MODULE: OPERATING SYSTEM INTRODUCTION 1 (1ST YEAR – S1)

Objectives:

1. Introduce students to practice the operating system UNIX.
2. At the end of the course; the students must be able to work in the UNIX environment (user mode) and they must be able to install UNIX (LINUX) in their machine.

Content:

Chapter 1: Introduction

- Operating systems introduction
- Operating systems history
- Introduction/history of UNIX

Chapter 2: UNIX Operating system presentation

- Unix users
- Unix structure
- Principles functions
- Shell

Chapter 3: Opening and closing session

- Opening session
- Creating/changing password
- Closing session

Chapter 4: Unix commands and operations

- Command syntax
- Basic commands

Chapter 5: Pipe and redirection

- Input/output
- Input/output redirection
- Error redirection
- Pipes

Chapter 6: Unix files and directories

- File types
- Reach a file (name a file, path)
- View a file
- Directory management commands
- File management commands

- Links (symbolics, physicals)
- Inodes
- Metacharacters
- Access rights
  - User identification
  - User rights definition (file/directory)
- Other commands (chmod, umask, group properties)

Chapter 7: UNIX filters

- Change data's file
  - Cat a file to parts (split)
  - Sort a file (sort)
  - String conversion (tr)
- Edit a file with criteria
  - edit a file from the end (tail)
  - edit a file from the beginning (head)
  - count the lines of a file (wc)
  - edit file field (cut)
  - files fusion (paste)
  - extraction of common lines of two files (comm)
- files comparison
  - two files comparison (cmp)
  - edit differences between two files (diff)
- the commands grep and find
  - Regular expressions
  - Grep command
  - Find command
- Process management
  - Process characteristics
  - View a process
  - Process management commands (launch a process, stop a process)

Chapter 8: system administration introduction

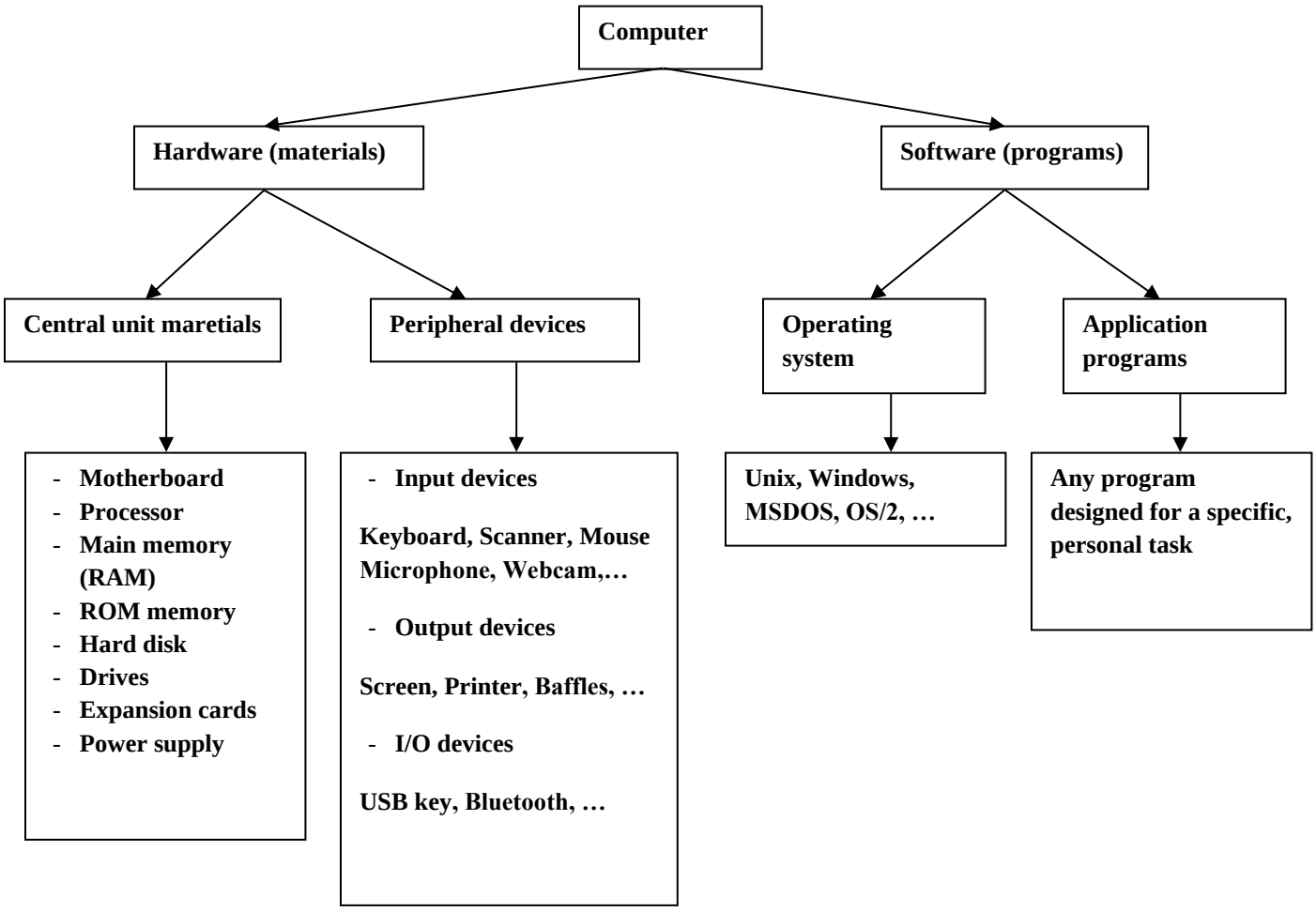
- UNIX installation
- UNIX account management

