Leet Code #26

↳ Day - 7

Removing Duplicates from a Sorted Array

Given an integer array 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 same. we then have to return the number of unique elements.

Approach

① We can use two variable and a for loop to remove duplicates from the array using in-place algorithm. In-place algorithm modifies the original data structure

without creating a new one.

② int j = 1;
int i = 1;

for(i; i<n; i++){
if (curr != prev)
prev = curr;
j++;
return j

We are starting from 1 instead of 0 because we need to check if the previous element is equal to the current element. If it is, then keep iterating otherwise overwrite the value starting from index '1' and increment j to come over to the next element in the array.

Example Illustration

[1, 1, 2] arr.size() => 3

i=1, j=1

for (i, i<3, i++)

↳ check arr[i] (1) != arr[i-1] (1)

↳ False. Do nothing.

Now i becomes '2'.

check arr[i] != arr[i-1]

(2) (1)

↳ TRUE so replace

arr[j] = arr[i]

[1, 1, 2] → [1, 2, 2]

j++;

This will keep a TRACK of no. of unique element.

After Iterating (End of for loop)

We Return (j)