Lebanese University

Faculty of Sciences

Lebanese University

Faculty of Sciences

I1100 E

Introduction to Computer Sciences

Department of Applied Mathematics

Fall Semester

2021 - 2022

Table of Contents

Table of figures

Course objective

General objectives

The overall purpose of a course is to describe in a holistic manner all the lasting changes that the teacher wishes to see in students during a course. Thus, this course aims to give, at the very beginning of training, a global overview of the field of computing, by introducing different sub-domains of computing (architecture, operating systems, languages, networks). It aims to provide students with a computer culture by instilling them with precise knowledge of the means used in computers and their methods of use while introducing them to new areas of computer knowledge and providing them with the necessary elements to better understand other computer courses.

Methodology

As students are a priori novices, the teacher should approach the different themes in a top-down approach by focusing on the functional aspect of each device (system, network, ...) to then give a precise description but not necessarily detailed of the latter. This description aims to make the student familiar with computer jargon and give him the opportunity to choose a configuration of computer equipment according to his needs.

Specific objectives

A specific objective describes, within the limits of a theme, what students must achieve during or because of a learning situation. Thus, at the end of the course, the student must know, if not master perfectly, a number of basic concepts relating to different sub-fields of computing. In particular, the student will have to:

• Know the components of a computer

• Know the components of a computer system

• Learn the basic concepts of computer networks

• Understand the basic concepts of the Web

• Understand the type of services, documents and information that can be found on the Internet

• Create static web pages

• Become aware of the role and duties of the computer scientist.

Chapter I: Introduction

I.1 Definitions

Computer Science

Computer science is the science that deals with the theory and methods of processing information in digital computers. In other words, computer science is the study of theoretical foundations of information and calculation, and their implementation and application with computers.

The purpose of information processing is to:

- Perform arithmetic and logical calculations
- Manage and store data
- Communicate and exchange information

Computer

A computer is a machine or device that processes information according to a set of predefined instructions or program. It interacts with the environment thanks to peripherals (screen, keyboard, modem, etc.). A computer is made of a set of electronic circuits that manipulate data in binary form or in bits.

Information

When data is processed, it is called information. Data can be any character, number, word, text, picture, sound or video. It is used as input for the computer system; it is a coded representation to be processed by the machine. Output data can be perceived as information.

I.2 Importance of using computer science in an enterprise

Computers have tremendously improved the way enterprises operate by offering several important advantages:

- **Better production**, for example by automating the production control, monitoring, and the inventory management.
- **Better administration** by memorizing, archiving, and automating payments.
- **Better decision** by using statistics, forecasts, analyzes and reports.
- **Better communication** among the staff by sharing different software applications, Emails, Intranet, etc.
- **Minimize expenses** and **maximize profitability**.

I.3 Areas of Application of Computer science

- **Finance** (stock exchange, banks, …)
- **Business** (e-commerce, marketing, management, data mining, …)
- **Science** (weather forecast, geography, physics, mathematics, …)
- **Medical** (pharmacy, patient history and monitoring, genetics, …)
- **Network supervision** (pipeline, flights, trains, trucks, transport, water distribution, …)
- **Operating supervision** (product manufacturing, nuclear power plants control, …)

I.4 Jobs in Computer Science

- **Database administrator**: is involved in the planning and development of the database as well as in troubleshooting any issues on behalf of the users.
- **Network administrator**: installs, maintains, repairs, and upgrades network and computer systems.
- **Network architect**: designs and builds data communication networks including local area networks (LANs), wide area networks (WANs), and intranets.
- **Software project manager/ IT project manager**: has the responsibility to deliver every project on time within the budget and scope.
- **Computer programmer:** writes programs in a variety of computer languages such as C++ and Java, debugs programs by testing for and fixing errors.
- **Mobile application developer:** designs and writes mobile application code.
- **Computer security expert:** maintains security infrastructures and designs, implements and tests compliant network security systems and solutions.
- **Computer instructor**: teaches the principles of computers to students at different levels.
- **Industrial computer scientist**: develops new products and solves computing problems.
- **System engineer**: implements new systems, correct software errors in existing systems and improve performance through hardware upgrades.
- **Telecommunications network engineer**: sets up the interaction between a computer network and communications equipment such as wireless telephony services, radio and satellite communications, internet and broadband technologies.
- **Computer maintenance technician**: runs diagnostic tests, installs updates to existing software, and sets up new equipment.
- **Software tester**: is involved in the quality assurance stage of software development and deployment.

I.5 History of Computer Science

The evolution of computer science can be divided into five generations.

First generation (1945 – 1955)

During this period, computers moved from electromechanical into electronic ones. These computers used **vacuum tubes** to control electron flow, as switches and amplifiers of electron signals.

Figure 1: Vacuum tubes

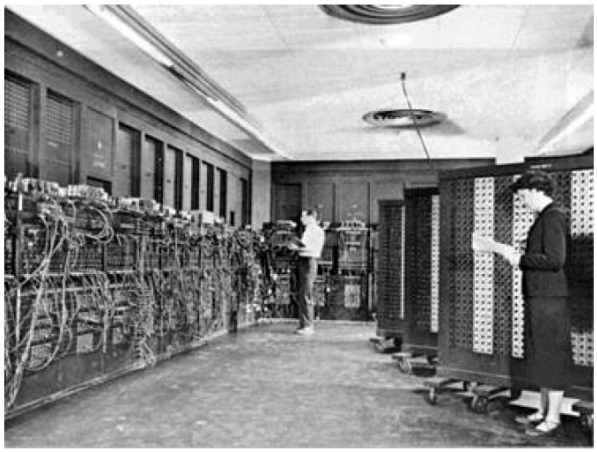


Figure 2: ENIAC (U.S. Army Photo)

ENIAC (Electronic Numerical Integrator and Computer) was amongst the earliest electronic general-purpose computers made. It was designed by John Mauchly and J. Presper Eckert of the University of Pennsylvania, U.S. and completed in 1945. ENIAC was originally intended for artillery calculations and ordered by the US army. It weighted more than 30 tons, occupied 167 m^2 (see Figure 2) and consumed 150kW of electricity. ENIAC could perform 5000 simple addition or subtraction per second and its calculations were in decimal. It used an IBM card reader as input and an IBM punch card for output.

Second generation (1955 – 1960)

In 1947, John Bardeen and Walter Brattain engineers at AT&T's Bell Labs in U.S. successfully tested the **transistors** thus paving the way for semiconductor revolution. In 1954, Texas Instruments produced silicon transistors. It is smaller, cheaper, and dissipates less energy than a vacuum tube. Transistors progressively replaced the vacuum tubes. As a result, simpler computers were designed. The use of mass memories such as tapes and magnetic disks became widespread.

Figure 3: Transistor

Third generation (1960 – 1970)

The **integrated circuits** (IC) were used to build more compact computers called minicomputers, which became more reliable and cheaper. Jack Kilby developed the first IC or microchip in 1958 at Texas Instruments. The IC principle is to manufacture in the same semiconductor block as small as possible, a maximum logic functions that can be accessed through connections distributed around the circuit. In 1959, Robert Noyce developed at Fairchild society an integrated circuit printed on a silicon surface. From 1958 to our present days, the concept of integrated circuit has been extraordinarily developed. The most famous application of IC was the microprocessor. Operating systems saw major advances during this period.

Fourth generation (1971 -)

This generation appeared around the 70's and is by far the most eventful. It is the most generation that marked the **microcomputer**. **Personal computers** (PC) appeared as the result of the **microprocessor** development. Intel 4004 (see Figure 4) was the first commercial microprocessor. Ted Hoff at Intel developed it in 1971. This processor is capable of processing a 4-bit data. It contains 2,300 transistors and can perform 60,000 operations per second. ARPANET was developed in 1970 as the first network connecting 23 computers. The sent of the first email was made by Ray Tomlinson in 1971. The theory of **databases** saw major advances with the work on relational databases. Many PCs were put on the market like Apple II and Commodore PET. In 1984, Apple launches Macintosh, the first personal computer exclusively graphical that operates with a mouse (see Figure5). In 1986, ARPAnet was infected with Brain the first computer **virus** that renames all the system booting disks with (C) Brain. Some people consider that we are still in the fourth generation since no component has revolutionized the computer science world ever since.

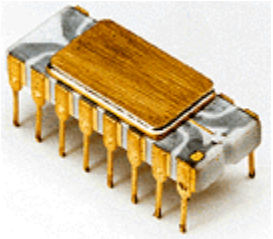


Figure 4 : Intel 4004 micro-processor

Figure 5 : Apple Macintosh

Fifth generation (early 90's)

Web was invented by Tim Berners-Lee in 1989. **Laptop** was invented in 1986, **GSM** (Global System for Mobile Communication) is a second generation for mobile phones appeared in 1991, smartphones existed starting from 2001, an example, Apple launched its first phone which was **IPhone** in 2007. In recent years, there has been a trend towards multiprocessors systems in order to improve computers performance and reliability.

Chapter II: Computer Structure

II.1 Von Neumann Architecture

At the end of 1945, John Von Neumann proposed the stored program concept and the construction of EDVAC computer (Electronic Discrete Variable Automatic Computer) as a "model machine of the computer". EDVAC was a binary computer rather than decimal, and was a stored-program computer. Von Neumann proposed to store both the program and data together in a memory. Thus, the computer gains flexibility and speed. The vast majority of modern computers follow the Von Neumann model.

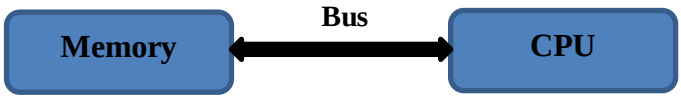


Figure 6: Von Neumann Model

The program is a sequence of instructions provided to the computer for an eventual execution by the processor. The role of the processor is to extract the instructions from the memory one by one, to interpret them, and to execute them until the problem is solved. The instructions transfer is made through buses.

The Von Neumann architecture divides the computer into four distinct parts (see Figure 7):

1) ***ALU (Arithmetic/Logic Unit)***: performs mathematical operations (+, -, ×, /) and logic operations (=, <, >, and, or, not) requested by program.
2) ***Control Unit***: interprets the instructions in memory and causes them to be executed.
3) ***Memory***: stores both data (variables) and a program (a set of instructions).
4) ***Input/output Units (I/O)***: allow the computer to communicate and interact with the outside world like a screen, a keyboard, etc.

The program to be executed is stored in memory during execution and program instructions are executed sequentially.

1) Arithmetic Logic Unit (ALU)

The **ALU** is the computer unit that is responsible for performing calculations. It is integrated into the CPU (i.e. the microprocessor). The ALU performs mathematical operations (+, -, ×, /), logic operations (and, or, not, xor), and comparison operations (=, <, >).

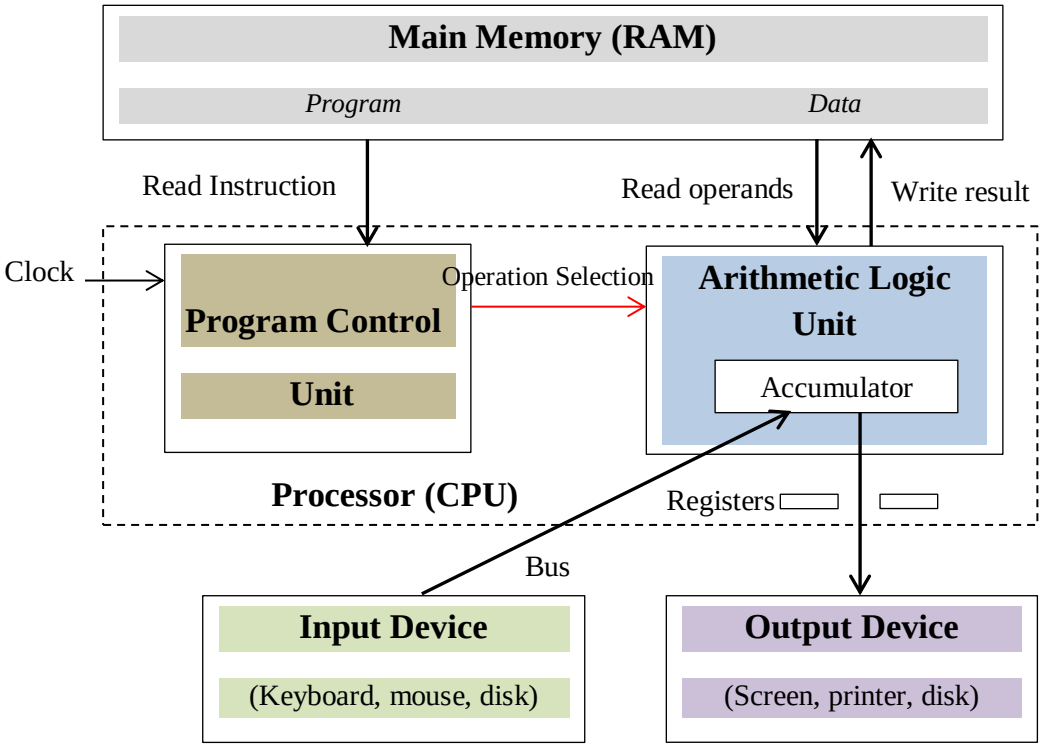


Figure7: Von Neumann Architecture

2) Control Unit

The control unit determines the order in which the instructions of a program should be executed and controls the retrieval of the proper operands. It governs the flow of information through the system by issuing control signals to different components.

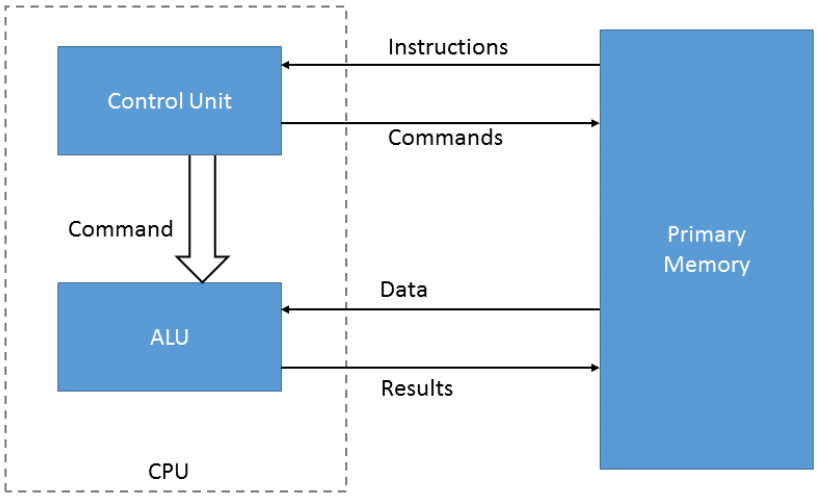


Figure 8: Order of the operations performed by the control

Processor (CPU)

CPU is the abbreviation for central processing unit commonly called processor, the CPU is the brain of the computer (see Figure 9) where most calculations take place. It is the unit that performs the program processing stored in the main memory inside a computer. CPU has two components: the arithmetic logic unit (ALU) which performs arithmetic logic operations (+, -, /, x, and, or, etc.) and the control unit (CU) which extracts instructions from memory and decodes and executes them, calling on the ALU when necessary.

