

PromptScratchpadOur Solution(s)Video Explanation

Run Code

Solution 1Solution 2Solution 3

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 package main
4
5 // O(n^2) time | O(n) space - where in is the length of the input array
6 func MinRewards(scores []int) int {
7     rewards := make([]int, len(scores))
8     fill(rewards, 1)
9     for i := 1; i < len(scores); i++ {
10         j := i - 1
11         if scores[i] > scores[j] {
12             rewards[i] = rewards[j] + 1
13             continue
14         }
15         for j >= 0 && scores[j] > scores[j+1] {
16             rewards[j] = max(rewards[j], rewards[j+1]+1)
17             j--
18         }
19     }
20     return sum(rewards)
21 }
22
23 func fill(arr []int, val int) {
24     for i := range arr {
25         arr[i] = val
26     }
27 }
28
29 func sum(arr []int) int {
30     sum := 0
31     for i := range arr {
32         sum += arr[i]
33     }
34     return sum
35 }
36
37 func max(i, j int) int {
38     if i > j {
39         return i
40     }
41     return j
42 }
43
```