AlgoExpert Quad Layout C++ 12px Sublime Monokai 00:00:00

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1

41

```
1\, // Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 3 #include <vector>
 4 #include <climits>
 5 using namespace std;
 7 vector<vector<int>> buildSequence(vector<int> array, vector<int> sequences,
                                        int currentIdx, int sum);
10 // O(n^2) time | O(n) space
11 vector<vector<int>> maxSumIncreasingSubsequence(vector<int> array) {
      vector<int> sequences(array.size(), INT_MIN);
      vector<int> sums = array;
13
14
      int maxSumIdx = 0;
      for (int i = 0; i < array.size(); i++) {</pre>
15
       int currentNum = array[i];
16
17
        for (int j = 0; j < i; j++) {
          int otherNum = array[j];
18
           \textbf{if} \ (\texttt{otherNum} \ \land \ \texttt{currentNum} \ \&\& \ \texttt{sums[j]} \ + \ \texttt{currentNum} \ \gt= \ \texttt{sums[i]}) \ \{ \\
19
20
            sums[i] = sums[j] + currentNum;
21
            sequences[i] = j;
22
23
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> sequences,
                                       int currentIdx, int sum) {
32
      vector<vector<int>>> sequence = {{}}, {{}}};
33
34
      sequence[0].push_back(sum);
      while (currentIdx != INT_MIN) {
35
        sequence[1].insert(sequence[1].begin(), array[currentIdx]);
36
        currentIdx = sequences[currentIdx];
37
38
39
      return sequence;
40 }
```