Mistake-> Didn’t read question properly.

Post order or preorder selection.

Root should be changed. So, if root is changed then how to traverse, so postorder with right first and then left is used.

class Solution:

def flatten(self, root):

"""

:type root: TreeNode

:rtype: void Do not return anything, modify root in-place instead.

"""

self.prev=None

self.temp=None

def dfs(root):

if root is None:

return None

dfs(root.right)

dfs(root.left)

root.left=None

root.right=self.prev

self.prev=root

self.temp=root

# print(self.temp.val,"local temp")

dfs(root)

class Solution:

def flatten(self, root):

"""

:type root: TreeNode

:rtype: void Do not return anything, modify root in-place instead.

"""

def helper(root):

if root is None:

return

left\_leaf = root

if root.left:

left\_leaf = helper(root.left)

left\_leaf.right = root.right

root.right = root.left

root.left = None

if root.right:

left\_leaf = helper(root.right)

return left\_leaf

helper(root)