Figure 9: Processor Example

The CPU also contains very fast and small memories called registers. A register may hold an instruction, a storage address, any kind of data, or temporary results.

3) Main memory (RAM)

The main memory is called Random Access Memory (RAM) is the area in a computer that stores programs and data for quick access by the processor (CPU). In addition, it holds operating system software necessary for a computer to function correctly. The computer can manipulate only data that is main memory. Therefore, every program you execute must be copied from a storage device into main memory. Everything in the memory is stored in binary.

The Buses

In computer architecture, a computer **bus** is a communication system that transfers data between components inside a computer. A bus consists of a set of wires that carry binary signals. There are three types of buses: a **data bus** for data transfer, an **address bus** for address transfer, and a **control bus** for control signals transfer such as read from memory.

Let *k* be the size of the address bus (i.e. the bus is made of k wires) and let *n* be the size of the data bus, the capacity of the main memory can be expressed either:

- in number of memory words: the capacity = 2^k memory words
- or in bits (bytes, Kilobytes, etc.): the capacity = 2^k * n bits

Example: Let k = 3 => the address bus is made of 3 wires (see Figure 10). Thus, the main memory has $2^k = 2^3 = 8$ words. Every word has a unique address. For example, the first word has the address

$A_2A_1A_0$ = 000. In the figure, the size of the data bus is n = 8, so the word size is 8 bits (D_7D_6 … D_0) and the memory capacity is = 8 x 8 = 64 bits.

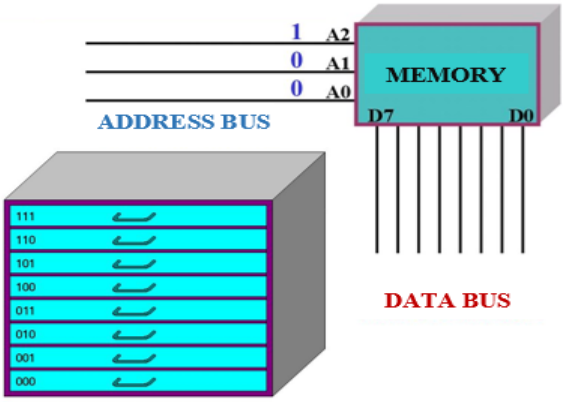


Figure 20: Address bus vs. Data bus

4) Input/output Units

Input/output units (or peripherals) are devices that allow the computer system to communicate and interact with the outside world and to store information (see Figure 11).

- ***Input units*** bring information into the computer that will be coded and later on processed by the CPU. Examples of input units: keyboard, mouse, scanner, webcam, microphone, cameras, joystick, touchpads, pen input, electronic whiteboard, barcode reader, card reader, CDs, sensors, and fingerprints reader.
- ***Output units*** are devices that bring information out of a computer after being processed. Examples of output units: monitor, printers, plotters, projectors, LCD projection panels, speakers, beepers.
- Some devices called ***input-output devices*** provide both input and output of data such as hard disks, diskettes, CD-RWs, pads, modem, touch screen, headsets, audio/sound cards, and network cards.

There are ***I/O controllers*** interface between the many different I/O devices and the internal system components: CPU, memory, and bus. A processor does not access external devices directly. Instead, the processor uses a programming interface to pass requests to an I/O controller, which translates the requests into the appropriate external signals.

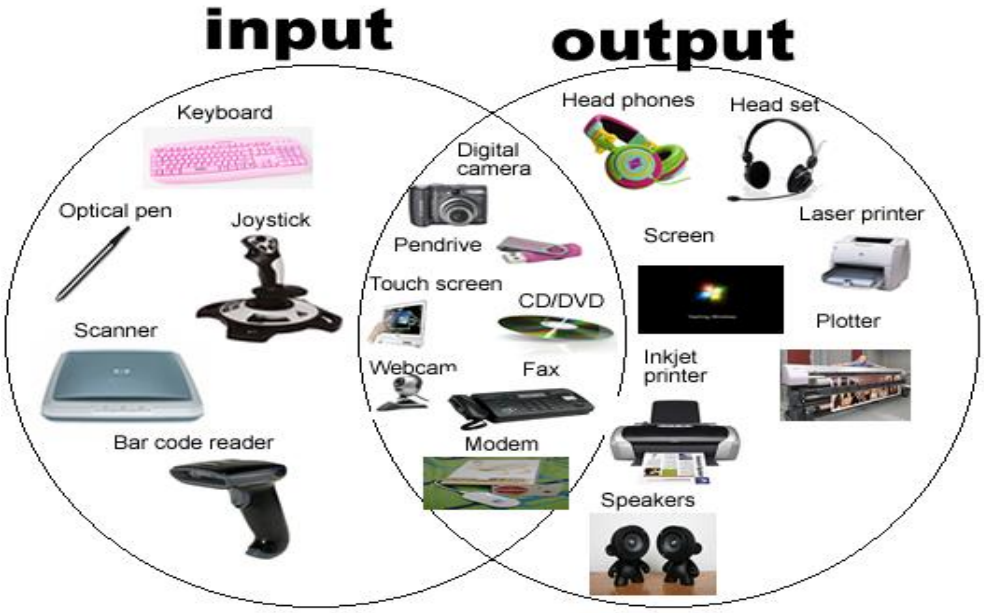


Figure 10 : Input/output Units

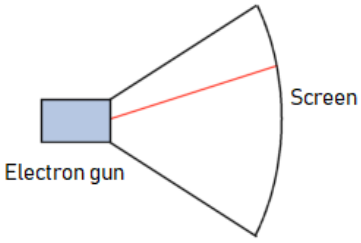
Input/output Unit Examples

Keyboard: A computer keyboard is the main input device for computers. It is composed of keys set that sends commands to the computer. There are QUERTY and AZERTY keyboards. User can enter a symbol, a character, a digit or execute a particular function.

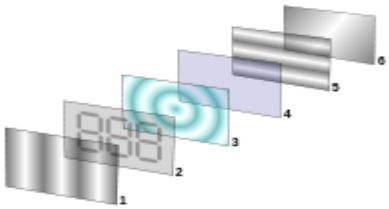
Computer screen: A computer screen or monitor is an output device that displays information in pictorial form. Monitors are connected to the computer via different types of connectors like VGA (Video Graphics Array), DVI (Digital Visual Interface), and HDMI (High-Definition Multimedia Interface) connectors.



The first computer monitors used **cathode ray tubes** (CRTs). These monitors are voluminous, heavy with high power consumption.




The **liquid crystal displays** (LCD) monitors replaced the CRTs. Nowadays, the monitors are found in laptops, smartphones, digital cameras, and desktops. They are low in power consumption, have a lighter weight and smaller physical size but has higher price versus a CRT.

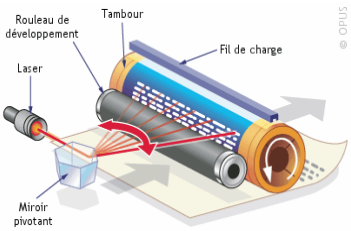
The performance of a monitor is measured by ***luminance*** (candelas per square meter cd/m^2), ***display resolution*** (number of distinct pixels in each dimension that can be displayed, it us quoted as width × height, example: 1024 × 768), ***Dot pitch*** (the distance between pixels of same color in millimeters), ***power consumption*** (in watts), ***Delta-E*** (color accuracy measured in Delta-E), ***Viewing angle*** (maximum angle at which images on the monitor can be viewed without excessive image degradation measured in degrees), and ***monitor size*** (many ways to measure the size by knowing ***image area*** [length×height] in inches note that 1 inch = 2.54 cm, ***aspect ratio*** (length:height) ex. 16:9, or the ***diagonal measurement***).

Computer printer: is an output device that allows a user to print items on paper such as letters, computer programs, and pictures. There are many types of printers:

- **Dot matrix printer**: uses a print head made of pins (between 9 and 24) that moves up-and-down motion on the page and prints by impact, striking an ink-soaked cloth ribbon against the paper. Every pin produces a dot and every character is made of multiple dots. The pins are driven forward by a tiny electromagnet. This printer generally prints one line of text at a time. It has many advantages such as low purchase cost, can handle multipart forms, cheap to operate, only needs fresh ribbons,

low repair cost and the ability to print on continuous paper. However, it is noisy, with low resolution, and not all can do color.

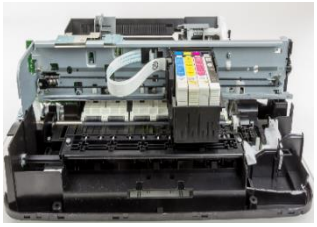
- **Inkjet printer**: creates digital images by propelling droplets of ink onto paper, plastic or other substrates. The ink comes in special ink cartridges which can be very expensive. It is the most commonly used type of printer, and ranges from small inexpensive consumer models to expensive professional machines. The Inkjet printer is used by people who print very little. It is quieter than dot matrix printer, prints good resolution photos between 300 and 720 DPI (Dots Per Inch), no warm-up time is needed before printing, it can be very cheap but the ink can also be rather expensive, and prints pages in color. It has an average printing speed and the ink in the cartridges may dry up.
- **Laser printer**: uses static electricity and a laser to melt the powder ink (called toner) from a cartridge onto the paper. Most do not print colors. Very often, it is 2 to 3 times faster and more economical to use than the Inkjet printers that use more ink. Laser printers typically cost a bit more than Inkjet and need more frequent maintenance and cleaning. Their print quality is not as good as Inkjet printers. It is good enough for text and charts but not ideal for photos.



Laser Printer

Dot matrix Printer

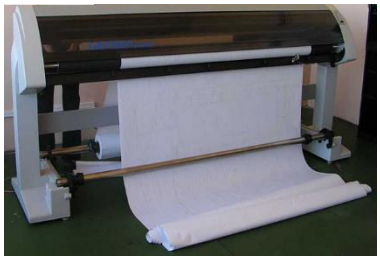
Inkjet Printer



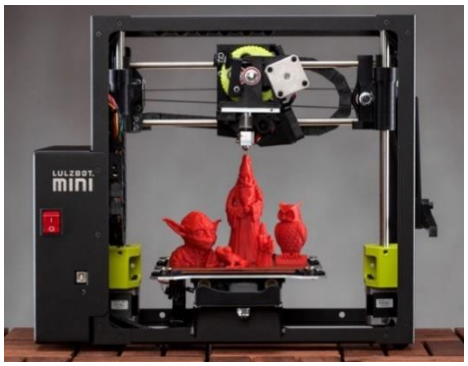
Inkjet Printer case removed

Laser Printer

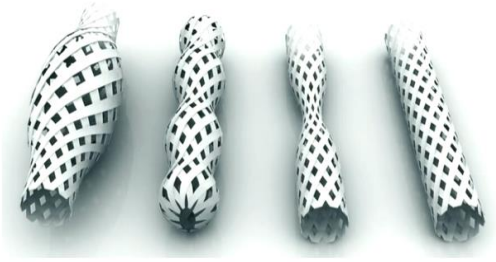
- **Plotter**: is a large format Inkjet printer usually used to print to very large paper.
- **3D & 4D printers**: *3D printing* or additive manufacturing (AM) is the process of building a three-dimensional object from computer-aided design (CAD) model or AMF file by successively adding material layer by layer. *4D printing* uses the same technique of 3D printing and adds the dimension of transformation over time. The printed product reacts with environment parameters (humidity, temperature, etc.) and changes its form accordingly.



Plotter



3D Printer



4D Printer

II.2 Motherboard

A motherboard is the main printed circuit board in a computer. It connects all the different components of the computer such as CPU, memory, connectors for input and output devices, and allows communication between them (see Figure 11).

