

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 import java.util.*; 4 import java.util.stream.*; 5 6 // O(n^2) time O(n) space - where n is the length of the input array 7 class Program { 8 public static int minRewards(int[] scores) { 9 int[] rewards = new int[scores.length]; 10 Arrays.fill(rewards, 1); 11 for (int i = 1; i < scores.length; i++) { 12 int j = i - 1; 13 if (scores[i] > scores[j]) { 14 rewards[i] = rewards[j] + 1; 15 } else { 16 while (j >= 0 && scores[j] > scores[j + 1]) { 17 rewards[j] = Math.max(rewards[j], rewards[j + 1] + 1); 18 j--; 19 } 20 } 21 } 22 return IntStream.of(rewards).sum(); 23 } 24 } 25</pre>		
Solution 1	Solution 2	Solution 3
<pre>1 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

1 class Program {

2 public static int minRewards(int[] scores) {

3 // Write your code here.

4 return -1;

5 }

6 }

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

```
10 #print
11 #print testFunction()
12 #print assertFunction(testFunction(), 10, 10) == 10
13 }
14
15 #print
16 #print testFunction()
17 #print assertFunction(testFunction(), 10, 10) == 10
18 }
19
20 #print
21 #print testFunction()
22 #print assertFunction(testFunction(), 10, 1, 1, 10) == 10
23 }
24
25
```

Run or submit code when you're ready.