**BST\_TECHNICAL MOCK TEST:**

**ANSWERS:**

1. Figure a
2. Figure b
3. 30,72,18,98,28,29,25,102,,90,96
4. 45,40
5. 16

**CODING:**

**6)** class Node

{

    int data;

    Node left, right;

    public Node(int item)

    {

        data = item;

        left = right = null;

    }

}

public class BinaryTree

{

    Node root;

    int getLeafCount()

    {

        return getLeafCount(root);

    }

    int getLeafCount(Node node)

    {

        if (node == null)

            return 0;

        if (node.left == null && node.right == null)

            return 1;

        else

            return getLeafCount(node.left) + getLeafCount(node.right);

    }

    public static void main(String args[])

    {

        BinaryTree tree = new BinaryTree();

        tree.root = new Node(1);

        tree.root.left = new Node(2);

        tree.root.right = new Node(3);

        tree.root.left.left = new Node(4);

        tree.root.left.right = new Node(5);

        System.out.println("The leaf count of binary tree is : "

        + tree.getLeafCount());

    }

}

**7)** class Node

{

    int data;

    Node left, right;

    Node(int item)

    {

        data = item;

        left = right = null;

    }

}

class BinaryTree

{

    Node root;

    void printSingles(Node node)

    {

    if (node == null)

      return;

    if (node.left != null && node.right != null)

    {

        printSingles(node.left);

        printSingles(node.right);

    }

    else if (node.right != null)

    {

        System.out.print(node.right.data + " ");

        printSingles(node.right);

    }

    else if (node.left != null)

    {

        System.out.print( node.left.data + " ");

        printSingles(node.left);

    }

}

    public static void main(String args[])

    {

        BinaryTree tree = new BinaryTree();

        tree.root = new Node(1);

        tree.root.left = new Node(2);

        tree.root.right = new Node(3);

        tree.root.left.right = new Node(4);

        tree.root.right.left = new Node(5);

        tree.root.right.left.right = new Node(6);

        tree.printSingles(tree.root);

    }

}