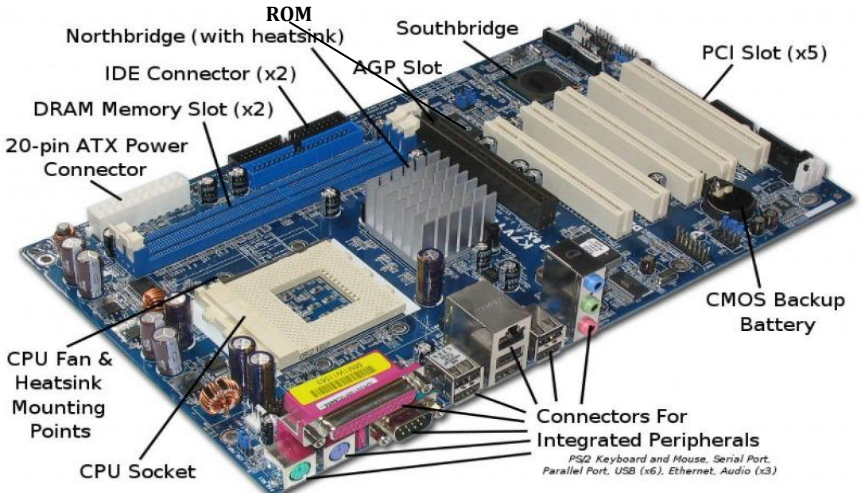


Figure 11: Motherboard

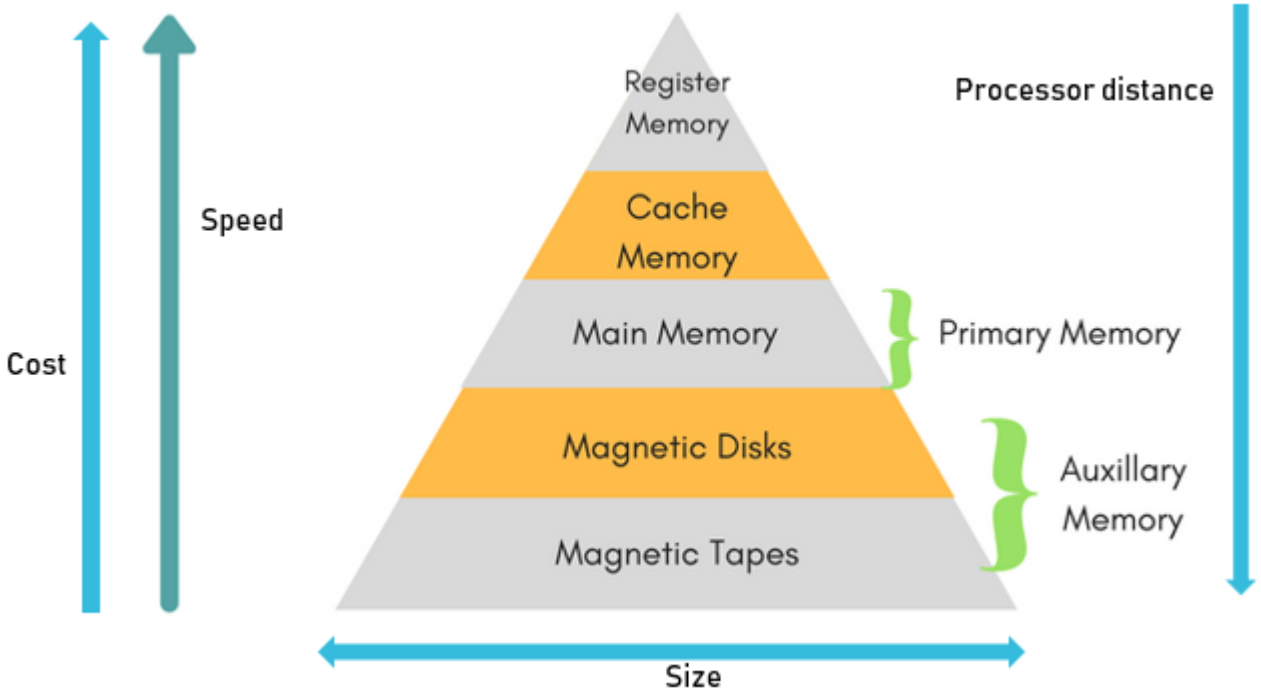
Parts of a Motherboard

Here are some of the more important parts of the motherboard:

- A **CPU socket**: the actually CPU is directly soldered onto the socket.
- A **power connector**: to distribute power to the CPU and other components.
- **Slots** for the system's **main memory**, typically in the form of DRAM chips.
- **Nothbridge chip** forms an interface between the CPU, the main memory and other components.
- **Southbridge chip** controls the input and output (I/O) functions. It is not connected directly to the CPU but to the Northbridge. The Northbridge and the Southbridge combined are referred to as the **chipset**.
- **Several connectors**, which provide the physical interface between input and output devices and the motherboard. The Southbridge handles these connections.
- **Slots** for one or more **hard drives** to store files. The most common types of connections are Integrated Drive Electronics (IDE) ans Serial Advanced Technology Attachment (SATA).
- A **read-only memory (ROM) chip** also called the BIOS, which contains the firmware or startup instructions for the computer system.
- A **slot for a video or graphic card**. There are a number of different types of slots, including the Accelerated Graphics Port (AGP) and Peripheral Component Interconnect Express (PCIe).
- Peripheral Component Interconnect **(PCI) slots** to connect hardware (for example different cards like sound, video, TV tuner, and network cards).

II.3 Different Types of Computer Memories

The memory is a device able to store, preserve, and extract information (coded in binary in a computer). Memories can be classified based on their capacity/size in Bytes, access time, volatility, access type (sequential and direct), and cost. Moving away from the processor, the memory access time and capacity increase and the cost per bit decreases. The memory hierarchy is shown below:



- **Registers**: found integrated in a CPU. A register consists of a small amount of fast storage. It provides the fastest way to access data and has the same speed of the CPU. A register can store a word of information such as an instruction, data like variables and information relative to an instruction (operands, parameters, results).The register size could be 8, 16, 32 bits. The exact number of registers depends on the type of the processor and typically varies between and a hundred. The number of registers that a CPU has and the size of each determine the power and speed of a CPU. Among the registers, the most important one is the accumulator register that stores arithmetic and logic results.
- **Cache**: is an intermediate memory between the CPU and the main memory. A cache is a smaller, faster memory, closer to a CPU, however larger than register. The cache contains copies of data from main memory frequently requested by the CPU. It is a SRAM memory with a size in powers of two 4, 8, 16, etc. KiB or MiB.
- **Primary memory**: or **main memory** is a fast memory but slower than registers. Its storage capacity is small and has a high cost per bit storage. Its size is between few hundreds of MB to few GB with a direct access. This memory is accessed directly by the processor and made up of RAM and ROM.

- **RAM (Random Access Memory)** is a temporary storage that stores program code and data that can be accessed, read, and written to by the processor (CPU) and other hardware devices. Its content can be modified (read and write). RAM is volatile; all its contents are lost when the power supply is interrupted. It can be further classified as **DRAM** (Dynamic RAM) and **SRAM** (Static RAM). SRAM is faster than DRAM and used as cache memory.
- **ROM (Read Only Memory)** is non-volatile and is more like a permanent storage for information. It stores the BIOS software, the bootstrap loader program to load and start the operating system when computer is turned on. **PROM** (Programmable ROM), **EPROM** (Erasable PROM), and **EEPROM** (Electrically Erasable PROM) are some commonly used ROMs.

ROM

RAM

- **Secondary memory**: also known as secondary storage or auxiliary storage is a non-volatile high capacity memory of several hundred MB to several hundred GB. It is the slowest and cheapest form of memory. This memory is not directly accessible by the CPU. A secondary memory is larger and slower than RAM. Secondary memory devices include **flash memory** that can be electrically erased and reprogrammed (USB flash drives, memory cards, and (SSD) solid state drives), **optical disc** for which data is read and written by lasers (CD-ROM, CD-RW, DVD, Blu-ray BD), **magnetic disk** is a circuit plate constructed of metal with magnetized material (HDD hard disk drive, floppy disk drive), and **magnetic tape** made of thin, magnetizable coating on a long, narrow strip of plastic film (Audio cassette, Video Home System VHS).

CD, DVD, Blu-ray

SSD

USB flash drive

VHS

Compact Cassette

Memory Cards

Hard disk drive (HDD)

A hard disk is an electromechanical data storage device that uses magnetic storage to store and retrieve digital information using one or more rigid rapidly rotating disks (platters) coated with magnetic material. The platters are paired with magnetic heads, usually arranged on a moving actuator arm, which read and write data to the platter surfaces (see Figure 12). Data is accessed in a random-access manner meaning that individual blocks of data can be stored or retrieved in any order and not only sequentially. HDDs are a type of non-volatile storage, retaining stored data even when powered off.

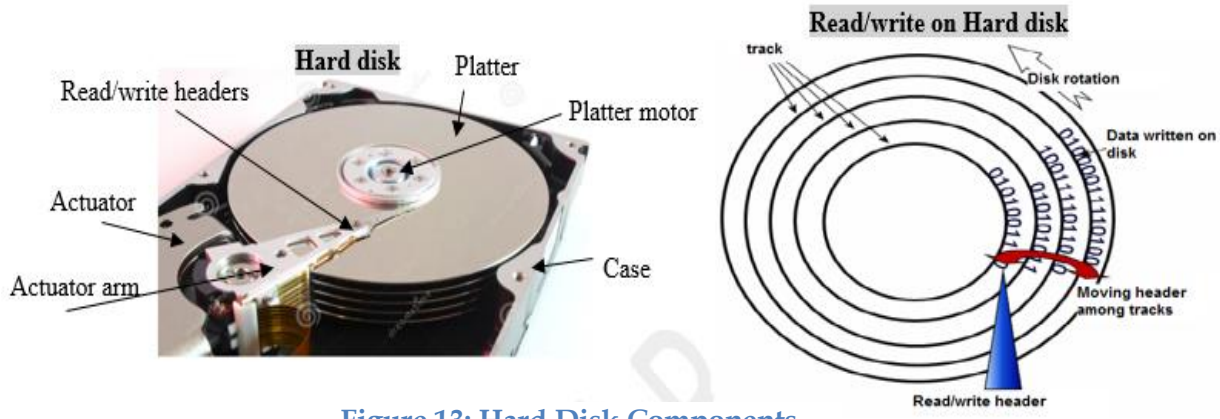


Figure 13: Hard Disk Components

Hard disk partitioning

Hard disk partitioning is the creation of one or more regions on a hard disk, so that an operating system can manage information in each region separately. These regions are called partitions. The disk stores the information about the partitions' locations and size in the partition table. A **partition** is a logical division of a hard disk that is treated as a separate unit by operating systems and file systems. The operating systems and file systems can manage information on each partition as if they were a distinct drive. You can install different operating systems on the drive, each system using a single partition and the partition file system. There are two types of partitions: primary partition and extended partitions (see Figure 13). Every partition is assigned a unique drive letter. Hard drive partitioning protects files by separating the operating system and program files from user files. This makes it easier to recover a

corrupted file system or operating system. If one partition is corrupted, other file systems may not be affected.

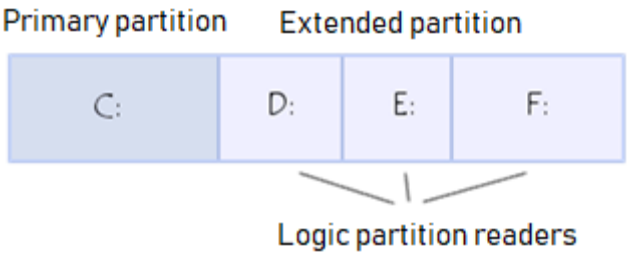
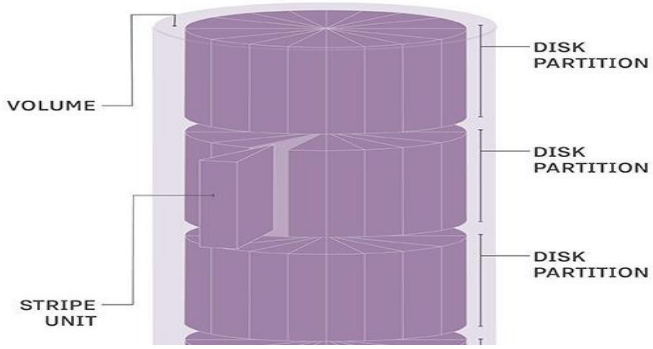


Figure 13: Hard Drive Partitions



II.4 Information Organization in Memory

The information processed by a computer can be of different types (text, numbers, images, audios, videos, etc.) but they are always represented and manipulated by the computer in binary form. Any information will be treated as a sequence of 0 and 1 in order to be processed by the electronic components of the computer (0 corresponds to electrical value off and 1 to on).

A **bit** is a BInary digiT. Bits are the fundamental unit of data storage in a computer. A bit can hold only one of two values 0 or 1 sometimes refer to false or true.

A **byte** = 8 bits. Because bits are so small, they are assembled into a group of eight to form a byte.

A **word** is piece of data of fixed size handled as a unit. It is composed of bits. The word size is the number of bits processed by the CPU at a time. Originally, word meant 16 bits. With modern computers word size can be 32 bits or 64 bits.

Larger units of information storage can be used like kilobyte (KiB or KB), megabyte (MiB, gigabyte (GiB), TeraByte (TB), etc. Usually, memory capacity is measured in MB and GB.

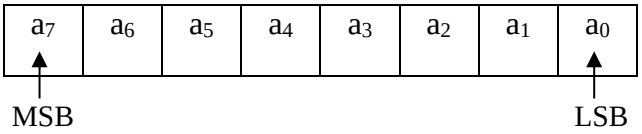
How to convert from a unit to another is shown below:

- 1 Byte = 8 bits.
- 1 Kilobyte = 1 KB = 2^{10} B = 1,024 Byte.
- 1 Megabyte = 1 MB = 2^{10} KB = 2^{20} Byte.
- 1 Gigabyte = 1 GB = 2^{10} MB = 2^{30} Byte.
- 1 Terabyte = 1 TB = 2^{10} GB = 2^{40} Byte.
- 1 Petabyte = 1 PB = 2^{10} TB = 2^{50} Byte, 1 Exabyte = 1 EB = 2^{10} PB
- 1 Zettabyte = 1 ZB = 2^{10} EB, 1 Yottabyte = 1 YB = 2^{10} ZB.

II.5 Information Coding

Information coding in computer is the process of converting information such as letter, word, sound, image, and video into binary form or representation in order to be processed. It consists of establishing a correspondence between the information external representation and the internal one in the computer, which is a sequence of bits.

Example: the code of the letter 'A' is 65 in decimal (= 0100 0001 in binary). We can represent large numbers by grouping series of bits. Two of these groupings are of importance Nibble (a nibble is a group of four bits) and Byte (a byte is a group of eight bits). The right most bit in a Byte is called least significant bit (LSB) and the left-most bit is called most significant bit (MSB). The MSB can also correspond to the sign bit of a signed binary number.



Character coding

A character could be a capital letter, a small letter, a digit, a symbol ($, ?, !, …) and initially was coded on 7 bits using the ASCII (American Standard Code for Information Interchange) code. The **ASCII code** can represent 2^7 = 128 characters. The Figure 14 shows the content of the ASCII code table. For example, the code of 'A' = 65, 'a'= 97, and '0' = 48 in decimal. The ASCII code was developed for the English language; so it does not contain accented characters. It was extended to 8 bits (1 byte) to code more characters and called extended ASCII or ANSI. The **ANSI code** contains 2^8 = 256 characters like à, È, ë, and Ç. Unicode is another character coding system proposed in 1991. It uses 16 bits to code a character, so 2^{16} characters can be coded using Unicode. The Unicode contains Chinese letters, Arabic letters, etc. For more information about Unicode, you can consult the http://www.unicode.org site.

