① 2Sum BST: Return true if there are two nodes in the tree that equal target number.

Algo.

① Simultaneously do inorder and reverse inorder and add to two stacks.

② Now, iterate the stacks and check sum. if sum > target pop from s2, else pop from s1.

```
boolean 2SumBST(Node root, int target){
    if(root == null) return false.

    Stack<Integer> s1 = new Stack<>();
    Stack<Integer> s2 = new Stack<>();

    inOrder(root, s1);
ReverseInOrder(root, s2);
    while(!s1.isEmpty() && !s2.isEmpty()) {
        int sum = s1.peek() + s2.peek();
        sum == target && (s1.peek() != s2.peek())
        if(sum == target) return true;
        if(sum > target) s1.pop();
        else s2.pop();
    }
    return false;
}
```