Adetoyinbo Adeyemi

Cohort 16.

NB: The A4 Summary of the Chapter is at the end of this document.

1.4

a. Input Unit

b. Programming

c. Assembly Languages and Assemblers

d. Output Unit

e. Memory Unit (or RAM) and Secondary Storage Unit

f. Arithmetic and Logic Unit (ALU)

g. Central Processing Unit (CPU)

h. High Level Languages

i. Machine Language

j. Central Processing Unit (CPU)

1.5

a. Java

b. Java Standard Edition (Java SE), Java Enterprise Edition (Java EE), Java Micro Edition (Java ME)

c. Bandwidth

d. Assembler

1.6

a. Edit, compile, Load, Verify and Execute – EcoLoVE

b. Integrated Development Environement (IDE)

c. Java Virtual Machine (JVM) and it executes Java programs.

d. Virtual Machine

e. Java Virtual Machine Class Loader

f. Bytecode Verifier.

1.7. Just in Time (JIT) translates bytecodes (that has been verified by bytecode verifier) into the computer's machine language such that when Java Virtual Machine encounters these compiled parts again, the faster machine-language code executes. The purpose of Just in Time Verification is to ensure that bytecodes are translated into machine language.

1.8. Wrist Watch: A wrist watch as an object is the reusable component of a clock. I got a wrist watch in December 2022 and within two months, I already changed the battery twice. The first person just replaced the battery and after using it for about 2 weeks, it stopped working again. The second person I took it to told me the battery is still good but that we needed to change the **engine**. The **attributes** of the wrist watch include the strap, the hook, the colour, the shape and design**. The behavior** of the my wrist watch is analog. As an analog watch, it measures different concepts of time, milliseconds, seconds, minutes and days. Unlike the replace I had to get for it which had different behavior (including measurement of days).

Method and Class: When the second repairer opened it, there were a lot of things I didn't understand. I just thought he would change the battery and the wrist watch would work fine. However, I discovered that there were different methods involved.

The wrist watch had three buttons, each of which performs different function, behind the buttons are set of other methods that ensure that the 'millisecond' doesn't miss a count, or the 'second' counter or the minute. Each of the buttons have their role. The Class includes these set of methods (The three buttons are a class – we can name them class button, .button).

Inheritance: This wrist watch could end up becoming a "Stop Timer" because of the different classes it has. It could be used, in the long run, to measure times in different capacities; milliseconds, seconds, minutes, hours.

Interface: The buttons beside the analog clock, the time counters (hands), the casing behind the wrist watch, the strap and hook – these things inform the functionality of what the wrist watch does. They allow me tell the wrist watch what I want it to do.

Encapsulation refers to the bonding of data with the methods that operate on the data. It is a process of hiding the values preventing unauthorized parties direct access to them. In this case, the watch has three buttons, encapsulation occurs as each button conceals its own function, same way the different methods inside the watch have their own secrets. It is also the same thing as Information hiding and is crucial for the engineering work.

Messages. Pressing one of the buttons sends a message to perform the task. For instance, the top button when pressed twice activates the milliseconds button which, the second repairer said is the best. On the other end, pulling out the middle button gives you the liberty of manually adjusting the wrist watch and pulling it back in starts the watch, the seconds and minutes handles.

Modelling is the actual work. The process through which the design and functionalities is mad.

2. Java Compilation Process.

Source codes are converted firstly to bytecodes so that Java Virtual Machines can understand and this represents the task to execute in that particular stage. The Java Virtual

Machines execute bytecodes. Java Virtual Machines are platform based in such a way that the computers that JVM.

3. Differentiate between JDK and JRE

JDK is used for development purpose whereas JRE is for running the java programs.

4. If the computer is so smart, why do we need a compiler?

We need a compiler for the execution phase of the source code. This is not possible as Java source codes are platform based, and not machine based.

5. Do a one page summary of the whole chapter.

ONE PAGE SUMMARY OF CHAPTER ONE

The chapter is an introduction to computers, the internet and Java. The chapter started with a brief introduction to Java. It introduced Java an object oriented programming platform that is widely used by software engineers around the world to build, or build upon their project. It also spoke about some of the devices that is built on Java; Mobile phones, Automobiles, Access Control Systems and even Televisions.

I learnt about the different Java editions which comprises of Java Standard Edition otherwise known as Java SE. It contains capabilities needed to develop desktop and server applications unlike Java Enterprise Edition which is geared towards development for large-scale distributed networking application and web-based applications. Java micro edition is geared towards smaller devices like smartwatches, smart meters etc.

I discovered the power computers and the prospect of it. Because computers are such a beauty, they process data under the control of series of instructions called Computer Programs. The chapter also discussed Supercomputers, and after further research, I discovered that Sunway Taihulight wasn't able to win the fastest, with United States "Frontier" the fastest right now. Frontier has 1.1 Exaflops.

The computer also has software and hardware component and the chapter discussed the different part of a computer hardware. Including input and output devices, the CPU, Memory Unit, Arithmetic Logic Unit and more. The computer also, was said to have Moore's Law… computer consists of various devices referred to as hardware (e.g., the keyboard, screen, mouse, hard disks, memory, DVD drives and processing units). Computing costs are dropping dramatically, owing to rapid developments in hardware and software technologies.

We proceeded into Data Hierarchy, important for knowing and understanding. The smallest data in the computer is Bits short for Binary Digits. Because it is hard to work with Bits, we prefer to work with Decimal Digits, Letters and Special Symbols. Java uses Unicode because the computer uses only 1 and 0s.

Programmers use different programming language to put out their instructions, some need translation tools while others do not. There are three various types of programming languages disussed in the chapter; Machine, Assembly and High Level Languages. Machine Languages are those ones that the computer easily understand because it is writing in the language of the computer. It consist of strings of numbers, 1s and 0s and it instruct computers to perform their operations one at a time. Machine languages are machine dependent.

Unlike Machine languages, Assembly Languages uses English-like abbreviations to represent elementary operations. Translator programs called assemblers were developed to convert early assembly-language programs to machine language at computer speeds.

High Level Languages brought about higher speed of computer programming. They were created to speed up programming process. They were developed in which single statements could be written to accomplish substantial tasks. Unlike machine language which were written to be easily understood by the computer, High Level Languages need Compilers to turn these platform-based languages into machine languages so that computer can process them.

The next thing the chapter discussed was Java and Object based programming where we used the analogy of the car and how these things form the whole car. Froom the pedal to the unseen classes and methods. Object Oriented Design, which explains the totality of Java, is focused on reusable objects in such a way that you do not need to code everything, you just need to use components that have been done before. The purpose is for programming and innovation to be faster and easier.

The chapter also introduced us to Operating Systems where we learnt that Linux, aside from having partners and a community, is also an open-end platform where it is easier to get support and for codes to be verified, edited and worked on by the community. We learnt that it is easier to innovate with Linux. Other Operating Systems include Windows and Apple IOs.

The chapter ended with the Java Development Environment, Test Driving a Java application, the Internet and World wide Web and Software Technologies.