14/10/20

Divyank Gupta
1BM18CS030

Write - Up

* AVL Tree insertion and deletion :-

→ Pseudo-code :- ——— ① Insertion

```
Node insert (Node node, int key)
{
    if (node == null)
        do return (new Node(key))
    if (key < node.key)
        do node.left ← insert (node.left, key)
    else if (key > node.key)
        do node.right ← insert (node.right, key)
    else
        do return node

    node.height = 1 + max (height(node.left), height (node.right))

    int balance ← getBalance (node).

    if (balance >1 && key < node.left.key)
        do return rightRotate (node)

    if (balance <-1 && key > node.right.key)
        do return leftRotate (node)

    if (balance >1 && key > node.left.key)
        do node.left ← leftRotate (node.left)
        return rightRotate (node)
```

[illegible]

Scanned with CamScanner

Divyank Gupta
1BM18CS030

```
    if (balance < -1 && Key < node.right.Key)
        do node.right ← rightRotate (node.right)
            return leftRotate (node)

    return node
}

Node deleteNode (Node root, int Key)                ---- (2) Deletion

{   if (root == null)
        do return root

    if (Key < root.Key)
        do root.left ← deletNode (root.left, Key)
    else if (Key > root.Key)
        do root.right ← deleteNode (root.right, Key)

    else
    {
        if (root.left == null || root.right == null)
            do  Node temp ← null
                if (temp == root.left)
                    temp ← root.right

                else
                    temp ← root.left

                if (temp == null)
                    do temp ← root ;
                       root ← null

                else
                    root ← temp

    else
        appey balancing code
}
```

[illegible]