

**BrianChiang\_tw**

# #1038 Binary Search Tree to Greater Sum Tree

Python O(n) solution by DFS approach.

---

## Hint:

 1.

Recall that node value of BST always follows the rule: **value of Right sub-tree  $\geq$  value of Current node  $\geq$  value of Left sub-tree**

2.

In addition, in-order traversal to BST generates the sequence with ascending order. In-order traversal ordering: ( Left sub-tree, Current node, Right sub-tree)

3.

With symmetry, **reversed in-order traversal** to **BST** gives us the **sequence** with **descending order**. Reversed in-order traversal ordering: ( Right sub-tree, Current node, Left sub-tree)

4.

Finally, we can **build greater tree by maintaining a variable** to **accumulate greater nodes' values**, based on reversed in-order traversal ordering.

---

## Algorithm:

### Step\_#1:

Maintain a **global variable** to **keep accumulation value** of greater nodes

### Step\_#2:

From root node, start **DFS traversal** with the **reversed in-order**: ( Right sub-tree, Current node, Left sub-tree)

Also, update accumulation value, and value of current node on each run.

---

## Implementation:

```
1 class Solution:
2
3     def helper( self, node: TreeNode):
4
5         if not node:
6             # empty node or empty tree
7             return None
8
9         else:
10            # DFS to next level with order:( right, current, left )
11            self.helper( node.right )
12
13            self.accumulation_sum += node.val
14            node.val = self.accumulation_sum
15
16            self.helper( node.left )
17
18            return node
19
20
21
```

```
22     def bstToGst(self, root: TreeNode) -> TreeNode:
23
24         # accumulation sum of nodes
25         self.accumulation_sum = 0
26
27         return self.helper( root )
```

---

## Related leetcode challenge:

[Leetcode #94 Binary Tree Inorder Traversal](#)

[Leetcode #98 Validate Binary Search Tree](#)

[Leetcode #538 Convert BST to Greater Tree](#)

---

## Reference:

[1] [Wiki: Inorder traversal of binary tree](#)

[2] [Wiki: Binary search tree](#)