codility

ning center

t Codility training tasks

Congratulations

You have completed a Codility training test.

Tweet this!
I scored 77% in #javascript on @Codility!
https://codility.com/demo/take-sample-test/missing_integer

Sign up for our newsletter!

Like us on Facebook!

Training ticket

Session

ID: training43N9ER-JYZ
Time limit: 120 min.

Status: closed

Created on: 2017-01-29 03:36 UTC
Started on: 2017-01-29 03:36 UTC
Finished on: 2017-01-29 04:27 UTC

Tasks in test	Correctness	Performance	Task score
1 MissingInteger Submitted in: JavaScript	60%	100%	77%

Test score

77%

77 out of 100 points

EASY

1. MissingInteger

Find the minimal positive integer not occurring in a given sequence.

score: 77 of 100

Task description

Write a function:

```
function solution(A);
```

that, given a non-empty zero-indexed array A of N integers, returns the minimal positive integer (greater than 0) that does not occur in A.

For example, given:

```
A[0] = 1
A[1] = 3
A[2] = 6
A[3] = 4
A[4] = 1
A[5] = 2
```

the function should return 5.

Assume that:

- N is an integer within the range [1..100,000];
- each element of array A is an integer within the range [−2,147,483,648..2,147,483,647].

Complexity:

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

Solution

Programming language used: JavaScript

Total time used: 51 minutes

Effective time used: 51 minutes

Notes: *not defined yet*

Task timeline

03:36:22 04:27:13

Code: 04:27:13 UTC, js, final, score: 77

show code in pop-up

```
1  /****
2
3  Write a function:
4
5  function solution(A);
6
7  that, given a non-empty zero-indexed array A of N inte
```

input arguments).

Elements of input arrays can be modified.


```
returns the minimal positive integer (greater than 0) t

For example, given:

  A[0] = 1
  A[1] = 3
  A[2] = 6
  A[3] = 4
  A[4] = 1
  A[5] = 2
the function should return 5.

Assume that:

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

expected worst-case time complexity is O(N);
expected worst-case space complexity is O(N), beyond i
Elements of input arrays can be modified.

Tests:

[-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5]  -> 6
[-5, -4, -3, -2, -1, 0, 1, 2, 4, 5]     -> 3
[-5, -4, -3, -2, -1, 0, 2, 3, 4, 5]     -> 1
[-5, -4, -3, -2, -1, 0, 1, 3, 4, 5]     -> 2
[-5, -4, -3, -2, -1, 1, 2, 3, 4, 5]     -> 6
[-5, -4, -3, -2, 0, 1, 2, 3, 4, 5]      -> 6
[-5, -4, -3, -1, 0, 1, 2, 3, 4, 5]      -> 6
[-1] -> 1
[-2] -> 1
[0] -> 1
[1] -> 2
[2] -> 1
[3] -> 2


****/


// you can write to stdout for debugging purposes, e.g
// console.log('this is a debug message');

function solution(A) {
    // write your code in JavaScript (Node.js 6.4.0)

    // If there is only one element, the min. missing
    if (A.length === 1) {
        if (A[0] < 1 ) return 1;
        if (A[0] === 1) return 2;
        return A[0] - 1;
    }

    // Sort the array
    A.sort(function(a, b){return a-b});

    var num1 = 0;
    var num2 = 0;

    // Find the first missing number starting with the
    for (i=0; i<A.length-1; i++) {

        // Skip all numbers that are < 1.
        if (A[i] < 0) continue;

        // We are either >= 1 at this point
        if (A[i] === 1 && i === A.length-1) return 2;

        num1 = A[i];
        num2 = A[i+1];

//        console.log("num1: ", num1, "num2: ", num2);

        if (num2 - num1 > 1) {
            // We found the missing number
            return num1 + 1;
        }
    }
```

```
88
89      return num2 + 1;
90
91  }
```

Analysis summary

The following issues have been detected: wrong answers.

For example, for the input `[4, 5, 6, 2]` the solution returned a wrong answer (got 3 expected 1).

Analysis

expand all	Example tests	
▶	example example (without minus)	✔ OK
expand all	**Correctness tests**	
▶	extreme_single a single element	✔ OK
▼	simple simple test	✘ WRONG ANSWER got 3 expected 1
	1. 0.142 s **WRONG ANSWER**, got 3 expected 1	
	2. 0.137 s **OK**	
	3. 0.139 s **WRONG ANSWER**, got 94 expected 1	
▶	extreme_min_max_int MININT and MAXINT (with minus)	✔ OK
▼	positive_only shuffled sequence of 0...100 and then 102...200	✘ WRONG ANSWER got 150 expected 1
	1. 0.148 s **OK**	
	2. 0.141 s **WRONG ANSWER**, got 150 expected 1	
▶	negative_only shuffled sequence -100 ... -1	✔ OK
expand all	**Performance tests**	
▶	medium chaotic sequences length=10005 (with minus)	✔ OK
▶	large_1 chaotic + sequence 1, 2, ..., 40000 (without minus)	✔ OK
▶	large_2 shuffled sequence 1, 2, ..., 100000 (without minus)	✔ OK
▶	large_3 chaotic + many -1, 1, 2, 3 (with minus)	✔ OK