Prompt   Scratchpad   Our Solution(s)   Video Explanation   Run Code

Solution 1   Solution 2   Solution 3

```go
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

package main

// O(n) time | O(n) space - where in is the length of the input array
func MinRewards(scores []int) int {
  rewards := make([]int, len(scores))
  fill(rewards, 1)
  for i := 1; i < len(scores); i++ {
    if scores[i] > scores[i-1] {
      rewards[i] = rewards[i-1] + 1
    }
  }
  for i := len(scores) - 2; i >= 0; i-- {
    if scores[i] > scores[i+1] {
      rewards[i] = max(rewards[i], rewards[i+1]+1)
    }
  }
  return sum(rewards)
}

func fill(arr []int, val int) {
  for i := range arr {
    arr[i] = val
  }
}

func sum(arr []int) int {
  sum := 0
  for i := range arr {
    sum += arr[i]
  }
  return sum
}

func max(i, j int) int {
  if i > j {
    return i
  }
  return j
}
```