Solution 1

```swift
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

class Program {
    class BST {
        var value: Int?
        var left: BST?
        var right: BST?

        init(value: Int) {
            self.value = value
            left = nil
            right = nil
        }
    }

    // O(n) time | O(d) space
    func validateBST(tree: BST) -> Bool {
        var minimum = Int(Int32.min)
        var maximum = Int(Int32.max)
        return validateBSTHelper(tree: tree, minimum: &minimum,
    }

    func validateBSTHelper(tree: BST?, minimum: inout Int, maxim
        if tree === nil {
            return true
        }

        if let tree = tree, let value = tree.value, value < mini
            return false
        }

        if var treeValue = tree?.value {
            let leftIsValid = validateBSTHelper(tree: tree?.left
```

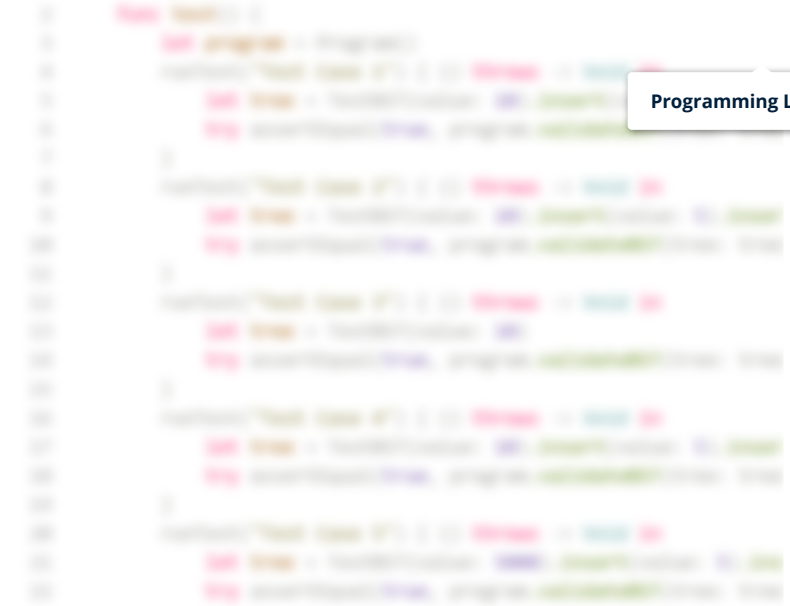
Solution 1 Solution 2 Solution 3

```swift
class Program {
    // This is an input class. Do not edit.
    class BST {
        var value: Int?
        var left: BST?
        var right: BST?

        init(value: Int) {
            self.value = value
            left = nil
            right = nil
        }
    }

    func validateBST(tree: BST) -> Bool {
        // Write your code here.
        return false
    }
}

```



Programming Language

Run or submit code when you're ready.