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

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1 Solution 2

33

```
_{\rm 1} \, # Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 3 # O(nlogn) time | O(n) space
    def longestIncreasingSubsequence(array):
        sequences = [None for x in array]
        indices = [None for x in range(len(array) + 1)]
        length = 0
        for i, num in enumerate(array):
            newLength = binarySearch(1, length, indices, array, num)
            sequences[i] = indices[newLength - 1]
10
11
            indices[newLength] = i
12
            length = max(length, newLength)
13
        return buildSequence(array, sequences, indices[length])
14
15
16 def binarySearch(startIdx, endIdx, indices, array, num):
17
        if startIdx > endIdx:
18
            return startIdx
19
        middleIdx = (startIdx + endIdx) // 2
20
        if array[indices[middleIdx]] < num:</pre>
21
22
23
           startIdx = middleIdx + 1
        else:
           endIdx = middleIdx - 1
24
        return binarySearch(startIdx, endIdx, indices, array, num)
25
26
27 def buildSequence(array, sequences, currentIdx):
28
        sequence = []
        while currentIdx is not None:
30
            sequence.append(array[currentIdx])
31
            currentIdx = sequences[currentIdx]
        return list(reversed(sequence))
```