LC 199. Binary Tree Right Side View

Question

Given a binary tree, imagine yourself standing on the *right* side of it, return the values of the nodes you can see ordered from top to bottom.

Example:

```
Input: [1,2,3,null,5,null,4]
Output: [1, 3, 4]
Explanation:

   1            <---
 /   \
2     3         <---
 \     \
  5     4       <---
```

Solution

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

class Solution:
    def rightSideView(self, root):
        """
        :type root: TreeNode
        :rtype: List[int]
        """
        #Solution
        if not root:
            return []
        stack = [(root, 0)]
        ans = []
        while stack:
            node, level = stack.pop()
            if not node:
                continue
            if len(ans) == level:
                ans.append(node.val)
```

```
            if node.left:
                stack.append((node.left, level + 1))
            if node.right:
                stack.append((node.right, level + 1))
        return ans
```