

# Codility

## CodeCheck Report: trainingUX3Z24-5GX

Test Name:

[Check out Codility training tasks](#)

Summary

Timeline

### Tasks summary

| Task                      | Time spent | Score |
|---------------------------|------------|-------|
| PermMissingElem<br>Python | 16 min     | 100%  |

### Total score



### Tasks Details

|      |  |            |             |             |
|------|--|------------|-------------|-------------|
| Easy | 1.   |            |             |             |
|      | <b>PermMissingElem</b>                           | Task Score | Correctness | Performance |
|      | Find the missing element in a given permutation. | 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:

```
def solution(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

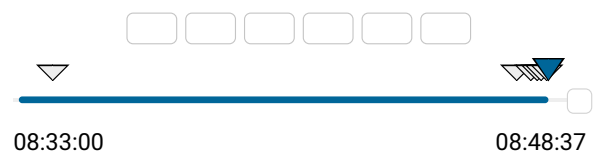
Programming language used: Python

Total time used: 16 minutes ?

Effective time used: 16 minutes ?

Notes: not defined yet

### Task timeline



Code: 08:48:36 UTC, py,

[show code in pop-up](#)

Write an **efficient** algorithm for the following assumptions:

- 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)].

Copyright 2009–2023 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

final, score: 100

```
1 # you can write to stdout for debugging purposes
2 # print("this is a debug message")
3
4 def solution(A):
5     # Implement your solution here
6     # pass
7     expected_sum = (len(A) + 1) * (len(A) + 1) // 2
8
9     actual_sum = sum(A)
10
11     missing_element = expected_sum - actual_sum
12
13     return missing_element
```

### Analysis summary

The solution obtained perfect score.

### Analysis

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

| expand all | Example tests   |      |
|------------|---|------|
| ▶          | example<br>example test   | ✓ OK |
| expand all | Correctness tests   |      |
| ▶          | empty_and_single<br>empty list and single element                 | ✓ OK |
| ▶          | missing_first_or_last<br>the first or the last element is missing | ✓ OK |
| ▶          | single<br>single element  | ✓ OK |
| ▶          | double<br>two elements  | ✓ OK |
| ▶          | simple<br>simple test   | ✓ OK |
| expand all | Performance tests   |      |
| ▶          | medium1<br>medium test, length = ~10,000                          | ✓ OK |
| ▶          | medium2<br>medium test, length = ~10,000                          | ✓ OK |
| ▶          | large_range<br>range sequence, length = ~100,000                  | ✓ OK |
| ▶          | large1<br>large test, length = ~100,000                           | ✓ OK |
| ▶          | large2<br>large test, length = ~100,000                           | ✓ OK |