**Final Exam – final.cpp**

This**two hour exam** consists of a single programming problem, focusing on material from the second half of the course. This code will be written in a "final.cpp" file provided to you. Comments will not be required except for a header, but may be included to increase clarity.

**You will be required to be on Zoom with your video turned on and your microphone muted.** Private communication with the instructor is allowed, but you will not be allowed to chat with other students during the exam.

**This is an open book exam**. All digital resources provided by the class (slides) as well as the textbook and class notes are permitted on the exam. However, **other outside materials, including cell phones and other computers, will not be permitted.** This also includes communicating with other students during the exam period. ***Students who violate these rules will be given a 0 for the exam.***

When you are completed with the exam, please submit the "final.cpp" file with your code onto the autograder. You will be given an extra 10 minutes after the exam to submit, so do not worry about rushing to submit on time. If there are any issues with submitting to the autograder, please send me an email with your code.

The assignments will be graded with both a manual and a automatic portion with some test runs. As a result, your final grade will not be available right when you finish.

**Assignment Instructions**

For this final, you will be writing a program that populates an array with numbers and allows the user to run some actions on the array. Your array will be initialized in your main function to a constant global value, MAX\_SIZE, which should be set to 20.

To perform the subtasks, you will be writing three functions:

* **fillArray**: This function takes in an array of integers and a reference to an integer representing the number of filled elements. In this function, the user is prompred to enter a positive number that is no greater than 20, and **this input will loop until the user enters a legal value**. This value will be filled inside of the reference parameter mentioned above. From here, the array will be filled up to "value" elements with **randomly generated numbers** between 1 and 100. **Make sure to set the random seed to 20 at the start of your main function for consistent results**.
* **displayArray**: This function takes in an array of integers and an integer representing the number of filled elements. Using a loop, all elements in the array are printed out to the user.
* **replaceNum**: This function takes in an array of integers, an integer representing the number of filled elements, an integer representing a number to replace, and one more integer that will replace the previous parameter. This function will search for the the number to replace in the array. If that number is found, then the value in the array is changed to the replacement number and the function returns true. If the number is not found, the function returns false.

Your main function should first fill the array, then it should provide a looping menu interface that lets the user choose between the choices as follows:

**(D)isplay, (R)eplace, Re(F)ill, or (Q)uit?**

Depending on the user's input, do the following actions:

* D: call displayArray
* R: Ask the user to "Enter a number to search for: " and to "Enter replacement: ". Call the replaceNum function, and if it returns true, report "Replacement sucessful." Otherwise, report "Replacement unsuccessful."
* F: Call fillArray.
* Q: Wish the user a good bye. If the user enters this, end the loop.
* Any other input: Report "Invalid answer - input again!"

Here is one sample run of the program, with user input bolded:

Enter a positive number no larger than 20: **3**  
(D)isplay, (R)eplace, Re(F)ill, or (Q)uit? **d**  
12  
59  
98  
(D)isplay, (R)eplace, Re(F)ill, or (Q)uit? **r**  
Enter a number to search for: **59**  
Enter replacement: **40**  
Replacement successful.  
(D)isplay, (R)eplace, Re(F)ill, or (Q)uit? **D**  
12  
40  
98  
(D)isplay, (R)eplace, Re(F)ill, or (Q)uit? **f**  
Enter a positive number no larger than 20: **5**  
(D)isplay, (R)eplace, Re(F)ill, or (Q)uit? **D**  
60  
97  
82  
63  
42  
(D)isplay, (R)eplace, Re(F)ill, or (Q)uit? **Q**  
Good bye!

Some notes:

* Make sure to account for both capital and lowercase input in your menu.
* You may write functions not mentioned here in your program, but no extra ones are required.