CHAPTER I: OPERATING SYSTEMS INTRODUCTION

I/ Introduction:

The designers' aim in creating the first computer was to make human life easier by automating all kinds of tasks. In the 40s, only one group of people designed, built, programmed and used the machine, because of its complexity (in other words, the use of computers was limited to this group). Over the following years, this complexity was gradually reduced, until the 80s, when the computer evolved to become within everyone's reach, while using an intermediary element to perform all tasks. This intermediary between the user and the computer is called the "operating system".

II/ Recall: the computer components



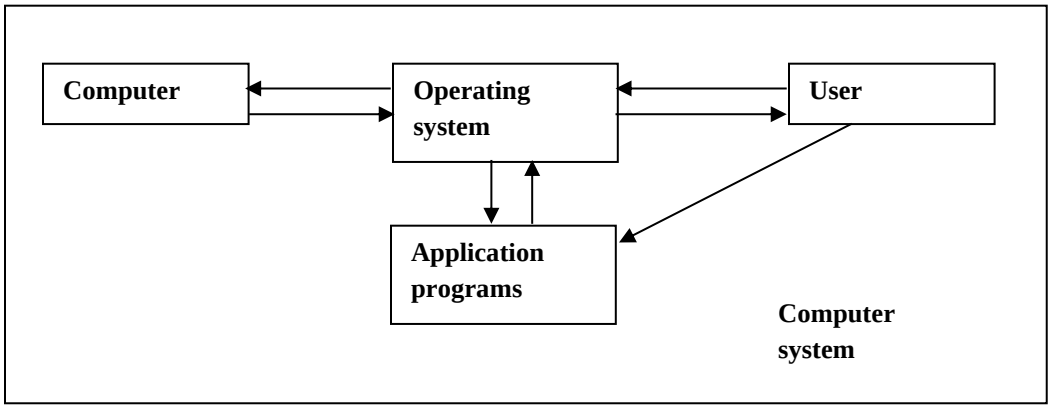
III/ Definition:

We can consider the definitions:

Definition 1: an operating system is the intermediary between the computer and the user.

Definition 2: An operating system is a set of essential programs that manage, control and operate the computer's resources.

Definition 3: an operating system acts as an intermediary between application programs and machine hardware.



Note: a computer system consists of hardware, OS, application programs and users

IV/ Resources managed by the OS:

The operating system manages two categories of resources:

a. Hardware resources: it manages the processor, the memories and I/O devices
b. Software resources: it manages:
   - Application programs
   - Users' data

V/ Operating system functions:

The OS can provide several tasks to ensure user convenience and execution efficiency:

a/ Program execution control:

The system takes charge of controlling the program execution stages (program on disk → loading into main memory → execution on the processor).

This task becomes more difficult when you are in a multitasking environment where you have to decide:

- Which program to run first

- For how long
- Which program to prioritize

b/ Memory management:

The OS needs to share all memory space between programs and data, knowing which spaces are free and which are occupied, in order to facilitate and speed up execution.

c/ Processor management:

The operating system must control the execution of a process at processor level. It must decide on the program scheduling policy during execution, as well as the execution time of each program (especially in time-sharing systems).

It manages status and execution priorities.

d/ Input/Output operations and devices management:

The OS provides I/O request service regardless of the complexity of the device and media used.

e/ File management:

The OS must enable users to access their files via specific operations.

f/ Resource sharing between users:

The operating system allocates each resource to the required user using a predefined policy. It manages access to the resource, as well as its release and recovery.

g/ Protect users from unexpected actions by other users:

Unexpected actions are the result of :

- The use of resources by different users.
- Simultaneous execution of several programs.
- Sharing a computer by several users.
- ….

h/ Hardware/software error detection and recovery:

The OS must be able to detect anomalies, report them and correct them if possible.

i/ Machine virtualization:

The OS must offer the user a virtual (abstract) machine in place of the real (physical) machine. This virtual machine provides the user with a command language (or graphical user interface) better suited to the instructions of the physical machine.

