二叉树的层序遍历 II

给定一个二叉树，返回其节点值自底向上的层序遍历。 （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序遍历为：

```
[
  [15,7],
  [9,20],
  [3]
]
```

```cpp
/**
 * Definition for a binary tree node.
 * struct TreeNode {
 *     int val;
 *     TreeNode *left;
 *     TreeNode *right;
 *     TreeNode() : val(0), left(nullptr), right(nullptr) {}
 *     TreeNode(int x) : val(x), left(nullptr), right(nullptr) {}
 *     TreeNode(int x, TreeNode *left, TreeNode *right) : val(x), left(left
), right(right) {}
 * };
 */

class Solution {
public:
    vector<vector<int>> levelOrderBottom(TreeNode* root) {
        vector<vector<int>> res;
```

```cpp
        queue<TreeNode*> q;
        if(root)
        {
            q.push(root);
        }
        while(!q.empty())
        {
            int row=q.size();
            vector<int> tmp;
            while(row--)
            {
                TreeNode* cur=q.front();
                q.pop();
                tmp.push_back(cur->val);

                if(cur->left)
                {
                    q.push(cur->left);
                }
                if(cur->right)
                {
                    q.push(cur->right);
                }
            }
            res.push_back(tmp);
        }
        reverse(res.begin(),res.end());
        return res;
    }
};
```