MSB / LSB		0 000	1 001	2 010	3 011	4 100	5 101	6 110	7 111
0	0000	NUL	DLE	SP	0	@	P	`	p
1	0001	SOH	DC1	!	1	A	Q	a	q
2	0010	STX	DC2	"	2	B	R	b	r
3	0011	ETX	DC3	#	3	C	S	c	s
4	0100	EOT	DC4	$	4	D	T	d	t
5	0101	ENQ	NAK	%	5	E	U	e	u
6	0110	ACK	SYN	&	6	F	V	f	v
7	0111	BEL	ETB	'	7	G	W	g	w
8	1000	BS	CAN	(	8	H	X	h	x
9	1001	HT	EM	)	9	I	Y	i	y
A	1010	LF	SUB	*	:	J	Z	j	z
B	1011	VT	ESC	+	;	K	[	k	{
C	1100	FF	FS	,	<	L	\	l	|
D	1101	CR	GS	-	=	M	]	m	}
E	1110	SO	RS	.	>	N	^	n	~
F	1111	SI	US	/	?	O	_	o	DEL

Figure 14: ASCII Code Table

Figure 15: ANSI Code Tables

Dec	Hex	Name	Char	Ctrl-char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	0	Null	NUL	CTRL-@	32	20	Space	64	40	@	96	60	`
1	1	Start of heading	SOH	CTRL-A	33	21	!	65	41	A	97	61	a
2	2	Start of text	STX	CTRL-B	34	22	"	66	42	B	98	62	b
3	3	End of text	ETX	CTRL-C	35	23	#	67	43	C	99	63	c
4	4	End of xmit	EOT	CTRL-D	36	24	$	68	44	D	100	64	d
5	5	Enquiry	ENQ	CTRL-E	37	25	%	69	45	E	101	65	e
6	6	Acknowledge	ACK	CTRL-F	38	26	&	70	46	F	102	66	f
7	7	Bell	BEL	CTRL-G	39	27	'	71	47	G	103	67	g
8	8	Backspace	BS	CTRL-H	40	28	(	72	48	H	104	68	h
9	9	Horizontal tab	HT	CTRL-I	41	29	)	73	49	I	105	69	i
10	0A	Line feed	LF	CTRL-J	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	VT	CTRL-K	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	FF	CTRL-L	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage feed	CR	CTRL-M	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	SO	CTRL-N	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	SI	CTRL-O	47	2F	/	79	4F	O	111	6F	o
16	10	Data line escape	DLE	CTRL-P	48	30	0	80	50	P	112	70	p
17	11	Device control 1	DC1	CTRL-Q	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	DC2	CTRL-R	50	32	2	82	52	R	114	72	r
19	13	Device control 3	DC3	CTRL-S	51	33	3	83	53	S	115	73	s
20	14	Device control 4	DC4	CTRL-T	52	34	4	84	54	T	116	74	t
21	15	Neg acknowledge	NAK	CTRL-U	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	SYN	CTRL-V	54	36	6	86	56	V	118	76	v
23	17	End of xmit block	ETB	CTRL-W	55	37	7	87	57	W	119	77	w
24	18	Cancel	CAN	CTRL-X	56	38	8	88	58	X	120	78	x
25	19	End of medium	EM	CTRL-Y	57	39	9	89	59	Y	121	79	y
26	1A	Substitute	SUB	CTRL-Z	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	ESC	CTRL-[	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	FS	CTRL-\	60	3C	<	92	5C	\	124	7C	|
29	1D	Group separator	GS	CTRL-]	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	RS	CTRL-^	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	US	CTRL-_	63	3F	?	95	5F	_	127	7F	DEL

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
128	80	Ç	160	A0	á	192	C0	└	224	E0	α
129	81	ü	161	A1	í	193	C1	┴	225	E1	ß
130	82	é	162	A2	ó	194	C2	┬	226	E2	Γ
131	83	â	163	A3	ú	195	C3	├	227	E3	π
132	84	ä	164	A4	ñ	196	C4	─	228	E4	Σ
133	85	à	165	A5	Ñ	197	C5	┼	229	E5	σ
134	86	å	166	A6	ª	198	C6	╞	230	E6	µ
135	87	ç	167	A7	º	199	C7	╟	231	E7	τ
136	88	ê	168	A8	¿	200	C8	╚	232	E8	Φ
137	89	ë	169	A9	⌐	201	C9	╔	233	E9	Θ
138	8A	è	170	AA	¬	202	CA	╩	234	EA	Ω
139	8B	ï	171	AB	½	203	CB	╦	235	EB	δ
140	8C	î	172	AC	¼	204	CC	╠	236	EC	∞
141	8D	ì	173	AD	¡	205	CD	═	237	ED	φ
142	8E	Ä	174	AE	«	206	CE	╬	238	EE	ε
143	8F	Å	175	AF	»	207	CF	╧	239	EF	∩
144	90	É	176	B0	░	208	D0	╨	240	F0	≡
145	91	æ	177	B1	▒	209	D1	╤	241	F1	±
146	92	Æ	178	B2	▓	210	D2	╥	242	F2	≥
147	93	ô	179	B3	│	211	D3	╙	243	F3	≤
148	94	ö	180	B4	┤	212	D4	╘	244	F4	⌠
149	95	ò	181	B5	╡	213	D5	╒	245	F5	⌡
150	96	û	182	B6	╢	214	D6	╓	246	F6	÷
151	97	ù	183	B7	╖	215	D7	╫	247	F7	≈
152	98	ÿ	184	B8	╕	216	D8	╪	248	F8	°
153	99	Ö	185	B9	╣	217	D9	┘	249	F9	∙
154	9A	Ü	186	BA	║	218	DA	┌	250	FA	·
155	9B	¢	187	BB	╗	219	DB	█	251	FB	√
156	9C	£	188	BC	╝	220	DC	▄	252	FC	ⁿ
157	9D	¥	189	BD	╜	221	DD	▌	253	FD	²
158	9E	Pts	190	BE	╛	222	DE	▐	254	FE	■
159	9F	ƒ	191	BF	┐	223	DF	▀	255	FF	

II.6 Numeral system (or system of numeration) and convert methods

A numeral system is a writing system for expressing numbers; that is a mathematical notation for representing numbers of a given set, using digits or other symbols in a consistent manner.

Decimal base

"Decem" (latin) means ten. Decimal means base 10. In any numeral system, given the base (often referred to as radix), the number of digits that can be used to count is fixed. For example, in the base 10 numeral system, the digits that can be used to count are 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9.

Example: consider the number 1234. It can be represented as

$1\times10^3 + 2\times10^2 + 3\times10^1 + 4\times10^0$ (positional system notation)

Binary base

"Binarius" (latin) means two. Binary means base 2. The digits that can be used to count in this number system are 0 and 1. The 0, 1 used in the binary system are called binary digits (bits). The bit is the smallest piece of information that can be stored in a computer. It can have one of two values 0 or 1. Think of a bit as a switch that can be either ON or OFF. Since only two values can be stored in a bit, we combine a series of bits to represent more information.

Example: consider the binary number $(1101)_2$. It can be represented as

$1\times2^3 + 1\times2^2 + 0\times2^1 + 1\times2^0$

This expanded notation also gives you the means of converting binary numbers directly into the equivalent decimal number = 8 + 4 + 0 + 1 = 13.

Example: consider the binary number 1101.101. It can be represented as:

$1\times2^3 + 1\times2^2 + 0\times2^1 + 1\times2^0 + 1\times2^{-1} + 0\times2^{-2} + 1\times2^{-3} = 13 + 0.5 + 0 + 0.125 = 13.625$

To convert from **binary to decimal**: expand using positional notation

$100101_2 = 1\times2^5 + 0\times2^4 + 0\times2^3 + 1\times2^2 + 0\times2^1 + 1\times2^0 = 32 + 0 + 0 + 1 + 0 + 1 = 37_{10}$

To convert from **decimal to binary**:

First method: determine largest power of 2 ≤ number; write template

$37 = ?\times2^5 + ?\times2^4 + ?\times2^3 + ?\times2^2 + ?\times2^1 + ?\times2^0$

$37 - 2 = 5$; $5 - 4 = 1$; $1 - 1 = 0$

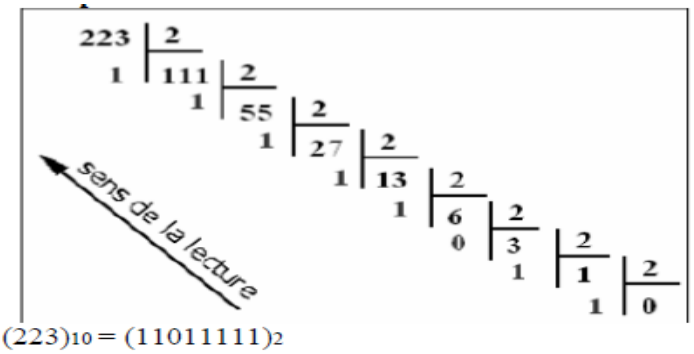
$37_{10} = 1\times2^5 + 0\times2^4 + 0\times2^3 + 1\times2^2 + 0\times2^1 + 1\times2^0 = 100101_2$

Second method: repeatedly divide by 2, consider remainder

37 / 2 = 18 R 1 (LSB) ↑
18 / 2 = 9 R 0
9 / 2 = 4 R 1
4 / 2 = 2 R 0
2 / 2 = 1 R 0
1 / 2 = 0 R 1 (MSB)

Read from bottom to top => 100101_2

Example: convert the decimal number 223 to binary.



$(223)_{10} = (11011111)_2$

Octal number system

"Octo" (latin) means eight. Octal is base 8 ($8 = 2^3$ digits, which means 3 bits per digit). To avoid writing down long binary words, it is often easier to use larger base systems. Two commonly used systems are octal and hexadecimal. The octal number system is base eight, i.e. values can be represented using an 8-symbol dictionary: 0-7.

To convert from **binary to octal**, binary numbers are grouped on 3-bits words such that: $000_2 = 0_8$, $001_2 = 1_8$, $011_2 = 3_8$, $100_2 = 4_8$, $101_2 = 5_8$, $110_2 = 6_8$, and $111_2 = 7_8$.

$001010000100111101_2 = 120475_8$ digit count in binary number not a multiple of 3 => pad with zeros on left.

To convert from **octal to binary**: $120475_8 = 001|010|000|100|111|101_2 = 001010000100111101_2$ discard leading zeros from binary number if appropriate.

To convert from **octal to decimal**: $24_8 = 2\times8^1 + 4\times8^0 = 20_{10}$.

To convert from **decimal to octal** (and from there to binary), repeat divide by 8:
20 / 8 = 2 remainder 4 (LSB) ↑
2 / 8 = 0 remainder 2 (MSB) Read from bottom to top => $20_{10} = 24_8$.

Example: convert from octal to decimal
$256.1_8 = 2\times8^2 + 5\times8^1 + 6\times8^0 + 1\times8^{-1} = 128 + 40 + 6 + 0.125 = 174.125_{10}$.

Hexadecimal number system

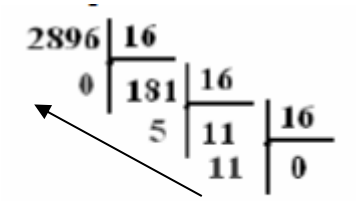
"Hexa" (Greek) => six and "decem" (Latin) => ten. Hexadecimal is base 16 ($16 = 2^4$ digits so 4 bits per digit). It is the most popular large-base system for representing binary numbers. Each symbol represents 4-bits (1 nibble), that can take one of 16 different values: the values 0-9 are represented by the digits 0-9, and the values 10-15 are represented by the capital letters A-F.

Bin	Hex	Dec	Bin	Hex	Dec	Bin	Hex	Dec	Bin	Hex	Dec
0000	0	0	0100	4	4	1000	8	8	1100	C	12
0001	1	1	0101	5	5	1001	9	9	1101	D	13
0010	2	2	0110	6	6	1010	A	10	1110	E	14
0011	3	3	0111	7	7	1011	B	11	1111	F	15

To convert from hexadecimal to decimal: expand using positional notation
$25_{16} = 2\times16^1 + 5\times16^0 = 32 + 5 = 37_{10}$

To convert from decimal to hexadecimal: use the division by 16 method
37 / 16 = 2 remainder 5
2 / 16 = 0 remainder 2 ↑ Read from bottom to top => $37_{10} = \overrightarrow{25_{16}}$

Example: convert $(2896)_{10}$ to hexadecimal



(2896)10 = (B50)16 (11 correspond à la lettre B).

Note: To convert from decimal to any base N, you can use successive divisions and consider the remainder from bottom to top. To convert from base N to hexadecimal, you can use the positional notation with radix N. Examples: $(7)_{10} = (12)_5$ 7 / 5 = 1 remainder 2 1 / 5 = 0 remainder 1 $(41)_6 = 4\times6^1 + 1\times6^0 = 24 + 1 = (25)_{10}$.

Signed numbers representation

Signed number representations are required to encode negative number in binary number systems. If the number is coded on n bits, the first bit is used for sign (0 = +, 1 = -). Two forms can be used: The one's complement form for a positive number in binary remains unchanged. The one's complement of a negative binary number is the complement of its positive counterpart, all we need to do is to change each bit in turn, 1 to 0 and vice versa.

Example: -5_{10} = ?1'c |-5| = 5 = 00000101 (on 8 bits) => one's complement = 11111010.

To convert from one's complement to decimal: we first check if the binary number is positive or negative by looking at the sign at the MSB. If MSB = 1, we write minus (-) then we invert the bits, and convert it to decimal and the result is (-) number in decimal.

The two's complement form is also used to represent signed numbers. The 2'c of a positive number remains unchanged. The 2'c of a negative binary number is = 1'c + 1.

Examples: -5_{10} = ?2'c, 1'c = 11111010 => 2'c = 111110101'c + 1 = 111110112'c.
$+7_{10}$ = ?2'c, 7 in binary on 8 bits = 000001112 since it's positive=> 2'c =000001112.

To convert from two's complement to decimal, we first check the sign bit, if it's 0 (+) then it is the same as binary. If it's 1 (-), we convert it into one's complement and then we add 1.

Examples: $\boxed{0}0001001_{2'c} = 00001001_{2}(\text{binary}) = 9_{10}$
$\boxed{1}0001010_{2'c} = -(01110101_{1'c} + 1) = -\ 01110110_{2} = -118_{10}$

EXERCISES

Exercise 1: Conversion between bases

a) $(123)_{10} = (\ ?)_{2}$ b) $(1000111)_{2} = (\ ?)_{10}$ c) $(3C)_{16} = (\ ?)_{10}$ d) $(70)_{8} = (\ ?)_{2}$

Exercise 2: Convert in binary the following numbers by coding those using 4 bits then 8 bits:

a) $(11)_{10} = (\ ?)_{2}$ b) $(17)_{10} = (\ ?)_{2}$ c) $(125)_{10} = (\ ?)_{2}$ d) $(268)_{10} = (\ ?)_{2}$

Exercise 3:
3.1) Convert from decimal to two's complement on 8 bits the following signed numbers:

a) $(+8)_{10} = (\ ?)_{2'c}$ b) $(-8)_{10} = (\ ?)_{2'c}$ c) $(-22)_{10} = (\ ?)_{2'c}$ d) $(-5)_{10} = (\ ?)_{2'c}$

3.2) Convert from two's complement to decimal on 8 bits the following signed numbers:

e) $(11101111)_{2'c} = (?)_{10}$ f) $(00011001)_{2'c} = (?)_{10}$ g) $(11110101)_{2'c} = (?)_{10}$ h) $(11111110)_{2'c} = (?)_{10}$

Chapter III: Operating System

Definition and role

An *operating system* is system software (or a set of programs) that manages computer hardware and software resources and provides common services for computer programs. Application programs usually require an operating system to function. For hardware functions such as input and output and memory allocation, the operating system acts as an intermediary between programs and the computer hardware.

