

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1	Solution 2	Solution 3
<pre>1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved. 2 3 using System; 4 using System.Linq; 5 6 // O(n^2) time   O(n) space - where n is the length of the input array 7 public class Program { 8     public static int MinRewards(int[] scores) { 9         int[] rewards = new int[scores.Length]; 10         Array.Fill(rewards, 1); 11         for (int i = 1; i &lt; scores.Length; i++) { 12             int j = i - 1; 13             if (scores[i] &gt; scores[j]) { 14                 rewards[i] = rewards[j] + 1; 15             } else { 16                 while (j &gt;= 0 &amp;&amp; scores[j] &gt; scores[j + 1]) { 17                     rewards[j] = Math.Max(rewards[j], rewards[j + 1] + 1); 18                     j--; 19                 } 20             } 21         } 22         return rewards.Sum(); 23     } 24 } 25</pre>		
Solution 1	Solution 2	Solution 3
<pre>1 public class Program { 2     public static int MinRewards(int[] scores) { 3         // Write your code here. 4         return -1; 5     } 6 } 7</pre>		

Our Tests

Custom Output

Submit Code

<pre>1 public class Program { 2     // ... 3     public static int MinRewards(int[] scores) { 4         // ... 5         return -1; 6     } 7 }</pre>	
---	--

```
18
19
20 (Test)
21 #addition = add TestNumber()
22 #addition = add TestNumber(10, 20)
23 #addition = Program.MainMethod()
24 #addition = 1
25 #Main.MainMethod()
26
27
28 (Test)
29 #addition = add TestNumber()
30 #addition = add TestNumber(10, 20)
31 #addition = Program.MainMethod()
32 #addition = 1
```

Run or submit code when you're ready.