package saisha;

class node {

int data;

Node left, right;

public node(int data) {

this.data = data;

left = right = null;

}

}

class Binarysearchtree {

Node root;

Binarysearchtree() {

root = null;

}

void insert(int data) {

root = insertRec(root, data);

}

Node insertRec(Node root, int data) {

if (root == null) {

root = new Node(data);

return root;

}

if (data < root.data) {

root.left = insertRec(root.left, data);

} else if (data > root.data) {

root.right = insertRec(root.right, data);

}

return root;

}

void inOrder() {

inOrderRec(root);

}

void inOrderRec(Node root) {

if (root != null) {

inOrderRec(root.left);

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

inOrderRec(root.right);

}

}

}

public class Node1 {

public static void main(String[] args) {

BinarySearchTree bst = new BinarySearchTree();

// Inserting elements into the BST

bst.insert(50);

bst.insert(30);

bst.insert(70);

bst.insert(20);

bst.insert(40);

bst.insert(60);

bst.insert(80);

// Printing the elements in in-order traversal

System.out.println("In-order traversal of the BST:");

bst.inOrder();

}

}