

Validate Binary Search Tree

To determine if a binary tree is a valid Binary Search Tree (BST), we use the key property:

- For every node, all values in the left subtree must be strictly less than the node's value, and all values in the right subtree must be strictly greater than the node's value.
- This condition must hold recursively for all nodes.

Approach: Recursive with Value Bounds

Idea:

- Carry minimum and maximum allowed values down the recursion:
 - Initially, $(-\infty, \infty)$.
 - For the left child: new range = (min, current node value)
 - For the right child: new range = (current node value, max)
- If any node violates the range, return false.

Complexity:

- Time: $O(n)$ - visits each node once.
- Space: $O(h)$ - recursion stack (h = tree height)

Why use long long?

- Node values can be -2^{31} to $2^{31}-1$. We use long long to avoid overflow when setting bounds beyond these values.