

Difficulty: ■ Category:   Successful Submissions: 43,997+

## Validate BST ○ ★

Write a function that takes in a potentially invalid Binary Search Tree (BST) and returns a boolean representing whether the BST is valid.

Each **BST** node has an integer **value**, a **left** child node, and a **right** child node. A node is said to be a valid **BST** node if and only if it satisfies the BST property: its **value** is strictly greater than the values of every node to its left; its **value** is less than or equal to the values of every node to its right; and its children nodes are either valid **BST** nodes themselves or **None** / **null**.

A BST is valid if and only if all of its nodes are valid **BST** nodes.

### Sample Input

```
tree = 10
      /  \
     5    15
    /  \  /  \
   2   5 13  22
  /       \
 1         14
```

### Sample Output

```
true
```

### Hints

Hint 1 ▼Hint 2 ▼Optimal Space & Time Complexity ▼