

Lowest Common Ancestor I

```
1 # Definition for a binary tree node.
2 # class TreeNode(object):
3 #     def __init__(self, x):
4 #         self.val = x
5 #         self.left = None
6 #         self.right = None
7 class Solution(object):
8     def lowestCommonAncestor(self, root, p, q):
9         """
10         :type root: TreeNode
11         :type p: TreeNode
12         :type q: TreeNode
13         :rtype: TreeNode
14         """
15         if not root:
16             return None
17         if root == p or root == q:
18             return root
19         left_node = self.lowestCommonAncestor(root.left, p, q)
20         right_node = self.lowestCommonAncestor(root.right, p, q)
21         if left_node and right_node:
22             return root
23         if left_node:
24             return left_node
25         return right_node
```