# Definition for a binary tree node.

# class TreeNode(object):

# def \_\_init\_\_(self, val=0, left=None, right=None):

# self.val = val

# self.left = left

# self.right = right

class Solution(object):

def rangeSumBST(self, root, low, high):

"""

:type root: TreeNode

:type low: int

:type high: int

:rtype: int

"""

def range\_sum(node):

if not node:

return 0

left\_sum, right\_sum = range\_sum(node.left), range\_sum(node.right)

if low <= node.val <= high:

stage\_sum = left\_sum + right\_sum + node.val

else:

stage\_sum = left\_sum + right\_sum

return stage\_sum

return range\_sum(root)