24 Lecture - CS301

Important Subjective

1. What is AVL Tree? Describe the steps to delete a node in AVL Tree.

Answer: AVL Tree is a self-balancing binary search tree in which the height of the left and right subtrees of any node differs by at most one. To delete a node in an AVL Tree, we perform the following steps:

- 1. Perform the standard BST delete operation for the node to be deleted.
- 2. Check the balance factor of all the nodes in the path from the deleted node to the root.
- 3. If the balance factor of any node becomes +2 or -2, perform the appropriate rotation to balance the tree.
- 4. What is the difference between the deletion of a node in a BST and AVL Tree?

Answer: The deletion of a node in a BST can lead to an unbalanced tree, while in AVL Tree, after the deletion of a node, the balance factor of all the nodes is checked, and the tree is rebalanced by performing appropriate rotations.

3. What are the different cases that can arise while deleting a node from an AVL Tree?

Answer: The different cases that can arise while deleting a node from an AVL Tree are:

- 1. The node to be deleted is a leaf node.
- 2. The node to be deleted has only one child.
- 3. The node to be deleted has two children.

4. How is the height of a node in an AVL Tree calculated?

Answer: The height of a node in an AVL Tree is calculated as the maximum height of its left and right subtrees plus one.

5. Why is AVL Tree preferred over other binary search trees?

Answer: AVL Tree is preferred over other binary search trees because it is self-balancing, which ensures that the height of the tree remains balanced, and the search, insertion, and deletion operations take O(log n) time complexity.

6. How is the balance factor of a node in an AVL Tree calculated?

Answer: The balance factor of a node in an AVL Tree is calculated as the difference between the height of its left and right subtrees.

7. What is the role of rotations in AVL Tree?

Answer: Rotations are performed in an AVL Tree to balance the tree after insertion or deletion of a node. There are four types of rotations: left-rotate, right-rotate, left-right-rotate, and right-left-rotate.

8. How is the balance of an AVL Tree maintained?

Answer: The balance of an AVL Tree is maintained by keeping the height difference of the left and right subtrees of each node within the range of -1 to +1. If the balance factor of a node becomes -2 or +2, appropriate rotations are performed to balance the tree.

9. What are the advantages of AVL Tree?

Answer: The advantages of AVL Tree are:

- 1. It ensures that the height of the tree is balanced.
- 2. Search, insertion, and deletion operations take O(log n) time complexity.
- 3. It is efficient in terms of time and space complexity.
- 4. What is the worst-case time complexity of the deletion operation in an AVL Tree?

Answer: The worst-case time complexity of the deletion operation in an AVL Tree is O(log n).