VI/ Operating system classes:

a/ Industrial process-oriented system:

This type of system is designed to control an industrial process, and to achieve this goal, the system must be able to perform the following additional functions:

- Regulation: maintaining operating parameters (temperature, flow, pressure, etc.) at the values required for production to run smoothly.
- Recording: periodically record the values of operating parameters, in order to keep track of the history, which can be used in studies targeting production improvement.
- Safety: to avoid disasters if certain parameters exceed their correct values. The system must be equipped with a safety function capable of stopping the entire process in the event of failure.

b/ Transaction-oriented system:

This category of system is designed to store and manage large volumes of information (databases), such as Air Algérie's reservation system or the CCP. The functions performed by this system are:

- Interactive transaction execution: a set of indivisible operations executed on databases.
- Competition management: information in databases is shared between different users, who can access it simultaneously. The role of this function is to manage access conflicts between users.

c/ Program creation and execution-oriented system:

This category of system allows users to create and run their own programs.

VII/ History and evolution of OS:

a/ Fully sequential systems (1950-1960):

Operating systems during this period were based on the working principle of running a single program at a time:

- Stand-alone programs: Initially, computer peripherals consisted of a card reader, a card punch and a printer. From the user's point of view, operation was fairly straightforward, since program execution involved placing a packet containing the "binary" form of the program, followed by the data, into the card reader. Once the program had been read, the computer would execute it, with the results in the form of a packet of punched cards, or printed lines.
- The job sequence monitor (1955): the user prepares a deck of cards representing his job, consisting of one or more steps. Each step corresponds to a program whose execution is requested by the user on a particular data set. This program may be one of the utilities stored on tape, such as a compiler, or the result of a previous step temporarily stored on tape.

b/ Systems introducing parallelism (1960-1965):

The most popular systems at the time were multi-programming systems, whose principle was to keep several jobs ready to run in memory and share resources between them. The processor is initially allocated to one job and the rest are in memory; as soon as the job performs an I/O, the processor is allocated to the next job.

c/ Time sharing systems (1965):

The processor in these systems is switched to the next process each time a Q (Quantum Time) delay expires.

c/ Real time systems:

These are specialized systems dedicated to specific applications, in particular control systems. They are used when there are response time requirements.

d/ Parallel systems:

These are highly coupled systems with more than one processor sharing the clock and memory.

e/ Distributed systems:

They are systems that use several processors, not necessarily identical, each with its own memory.

f/ Embedded systems:

Are systems for mobile terminals.

VIII/ Examples of OS:

Several operating system variants have appeared since the 50s, the most famous of which are:

1/ MS-DOS: Microsoft Disk Operating system

It's a single-user system, known since the early '80s, designed for microcomputers (the first IBM-PC 8088), conceived by Bill Gates and now associated with Windows.

2/ Windows Operating system

A system designed by Microsoft with a graphical interface, its name refers to the use of a window for each execution. It is designed for several architectures (PC, workstation, laptop, client/server network, etc.).

It has gone through several versions, as for PC: 1.0, 2.0, 3.10, 3.11, 9x, 2000, Me, XP, Vista, Seven, 10, 11, also: Windows Server, NT for multi-user and client/server architecture.

3/ UNIX Operating system: Uniplexed Information and Computer Service

A family of time-sharing systems proposed for most architectures, used in certain industrial and research environments.

4/ Android operating system:

The Android operating system is a mobile operating system that was developed by Google (GOOGL) to be primarily used for touchscreen devices, cell phones, and tablets. Its design lets users manipulate the mobile devices intuitively, with finger movements that mirror common motions, such as pinching, swiping, and tapping. Google also employs Android software in televisions, cars, …

5/ IOS operating system:

IOS stands for iPhone operating system. It is a proprietary mobile operating system of Apple for its handheld. It supports Objective-C, C, C++, and Swift programming languages. It is based on the Macintosh OS X. After Android, it is the world's second most popular mobile operating system. Many of Apple's mobile devices, including the iPhone, iPad, and iPod, run on this operating system. To control the device, iOS employs a multi-touch interface, such as sliding your finger across the screen to advance to the next page or pinching your fingers to zoom in or out of the screen.

APPENDIX:

1/ Central unit:

Motherboard: This is an electronic board that connects all the PC's components and enables the flow of information.

Processor: This is the brain of the machine, executing all the operations requested and providing the expected responses.

Random Access Memory RAM: This is the processor's working area, where the information needed for processing is stored. It's a volatile memory that's erased whenever there's a power cut.

Hard disk: This is where information is stored; it's a permanent memory.

CD/DVD and floppy disk drives: These drives provide access to external disks (floppy disk, DVD and CD-ROM).

Expansion cards: These are cards that are added to the motherboard to provide other functions not available on the motherboard (e.g. network card, sound card, graphics card, etc.).

2/ Peripheral devices:

Input devices:

These are the elements of the PC, such as: keyboard, mouse, scanner, modem...etc. used to enter data.

Output devices:

These are the elements of the PC that allow information to be output, such as: screen, printer, speakers...

Input/output devices:

These are the elements of the PC that enable information to be input and output, such as USB keys, digital cameras and scanner printers,