Find the Zero in Sorted Array

You are given an array of size $N+1$. The array contains **all** the integers from 0 to N (both inclusive). The array is completely sorted in ascending order with one exception - the number zero (0) is placed at an arbitrary position. You are expected to find the position of zero in this sorted array.

Note: There is one and only one occurrence of zero in this sorted array, and the array is **0-indexed**.

Refer to these example test cases for further explanation:

TEST CASE 1
$N = 6$
Input Array: $[1, 2, 0, 3, 4, 5, 6]$
Output: 2
Explanation: The position of zero in the array is 2

TEST CASE 2
$N = 3$
Input Array: $[1, 2, 3, 0]$
Output: 3
Explanation: The position of zero in the array is 3

Expected Time Complexity: $O(\lg(N+1))$, where $N+1$ is the size of the input array.

Input Format:
The first line of input file contains S - the number of sub-test-cases.
In each sub-test-case, the first line contains N - the size of the array, and the next line contains $N+1$ space-separated integers.

(After reading this question, follow the steps described in `Instructions.pdf` to see the input and the expected output for each test case.)

Instructions:
You have been provided with a template file containing 2 functions - *main*

and *solve*. The *main* function already takes the input S, which is the number of sub test cases. For each sub test case, you have to take the array as input and call the *solve* function. The *solve* function should find the position of zero and print it on the terminal in $O(\lg(N+1))$ time, where $N+1$ is the size of the input array.

Follow the steps described in `Instructions.pdf` to see the input and the expected output for each test case.