codility

Training center
Check out Codility training tasks

Candidate Report: Anonymous

Test Name:

SUMMARY TIMELINE

Test Score

100 out of 100 points

100%

Tasks in Test

	Time Spent	Task Score
PermMissingElem Submitted in: C++	7 min	100%

TASKS DETAILS

1. PermMissingElem
EASY
Find the missing element in a given permutation.

Task Score	Correctness	Performance
100%	100%	100%

Task description

An 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:

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

that, given an array A, returns the value of the missing element.

For example, given array A such that:

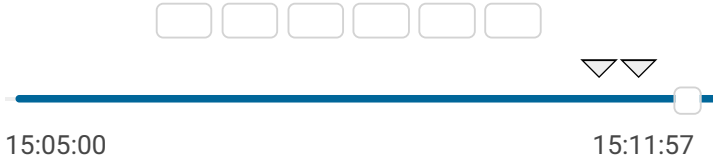
```
A[0] = 2
A[1] = 3
A[2] = 1
A[3] = 5
```

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

Solution

Programming language used:	C++
Total time used:	7 minutes
Effective time used:	7 minutes
Notes:	*not defined yet*

Task timeline



Code: 15:11:57 UTC, show code in pop-up

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) (not counting the storage required for input arguments).

cpp, final, score: **100**

```cpp
// you can use includes, for example:
// #include <algorithm>

// you can write to stdout for debugging purpc
// cout << "this is a debug message" << endl;

int solution(vector<int> &A) {
    // write your code in C++14 (g++ 6.2.0)
    int sum = ((1+ A.size()+1)*(A.size()+1)/2)

    for(int i = 0; i < A.size(); i++) {
        sum-=A[i];
    }
    return sum;
}
```

Analysis summary

The solution obtained perfect score.

Analysis

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

expand all	**Example tests**	
▶	example example test	✔ OK
expand all	**Correctness tests**	
▶	empty_and_single empty list and single element	✔ OK
▶	missing_first_or_last the first or the last element is missing	✔ OK
▶	single single element	✔ OK
▶	double two elements	✔ OK
▶	simple simple test	✔ OK
expand all	**Performance tests**	
▶	medium1 medium test, length = ~10,000	✔ OK
▶	medium2 medium test, length = ~10,000	✔ OK
▶	large_range range sequence, length = ~100,000	✔ OK
▶	large1 large test, length = ~100,000	✔ OK

▶	large2 large test, length = ~100,000	✔ OK