Problem 1060: Missing Element in Sorted Array

Problem Information

Difficulty: Medium
Acceptance Rate: 59.56%
Paid Only: Yes
Tags: Array, Binary Search

Problem Description

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 `kth` 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 * 104` * `1 <= nums[i] <= 107` * `nums` is sorted in **ascending
order,** and all the elements are **unique**. * `1 <= k <= 108`

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

Code Snippets

C++:

```cpp
class Solution {
public:
int missingElement(vector<int>& nums, int k) {


}
};
```

Java:

```java
class Solution {
public int missingElement(int[] nums, int k) {


}
}
```

Python3:

```python
class Solution:
def missingElement(self, nums: List[int], k: int) -> int:
```