# ***LAB TEST 01 – Building and Debugging a C Program***

## **PURPOSE:**

The purpose of this test is to use Visual Studio environment to build a C program with function pointers and examine it under the debugger. Various aspects of the debugger functionality will be used and learned.

## **TOOLS AND COMPONENTS:**

Visual Studio 2022

Sample Bubble Sort program with function pointers.

## **GENERAL KNOWLEDGE:**

Complex programs that use data and function pointers are best examined by using a debugger. The sample Bubble Sort program makes use of function pointers and hence it is a good candidate to be examined under a debugger. The free Community version of Visual Studio 2022 will be used to build it for debugging and the integrated debugger will be used to debug it.

The sample program takes a hard coded array of integers and provides ascending/descending sorting of that array depending upon the input (1=ascending, 2=descending). Both the original and the sorted arrays are printed out.

In order to perform sorting a comparison function is needed to compare the neighboring elements of the array. For this sample, there are two comparison functions, one for ascending and another one for descending sorting. A function pointer is used to select the right one depending upon the user input.

## **PRELIMINARY WORK:**

Make sure Visual Studio 2022 is installed on the test computer. Unzip the test program zip file in a suitable directory. Double-click the solution file in the directory to start Visual Studio with the sample project.

Build the project with the “Debug” option and check the executable file in the Debug directory. Then, the program is ready for debugging.

## **PERFORMING THE EXPERIMENT:**

1. Press the F10 key to start the debugging of the program. Observe that an Output Window is created, the program file is opened in the IDE and the yellow arrow on the left points to the start of the main function.
2. Keep pressing the F10 key to step through the program and observe the contents of the Output Window.
3. When prompted for entering a selection for sorting in the Output Window, enter 1 or 2, go back to the IDE and keep pressing F10 key to continue program execution. Observe that the F10 key does not go into functions.
4. Press F5 key to run the program to the end.
5. Repeat steps 1,2 and 3 above. In the IDE, click on the line where “bubble” is called in the main function and press F9 key to insert a (red) breakpoint. (There are two lines!)
6. Press F5 key to run to the breakpoint.
7. Press F11 to go into the “bubble” function. Press F10 a few times while in that function. Then, press SHIFT+F11 to step out of the “bubble” function and to go back to the next line in the main function.
8. Use F9 and F5 keys to place breakpoints at other lines of the program and to go to those lines quickly.
9. Open up a “Watch” window using the “Debug->Windows->Watch” menu from the top.
10. Enter the variable name “a” into the Name field in the Watch window to observe the array.
11. Observe the (start) address of the array and its contents. Repeat this before and after sorting. (Restart the program if necessary and put breakpoints to appropriate lines to make the required observations.)
12. Expend the array a in the watch window and change one of its entries. Run the program to the end and see how that change is reflected in the output. (The address of the array may not be modified as it is assigned by the compiler at the compilation time. However, if a simple pointer is declared rather than an array, it may be changed.)
13. Right click on of the breakpoints and observe that it may be disabled/enabled from the dropdown menu. Also, observe that a condition may be attached to the breakpoint. (Hence, the program execution will stop at the breakpoint if a certain condition is satisfied.)
14. Observe other menu items under the Debug in the at the top.
15. Make a list of all functions keys you have learned.