Description | Solution | Discuss (362) | Submissions

1060. Missing Element in Sorted Array

Medium 1109 45 Add to List Share

Given an integer array `nums` which is sorted in **ascending order** and all of its elements are **unique** and given also an integer `k`, return the `k^th` missing number starting from the leftmost number of the array.

Example 1:

```
Input: nums = [4,7,9,10], k = 1
Output: 5
Explanation: The first missing number is 5.
```

Example 2:

```
Input: nums = [4,7,9,10], k = 3
Output: 8
Explanation: The missing numbers are [5,6,8,...], hence the third missing number is 8.
```

Example 3:

```
Input: nums = [1,2,4], k = 3
Output: 6
Explanation: The missing numbers are [3,5,6,7,...], hence the third missing number is 6.
```

Constraints:

- `1 <= nums.length <= 5 * 10^4`
- `1 <= nums[i] <= 10^7`
- `nums` is sorted in **ascending order,** and all the elements are **unique**.
- `1 <= k <= 10^8`

Follow up: Can you find a logarithmic time complexity (i.e., `O(log(n))`) solution?

Accepted 76,958 | Submissions 139,021

Seen this question in a real interview before? Yes No

Companies

0 ~ 6 months 6 months ~ 1 year 1 year ~ 2 years

Facebook | 7 eBay | 2

Related Topics

Array Binary Search

Hide Hint 1

First define a function f(x) that counts the number of missing elements until x.

Hide Hint 2

Then use binary search with the given function f(x) to find the kth missing element.

Java | Autocomplete

```java
class Solution {
  int missing(int idx, int[] nums) {
    return nums[idx] - nums[0] - idx;
  }
  public int missingElement(int[] nums, int k) {
    int n = nums.length;
    if (k > missing(n - 1, nums))
      return nums[n - 1] + k - missing(n - 1, nums);
    int left = 0, right = n - 1, pivot;
    while (left != right) {
      pivot = left + (right - left) / 2;

      if (missing(pivot, nums) < k) left = pivot + 1;
      else right = pivot;
    }
    return nums[left - 1] + k - missing(left - 1, nums);
  }
}

class Solution {
  int missing(int idx, int[] nums) {
    return nums[idx] - nums[0] - idx;
  }

  public int missingElement(int[] nums, int k) {
    int n = nums.length;
    if (k > missing(n - 1, nums))
      return nums[n - 1] + k - missing(n - 1, nums);

    int idx = 1;
    while (missing(idx, nums) < k) idx++;
    return nums[idx - 1] + k - missing(idx - 1, nums);
  }
}
```