AlgoExpert Quad Layout JavaScript 12px Sublime Monokai 00:00:00

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1

32

```
1\, // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
 3 // O(n^2) time | O(n) space
  \  \, \text{4} \quad \text{function } \text{maxSumIncreasingSubsequence(array)} \, \, \{ \\
    const sequences = new Array(array.length);
     const sums = array.map(num => num);
     let maxSumIdx = 0;
      for (let i = 0; i < array.length; i++) {</pre>
       const currentNum = array[i];
10
       for (let j = 0; j < i; j++) {
11
         const otherNum = array[j];
          if (otherNum < currentNum && sums[j] + currentNum >= sums[i]) {
12
13
           sums[i] = sums[j] + currentNum;
           sequences[i] = j;
14
15
16
17
        if (sums[i] >= sums[maxSumIdx]) maxSumIdx = i;
18
19
      return [sums[maxSumIdx], buildSequence(array, sequences, maxSumIdx)];
20 }
21
22 function buildSequence(array, sequences, currentIdx) \{
     const sequence = [];
      while (currentIdx !== undefined) {
24
25
       sequence.unshift(array[currentIdx]);
26
       currentIdx = sequences[currentIdx];
27
28
     return sequence;
29 }
30
31 exports.maxSumIncreasingSubsequence = maxSumIncreasingSubsequence;
```