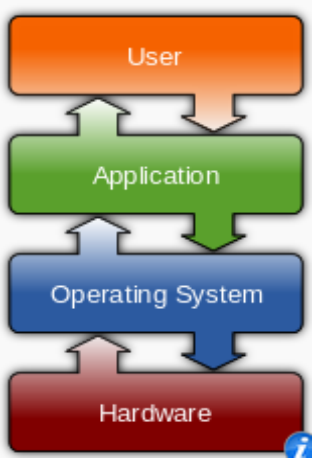


Figure 4: Logical layers of a computer

The major functions of the operating systems are:

- The processor management that performs the process scheduling operations (task in execution).
- The management of the process update operations as well as their synchronization and the communication between them.
- The main memory management, which essentially consists in carrying out the memory allocation operations and the follow-up of the memory occupation.
- The management of the secondary memories.
- The Input/output management that transfers the information between the CPU, the Input/Output, and the network.
- The management of networks.
- The management of the users' commands.

The most popular operating systems are Windows, Linux, Mac OS, Android, and iOS.

The components of an operating system (structure)

The operating system consists of a set of applications to manage the interactions with the hardware (see Figure 6). Among this set of software we generally distinguish the following components:

- The **kernel** is a program that constitutes the central core of an operating system. It manages the computer resources (CPU, memory, Input/output) and enables the communication between the hardware and software. The kernel is responsible for all major activities of an operating system. Its main function is the process execution, and managing the input/output, the memory, and files.
- A **command interpreter (shell)** allows the communication with the operating system via a command language in order to let the user to control the input/output units.
- **File system** controls how data is stored and retrieved and allows you to save files in a tree.

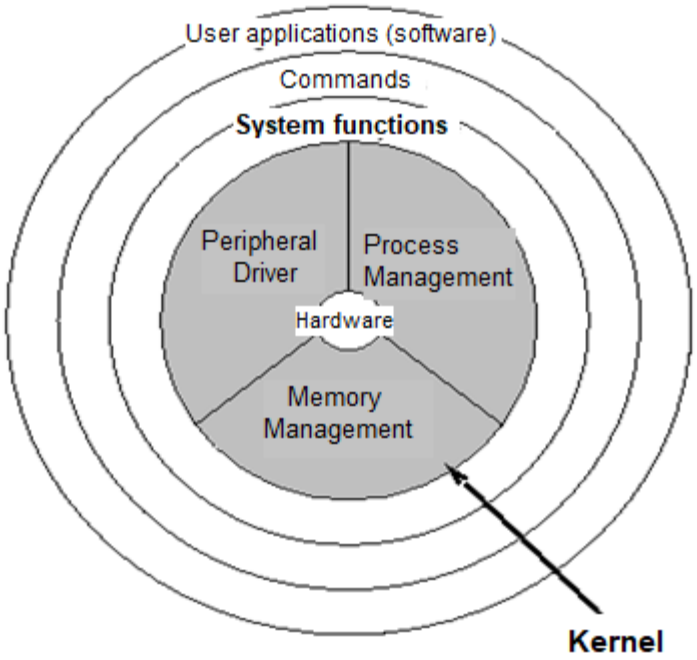


Figure 5: Operating System Structure

Particular characteristics of an operation system

An operating system is called **multitasking** (or multithreaded) when it can perform more than one computer task (also called process) at a time.

Multitasking is a technique that consists of running multiple processes concurrently in parallel. These processes share common processing resources such as a CPU. In the case of a computer with single CPU, only one job can be processed at a time. The processes are executed one after another through a single CPU by time sharing. In this case, a scheduler schedules and decides which task should be the

running task and when a waiting task should get turn. Multitasking makes the best possible use of available hardware at any given instance of time and improves the overall efficiency of computer system.

Embedded systems are operating systems with a dedicated function intended to operate on small machines (PDA, digital watches, and MP3 players) or embedded as part of a complete device often including hardware and mechanical parts (traffic lights, factory controllers, hybrid vehicles, avionics, robots, etc.).

Booting the computer

At the time of the motherboard construction, a system software called the BIOS is stored in the ROM (Read Only Memory). The BIOS is sometimes called firmware. Before the 1990s, the BIOS was stored on ROM chips that could not be modified. As their complexity increased and the need to update them became necessary, the BIOSes were stored on EEPROM or flash memories that could be modified. The update operation of the BIOS is called "Flashing the BIOS".

When you turn on the power to a computer, the first program that runs is the BIOS. It examines the system hardware to make sure that everything is functioning properly. The BIOS detects the hard disk, configures the components, and searches for the booting unit. This unit is the part where to look for an operating system. In most cases, the operating system is in the master boot record (MBR) of the hard disk. Finally, the BIOS loads the operating system into the RAM memory and allows it to begin operating. The memory will hold the operating system, user information and applications.

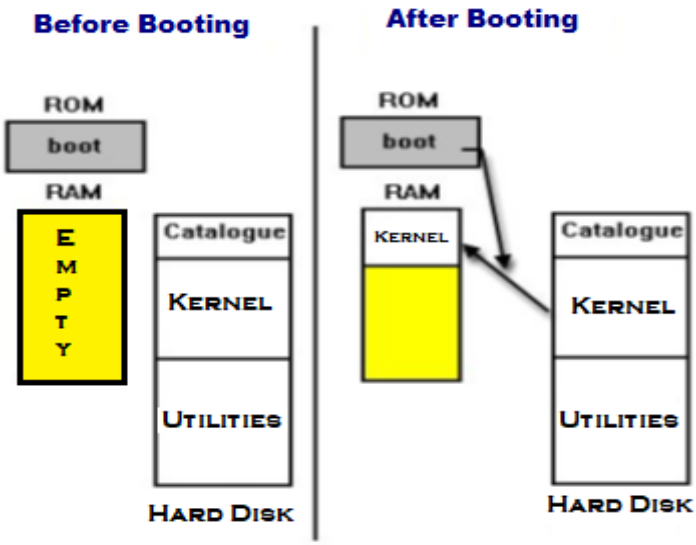


Figure 6: Computer Booting

Major functions of an operating system

The main purpose of an operating system is to manage the components of the computer. There are therefore four tasks that correspond to the management of the hard disk, the central memory, the processor, the input/output units, and the network.

The second memory management (Hard disk)

The hard disk is an auxiliary support that aims to permanently store the programs running in the main memory (the latter being volatile). The programs are stored as files on the hard disk and organized into directories. The second memory management includes the management of files via a file management system. The file system offers operations to create, copy, read, delete these files. It also manages the space occupied by the file as well as the free space. Finally, the file system supports the share and the protection of the files in a multi-user environment.

The main memory management

The main memory (RAM) is a large space organized into words (set of bytes) and used to store the data to be processed by the CPU. The CPU loads the instructions to be executed in the CPU registers from addresses in the main memory. Similarly after execution, the results are stored in the main memory. This involves managing the allocation of this memory to the programs (allocation and freeing memory), the addressing rules, and ensuring that the programs in the memory do not interfere with each other.

The processor management

The main task of an operating system is the allocation of the processor to the processes (running programs). It decides which process is running at a given time, when to stop the process, what process will be running next, and a process needs as resources for its execution. The operating system must handle two types of processes: its own and those of the users. It also provides primitives for the communication between processes and their synchronization.

The input/output management

The operating system takes in charge the transfer of the information between the CPU, the input/output (keyboard, screen, mouse, etc.) through drivers, and the network. When a running process requests a data from an input/output unit, the operating system interrupts this process and allocates the CPU to another process. The operating system also ensures an effective sharing of resources between programs; each application gets the necessary resources it needs. This sharing maximizes the functionality of the overall system.

Network management

Modern operating systems incorporate features for network management. These systems offer user file sharing, file protection, identification of users and machines connected to the network.

File System Management (FSM)

The main tasks of a file system management are:

- Store files on a disk (distributing them in different blocks)
- Organize and structure hierarchically all the files (directories)
- Allow to create, modify or delete a file.
- Allow to change the hierarchy of files.
- Manage the hard disk free space.
- Manage files in a multi-user environment.

The file concept

A file is a logical storage unit made available to users for storing their data: it is the allocation unit. The operating system establishes a correspondence between the file and the binary system used during the storage in a transparent way for users. In a file, you can write text, programs, add images, and do calculations. Files are usually created by users but sometimes they are generated by the system or some tools. In order to differentiate between files, each file has a set of attributes that describes it. These attributes include name, extension, date and creation time or last modification time, size, and protection type (see Figure 9). Some attributes are specified by the user, others are completed by the system. From the file system's point of view, the file is a collection of blocks on the hard disk (see Figure 8).

Figure 8: Some File Attributes

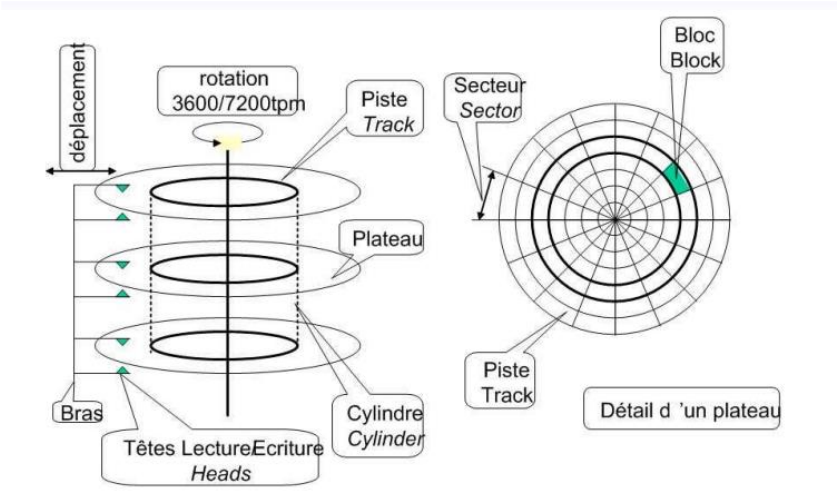


Figure 7 : Hard Disk Structure

The directory concept

A directory (see Figure 10) is an entity created for the organization of files. To be found, each file must have a unique name. With the multitude of files created, the operating systems needs an organization to structure these files and to be able to access them quickly. This organization is made via directories also called folders. Nowadays, all the file systems adopt a hierarchal structure.

Figure 9: Directory

The structure of folders and files under Windows (Tree)

As Figure 10 shows, Windows automatically creates on the main hard disk: (1) multiple directories, (2) subdirectories, (3) also names with a letter the hard disks and the external connected memories.

Figure 10: Tree of Directories in Windows

The hard disk is divided into folders:

- The programs folder (contains Words, Excel, OpenOffice, …),
- The Windows folder (contains Paint, Solitaire, …),
- The Users folder (2): contains a folder per user name.
- (3) Each user folder is also divided and contains the following folders: (4) Desktop, Documents, Images, Music, Downloads, Videos, Favorites, etc…

Windows allows you to create your own folders in which you will save your documents or files that have the same theme. For example, You can create in the Documents folder the subfolders Course, Private, and House. The subfolder Course can be divided into two folders: Computer and Painting.

You can also create in the subfolder Private the folders: Mail, Children, and Hobbies. This hierarchy of directories is called a directory tree structure. It can be represented horizontally or vertically. **NB**: The access path of a document takes into account the tree structure where it is located.

MS-DOS (Microsoft disk operating system)

The operating system allows the user to execute commands (often to launch an application). A user can communicate with the operating system (the Kernel functions) either via a command language interface or via a graphical user interface (GUI).

MS-DOS (Microsoft Disk Operating System) is an operating system developed by Microsoft operating in single-tasking and single-user mode. It is equipped with a command-line interface. The table below presents some basic commands of MS-DOS:

Command	Description
cd directory_name	Change the current directory
copy source_file destination_file	Copy a file to a destination
del file_name	Delete a file
mkdir, md directory_name	Create a directory
rmdir, rd directory_name	Delete a directory
type nul > file_name	Create a file
move file_name destination	Move a file to a specific location

Examples of MS-DOS commands:

1. cd C:\Users cd .. : go to the parent directory ⇔ cd C:\
2. copy C:\Users\A C:\Users\B : copy the content of the directory A into B
3. copy C:\Users\A\MyFile.txt C:\Users
4. del C:\Users\MyFile.txt
5. del C:\Users\A: delete the content of the directory A
6. mkdir C:\Users\REP rmdir C:\Users\REP
7. move C:\Users\MyFile.txt C:\Users\A : copy a text file into the directory A

Use Case: Windows (On Computer)

File and directory management (creation, delete, etc.)

User management

Software management

Exercices

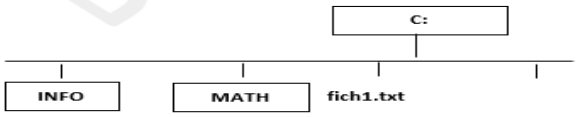
Exercise 1:

Use the MS-DOS commands to create a directory R1 in the C partition and two text files F1.txt and F2.txt in R1. Then create a directory R2 in the D partition, and copy all the files of R1 into R2.

Exercise 2:

Write the MS-DOS commands that allow to:

1. Create two directories INFO and MATH, and a file fich.txt in the C partition.
2. Create two subdirectories INFO1 and INFO2 in the INFO directory.
3. Move the file fich.txt into the MATH directory.
4. Delete the directory INFO1 and the file fich.txt.

