538. Convert BST to Greater Tree

Difficulty Easy 867 64 Discuss Favorite Share

Description | Hints | Solution | Submissions

Given a Binary Search Tree (BST), convert it to a Greater Tree such that every key of the original BST is changed to the original key plus sum of all keys greater than the original key in BST.

Example:

```
Input: The root of a Binary Search Tree like this:
              5
            /   \
           2     13

Output: The root of a Greater Tree like this:
             18
            /   \
          20     13
```

Contributor

```java
public class L538 {

    public class TreeNode{
        int val;
        TreeNode left;
        TreeNode right;
        public TreeNode(int val) {
            this.val = val;
        }
    }
    /*
     * 这题可以借鉴中序遍历的思想，但是不同的在于，它要把整颗树中比自己大的点加起来，对于BST
     * 中的点，比自己大的点只有可能是自己右边的点，分治法先找到右边最大的，保存一个global的
     * sum，从右开始做的特殊的中序遍历到的每一个点的值都是自己本身加上之前遍历过的所有点的和。
     */
    private int sum = 0;

    public TreeNode converBST(TreeNode root) {
        if(root == null)
            return null;
        helper(root);
        return root;
    }

    private void helper(TreeNode root) {
        if(root == null)
            return;
        helper(root.right);
        root.val += sum;
        sum = root.val;
        helper(root.left);
    }
}
```