```
/*
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）。

例如，给定一个 3叉树 ：

        1
  3     2     5
5  6

返回其层序遍历：

[
     [1],
     [3,2,4],
     [5,6]
]

说明:
     树的深度不会超过 1000。
     树的节点总数不会超过 5000。
来源：力扣（LeetCode）
链接：https://leetcode-cn.com/problems/n-ary-tree-level-order-traversal
著作权归领扣网络所有。商业转载请联系官方授权，非商业转载请注明出处。

*/
```

分析：

- 多叉树的层序遍历通常思维为使用队列遍历来进行迭代遍历(方法一)

方法一:C++,queue

```
/*
// Definition for a Node.
class Node {
public:
    int val;
    vector<Node*> children;

    Node() {}

    Node(int _val, vector<Node*> _children) {
        val = _val;
```

```
        children = _children;
    }
};
*/

class Solution
{
    public:
        vector<vector<int>> levelOrder(Node* root)
        {
            vector<vector<int>>     ret_val             ;
            vector<int>             level_vector        ;
            queue<Node*>            qn                  ;
            Node*                   temp        = NULL  ;
            int                     num_1       = 0     ;
            int                     num_2       = 0     ;
            int                     i           = 0     ;
            int                     j           = 0     ;

            if(root==NULL)
            {
                return ret_val;
            }

            qn.push(root)   ;
            num_1 = 0       ;
            num_2 = 1       ;
            while(1)
            {
                level_vector.clear();
                num_1   =   num_2   ;
                num_2   =   0       ;
                for(i=0;i<num_1;i++)
                {
                    temp = qn.front();
                    for(j = 0; j < temp->children.size();j++)
                    {
                        qn.push(temp->children[j]);
                        num_2++;
                    }
                    level_vector.push_back(temp->val);
                    qn.pop();
                }
                ret_val.push_back(level_vector);
                if( num_2 == 0 )
                {
                    break;
                }
            }
            return ret_val;
       }
};
```

```
/*
执行结果：
通过
显示详情
执行用时 :388 ms, 在所有 C++ 提交中击败了9.38%的用户
内存消耗 :33.4 MB, 在所有 C++ 提交中击败了95.49%的用户
*/
```

AlimyBreak
2019.07.29