Open Feedback Dialog

codility

Training center

eck out Codility training tasks

Congratulations

You have completed a Codility demo.

Tweet this!
I scored 100% in #java on @Codility!
https://codility.com/demo/take-sample-test/perm_missing_elem

Sign up for our newsletter!

Like us on Facebook!

Demo ticket

Session

ID: demoRYFQPX-8BP
Time limit: 120 min.

Status: closed

Created on: 2015-03-20 04:26 UTC
Started on: 2015-03-20 04:26 UTC
Finished on: 2015-03-20 04:39 UTC

Tasks in test	Correctness	Performance	Task score
1 PermMissingElem	100%	100%	100%

Test score

100%

100 out of 100 points

EASY

1. PermMissingElem

Find the missing element in a given permutation.

score: 100 of 100

Task description

A zero-indexed array A consisting of N different integers is given. The array contains integers in the range [1..(N + 1)], which means that exactly one element is missing.
Your goal is to find that missing element.
Write a function:

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

that, given a zero-indexed array A, returns the value of the missing element.
For example, given array A such that:

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

the function should return 4, as it is the missing element.
Assume that:

- N is an integer within the range [0..100,000];
- the elements of A are all distinct;
- each element of array A is an integer within the range [1..(N + 1)].

Complexity:

- expected worst-case time complexity is O(N);
- expected worst-case space complexity is O(1), beyond input storage (not counting the storage required for input arguments).

Elements of input arrays can be modified.

Solution

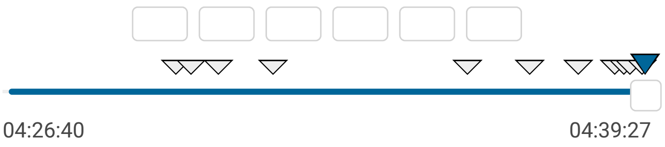
Programming language used: Java

Total time used: 13 minutes

Effective time used: 13 minutes

Notes: *not defined yet*

Task timeline



Code: 04:39:27 UTC, java, final, score: **100.00**
show code in pop-up

```
'/ you can also use imports, for example:
'/ import java.util.*;

'/ you can use System.out.println for debugging purposes
'/ System.out.println("this is a debug message");

:lass Solution {
    public int solution(int[] A) {
        // write your code in Java SE 8
        int actual = 0;
        int expected = 0;
        for(int i = 0; i < A.length; i++){
                actual += A[i];
                expected += i+1;
            }
            expected += A.length + 1;
            return expected - actual;

    }

```

Analysis

Detected time complexity:

O(N) or O(N * log(N))

test	time	result
Example tests		
example example test	1.052 s	**OK**
Correctness tests		
empty_and_single empty list and single element	1.048 s	**OK**
missing_first_or_last the first or the last element is missing	1.052 s	**OK**
single single element	1.048 s	**OK**
double two elements	1.048 s	**OK**
simple simple test	1.052 s	**OK**
Performance tests		
medium1 medium test, length = ~10,000	1.064 s	**OK**
medium2 medium test, length = ~10,000	1.072 s	**OK**
large_range range sequence, length = ~100,000	1.168 s	**OK**
large1 large test, length = ~100,000	1.300 s	**OK**
large2 large test, length = ~100,000	1.196 s	**OK**

Training center