26. Remove Duplicates from Sorted Array

Easy | Topics | Companies | Hint

Given an integer array `nums` sorted in **non-decreasing order**, remove the duplicates **in-place** such that each unique element appears only **once**. The **relative order** of the elements should be kept the **same**.

Consider the number of *unique elements* in `nums` to be `k`. After removing duplicates, return the number of unique elements `k`.

The first `k` elements of `nums` should contain the unique numbers in **sorted order**. The remaining elements beyond index `k - 1` can be ignored.

Custom Judge:

The judge will test your solution with the following code:

```
int[] nums = [...]; // Input array
int[] expectedNums = [...]; // The expected answer with correct length

int k = removeDuplicates(nums); // Calls your implementation

assert k == expectedNums.length;
for (int i = 0; i < k; i++) {
    assert nums[i] == expectedNums[i];
}
```

```
int removeDuplicates(int* nums, int numsSize) {
    if (numsSize == 0)
        return 0;

    int k = 1;

    for (int i = 1; i < numsSize; i++) {
        if (nums[i] != nums[i - 1]) {
            nums[k] = nums[i];
            k++;
        }
    }
    return k;
}
```

Output:

Input

nums =
[1,1,2]

Output

[1,2]

Expected

[1,2]