# 102. Binary Tree Level Order Traversal

作者：qianrong wu

# 思路

层序遍历用队列的先进先出特性来记录每一层的节点，保证对每层的每个节点都处理到其子节点，并将值记录下来。队列用到Queue这个类，offer方法可以添加一个元素，peek方法获取队首的元素，poll方法会从队首移除一个元素并获取它。

# Python Code

```
# Definition for a binary tree node.
# class TreeNode(object):
#     def __init__(self, x):
#         self.val = x
#         self.left = None
#         self.right = None

class Solution(object):
    def levelOrder(self, root):
        """
        :type root: TreeNode
        :rtype: List[List[int]]
        """
        res = []
        if not root: return res
        queue = [root]
        while queue:
            # how many node in the same level
            n = len(queue)
            sublevel = []
            # put all nodes from the next level in the queue
            for i in range(n):
                if queue[-1].left:
                    queue.insert(0, queue[-1].left)
                if queue[-1].right:
                    queue.insert(0, queue[-1].right)
                sublevel.append(queue.pop().val)
            res.append(sublevel)
        return res
```

# 总结

算法的时间复杂度是就结点的数量O(n)，空间复杂度是一层的结点数，也是O(n)