Solution 1 Solution 2

Your Solutions

Run Code

Our Solution(s) Run Code

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
   #include <vector>
   using namespace std;
   vector<int> getSmaller(vector<int>);
   vector<int> getBiggerOrEqual(vector<int> array);
10 // O(n^2) time | O(n^2) space - where n is the number of
11 // nodes in each array, respectively
12 bool sameBsts(vector<int> arrayOne, vector<int> arrayTwo) {
     if (arrayOne.size() != arrayTwo.size())
14
       return false;
15
     if (arrayOne.size() == 0 && arrayTwo.size() == 0)
16
17
       return true;
18
19
     if (arrayOne[0] != arrayTwo[0])
20
       return false;
21
22
     vector<int> leftOne = getSmaller(arrayOne);
23
     vector<int> leftTwo = getSmaller(arrayTwo);
     vector<int> rightOne = getBiggerOrEqual(arrayOne);
25
     vector<int> rightTwo = getBiggerOrEqual(arrayTwo);
26
27
     return sameBsts(leftOne, leftTwo) && sameBsts(rightOne, rightTwo);
28 }
29
30
   vector<int> getSmaller(vector<int> array) {
31
     vector<int> smaller = {};
     for (int i = 1; i < array.size(); i++) {</pre>
32
```

if (array[i] < array[0])</pre>

33

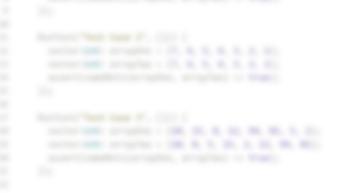
```
Solution 1 Solution 2 Solution 3
```

```
#include <vector>

using namespace std;

bool sameBsts(vector<int> arrayOne, vector<int> arrayTwo) {
   // Write your code here.
   return false;
}
```

 Our Tests
 Custom Output
 Submit Code



Run or submit code when you're ready.