**Matric Number**: 222515

**Name:** Sunday, Isaac Andikan

**Course Code:** CSC235

**A BRIEF INTRODUCTION TO THE UNIX OPERATING SYSTEM and the LINUX KERNEL**

Developed in the year 1969, the UNiplexed Information Computing System (UNIX) operating system has come into wide use. In an effort to establish a time-sharing computer system, AT&T Corporation's Bell Laboratories created UNIX in the late 1960s, which has become the operating system of choice for engineering and scientific workstations for reasons such as being lightweight, secure, open-source etc.

***LINUX*** - Like other operating systems like Microsoft Windows, Apple Mac OS, iOS, Google Android, etc., Linux is an open-source operating system. An operating system is a piece of software that makes it possible for computer hardware and applications to communicate. It transports data for the processor to process, and it delivers output to the hardware for display.

Linux is an operating system that resembles Unix but is not Unix. The Linux operating system is a continuation of the Unix design principles.

**SOFTWARE FUNCTIONAL REQUIREMENTS**

An explanation of the service that the software must provide is contained in a functional requirement (FR). It describes a piece of software or a software system. A function is nothing more than the inputs, behavior, and outputs of the software system. A system's likely function can be determined by a computation, data manipulation, business process, user interaction, or any other specialized feature.

**WHY IS UNIX PREFERRED AT SOME POINTS**

The UNIX operating system has readily available tools and Application Programming Interfaces, that aid software development.

**WHY DOES UNIX BEING REFERRED TO AS A SCIENTIST OS**

Unix is referred to as a scientist OS, mainly because of the readily available tools that best suit scientific practices.

**WHAT TYPE OF PROGRAMMING LANGUAGE IS C?**

C is an imperative procedural language

**STRUCTURE OF A C PROGRAM**

The basic structure of a C programming language is divides into 5 parts:

1) Documentation: This section usually involves description and details of what the program does. It is usually presented in the form of comments.

Eg. // description, name of the program, programmer name, date, time etc.

2) Preprocessor Section: All header files of the programs are put in the preprocessor section. In-built fuctions can be accessed through them.

E.g. #include<stdio.h> #include<math.h>

3) Global Declaration: The global declaration section is one where global variable, functions and static variables are defined. Variables defined in the scope can be used anywhere in your code.

E.g. int a = 0 char s = "Hello World"

4) Main() Function: Every C program must have a main() function. Operations like declaration and execution are performed inside the curly braces of the main program.

E.g. void main()

5) Sub-Programs: User defined functions are contained in this section.These are specified as per the requirements of the programmer.

E.g. int sum(int x, int y)

{

return x+y;

}

**HOW CAN I CREATE A C PROGRAMING FILE ON THE OS**

Create a new file on the os with desired name, then save it with the “.c” extension. This tells the operating system what compiler to run the file with.