43. Remove Duplicates from Sorted Array

Given an integer array nums sorted in non-decreasing order, remove the duplicates inplace such that each unique element appears only once. The relative order of the elements

should be kept the same.

Since it is impossible to change the length of the array in some languages, you must instead

have the result be placed in the first part of the array nums. More formally, if there are k

elements after removing the duplicates, then the first k elements of nums should hold the

final result. It does not matter what you leave beyond the first k elements.

Return k after placing the final result in the first k slots of nums.

Do not allocate extra space for another array. You must do this by modifying the input

array in-place with O(1) extra memory.

Code:

```python
def removeDuplicates(nums):
    if not nums:
        return 0

    k = 1
    for i in range(1, len(nums)):
        if nums[i] != nums[i - 1]:
            nums[k] = nums[i]
            k += 1
    return k
nums1 = [1, 1, 2]
k1 = removeDuplicates(nums1)
print(k1, nums1[:k1])
nums2 = [0, 0, 1, 1, 1, 2, 2, 3, 3, 4]
k2 = removeDuplicates(nums2)
print(k2, nums2[:k2])
```

Output:

```
2 [1, 2]
5 [0, 1, 2, 3, 4]
```

Time Complexity:

- T(n)= O(n)