**Add all greater values to every node in a BST:-**

Given a BST, modify it so that all greater values in the given BST are added to every node.

**Example 1:**

**Input:**

  50

  / \

  30 70

  / \ / \

  20 40 60 80

**Output:** 350 330 300 260 210 150 80

**Explanation:**The tree should be modified to

following:

  260

  / \

330 150

/ \ / \

  350 300 210 80

**Example 2:**

**Input:**

  2

  / \

  1 5

  / \

  4 7

**Output:** 19 18 16 12 7

**Your Task:**  
You don't need to read input or print anything. Your task is to complete the function **modify()**which takes one argument: root of the BST. The function should contain the logic to modify the BST so that in the modified BST, every node has a value equal to the sum of its value in the original BST and values of all the elements larger than it in the original BST. Return the root of the modified BST. The driver code will print the inorder traversal of the returned BST/  
  
**Expected Time Complexity:**O(N)  
**Expected Auxiliary Space:**O(Height of the BST).

**Constraints:**  
1<=N<=100000