

Demo ticket

Session

ID: demoY4ZNVJ-K2H
Time limit: 120 min.

Status: closed

Created on: 2014-03-16 22:14 UTC
Started on: 2014-03-16 22:14 UTC
Finished on: 2014-03-16 22:20 UTC

Tasks in test

Task score

Test score

100%

100 out of 100 points

EASY

1. PermCheck

Check whether array A is a permutation.

score: 100 of 100



Task description

A non-empty zero-indexed array A consisting of N integers is given. A *permutation* is a sequence containing each element from 1 to N once, and only once.

For example, array A such that:

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

is a permutation, but array A such that:

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

is not a permutation.

The goal is to check whether array A is a permutation. Write a function:

```
int solution(vector<int> &A);
```

that, given a zero-indexed array A, returns 1 if array A is a permutation and 0 if it is not.

For example, given array A such that:

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

the function should return 1.

Given array A such that:

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

the function should return 0.

Assume that:

Solution

Programming language used: C++

Total time used: 7 minutes

Effective time used: 7 minutes

Notes: correct functionality and scalability

Task timeline



22:14:16

22:20:21

Code: 22:20:21 UTC, cpp, final, score: 100.00

```
01. // you can also use includes, for example:
02. // #include <algorithm>
03. int solution(vector<int> &A) {
04.     vector<long long> C(A.size(), 0);
05.
06.     for (int i = 0; i < (int)A.size(); i++) {
07.         if (A[i] > (int)A.size() || A[i] < 1)
08.             return 0;
09.         else {
10.             if (C[A[i]-1] != 0)
11.                 return 0;
12.             else
13.                 C[A[i]-1] += 1;
14.         }
15.     }
16.     return 1;
17. }
```

Analysis

Assume that:

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

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 input arguments).

Elements of input arrays can be modified.

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

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

test	time	result
example1 the first example test	0.020 s.	OK
example2 the second example test	0.020 s.	OK
extreme_max single element with maximal value	0.020 s.	OK
single single element	0.020 s.	OK
double two elements	0.020 s.	OK
antiSum1 total sum is corret (equals $1 + 2 + \dots N$), but it is not a permutation, $N = 3$	0.020 s.	OK
medium_permutation permutation, $N = \sim 10,000$	0.020 s.	OK
antiSum2 total sum is corret (equals $1 + 2 + \dots N$), but it is not a permutation, $N = \sim 100,000$	0.040 s.	OK
large_permutation large permutation, $N = \sim 100,000$	0.040 s.	OK
large_range sequence 1, 2, ..., N, $N = \sim 100,000$	0.040 s.	OK
extreme_values all the same values, $N = \sim 100,000$	0.030 s.	OK

Training center