# 【题目】

给定一个二叉树，返回其节点值自底向上的层次遍历。 （即按从叶子节点所在层到根节点所在的层，逐层从左向右遍历）

例如：

给定二叉树 [3,9,20,null,null,15,7],

3

/ \

9 20

/ \

15 7

返回其自底向上的层次遍历为：

[

[15,7],

[9,20],

[3]

]

# 【分析】

这个题目很明显是树的层次遍历，不够时对输出结果有了要求，一是要求按成输出，其次是按照层次的倒序输出。根据题目要求，我们可以识别到其知识点原型为树的宽度遍历(BFS)和堆的数据结构(FILO，先进后出)。



其构建两个队列(parentList,CildrenList)和个栈(Stack)，其原理为：

1：将跟节点加入parentList中，此时childrenList为空。

2：遍历partenList列表，做两件事一是将其子节点加入到childrenList中，二是将节点数据push到Stack中。并将partenList中的数据清空

3：将childrenList中的所有数据拷贝到partenList中，重复步骤2的动作，直到childrenList列表为空

4：将Stack中的数据pop出来，即满足题目的要求。

# 【代码】

具体代码实现参：<git@github.com:zhexueren/leetcode.git>

leetcode/src/leetcode/sloution1/