| **Set A**  def left\_energy\_signature(root):  if root is None:  return 0  current\_sum = 0  if root.left is not None and root.right is None:  left\_child = root.left  if left\_child.left is None and left\_child.right is None:  current\_sum = root.elem  left\_sum = left\_energy\_signature(root.left)    right\_sum = left\_energy\_signature(root.right)  return current\_sum + left\_sum + right\_sum | **Set B**  def right\_energy\_signature(root):    if root is None:  return 0  current\_sum = 0  if root.left is None and root.right is not None:  right\_child = root.right  if right\_child.left is None and right\_child.right is None:  current\_sum = root.elem    left\_sum = right\_energy\_signature(root.left)  right\_sum = right\_energy\_signature(root.right)    return current\_sum + left\_sum + right\_sum |
| --- | --- |

| **Part** | **Marks** |
| --- | --- |
| Correct Base Case Handling | 1 |
| Current Sum Initialization & Updating | 2 |
| Left Child Check (Root), Right Child Check (Root) | 3 |
| Left Child Check (Child), Right Child Check (Child) | 3 |
| Recursive Left Child Check | 2 |
| Recursive Right Child Check | 2 |
| Returning Combined Sum | 2 |
|  | 15 |