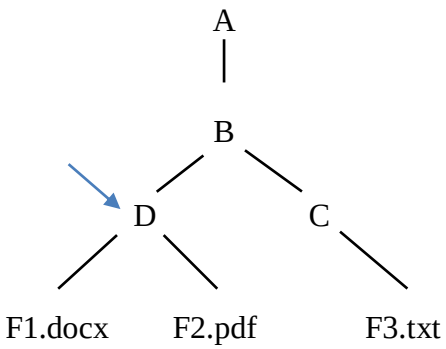


Exercise 3:

The current directory is "C:\R1". Use the MS-DOS to do the following actions: Create two subdirectories R11 and R12 in the directory R1. Create a file F11.txt in the directory R12. Without changing a directory, copy the file F11.txt into the directory R11. Go to the directory R11. Remove the file F11.txt from the directory R12. Finally delete the directory R12.

Exercise 4:

4.1) Use the MS-DOS commands to create the following directory and file tree:



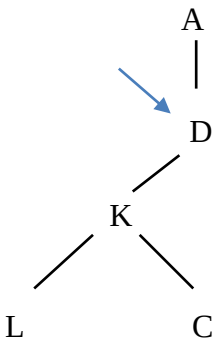
4.2) Make D the current directory, create a file F4.pdf in the directory B.

4.3) Move all the files in the directory D into the directory B.

4.4) Delete all the subdirectories in the directory B. I should empty & delete all subdirectories in B.

4.5) Copy the directory B in the directory A

Exercise 5: change current directory



Suppose that the current the directory is D, and given the above directory tree. Use the MS-DOS commands to:

5.1) Go to the parent directory A.

5.2) Go back to the directory D, and then to the directory K.

5.3) Create a subdirectory M in the directory L.

5.4) Create a subdirectory E in the directory D.

5.5) Delete the two directories C and L.

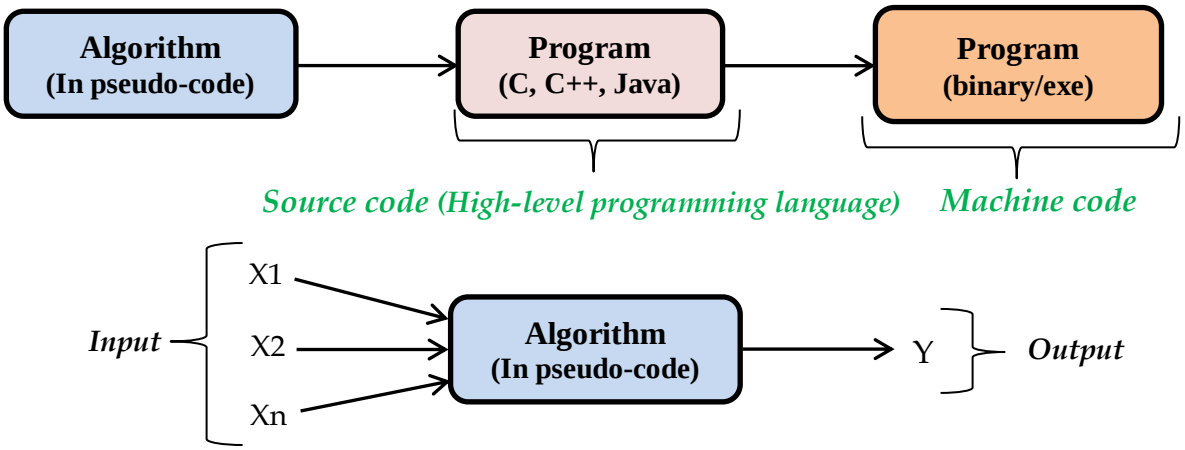
Chapter IV: Software

Software vs. Algorithm

Computer program (also called **application** or **software)** is a set of computer instructions or commands that tell the computer how to work. It consists of one or many files containing textual commands also called instructions. This set of files constitutes the source code of the computer program. Programming or coding means writing the source code of a program. Languages that programmers use to write source code are called high-level programming languages. These languages are close to natural languages (English, French, etc.) and they use special keywords and syntax the equivalent of vocabulary and grammar in a human language. Then, the high-level languages are translated using a compiler or an interpreter or both into a machine language that contains a series of binary codes that are understood directly by a computer's CPU.

An **algorithm** is simply a series of steps you follow when solving a problem. It lists the steps in the order they should be performed. An algorithm has a definite beginning and a definite end, and a finite number of steps. An algorithm produces the same output information given the same input information, and several short algorithms can be combined to perform complex tasks such as writing a computer program. Flowcharts and pseudo-code are two ways to describe algorithms. A flowchart is a diagrammatic description of an algorithm and a pseudo-code is a textual one. A flowchart is convenient for small algorithms.

In order to solve a problem, we usually write an algorithm in pseudo-code then the algorithm will be written in one of the programming languages in order to be executed on a computer. A pseudo-code is an intermediate language between natural language like English, French, etc. and a programming language like C, C++, Java, etc.



Algorithm example: Write an algorithm that takes three numbers from the users, computes and displays their summation.

Algorithm: SUM;
Variables: integer x, y, z, s;
begin
    Write("Enter three numbers:"); // *displays a message on the screen*
    Read(x, y, z) // *takes data from keyboard and stores them in variables in memory*
    s = x + y + z; // = *represents an* ***assignment operator*** *that assigns a value to s*
    Write("The summation is ", s);
end

Program example: Write the previous algorithm in a C programming language.

```
# include <stdio.h>
int main()
{ int x, y, z, s;
  printf("Enter three numbers:");
  scanf("%d %d %d", &x, &y, &z);
  s = x + y + z;
  printf("The result of the summation is %d", s);
return 0;}
```

Software Categories

Computer software can be put into categories based on common function or field of use or copyright status.

Functions

- **Application software** is the general designation of computer programs for performing tasks. It may be general purpose (word processing, web browsers, etc.) or have a specific purpose (accounting, truck scheduling, etc.). Application software contrasts with system software.
- **System software** is a generic term referring to the computer programs used to start and run computer systems including diverse application software and networks.
- **Computer programming** tools such as compilers and linkers are used to translate and combine computer program source code and libraries into executable RAMs.

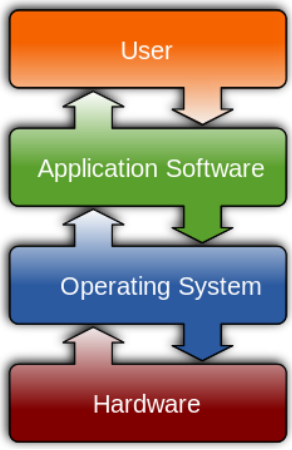


Figure 11: Application Software & Operating System

Copyright status

Software can be categorized according to its license type (copyright status):

1. **Open source software** is software with its source code made available under a certain license. The source code is open and can be modified as required. In general, it is gratis but sometimes it comes with a fee that depends on its license. Example: Linux operating system.
2. **Freeware** is software available for download and distribution without initial payment. It is cost free and does not provide its source code. It is copyrighted, so other people cannot market the software as their own. Examples: minor program updates and small games like Adobe PDF, and Yahoo messenger. Shareware software is free during a test period (often 1 month) and you have to purchase a license afterwards. Examples: WinZip, Cuteftp, and GetRight.
3. **Commercial open source software** is software produced for sale or that serves commercial purposes. Its source code is available and you have to pay a fee to use it. Examples: Netbeans, and MySQL Enterprise.
4. **Proprietary software** is non-free software for which the software's publisher retains intellectual property rights usually copyright of the source. So the source code is not provided. Examples: Microsoft Windows, Adobe Flash Player, PS3 OS, iTunes, Adobe Photoshop, Google Earth, macOS, Skype, and WinRAR.

	Available Code Source	**Unavailable Code Source**
Free	1. Open Source Free	2. Freeware Shareware
Pay	3. Commercial Open Source	4. Proprietary Software

Algorithm Flowcharts and Pseudo-code

This section presents the symbols of a flowchart and their equivalence in pseudo-code used to describe an algorithm.

Definitions

A **flowchart** is the graphical or pictorial representation of an algorithm with the help of different symbols, shapes and arrows in order to demonstrate a process or a program. It details the sequence of operations and decisions required to create output.

Pseudo-code is a textual and an informal way to describe an algorithm with a simple vocabulary. It does not require a prior knowledge of programming language syntax or underlying technology considerations. The pseudo-code is not an actual programming language, so it cannot be compiled into an executable program. You can write pseudo-code on a simple sheet of paper without the use of a computer.

Variable definition

A variable is a storage location (memory location). In a program, it may be necessary to temporarily store certain values (data). Values are stored in variables (memory locations). In other words, a variable is a means of referring to a memory location used in a computer program. This memory location contains values that could be text, number or complex data type (see Figure 13).

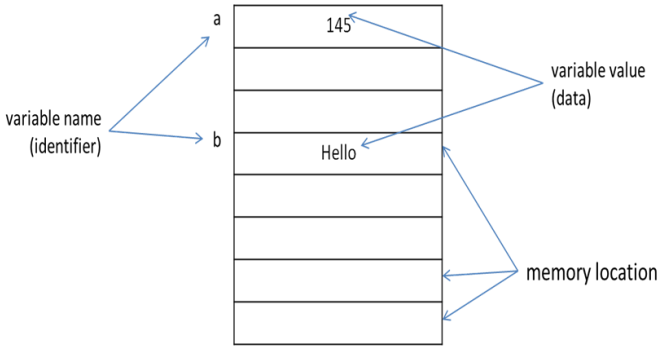


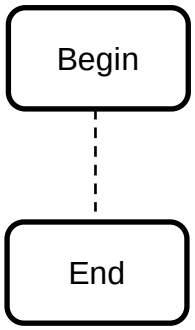
Figure 12: Variables in memory

Flowchart symbols

In this section, we will present the symbols used in a flowchart and their equivalents in a pseudo-code.

Begin and End

Flowchart	Pseudo-Code

	`Begin algorithm MY_ALGO`
	`...`
	`End algorithm MY_ALGO`

Input and Output

Flowchart	Pseudo-Code
	`Read...` `Write...`

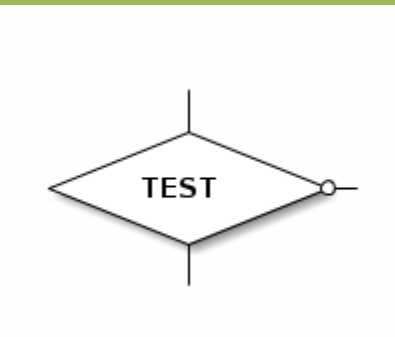
In an algorithm, we use two operations **Read()** and **Write()** to take a data from the user entered using an input like the keyboard and to display some data on the output like the screen. Consider x as number.

- ***Read(x)***: the Read operation takes some value entered by the user on the keyboard and puts the value in the variable x that is stored in memory. Sometimes READ or GET are also used.
- ***Write("Message")***: the Write operation displays Message on the screen. Sometimes PRINT is also used.
- ***Write(x)***: the Write operation displays the value of x on the screen.
- ***Write("Message", x)***: the Write operation displays Message followed by the value of x.

Example: int x = 7; Write(x, "is a positive number."); => 7 is a positive number.

Test: conditional statement (if ...else ...)

Flowchart	Pseudo-Code

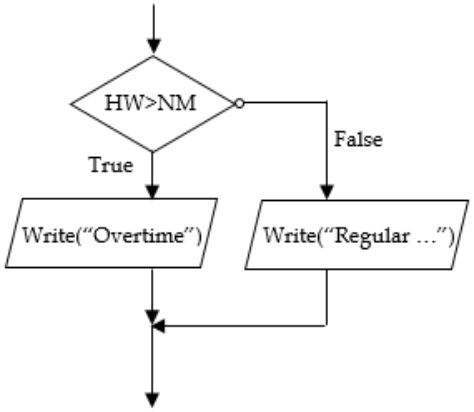
	`IF TEST THEN`
	`  \  ...`
	`ELSE`
	`  \  ...`
	`ENDIF`

In a conditional statement you make a test. The result of the test is a Boolean – either True or False. If the result of the test is True you take a certain course of action and if the result of the test is False you take another course of action. The circle that precedes a branch indicates the else sequence.

In pseudo-code:

IF condition **THEN**
  sequence1 // is a sequence of instructions or actions
ELSE
  sequence2
ENDIF

If the condition is True sequence1 is executed, otherwise sequence2 is executed. The ELSE sequence is optional.



Example1: **IF** HoursWorked > NormalMax **THEN**
  WRITE("Overtime");
ELSE
  WRITE("Regular time");
ENDIF

Action/Operation/Statement: Processing

Flowchart	Pseudo-Code
	`Statement...`

Any elementary action, operation, or statement is represented using a rectangle. An important operator is the assignment operator (=) which assigns the value of its right operand to its left operand. Example: x = 3 assigns the value of 3 to x. Other operators are presented in the tables below.

