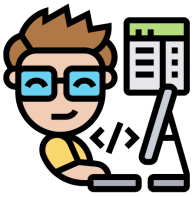


DAILY PROGRAMMING

CHALLENGE

WISDOM SPROUTS
IT Training Hub

Check if a Binary Tree is a Binary Search Tree (BST)

You are given the root of a binary tree. Your task is to determine whether the tree is a valid Binary Search Tree (BST). A binary search tree is defined as a tree where:

- Every node's left subtree contains only nodes with values less than the node's value.
- Every node's right subtree contains only nodes with values greater than the node's value.
- Both the left and right subtrees must also be binary search trees.

Input:

- A binary tree represented by its root node.

Output:

- Return true if the binary tree is a valid BST, otherwise return false.

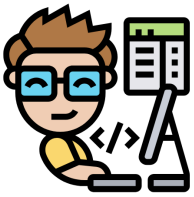
Examples:

- Example 1
  Input: root = [2, 1, 3]
  Output: true
  Explanation:
  - The tree is as follows

  - The tree satisfies the BST property
    - Node 1 (left of 2) is less than 2.
    - Node 3 (right of 2) is greater than 2.

Constraints:

- The number of nodes in the binary tree is in the range $[1,10^4]$.
- The node values are in the range $[-10^5,10^5]$



DAILY PROGRAMMING

CHALLENGE

WISDOM SPROUTS
IT Training Hub

Test Cases:

1. Input: root = [2, 1, 3]
   Output: true
2. Input: root = [5, 1, 4, null, null, 3, 6]
   Output: false
3. Input: root = [10, 5, 15, null, null, 6, 20]
   Output: false

Edge Cases:

1. Tree with only one node.
2. Tree with repeated values.
3. Tree where a subtree violates the BST property at deeper levels.