

Test Name:

Summary

Timeline

Feedback

Tasks summary

Task	Time spent	Score
Distinct C#	2 min	75%

Total score

75%

Tasks Details

Easy	1. Distinct	Task Score	Correctness	Performance
	Compute number of distinct values in an array.	75%	100%	0%

Task description

Write a function

```
class Solution { public int solution(int[] A); }
```

that, given an array A consisting of N integers, returns the number of distinct values in array A.

For example, given array A consisting of six elements such that:

```
A[0] = 2    A[1] = 1    A[2] = 1
A[3] = 2    A[4] = 3    A[5] = 1
```

the function should return 3, because there are 3 distinct values appearing in array A, namely 1, 2 and 3.

Write an **efficient** algorithm for the following assumptions:

- N is an integer within the range [0..100,000];
- each element of array A is an integer within the range [−1,000,000..1,000,000].

Copyright 2009–2020 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

Solution

Programming language used: C#

Total time used:

2 minutes

?

Effective time used:

2 minutes

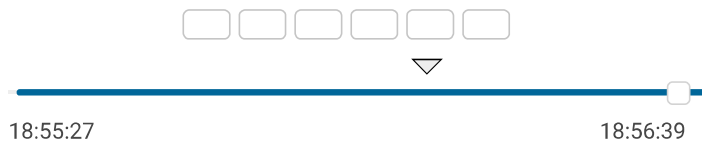
?

Notes:

not defined yet

Task timeline

?



Code: 18:56:39 UTC, cs, final,
score: 75

show code in pop-up

```
1 using System;
2 using System.Linq;
3 using System.Collections.Generic;
4 // you can also use other imports, for example:
5 // using System.Collections.Generic;
6
7 // you can write to stdout for debugging purposes, e.g.
8 // Console.WriteLine("this is a debug message");
9
10 class Solution {
11     public int solution(int[] A)
12     {
13         List<int> lst = new List<int>();
14         if (A == null || A.Length == 0)
15             return 0;
16         if (A.Length == 1)
17             return 1;
18         lst.Add(A[0]);
19         int len = A.Length;
20         for (int i = 1; i < len; i++)
21         {
22             if (!lst.Contains(A[i]))
23                 lst.Add(A[i]);
24         }
25         return lst.Count;
26     }
27 }
28 }
```

Analysis summary

The following issues have been detected: timeout errors.

Analysis ?

Detected time complexity: **$O(N^2)$**

Example tests	
▶ example1	✓ OK
example test, positive answer	
Correctness tests	
▶ extreme_empty	✓ OK
empty sequence	
▶ extreme_single	✓ OK
sequence of one element	
▶ extreme_two_elems	✓ OK
sequence of three distinct elements	
▶ extreme_one_value	✓ OK
sequence of 10 equal elements	
▶ extreme_negative	✓ OK
sequence of negative elements, length=5	
▶ extreme_big_values	✓ OK
sequence with big values, length=5	
▶ medium1	✓ OK
chaotic sequence of value sfrom [0..1K], length=100	
▶ medium2	✓ OK
chaotic sequence of value sfrom [0..1K], length=200	

<div>▶ medium3</div> <div>chaotic sequence of values from [0..10], length=200</div>	✓ OK
collapse all Performance tests	
<div>▼ large1</div> <div>chaotic sequence of values from [0..100K], length=10K</div>	<div>✗ TIMEOUT ERROR</div> <div>running time: 0.172 sec., time limit: 0.100 sec.</div>
1. 0.172 s TIMEOUT ERROR, running time: 0.172 sec., time limit: 0.100 sec.	
<div>▼ large_random1</div> <div>chaotic sequence of values from [-1M..1M], length=100K</div>	<div>✗ TIMEOUT ERROR</div> <div>Killed. Hard limit reached: 6.000 sec.</div>
1. 6.000 s TIMEOUT ERROR, Killed. Hard limit reached: 6.000 sec.	
<div>▼ large_random2</div> <div>another chaotic sequence of values from [-1M..1M], length=100K</div>	<div>✗ TIMEOUT ERROR</div> <div>Killed. Hard limit reached: 6.000 sec.</div>
1. 6.000 s TIMEOUT ERROR, Killed. Hard limit reached: 6.000 sec.	

The PDF version of this report that may be downloaded on top of this site may contain sensitive data including personal information. For security purposes, we recommend you remove it from your system once reviewed.