```cpp
class Solution {
public:
    vector<vector<int>> levelOrder(TreeNode* root) {
        // 结果保存
        vector<vector<int>> result;
        // 定义队列
        queue<TreeNode*> duilie;
        if (root == nullptr) {
            return result;
        }
        // 初始化队列
        duilie.push(root);

        while (!duilie.empty()) {
            vector<int> temp;
            int k = duilie.size(); // 当前层的节点数
            for (int i = 0; i < k; i++) {
                TreeNode* treenode = duilie.front();
                duilie.pop();

                // 保存temp
                temp.push_back(treenode->val);

                // 入队左子节点
                if (treenode->left != nullptr) {
                    duilie.push(treenode->left);
                }
                // 入队右子节点
                if (treenode->right != nullptr) {
                    duilie.push(treenode->right);
                }
            }
            result.push_back(temp);
        }

        return result;
    }
};```
```