

**Discipline of Information Technology,**

**Mathematics and Statistics**

**College of Science, Health, Engineering and Education**

**ICT374 LAB ASSIGNMENT DECLARATION**

**Surname:** Khan **Given Names:** Mohsin Ali

**Due Date: 23/09/2023** **Date Submitted: 29/09/2023**

**Your assignment should meet the following requirements. Please confirm this (by ticking boxes) before submitting your assignment.**

* I have read and understood the Requirements for Documentation and Submission of Lab Assignments (the requirements)
* **This submission is compliant to the above requirements**
* I have included all relevant Linux source code, executables and test files in the tar archive. The file names are chosen according to the above requirements.
* I have kept another copy of this submission and associated programs and files in a safe place.

**Please make your declaration for each question or exercise by writing YES in the last column if the question or exercise is fully completed and all relevant files for the question are included in this submission. Otherwise write No.**

**A question or exercise is fully completed when all requirements are met (eg, the program compiles and produces results). A fully completed question or exercise is not always 100% correct.**

**Batch Number: ICT374 A**

|  |  |  |
| --- | --- | --- |
| **Lab Number** | **Question/Exercise Number** | **Is this question or exercise fully completed?** |
| 1 | Theory Exercise 2 | Yes |
| 1 | C Programming 2 | Yes |
| 1 | C Programming 3 | Yes |
|  |  |  |
|  |  |  |

## Theory Exercise 2

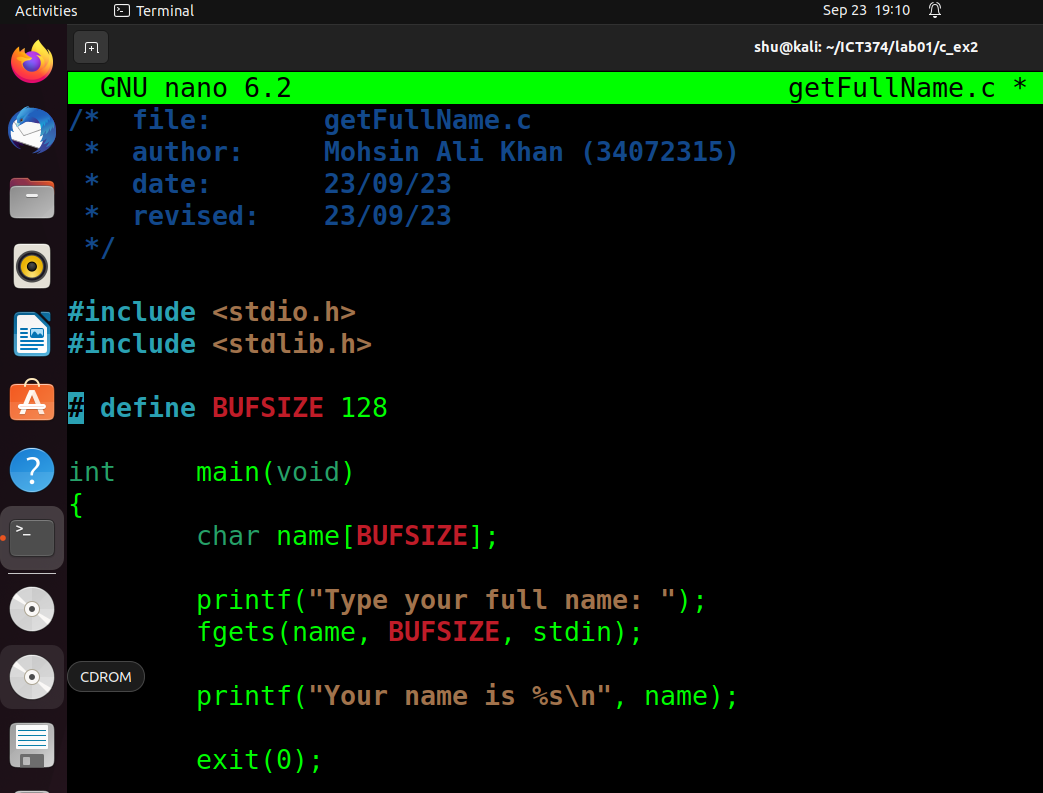
The operating system on your computer system shares the processor (assuming there is only one processor) with many other programs. However, we understand that at any given moment, the processor can only execute the instructions from one program. Explain how the operating system and several other programs can run on the same processor at the same time.

Answer: The operating system uses a technique called preemptive multitasking. This allows programs to run concurrently on a single processor.

The processor executes instructions from one program for about 10-100 milliseconds. After that, the operating system suspends the execution of the current program and saves its state. The operating system then loads the another program that is waiting to run and executes it for 10-100 milliseconds. This continues with the processor switching between programs frequently enough that all programs seems like they are executing simultaneously.

## C Programming Exercise 2

Code for getFullName.c program.



Compiling the program with gcc and -Wall -Wextra -Werror flags. -Wall checks for all compiler error messages, -Wextra checks for extra error messages and -Werror treats all warnings as errors and fails compilation. The flag -o is for output and it allows me to choose the name I want to give to me executable file after compiling. To run the program from your terminal just type “./getFullName” and the program will run.

