import java.util.Stack;

public class BinaryTree {

private TreeNode root;

private class TreeNode {

private TreeNode left;

private TreeNode right;

private int data; // Can be any generic type

public TreeNode(int data) {

this.data = data;

}

}

public void inOrder() {

if(root == null) {

return;

}

Stack<TreeNode> stack = new Stack<>();

TreeNode temp = root;

while(!stack.isEmpty() || temp != null) {

if(temp != null) {

stack.push(temp);

temp = temp.left;

} else {

temp = stack.pop();

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

temp = temp.right;

}

}

}

public void createBinaryTree() {

TreeNode first = new TreeNode(9);

TreeNode second = new TreeNode(2);

TreeNode third = new TreeNode(3);

TreeNode fourth = new TreeNode(4);

root = first; // root ---> first

first.left = second;

first.right = third; // second <--- root ---> third

second.left = fourth;

}

public static void main(String[] args) {

BinaryTree bt = new BinaryTree();

bt.createBinaryTree();

bt.inOrder();

}

}