Name: Adesola Adeoluwa
Matric Number: 223765
Course code: CSC235

Github repository

1. THE UNIX OPERATING SYSTEM

The Unix operating system was built by Ken Thompson, Dennis Ritchie at AT&T Bell laboratories in 1970. It was originally meant to be used by programmers to develop programs.

Linux is an open-source Unix operating system that was released by Linus Torvalds in 1991. It has many distributions/flavours that have their specific users. A few of these flavours are highlighted below:

a. Ubuntu - This flavour is suitable for all types of users. It comes with a lot of tools/packages in the default repository. It is beginner-friendly
b. Kali linux - suitable for cyber security, digital forensics, penetration testing and ethical hacking
c. openSUSE - openSUSE is suitable for developers and system administrators with in-built functionalities to help you manage systems and utilise containerization technologies included.
d. Linux mint - works out of the box with full multimedia support and is extremely easy to use
e. Red hat enterprise linux - a linux flavour designed for enterprise and commercial purposes.
f. Arch linux - a lightweight distro for advanced linux users which gives users the freedom to configure their systems to their preference.

2. SOFTWARE FUNCTIONAL REQUIREMENTS

These are product features that developers must implement in a software for users to use the software with ease. Software functional requirements should include:
i. Clear descriptions of workflows for the developers
ii. Clear descriptions of operations and data logic
iii. Define administration/authorization access

Having software functional requirements is important because it helps the developers visualise their goals and helps to guide against some bugs in early stages. It also helps developers ensure the software is doing exactly what it was built for.

3. WHY IS UNIX OFTEN PREFERRED AT SOME POINTS?

i. Security: Unix systems are more secure than Windows systems.

ii. Ability to run services/automation. Scripts can be written and set up to run at specific times on linux systems.

iii.Flexibility: Unix(linux) operating system can be run on almost any machine, as it requires minimal hardware features to run perfectly. Running a linux flavour on any old machine would “revive” that machine as long as the minimum hardware requirements are met.

iv. Reliable: Unix systems do not need to be restarted in order to implement/download an update, unlike Windows where you can be denied access from doing any work when your system is being updated.

v. It is free and open-source: No need for any licence to run linux on your machine and the source code is readily available for any and every one to take a look.

4. WHY IS UNIX REFERRED TO AS THE SCIENTIST OS?

Unix is referred to as the scientist operating system because a lot of tools needed by scientists are readily made available for use at any time. Unix provides an environment that is suited for developers.

5. WHAT TYPE OF PROGRAMMING LANGUAGE IS C

The C programming language is a procedural language, i.e it is a language that consists of a set of functions that must be executed in a particular order to accomplish a program. It is a statically typed language, meaning that all data types should be known at compile time.

6. DETAILED STRUCTURE OF A COMPLETE C PROGRAMMING LANGUAGE

A C program comprises 6 sections. They are: Link, Definition, Documentation, Global Declaration, subprograms and main function. The main section is compulsory and all other sections are optional.

Link - This section is where all header files(different functions from the libraries) are included.

Definition - In this section, constants are defined using the define keyword. E.g. “#define secondsPerMinute = 60

Documentation - This is written in form of comments. It contains information like the programmers’ name(s), description of the program, and any other useful information.

Global Declaration - Consists of global variables, functions

Subprograms - This consists of all user-defined functions. These functions are called in the main function

Main function - This is the entry point/starting point for execution of any C program and a compulsory function.

7. HOW CAN I CREATE A C PROGRAMMING FILE ON THE OS

To create a C programming file on the linux OS, there are various ways. One of those ways is typing “touch filename.c” in the linux terminal; after this command is executed, an empty c programming file is created and then, you can proceed to use your preferred editor to write your code. Another way is to use a terminal text editor, e.g. vim, nano, emacs; the way this is done is to use “text-editor filename.c”, then a new c programming file is created, if it does not exist, and then a text editor is opened in the terminal where you can proceed to type your code. For example, “nano hello_world.c”

After the file has been created and populated with code, you compile the code by executing “gcc filename.c” in your terminal. This compiles the code to a binary file with “a.out” as the default name. You can change the name of this binary output when compiling using a -o flag. For example, “gcc filename.c -o binary_file”. Now, the code has been compiled to binary_file and you can run the binary file by executing “./binary_file” or “./a.out” in your terminal