COMP-1400 Fall 2021
Assignment #3
(Due End of Wednesday, Dec. 1, 2021)

OBJECTIVE:

Through three assignments, we will learn how to apply the concepts learned in this course to develop a simple command-line calculator using programming language C.

In the third assignment, we are going to complete the calculator program we developed in the first two assignments. This version will cover **array variables** and **user-defined functions**.

Part A:

Add a new functionality to this calculator as follows:

1- The retult of each successful calculation in each iteration must be stored in the memory, in a way that is retrivable in the future. For example, assume the user runs the program and calculates the following:

> [1] 1 + 2.5 = 3.5
>
> [2] 5*6 = 30
>
> [3] max(8, 98) = 98
>
> [4] square root of 9 = 3
>
> [5] 3 ^ 2 = 9

You must store the results of these calculations, i.e. 3.5, 30, 98, 3, and 9 in an array.

2- A new item is added to the list of options, "select M to see the memory". If the user selects M, the program will show the number of items which have been stored in the memory. If the number of items is zero, a message will be prompted and the program will show the main list of available options. Otherwise the user can choose any item in a range of 1 to the number of items, and see its corresponded value. The user also can choose to see the whole list of values by entering zero. This process continues unless the user choses to return to the main list of options by entering X.

For example, following the above sample calculations, when the user selects M, the program will show that there are 5 items available in the memory. Then if the user enters 1, the program will show 3.5 as the output, and asks the user if s/he wants to see another item in the memory, the whole list, or return to the main menu.

3- Add another option to your calculator, R, such that by entering this character by the user, the momory will be erased.

Part B:

4- Write a function to show the list of available options (B,U,V,A, M, R, X). Then whenever you need it in the main program, call that function. This function should return the user selection as a single uppercase character, B, U, V, A, M, R, X. If user enters an invalid character, the function needs to show a proper message and gets another input. This means, the function won't exit unless a valid character is selected.

5- Convert cases B, U, V, A, M, R, X to separate functions and call them in the main program, whenever is needed. For example, for case B, you need to write a function that receives three parameters, two float numbers and one operator (character), and returns the result of such calculation in a float format. The function also will check the parameters and show

proper messages if any of them are invalid, as described in assignment 2. Then whenever, the user selects B in the main program, the inputs will receive from the user and then this function will be called and processed.

Save the file in the format of a C source file as **clc_v3.c** in your working directory, i.e., **/home/yourUserName/F2021/comp1400**.

As you are already in the working directory as completing UNIX Exercise, you can now compile the C program with the following command to produce an executable.

> **gcc -Wall clc_v3.c -o clc.out**

You can run the executable file with the following command.

> **./clc.out**

Submission Instruction:

You have to submit your solution as **A SINGLE C FILE** into Blackboard before the deadline.

PLEASE DO NOT submit any other types of files, such as compiled files, files associated with specific software, etc. Therefore, for this assignment you just need to submit the following file:

- One text file, named "clc_v3.c", which should be a simple text file, containing your c program.

Submission of other file types will make problems for grading your assignments, and you will lose a partial or complete mark.

Good Luck 😊