

# N叉树

## 思路

- 跟二叉树的前序遍历相似，不同在于处理（children）
- 迭代法中，将children压入栈中，注意顺序，从后往前压入栈中
- 递归法中，自动从最左边的children开始遍历

## 递归

```
1 class Solution {
2 public:
3     vector<int> res;
4     vector<int> preorder(Node* root) {
5         if(!root) return res;
6         res.push_back(root -> val);
7         for(auto i : root -> children){
8             preorder(i);
9         }
10        return res;
11    }
12 };
13
```

## 迭代

```
1 class Solution {
2 public:
3     vector<int> preorder(Node* root) {
4         vector<int> res;
5         if(!root) return res;
6         stack<Node*> stk;
7         stk.push(root);
8         while(!stk.empty()){
9             Node* tmp = stk.top();
10            stk.pop();
11            res.push_back(tmp -> val);
12            for(int i = tmp -> children.size()-1; i >= 0; --i){
13                stk.push(tmp -> children[i]);
14            }
15        }
16        return res;
17    }
18 };

```

