98. Validate Binary Search Tree

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.

```java
class Solution {
    public boolean isValidBST(TreeNode root) {

        return helper(root, null, null);
    }

    public boolean helper(TreeNode root, Integer min, Integer max) {

        if (root == null) {
            return true;
        }

        if ((min != null && root.val <= min)
        || (max != null && root.val >= max)) {
            return false;
        }

        return helper(root.left, min, root.val)
        && helper(root.right, root.val, max);
    }
}
```