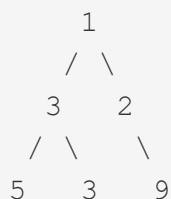


您需要在二叉树的每一行中找到最大的值。

示例：

输入：



输出： [1, 3, 9]

思路，该题为明显的广度遍历的题目，使用数组将每层遍历的节点记录下来，一次遍历一层，得到最大值，存放在返回的数组，即可得到答案。

代码

```
/**
 * Definition for a binary tree node.
 * struct TreeNode {
 *     int val;
 *     TreeNode *left;
 *     TreeNode *right;
 *     TreeNode(int x) : val(x), left(NULL), right(NULL) {}
 * };
 */
class Solution {
public:
    vector<int> largestValues(TreeNode* root) {
        vector<TreeNode*> nums;
        vector<int> result;
        if(root==NULL){
            return result;
        }
        nums.push_back(root);
        for(int i=0;i<nums.size();){
            int j=nums.size();
            int max=nums[i]->val;
            while(i<j){
                if(max<nums[i]->val){
                    max=nums[i]->val;
                }
                if(nums[i]->left!=NULL){
                    nums.push_back(nums[i]->left);
                }
                if(nums[i]->right!=NULL){

```

```
        nums.push_back(nums[i]->right);
    }
    i++;
}
result.push_back(max);
}
return result;
}
};
```