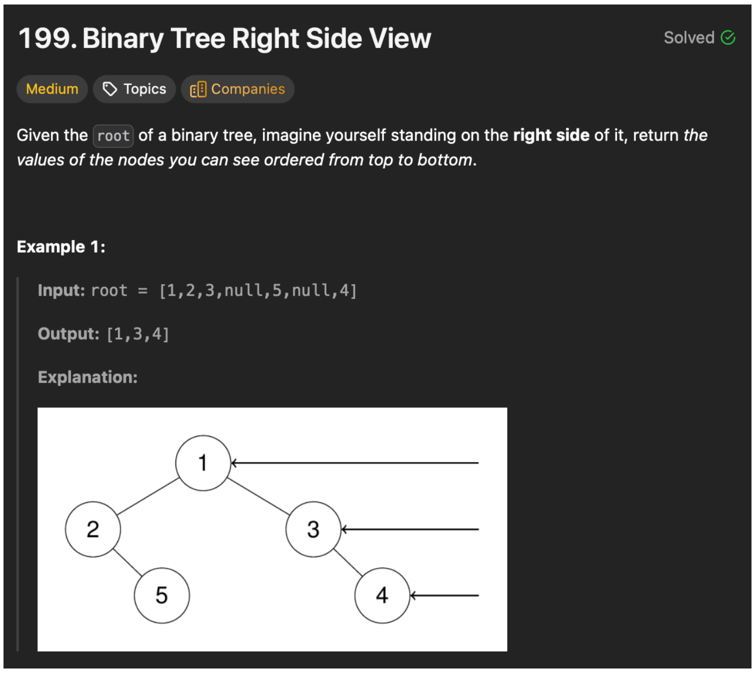


199. Binary Tree Right Side View

Solved ✓

Medium | Topics | Companies

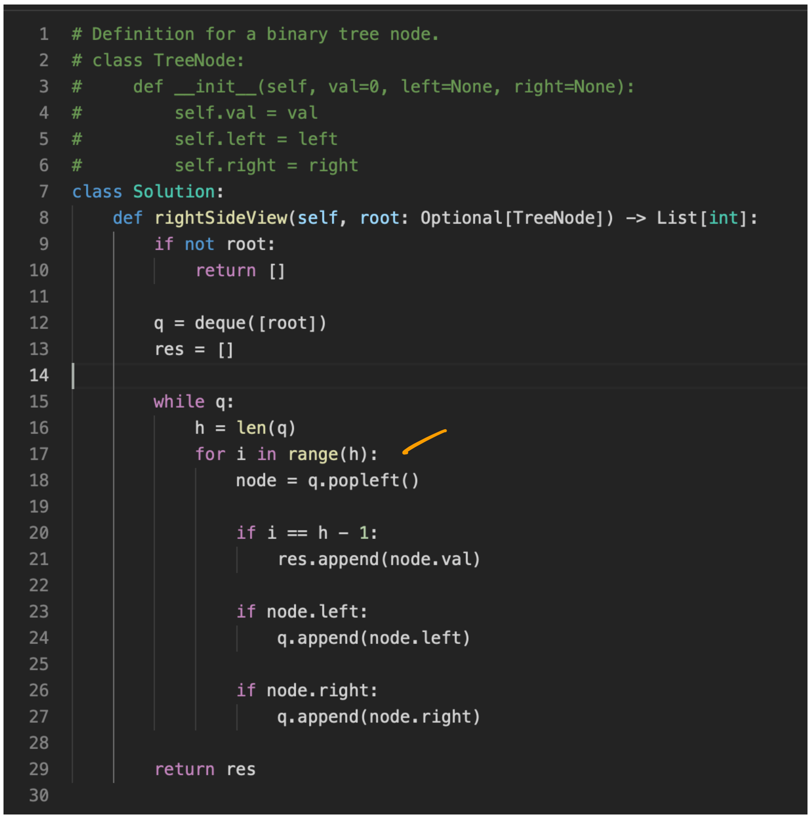
Given the `root` of a binary tree, imagine yourself standing on the **right side** of it, return *the values of the nodes you can see ordered from top to bottom.*

Example 1:

Input: root = [1,2,3,null,5,null,4]

Output: [1,3,4]

Explanation:



```python
# Definition for a binary tree node.
# class TreeNode:
#     def __init__(self, val=0, left=None, right=None):
#         self.val = val
#         self.left = left
#         self.right = right
class Solution:
    def rightSideView(self, root: Optional[TreeNode]) -> List[int]:
        if not root:
            return []

        q = deque([root])
        res = []

        while q:
            h = len(q)
            for i in range(h):
                node = q.popleft()

                if i == h - 1:
                    res.append(node.val)

                if node.left:
                    q.append(node.left)

                if node.right:
                    q.append(node.right)

        return res

```

- BFS + height measurement.

- for i in range(h):

  iterate through each height.

  if i == h-1,

  then it's the last node of this height, AKA the right most.

- add child from left to right, make sure the right most is always the tail.