



Congratulations

You have completed a Codility demo.

Tweet this!

I scored 100% in #ruby 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: demoGS7ZJR-2H9
Time limit: 120 min.

Status: closed

Created on: 2014-12-11 10:31 UTC
Started on: 2014-12-11 10:31 UTC
Finished on: 2014-12-11 10:43 UTC

Tasks in test

1 |  PermMissingElem

Correctness

100%

Performance

100%

Task score

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:

```
def solution(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.

Copyright 2009–2014 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

Solution

Programming language used: Ruby

Total time used: 12 minutes

Effective time used: 12 minutes

Notes: not defined yet

Task timeline



10:31:57

10:43:23

Code: 10:43:23 UTC, rb, final, score: 100.00

```
1 # you can use puts for debugging purposes, e.g.  
2 # puts "this is a debug message"  
3  
4 def solution(a)  
5   # write your code in Ruby 1.9.3  
6   h = {}  
7   a.each do |n|  
8     h[n] = true  
9   end  
10  (1..a.count+1).each do |n|  
11    return n unless h[n]  
12  end  
13 end
```

Detected time complexity:
O(N) or O(N * log(N))

test	time	result
Example tests		
example example test	0.060 s	OK
Correctness tests		
empty_and_single empty list and single element	0.056 s	OK
missing_first_or_last the first or the last element is missing	0.060 s	OK
single single element	0.060 s	OK
double two elements	0.056 s	OK
simple simple test	0.060 s	OK
Performance tests		
medium1 medium test, length = ~10,000	0.080 s	OK
medium2 medium test, length = ~10,000	0.080 s	OK
large_range range sequence, length = ~100,000	0.144 s	OK
large1 large test, length = ~100,000	0.268 s	OK
large2 large test, length = ~100,000	0.160 s	OK

Training center