

98. Validate Binary Search Tree ★

[Question](#)[Editorial Solution](#)[My Submissions \(/problems/validate-binary-search-tree/submissions/\)](/problems/validate-binary-search-tree/submissions/)

Total Accepted: **118362** Total Submissions: **541429** Difficulty: **Medium** Contributors: **Admin**

Given a binary tree, determine if it is a valid binary search tree (BST).

Assume a BST is defined as follows:

- The left subtree of a node contains only nodes with keys **less than** the node's key.
- The right subtree of a node contains only nodes with keys **greater than** the node's key.
- Both the left and right subtrees must also be binary search trees.

Example 1:

```
  2
 / \
1   3
```

Binary tree [2,1,3] , return true.

Example 2:

```
  1
 / \
2   3
```

Binary tree [1,2,3] , return false.

Subscribe (/subscribe/) to see which companies asked this question

[Show Tags](#)[Show Similar Problems](#)

Have you met this question in a real interview?

Discuss (<https://leetcode.com/discuss/questions/oj/validate-binary-search-tree>)

Pick One (/problems/random-one-question/)

Notes

C++



</>

```
1 /**
2  * Definition for a binary tree node.
```

```

3  * struct TreeNode {
4  *      int val;
5  *      TreeNode *left;
6  *      TreeNode *right;
7  *      TreeNode(int x) : val(x), left(NULL), right(NULL) {}
8  * };
9  */
10 class Solution {
11 public:
12     bool isValidBST(TreeNode* root, long long high, long long low) {
13         if (!root) return true;
14         if (root->left)
15             if (root->val <= root->left->val)
16                 return false;
17         if (root->right)
18             if (root->val >= root->right->val)
19                 return false;
20         if (root->val >= high) return false;
21         if (root->val <= low) return false;
22
23         return isValidBST(root->left, root->val, low) && isValidBST(root->right,
24     }
25
26     bool isValidBST(TreeNode* root) {
27         return isValidBST(root, (long long)INT_MAX+1, (long long)INT_MIN-1);
28     }
29 };

```

Custom Testcase ☐

Run Code

Submit Solution

✉ Send Feedback (<mailto:admin@leetcode.com?subject=Feedback>)

Frequently Asked Questions (/faq/) | Terms of Service (/tos/)

[Privacy](#)

Copyright © 2016 LeetCode

Notes