Arithmetic Operator	***Name***	***Example***
+	Addition	x=4; y=2; z = x + y; // gives 6
-	Subtraction	a=3.5; b = 0.5; c = a –b; // gives 3.0
*	Multiplication	X=2; Y= 2; Z = X*X*Y; // gives 8
/	Division	a = 5.0; b = 2; c = a/b; // gives 2.5
%	Remainder	a = 5; b = 2; c = a%b; // gives 1

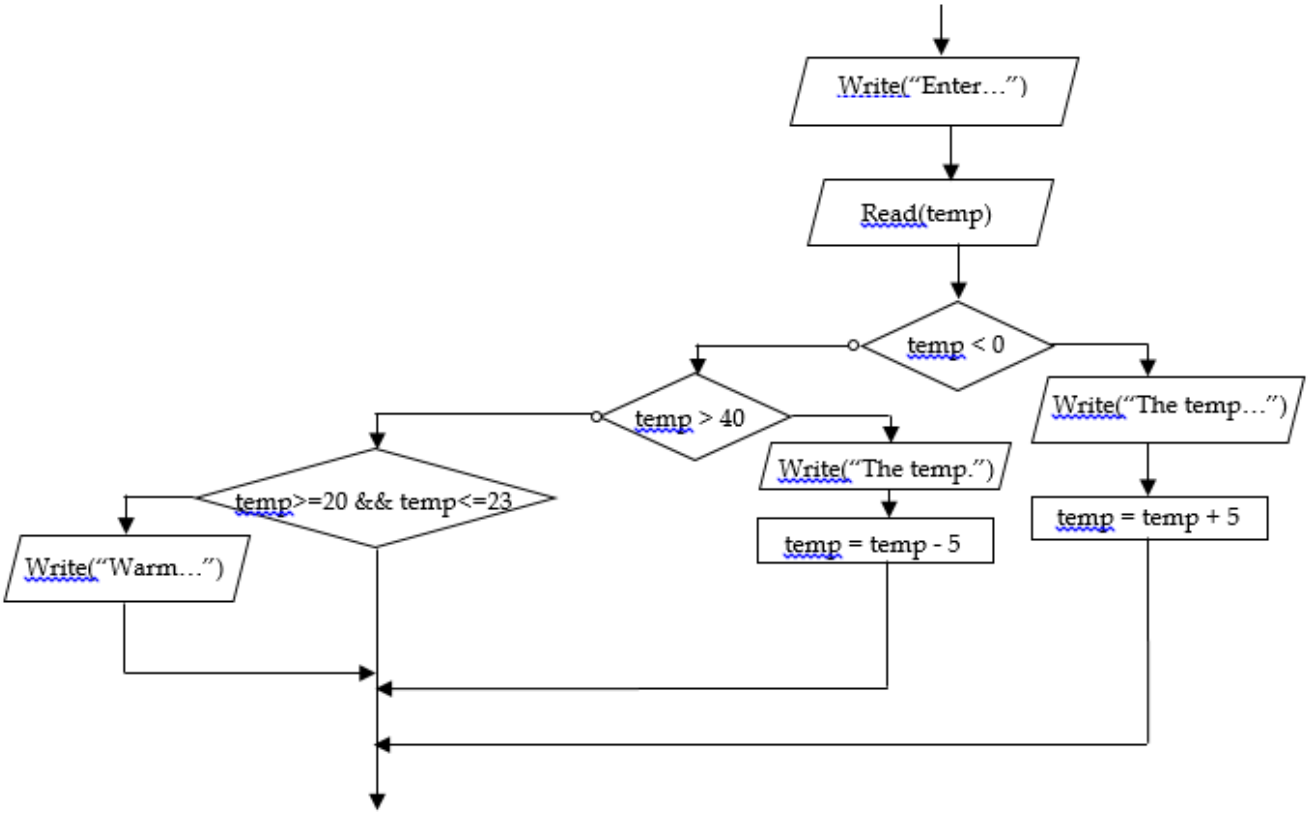
Relational Operator	***Name***	***Example***
==	Equality	x=4; y = 2 ; x == y // gives 0 = false
!=	Inequality	x = 4; y = 2; x != y // gives 1 = true
<	Less Than	a = 5; b = 1; a < b // gives 0
<=	Less Than or Equal	a =5; b = 5; a <= b; // gives 1
>	Greater	X = 1; Y = 0; X>Y // gives 1
>=	Greater or Equal	X= 1; Y = 0; X>= Y+2 // gives 0

Logical Operator	***Name***	***Example***
NOT	Logical Negation	x=4; NOT(x) // gives 0 = false
AND	Logical And	a=3 ; b = 2; (a==3) AND (b <=2) // gives 1
OR	Logical Or	X=2; Y=8 ; (X != 2) OR (Y==8) // gives 1

Example 2:

```
int temp;
   WRITE("Enter the current temperature");
    READ(temp);
    IF (temp < 0) THEN
        WRITE("The temperature ", temp, " is below zero. It is cold");
```

```
    temp = temp + 5;
ELSE IF (temp >40) THEN
    WRITE("The temperature ", temp, " is over forty. It is very hot");
    temp = temp - 5;
        ELSE IF (temp >=20 AND temp <=25) THEN
                WRITE("Warm temperature, do not change!");
                ENDIF
         ENDIF
 ENDIF
```



EXERCISES

Exercise1: Write an algorithm that takes from the user two integer numbers and finds the largest among these two.

Exercise2: Write an algorithm that takes from the user an integer number x and determines whether x is positive, negative or null.

Exercise3: Write an algorithm that takes from the user three integer numbers a, b, and c and finds the largest number among these three.

Exercise4: Write an algorithm that reads two adjacent sides of a rectangle and calculates its area. In addition, the algorithm determines whether the rectangle is a square or not.

Exercise5: Write an algorithm that takes from the user an amount in Lebanese Pound (LBP), converts it into US Dollar ($) and Euro (€) in this order. Finally, the algorithm displays the following message: amount in LBP = amount in € = amount in $.

Exercise6: Write an algorithm that takes from the user two real numbers *a* and *b* and swaps their value. Example: a = 3.1, b = 7.0 after swapping a = 7.0, b = 3.1.

Exercise7: Write an algorithm that takes from the user three integer numbers x, y, and z and swap their value.

Exercise8: Write an algorithm that stores the character 'I' in a variable and takes from the user a character x, then compares both characters, and displays "TRUE" when they are equal, "FALSE" otherwise.

Exercise9: Write an algorithm that takes from the user an integer number x, then display the value –x if x < 0, the value of x*x if x is in the range of 0 and 1 inclusive, and the value of 2*x if x is greater than 1.

Exercise10: Write an algorithm that takes from the user an integer number x, changes the value of x into its absolute value, and displays its new value at the end.

Exercise11: Write an algorithm that takes from the user an integer x that represents a numerical grade and computes its equivalent letter grade. The algorithm must only use tests with a simple condition (without AND and OR). In the case of x is in [90, 100], the letter 'A' is assigned to Grade. In the case of x is in [80, 90[, 'B' is assigned to Grade. In the case of x is in [70, 80[, 'C' is assigned to Grade. In the case of x is in [60, 70[, 'D' is assigned to Grade. In the case of x is less than 60, 'F' is assigned to Grade. At the end, the value of Grade is displayed on the screen.

Chapter V: Databases

Definitions

Database

A **database** (DB) is a collection of data that is organized so that it can be easily accessed, managed, and updated. Examples of database applications include computerized library systems, flight reservation systems, and many content management systems that store websites as collections of webpages in a database.

Databases offer many benefits such as reducing the amount of time spent adding, deleting, and changing data easily, improving the quality and consistency of information, reducing and controlling of data redundancy, sharing of data and multiusers system, support for multiple views of data, easier to maintain data security, and the ability to recover from crashes and errors.

Database Management System (DBMS)

A **database management system** (DBMS) is a computer software that enables you to create, query, update, and delete databases and managing the content of the databases i.e. to access, create, read or select, update, and delete data referred as CRUD operations and the access to these data in the databases. The SGBD allows also defining some constraints on the data called integrity constraints, defining data access and security (passwords, authorizations), and providing data recovery (backups, logs). The DBMS interacts with the user, other applications and the database itself to capture and analyze data.

Some of the well-known DBMS include Microsoft Access, **MySQL**, PostgreSQL, Microsoft SQL Server, **Oracle**, Sybase and IBM DB2.

Most DBMSs are based on a client-server model (see Figure 14). In this model, the database is stored on a server that is only used for that database and to interact with this database. A software called “client” is installed on a computer interrogates the server and gets the answer from it.

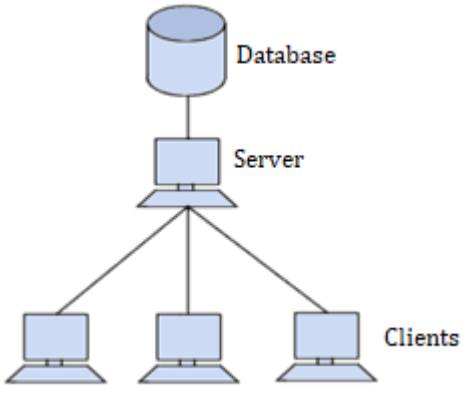


Figure 13: Example of an Architecture containing a Database

There are many types of DBMSs. They range from small systems that run on personal computers to huge systems that run on mainframes (computers that support hundreds of users simultaneously). DBMS are often classified according to the database model that they support; the most popular database systems since the 1980s have all supported the relational model as represented by the Structured Query Language (SQL).

Structured Query Language

Structured Query Language (SQL) is a language that allows communicating with databases (DB). This language is widely used by Web developers to communicate with the data of a Website. It enables selecting, inserting, updating, and deleting data from a database. SQL can also be used to manage DBs, i.e. create, delete or update databases.

SQL SELECT

The SELECT command enables to extract data from a database and returns a set of records in a table. This command can select one or many columns of a table by separating them by a comma (,). The general form of SELECT is as follows:

SELECT column_name
FROM table
WHERE one or many conditions

Example of a database of one table called « Client » that contains information about enterprise clients.

Id	FirstName	LastName	City	Salary
1	Pierre	Dupont	Paris	2,000$
2	Marine	Durant	Nantes	1,000$
3	Pierre	Martin	Marseille	1,500$

SELECT City **FROM** Client;

City
Paris
Nantes
Marseille

SELECT FirstName, LastName **FROM** Client;

FirstName	LastName
Pierre	Dupont
Marine	Durant
Pierre	Martin

In order to select all the columns of a table, we write:

SELECT * **FROM** Client;

To select different values from a column, we use the DISTINCT keyword:

SELECT DISTINCT FirstName **FROM** Client;

FirstName
Pierre
Marine

Find all the clients that live in Paris, we use the WHERE clause that filters the rows:

SELECT * FROM Client **WHERE** City = 'Paris';

Fin the maximum salary of clients, we use the MAX() function:

SELECT MAX(Salary) **FROM** Client;

MAX(Salary)
2,000$

We can also use **MIN, AVG, COUNT,** and **SUM** functions as shown next.

SELECT COUNT(*) FROM Client ;

COUNT(*)
3

This query counts the number of records in the Client table and returns the answer = 3.

You can also rename a column in the query result using the AS keyword:

SELECT COUNT(*) AS ClientsNumber **FROM** Client ;

ClientsNumber
3

SELECT COUNT(FirstName) **FROM** Client ;

This query counts the number of clients having a FirstName. The answer here is 3 but it could be different when a client does not have a name in the table. An attribute without a value is considered as NULL value, which means empty.

SELECT COUNT(FirstName) FROM Client **WHERE** FirstName="Mary";

COUNT(FirstName)
NULL

SELECT COUNT(DISTINCT FirstName) **FROM** Client ; The answer is 2

SELECT SUM(Salary) **FROM** Client;
The query sums up the salaries and returns the answer is 4500$

SUM(Salary)
4,500$

SELECT AVG(Salary) **FROM** Client **WHERE** FirstName = 'Pierre';
The query computes the average of salaries for clients having the first name ="Pierre", it is (2,000 + 1,500)/2, and the answer is 1,750$.

There are many comparison operators that can be written in WHERE such as equal =, different <> or **!=**, greater than >, less than <, greater or equal >=, less than or equal <=, **IN, BETWEEN, NOT BETWEEN, LIKE,** and **IS NULL.**

In order to make complex conditions, we write in the WHERE clause either **OR** or **AND** operator. These operators are used to filter records based on more than one condition:

- The **OR** operator displays a record if any of the conditions separated by OR is TRUE.
- The **AND** operator displays a record if all the conditions separated by AND are TRUE.

The **NOT** operator displays a record if the condition is NOT TRUE.

SELECT Id, FirstName **FROM** Client **WHERE** City='Paris' **OR** City='Nantes';

The OR operator works as a union of records. First, it selects the records that satisfy the first condition, i.e. the Clients that have City = 'Paris'. Then, it selects the Clients that have City ='Nantes'. The final answer of the query is the union of both these records.

Id	FirstName
1	Pierre

Id	FirstName
2	Marine

⇨

Id	FirstName
1	Pierre
2	Marine

SELECT Id, FirstName, Salary **FROM** Client **WHERE** FirstName ='Pierre' **AND** Salary> 1500;

The AND operator works as an intersection of common records. First, it selects the records that satisfy the first condition, then the second one. Finally, it keeps only the common records.

Id	FirstName	Salary
1	Pierre	2,000
3	Pierre	1,500

Id	FirstName	Salary
1	Pierre	2,000

⇨

Id	FirstName	Salary
1	Pierre	2,000

SELECT FirstName **FROM** Client **WHERE** LastName= 'Dupont' **OR** LastName = 'Durant' **OR** LastName = 'Matrin';

This query is equivalent to the following query:

SELECT FirstName **FROM** Client **WHERE** LastName **IN** ('Durant', 'Dupont', 'Martin');

You can use the IN operator when the column value is one of many values of a list. The list could be a list of numbers, a list of dates, a list of names, etc.

SELECT Id, LastName **FROM** Client **WHERE** Salary **BETWEEN** 100 **AND** 1000;

The Between operator is used when you want to select the value of a column from a range of values or an interval. The interval could be an interval of numbers or dates.

SELECT Id, LastName **FROM** Client **WHERE** Salary **NOT BETWEEN** 100 **AND** 1000;

SELECT FirstName, LastName **FROM** Client **WHERE** LastName **LIKE** "%D"; => LastName ends with letter D.

SELECT FirstName, LastName **FROM** Client **WHERE** LastName **LIKE** "D%"; => LastName starts with letter D.

SELECT FirstName, LastName **FROM** Client **WHERE** LastName **LIKE** "%D%"; => LastName contains letter D.

SELECT FirstName, LastName **FROM** Client **WHERE** LastName **LIKE** "D%t"; => LastName starts with D and ends with t.

SELECT FirstName, LastName **FROM** Client **WHERE** LastName **LIKE** "A_C"; => The underscore "_" character will be replaced by any character such as ABC, AEC, etc. It means that the Last name contains exactly 3 characters, the first one is A, the last one is C, and a character.

EXERCISES

Exercise 1: Consider relational database (DB) storing information about Master students and their supervisors. The DB is made of only one relation (table) called R shown below:

St_ID	St_Name	St_Age	St_City	Major	Sup_Name	Sup_City	End_Date
123	Avi	23	Zahlé	Economy	Samuelson	Beyrouth	01/02/2017
123	Avi	23	Beyrouth	Math	Nash	Byblos	10/10/2015
456	Darin	28	Tripoli	Math	Nash	Byblos	12/12/2016
789	Len	25	Baalbek	Physics	Einstein	Princeton	12/01/2016
999	Richard	26	Zahlé	Physics	Aline	Zahlé	04/31/2004

Write SQL queries that answer the following questions:

1) Find the student names that live in Zahlé or having a major Math.

2) Find the name and the city of supervisors that supervise students that live in Baalbek

3) Give all the information about the oldest student

4) How many students are registered in Physics?

5) Find the name and the city of students supervised by Nash.
6) Find the name of students and supervisors that live in the same city.

7) Give the major of students that live in a city where the name starts with the letter 'Z' and their age is greater than 23 years old or the city name of the supervisor that ends with the letter 's' or that will end their Master next year.

8) Give the cities of supervisors that supervise the students that live in cities where their name starts with the letter B.

