AlgoExpert

24

26

27 28

29 30 31

33

Quad Layout

Java

14рх

Your Solutions

Sublime

Monokai

00:00:

Run Code

Our Solution(s) Run Code

```
Solution 1 Solution 2
 1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
   import java.util.*;
   class Program {
     // O(n^2) time | O(n^2) space - where n is the number of
      \ensuremath{//} nodes in each array, respectively
      public static boolean sameBsts(List<Integer> arrayOne, List<Integer</pre>
        if (arrayOne.size() != arrayTwo.size()) return false;
10
        if (arrayOne.size() == 0 && arrayTwo.size() == 0) return true;
11
12
        if (arrayOne.get(0).intValue() != arrayTwo.get(0).intValue()) ret
14
15
        List<Integer> leftOne = getSmaller(arrayOne);
        List<Integer> leftTwo = getSmaller(arrayTwo);
16
17
        List<Integer> rightOne = getBiggerOrEqual(arrayOne);
        List<Integer> rightTwo = getBiggerOrEqual(arrayTwo);
18
19
20
        return sameBsts(leftOne, leftTwo) && sameBsts(rightOne, rightTwo);
21
22
23
      public static List<Integer> getSmaller(List<Integer> array) {
```

List<Integer> smaller = new ArrayList<Integer>();

if (array.get(i).intValue() < array.get(0).intValue()) smaller.a</pre>

public static List<Integer> getBiggerOrEqual(List<Integer> array) {
 List<Integer> biggerOrEqual = new ArrayList<Integer>();

for (int i = 1; i < array.size(); i++) {</pre>

for (int i = 1; i < array.size(); i++) {</pre>

return smaller;

```
Solution 1 Solution 2 Solution 3

1 import java.util.*;
2
3 class Program {
4   public static boolean sameBsts(List<Integer> arrayOne, List<Integer>
5   // Write your code here.
6   return false;
7   }
8 }
```

 Our Tests
 Custom Output
 Submit Code



Run or submit code when you're ready.