class Node{

public int data;

public Node left;

public Node right;

public Node(int data){

this.data=data;

this.left = null;

this.right = null;

}

}

class Tree{

public Node root = null;

void insert(int data){

Node newnode = new Node (data);

if(root == null)

{

root = newnode;

}

else{

Node temp = root;

while(true)

{

if (temp.data > data){

if(temp.left == null){

temp.left = newnode;

break;

}

else{

temp = temp.left;

}

}

else{

if(temp.right== null){

temp.right =newnode;

break;

}

else{

temp = temp.right;

}

}

}

}

}

void inorder(Node root){

if (root!=null){

inorder(root.left);

System.out.print(root.data+ " ");

inorder(root.right);

}

}

void preorder(Node root){

if (root!=null){

System.out.print(root.data+ " ");

inorder(root.left);

inorder(root.right);

}

}

void postorder(Node root){

if (root!=null){

inorder(root.left);

inorder(root.right);

System.out.print(root.data + " ");

}

}

}

public class Main

{

public static void main(String[] args) {

Tree t = new Tree();

t.insert('a');

t.insert('d');

t.insert('z');

t.insert('n');

t.insert('x');

t.insert('v');

t.insert('j');

t.insert('s');

System.out.println("The Inorder:");

t.inorder(t.root);

System.out.println();

System.out.println("The Preorder:");

t.preorder(t.root);

System.out.println();

System.out.println("The Postorder:");

t.postorder(t.root);

}

}