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.

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)].

def solution(A):

A\_len = len(A)

arr = [0] \* (A\_len+1)

for el in A:

arr[el-1] = 1 # array 0 based

for i, item in enumerate(arr):

if item == 0:

return i+1 # bring back the value

return -1

deger = [2, 3, 1, 5]

car\_1 = solution(deger)

print(car\_1)

or

def solution(A):

N = len(A)

return (((N + 1) \* ((N + 1) + 1) // 2) - sum(A))

or

XOR GATE Solution

def solution(A):

missing\_element = len(A)+1

for idx,value in enumerate(A):

missing\_element = missing\_element ^ value ^ (idx+1)

return missing\_element