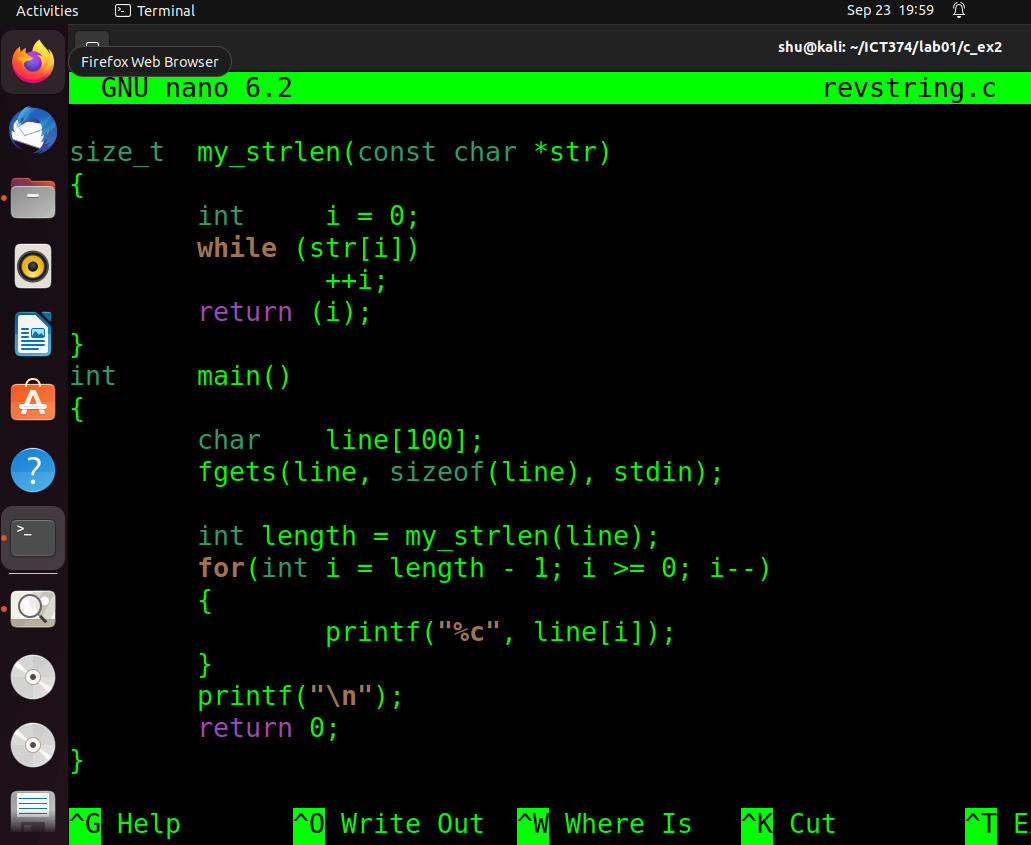
## 

The program asks for user to enter the full name from the terminal directly. After I have entered my full name the program prints out my full name using printf followed by a newline.

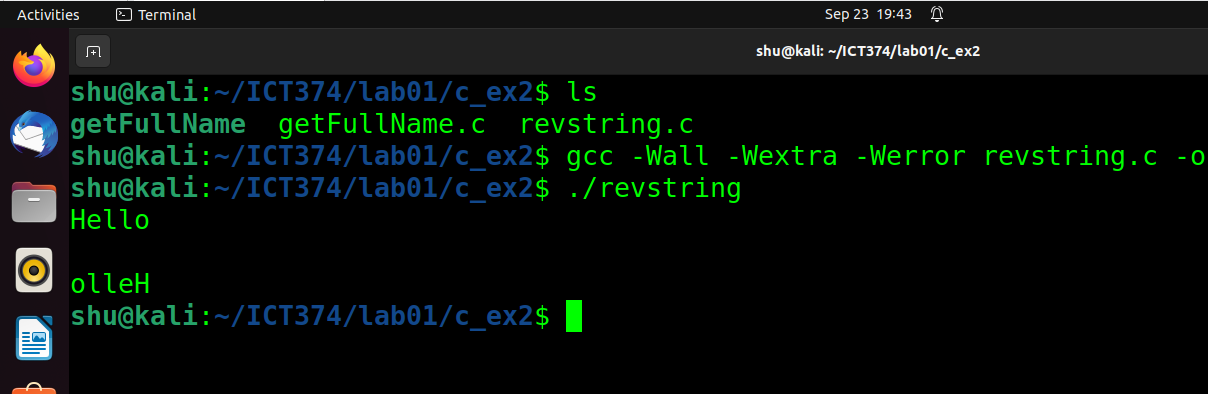
## 

## C Programming Exercise 3

Code for revstring.c program. I added my own simple implementation for the strlen function.



Compiling the program with gcc and -Wall -Wextra -Werror flags. -Wall checks for all compiler error messages, -Wextra checks for extra error messages and -Werror treats all warnings as errors and fails compilation. The flag -o is for output and it allows me to choose the name I want to give to me executable file after compiling. To run the program from your terminal just type “./revstring” and the program will run.



After you run the program, you can enter a string that will be reverse. For example, Hello is reversed to olleH

