```
class BTNode:
  def __init__(self, elem):
    self.elem = elem
    self.right = None
    self.left = None
```

SET-A:

Rubric:

1. Calculate left,right sub tree submission (2.5*2)
2. Use condition for level checking and calculate summation correctly for even odd levels (5)
3. For set A add the calculated summation of left sub tree and right sub tree of root, for set B multiple the calculated summation of left sub tree with right sub tree of root (5)

```
def leftRightAddition(root):
  if root == None:
    return None
  return sum(root.left) + sum(root.right)

def sum(root,level = 1):
  if root == None:
    return 0
  if leve


#Driver Code
root=BTNode(1)
#Write other nodes by yourself from the given tree of Doc File
node1 = BTNode(2)
node2 = BTNode(3)
node3 = BTNode(4)
node4 = BTNode(5)
node5 = BTNode(6)
node6 = BTNode(7)


root.left = node1
root.right = node2
node1.left = node3
node1.right = node4
node2.left = node5
node2.right = node6



print(leftRightAddition(root)) #This should print 17
```

```
28
```

Rubric:

1. Calculate left,right sub tree submission (2.5*2)
2. Use condition for level checking and calculate summation correctly for even odd levels (5)
3. For set A add the calculated summation of left sub tree and right sub tree of root, for set B multiple the calculated summation of left sub tree with right sub tree of root (5)

SET-B:

```
def leftRightMultiplication(root):
  if root == None:
    return 0

  left_sum = sum(root.left,1)
  right_sum = sum(root.right,1)

  return left_sum * right_sum


def sum(node,level):

  if node == None:
    return 0

  if level %2 ==0:
    elem = -node.elem
  else:
    elem = node.elem


  return elem+sum(node.left, level+1)+sum(node.right, level+1)




#Driver Code
root=BTNode(1)
#Write other nodes by yourself from the given tree of Doc File
node1 = BTNode(2)
node2 = BTNode(3)
node3 = BTNode(4)
node4 = BTNode(5)
node5 = BTNode(6)
node6 = BTNode(7)


root.left = node1
root.right = node2
node1.left = node3
node1.right = node4
node2.left = node5
node2.right = node6



print(leftRightMultiplication(root))
```

```
70
```

Start coding or generate with AI.