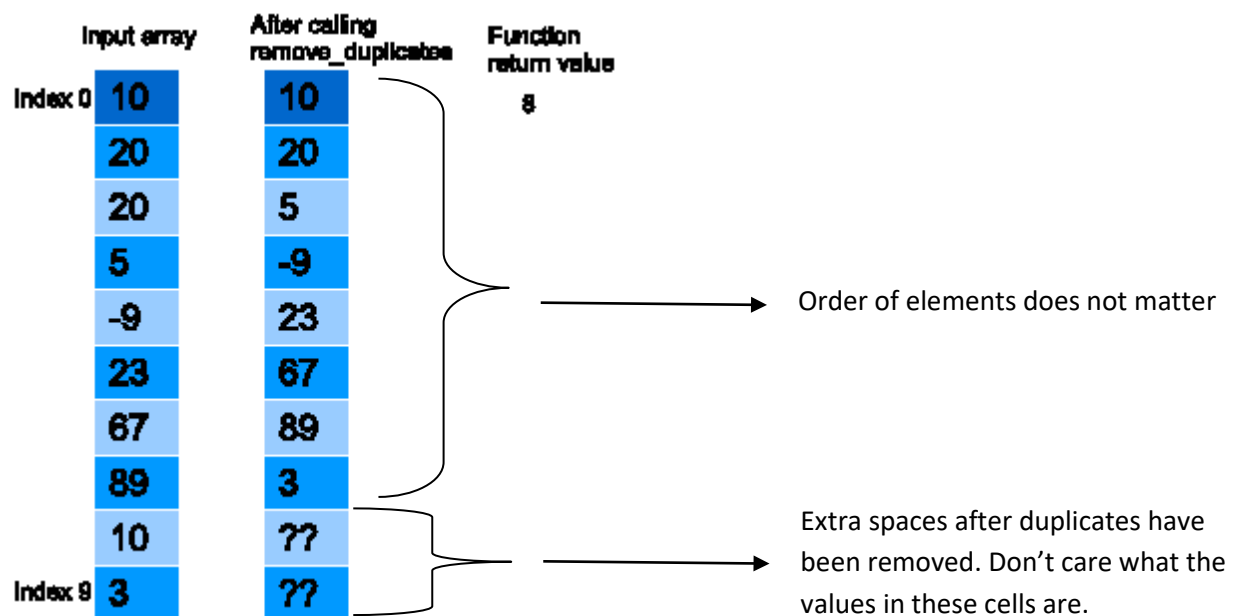


Submission Instructions:

Create a folder called <LastnameFirstname> (E.g. SawyerTom). Name the file for problem 1 as 01.cpp, that of problem 2 as 02.cpp, etc. Place them in the folder <LastnameFirstname>. Compress (zip) the folder to get a file called <LastnameFirstname>.zip. Upload this file to **blackboard** by the due date.

Problem 1 (2 points)

Write a function called `remove_duplicates` that takes an array of integers and the size of the array. The function removes all duplicate entries in the array, re-orders the array and returns the used number of spaces in the resulting array. See figure below for what is expected. Note that the function should not create another array or use any temporary array for storage. Demonstrate using a main function that it works.



Problem 2 (3 points)

Use dynamic arrays to solve this problem.

Consider an array 'a' that was created using dynamic memory allocation. Initialize the array with some values. Write a function called "insert_after" that takes the array, its current size, its capacity, a number in the array after which to insert, and the actual number to insert:

```
int* insert_after(int* a, int csz, int cap, int after, int val);
```

Assume that "after" exists in the array.

The challenge here is that the array may not have the necessary capacity to insert val. If the array is full, copy the contents of the input array into a bigger array that is dynamically created in the function. In this scenario, the function will return the address of the new array after inserting. If the array has the necessary capacity, simply insert the number and return NULL to indicate that there was no need to create a new array.

Because you will need dynamic arrays, ensure that there is no memory leak – i.e., your program deallocates memory properly.

Problem 3 (5 points)

Design and implement an ADT called Car. Support the following operations: ability to set make and model, ability to shift gear up or down (assume a max of 6-speed), ability to shift lanes (left or right; assume a max of three lanes), accelerate and decelerate (in units of 5 mph), and display the current state of the car (show the make, model, current gear number, current lane and speed). Instantiate some Car objects in your main function and take them for a spin.