

4月24日日报

本日学习内容

1. 学习二叉树的二叉树迭代遍历和层序遍历，完成随想录3 -5
2. 完成二叉树遍历方式的总结博客

今日算法题

题目1: [429. N 叉树的层序遍历](#)

429. N 叉树的层序遍历

中等

🔖 相关标签

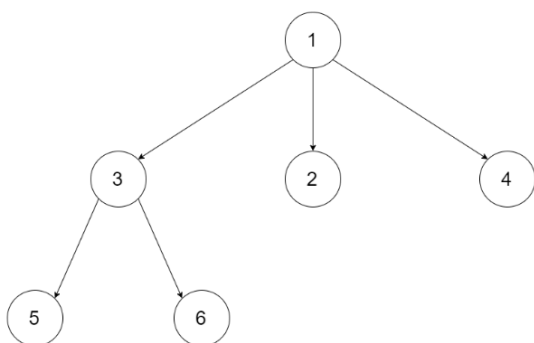
🏢 相关企业

Aa

给定一个 N 叉树，返回其节点值的层序遍历。（即从左到右，逐层遍历）。

树的序列化输入是用层序遍历，每组子节点都由 null 值分隔（参见示例）。

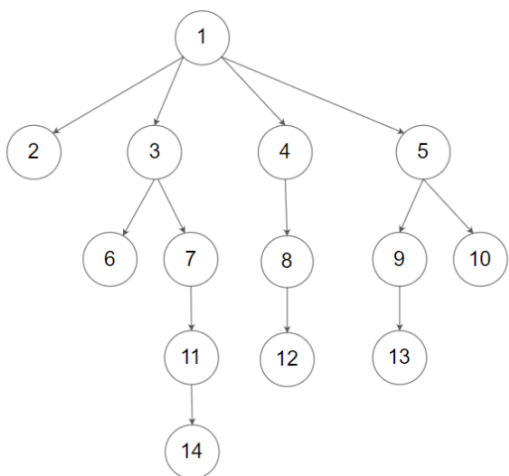
示例 1:



输入: root = [1,null,3,2,4,null,5,6]

输出: [[1],[3,2,4],[5,6]]

示例 2:



```
class Solution {
public:
    vector<vector<int>> levelOrder(Node* root) {
        if (root == nullptr)
            return {};
        vector<vector<int>> ans;
        queue<Node*> q;
        q.push(root);
        while (!q.empty()) {
            vector<int> vals;
            for (int n = q.size(); n--;) { // 处理当前层的n个节点
                auto node = q.front();
```

```
        q.pop();
        vals.push_back(node->val);
        for (auto c : node->children) { // 遍历所有孩子
            q.push(c);
        }
    }
    ans.emplace_back(vals);
}
return ans;
};
```

题目2: [515. 在每个树行中找最大值](#)

515. 在每个树行中找最大值

已解

中等

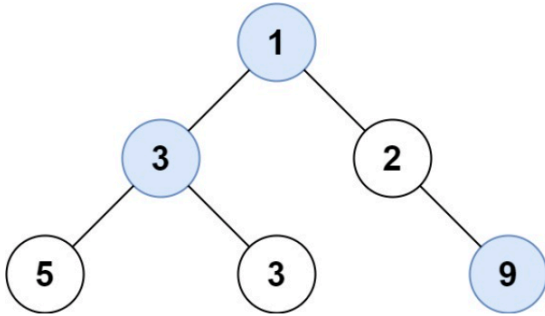
🔖 相关标签

🔒 相关企业

Ax

给定一棵二叉树的根节点 `root`，请找出该二叉树中每一层的最大值。

示例1:



输入: `root = [1,3,2,5,3,null,9]`

输出: `[1,3,9]`

示例2:

输入: `root = [1,2,3]`

输出: `[1,3]`

提示:

- 二叉树的节点个数的范围是 `[0, 104]`
- `-231 <= Node.val <= 231 - 1`

面试中遇到过这道题? 1/5

👍 🗑

```
class Solution {
public:
    vector<int> largestValues(TreeNode* root) {
        queue<TreeNode*> que;
        if (root != NULL)
            que.push(root);
        vector<int> result;
        while (!que.empty()) {
            int size = que.size();
            int maxValue = INT_MIN; // 取每一层的最大值
            for (int i = 0; i < size; i++) {
                TreeNode* node = que.front();
```

```
        que.pop();
        maxValue = node->val > maxValue ? node->val : maxValue;
        if (node->left)
            que.push(node->left);
        if (node->right)
            que.push(node->right);
    }
    result.push_back(maxValue); // 把最大值放进数组
}
return result;
}
};
```

明日学习计划：

1. 继续学习git操作分离head
2. 学习反转二叉树，继续二叉树的习题，深入理解递归和迭代
3. 继续完成hot100数组部分，并更新原博客