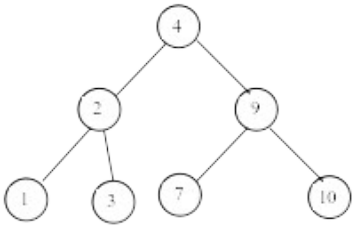
11.In the BinaryTree class, the createNewNode() method creates a TreeNode object. This method is defined in BinaryTree. It is overridden in the AVLTree class to create an AVLTreeNode.

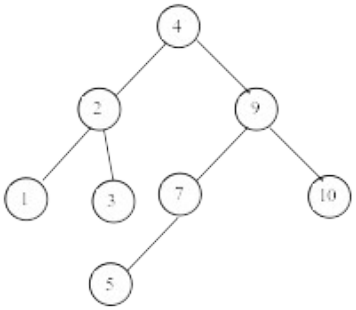
12. updateHeight(AVLTreeNode<E>) is invoked to update the height of a node. It is invoked to rebalance the tree. balanceFactor is invoked to check the balance factor of a node. It is invoked when a path is rebalanced. balancePath is invoked along the path where a new node is inserted or a node is deleted.
13. All data fields defined in the BST class are inherited in the AVLTree class. The AVLTree class does not define new data fields.

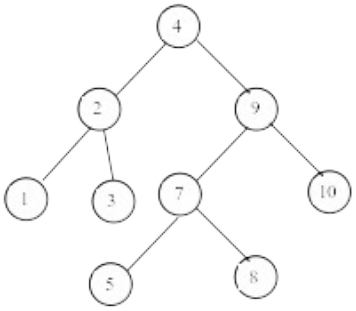
14.No.

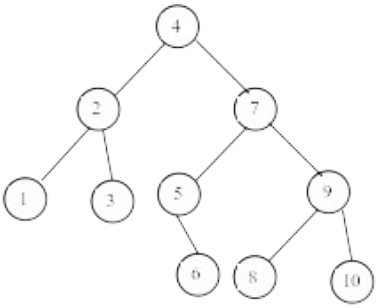
15.




4
2
9
1
3
7
10


4
2
9
1
3
7
10
5


4
2
9
1
3
7
10
5
8


4
2
7
1
3
5
9
6
8
10