AlgoExpert Quad Layout Go 14px Sublime 00:00:00

Our Solution(s) Run Code

Solution 1

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

package main

import "math"

type BST struct {
  Value int

  Left  *BST
  Right *BST
}

// O(n) time | O(d) space
func (tree *BST) Validate() bool {
  return tree.validate(math.MinInt32, math.MaxInt32)
}

func (tree *BST) validate(min, max int) bool {
  if tree.Value < min || tree.Value >= max {
    return false
  }
  if tree.Left != nil && !tree.Left.validate(min, tree.Value) {
    return false
  }
  if tree.Right != nil && !tree.Right.validate(tree.Value, max)
    return false
  }
  return true
}
```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```go
package main

type BST struct {
  Value int

  Left  *BST
  Right *BST
}

func (tree *BST) Validate() bool {
  // Write your code here.
  return false
}
```

Our Tests

Custom Output Submit Code

```go
func NewBST(root int, values ...int) *BST {
  tree := &BST{Value: root}
  for _, value := range values {
    tree.Insert(value)
  }
  return tree
}

func (tree *BST) Insert(value int) {
  if value < tree.Value {
    if tree.Left == nil {
      tree.Left = &BST{Value: value}
    } else {
      tree.Left.Insert(value)
    }
  } else {
    if tree.Right == nil {
      tree.Right = &BST{Value: value}
    } else {
      tree.Right.Insert(value)
```

Run or submit code when you're ready.