

NC45 实现二叉树先序，中序和后序遍历

中等 通过率: 41.27% 时间限制: 3秒 空间限制: 256M

知识点: 栈 树 哈希

题目 题解(84) 讨论(205) 排行

描述

分别按照二叉树先序，中序和后序打印所有的节点。

示例1

输入: {1,2,3}

返回值: [[1,2,3],[2,1,3],[2,3,1]]

备注:

$n \leq 10^6$

关联企业

关联职位

算法知识视频讲解

复制 收藏 分享 评论

Java

核心代码模式

```
10
11 public class Solution {
12     List<Integer> pre=new ArrayList<>();
13     List<Integer> in=new ArrayList<>();
14     List<Integer> post=new ArrayList<>();
15     ...
16     public int[][] threeOrders (TreeNode root) {
17         //write code here
18         if(root==null){
19             return null;
20         }
21         List<List<Integer>> list=new ArrayList<>();
22         first(root);
23         second(root);
24         third(root);
25         list.add(pre);
26         list.add(in);
27         list.add(post);
28         int[][] res=new int[list.size()][list.get(0).size()];
29         for(int i=0;i<res.length;i++){
30             for(int j=0;j<res[0].length;j++){
31                 res[i][j]=list.get(i).get(j);
32             }
33         }
34         return res;
35     }
36     public void first(TreeNode root){
37         if(root==null){
38             return;
39         }
40         pre.add(root.val);
41         first(root.left);
42         first(root.right);
43     }
44     ...
45     public void second(TreeNode root){
46         if(root==null){
47             return;
48         }
49         second(root.left);
50         in.add(root.val);
51         second(root.right);
52     }
53     public void third(TreeNode root){
54         if(root==null){
55             return;
56         }
57         third(root.left);
58         third(root.right);
59         post.add(root.val);
60     }
61 }
```

运行结果 自测输入