AlgoExpert

Solution 1

**Quad Layout** 

++

14рх

Sublime

Solution 1 Solution 2 Solution 3

Monokai

00:00:

Our Solution(s) Run

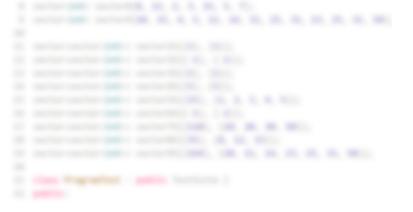
```
Run Code
```

```
Your Solutions Run Code
```

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
   #include <vector>
   #include <climits>
 5 using namespace std;
   vector<vector<int>> buildSequence(vector<int> array, vector<int> sequ
                                      int currentIdx, int sum);
10 // O(n^2) time | O(n) space
11 vector<vector<int>> maxSumIncreasingSubsequence(vector<int> array) {
12
     vector<int> sequences(array.size(), INT_MIN);
13
     vector<int> sums = array;
14
     int maxSumIdx = 0;
15
     for (int i = 0; i < array.size(); i++) {</pre>
16
       int currentNum = array[i];
17
       for (int j = 0; j < i; j++) {</pre>
18
         int otherNum = array[j];
19
         if (otherNum < currentNum && sums[j] + currentNum >= sums[i]) {
20
           sums[i] = sums[j] + currentNum;
21
           sequences[i] = j;
24
        if (sums[i] >= sums[maxSumIdx]) {
25
         maxSumIdx = i;
26
27
28
     return buildSequence(array, sequences, maxSumIdx, sums[maxSumIdx]);
29 }
30
31
   vector<vector<int>> buildSequence(vector<int> array, vector<int> seque
32
                                      int currentIdx, int sum) {
     vector<vector<int>> sequence = {{}}, {}};
33
```

```
1 #include <vector>
2 using namespace std;
3
4 vector<vector<int>> maxSumIncreasingSubsequence(vector<int>> array) {
5    // Write your code here.
6    return {};
7 }
8
```

 Our Tests
 Custom Output
 Submit Code



Run or submit code when you're ready.