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

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

```
Solution 1 Solution 2
```

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
 3 package main
 5 // O(nlogn) time | O(n) space
 6 func LongestIncreasingSubsequence(array []int) []int {
      sequences := make([]int, len(array))
      indices := make([]int, len(array)+1)
      for i := range array {
       sequences[i] = -1
10
11
        indices[i] = -1
12
13
      length := 0
      for i, num := range array {
14
        newLength := binarySearch(1, length, indices, array, num)
15
16
        sequences[i] = indices[newLength-1]
        indices[newLength] = i
17
18
        length = max(length, newLength)
19
20
      return buildSequence(array, sequences, indices[length])
21 }
22
23 \, func binarySearch(startIndex, endIndex \, int, indices, array []int, num \, int) \, int {
24
      if startIndex > endIndex {
25
        return startIndex
26
27
      middleIndex := (startIndex + endIndex) / 2
      if array[indices[middleIndex]] < num {</pre>
28
29
       startIndex = middleIndex + 1
30
      } else {
31
        endIndex = middleIndex - 1
32
33
      return binarySearch(startIndex, endIndex, indices, array, num)
34 }
35
    func buildSequence(array, sequences []int, index int) []int {
36
37
     out := []int{}
38
      for index != -1 {
39
       out = append(out, array[index])
40
       index = sequences[index]
41
42
      reverse(out)
43
      return out
44 }
45
    func max(arg int, rest ...int) int {
      for _, num := range rest {
47
      if num > arg {
48
49
         arg = num
50
51
52
      return arg
53 }
54
55 func reverse(numbers []int) {
56  for i, j := 0, len(numbers)-1; i < j; i, j = i+1, j-1 {</pre>
57
       numbers[i], numbers[j] = numbers[j], numbers[i]
59 }
```