9) Find the major of students and the name of their supervisor that terminated their Master in 2016.

10) Find the information about students having an age either 26 or 28 or 25 and that live in Tripoli.

11) Give the name of students and their major for students that they have already finished their Master.

Exercise 2: Consider a relational database made of three relations (tables) books, students, and libraries presented below:

Lib_Name	Lib_address
Du Liban	Zouk Mosbeh
Antoine	Sin el Fil
LDLP	Beyrouth

Book_ID	Title	Author	Category	Year	Lib_Name
1	AA	X	Fiction	2000	Du Liban
2	BB	Y	Nature	2010	Antoine
3	CC	X	Arts	2008	Antoine
4	DD	Y	Fiction	2010	Du Liban
5	EE	Z	Cooking	2015	Antoine

St_ID	Major	Gender
123	Math	Male
124	Informatics	Female
125	Physics	Male

Give the question for every SQL query and its execution result:

1) SELECT COUNT(*)
   FROM Student
   WHERE Gender =’Male’ AND major = ‘Physics’;

2) SELECT Title
   FROM Book
   WHERE Year BETWEEN 2008 AND 2010 OR Category =’Fiction’;

3) SELECT DISTINCT Author
   FROM Book
   WHERE Lib_Name LIKE ‘A%e’;

4) SELECT Title, Category
   FROM Book
   WHERE Category IN (‘Fiction’, ‘Arts’, ‘Cooking’) AND Lib_Name = ’Du Liban’;

5) SELECT COUNT(DISTINCT Lib_Name)
   FROM Book;
   SELECT COUNT(DISTINCT Lib_Name)
   FROM Library;

6) SELECT *
   FROM Book
   WHERE Year = 2010 AND Lib_Name = ‘Du Liban’;

7) SELECT *
   FROM BOOK
   WHERE Year = 2010 OR Lib_Name = ‘Du Liban’;

8) SELECT Author
   FROM Book
   WHERE Category < > ‘Fiction’ AND Year > 2008;

9) SELECT MAX(St_ID), Major
   FROM Student
   WHERE Gender = ‘Female’;

10) SELECT Lib_Name, Lib_adress
    FROM Library
    WHERE Lib_Name NOT IN (SELECT Lib_Name
                           FROM Book);

Chapter VI: Computer Networks

Computer network definition

A **computer network** is a set of connected computers and other hardware devices such as printers, modems, and routers that are linked together via Ethernet cable or wirelessly through radio waves. These connected computers and devices can communicate with each others, and thereby exchange data, voice and video traffic, and can share resources, like access to the internet, printers, file servers, and others.

Computer networks offer also other benefits such as increasing storage capacity so you can access files and multimedia (images, music) stored on remote machines, improving communication between people via emails, chats, etc., facilitating many types of games and entertainment that can be played by multi-players on local networks or online.

Importance of a network

A computer is a machine for manipulating data. Man, as a communicator, quickly realized the value of linking these computers together so that information could be exchanged. A computer network can serve several purposes:

- Sharing resources (files, applications or materials, internet connection, etc.).
- Communication between people (e-mail, live chat, etc.).
- Communication between processes (between industrial computers for example).
- The guarantee of uniqueness and universality of access to information (networked databases).
- Lower costs by sharing data and devices.
- Standardization of applications.
- Access to data in a timely manner.
- Communication and organization more efficient.
- The multiplayer video game.

Similarities between types of networks

The different types of networks usually have the following points in common (see Figure 41):

- Servers: Computers that provide shared resources to users through a network server.
- Clients - Computers that access shared resources provided by a network server.
- Connection support: determines how computers are connected to each other.
- Shared data: files accessible on network servers.
- Printers and other shared devices: files, printers, or other items used by network users.

Network equipments

Many equipments are used in building a computer network. They are presented below:

Equipment	Description
Network Card 	A Network Interface Card (NIC) is an electronic device that connects a computer to a computer network usually a LAN. It may enable a wired connection (such as Ethernet) or a wireless connection (such as Wi-Fi) to a local area network.
Concentrator (Hub) 	A network hub is a hardware device commonly used to connect segments of a LAN. The hub contains multiple ports. When a packet arrives at one port, it is copied to the other ports so that all segments of the LAN can see all packets.
Switch 	A network switch is a computer networking device that connects devices together on a computer network by using packet switching to receive, process, and forward data to the destination device.
Router 	A router is a small electronic device that connects a local network to other networks, commonly two LANs or WANs or a LAN and the Internet. A router forwards/directs data packets along networks.
Repeater 	Network repeaters receive and retransmit incoming electrical, wireless or optical signals. Repeaters attempt to preserve signal integrity and extend the distance over which data can safely travel.

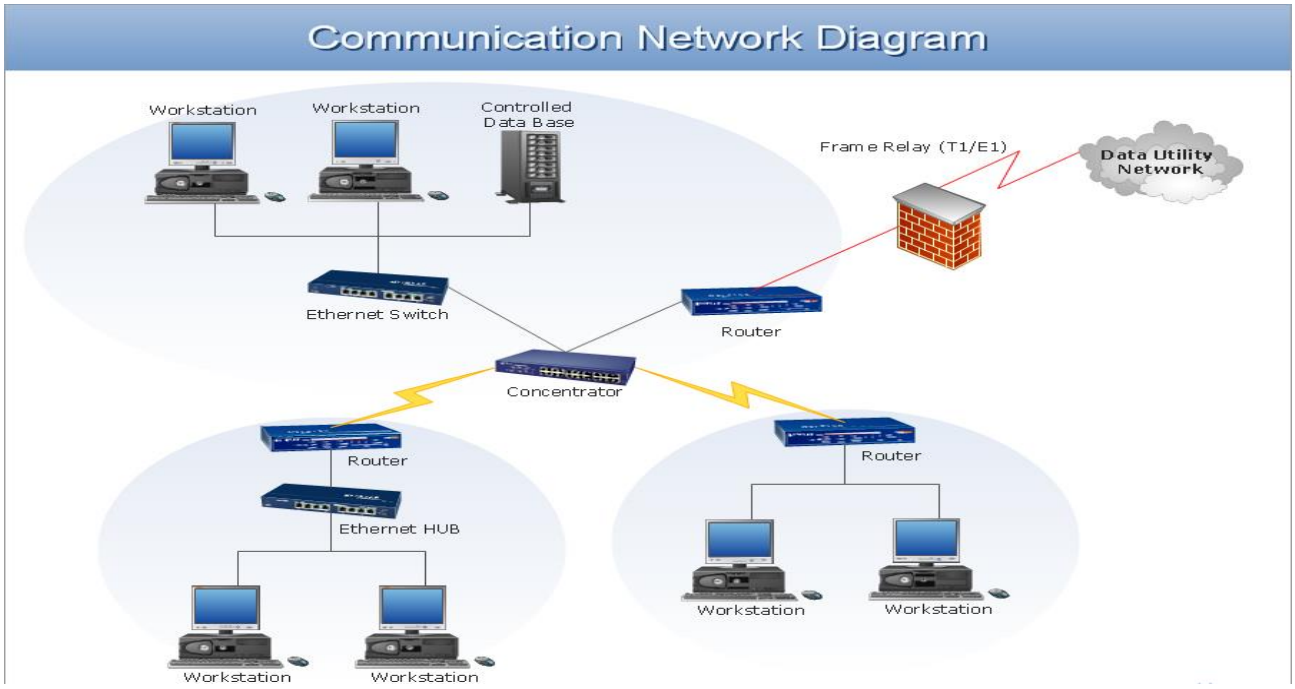


Figure 14: Communication Network Diagram

Computer network types

A network can be characterized by its physical capacity (number of connected computers), the distances that separate the computers (geographical area) or its organizational purpose. There are three main types of computer networks:

1. Local Area Network (LAN)

LAN is a network that connects computers and devices in a limited geographical area such as home, office building, or closely positioned group of buildings (**1 meter to 1 km**). The network transmission speed is between 10MB/s to 1 GB/s. The network capacity does not exceed (**2 to 1000 computers**).

2. Metropolitan Area Network (MAN)

MAN is a large computer network that usually spans a city or a large campus. The MAN network is the union of several local area networks (LAN) within the same perimeter of a very large company, region or city that connects computers within **10 – 25 km** (few kms to dozen kms). The MAN networks usually use fiber optic. For example, a university or college can have a MAN that connects several local area networks located in an area of 1 km^2.

3. Wide Area Network (WAN)

A WAN network is a computer network that covers a large geographic area such as city, country or spans even intercontinental distances (**several hundreds of kilometers**). A WAN uses a communications

channel that combines many types of media such as telephone lines, cables, and air waves. The largest WAN is the Internet. A wide area network consists of a collection of interconnected LANs.

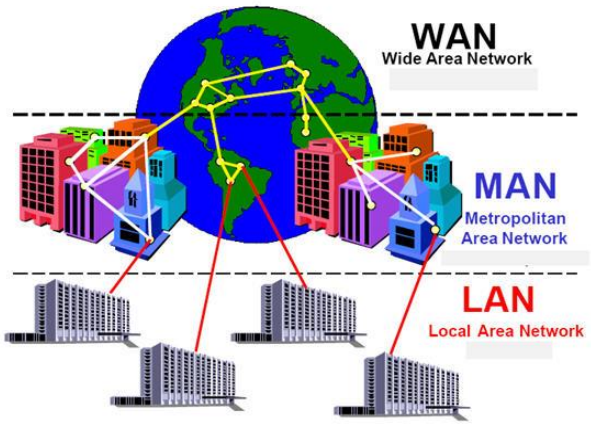
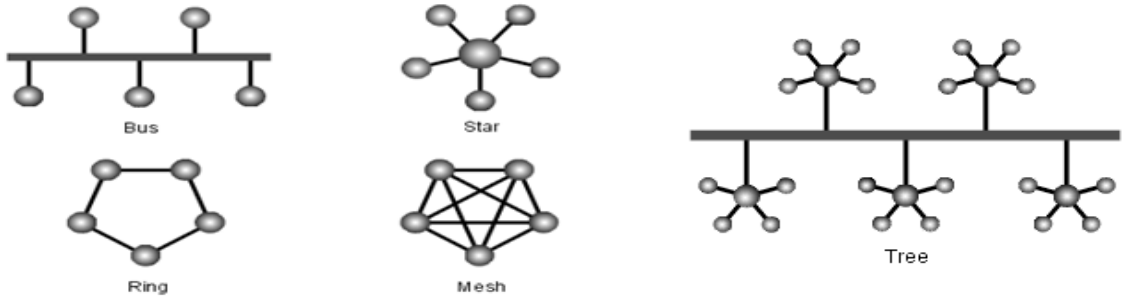


Figure 15: Network Types

Network characteristics

In addition to these types, the following characteristics are also used to categorize different types of networks:

- **Topology**: the geometric arrangement of a computer. Common topologies include a bus, star, ring, mesh, and tree as shown in the figure below.



  - **Bus**: each node is connected to a single cable. A signal travels in both directions to all machines connected on the bus cable until it finds the intended recipient.
  - **Star**: each network host is connected to a central hub with a point-to-point connection. The hub represents a single point of failure.
  - **Ring**: a ring topology is a bus topology in a closed loop. Data travels around in one direction. When one node sends data to another, the data passes through each intermediate note on the ring until it reaches its destination.
  - **Mesh**: the nodes in this network connect directly and no-hierarchically to as many other nodes as possible and cooperate with one another to efficiently route data from/to clients.

  - **Tree**: A tree network, or star-bus network, is a hybrid network topology in which star networks are interconnected via bus networks. Tree networks are hierarchical, and each node can have an arbitrary number of child nodes.
- **Protocol**: the communication protocol defines a common set of rules and signals that computers on the network use to communicate. One of the most popular protocols for LANs is called Ethernet. Popular protocols include File Transfer Protocol (FTP), TCP/IP (Transmission Control Protocol/Internet Protocol), User Datagram Protocol (UDP), Hypertext Transfer Protocol (HTTP), Post Office Protocol (POP3), Internet Message Access Protocol (IMAP), and Simple Mail Transfer Protocol (SMTP).
- **Architecture**: Networks can be broadly classified as using either a Peer-to-Peer network or client/server architecture.

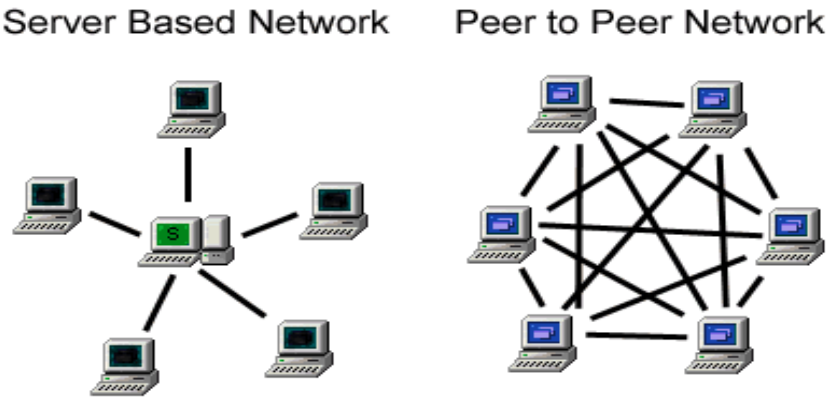


Figure 16: Client-server Network VS. Peer-to-Peer Network

  - **Client/server network**: is a computer network in which one centralized powerful computer called the server is a hub to which many less powerful personal computers or workstations called clients are connected. The clients run programs and access data that are stored on the server.
  - **Peer-to-Peer (P2P) network**: the peers are computer systems which are connected to each other via Internet. Each computer acts as both client and server, so that each can exchange and email directly with every other computer on the network without the need of a central server.

Internet, Intranet and Extranet

Internet

The internet is a global network connecting millions of computers. It is a massive network of networks (internetwork) in which any computer can communicate with any other computer as long as they are both connected to the Internet and that transmit data using the standard Internet Protocol (IP). It is a global system of interconnected governmental, academic, corporate, public, and private computer networks.

Intranet

An intranet is a private network that is contained within an enterprise. It may consist of many interlinked local area networks (LANs). Its main purpose is to share company information and computing resources among employees. An intranet is also anything behind the router on a local area network.

Extranet

An extranet can be viewed as part of a company's intranet that is extended to users outside the company like suppliers, vendors, partners, or customers. Extranet is required for normal day-to-day business activities. For example, placing order to registered vendors, billing & invoices, payments, etc.

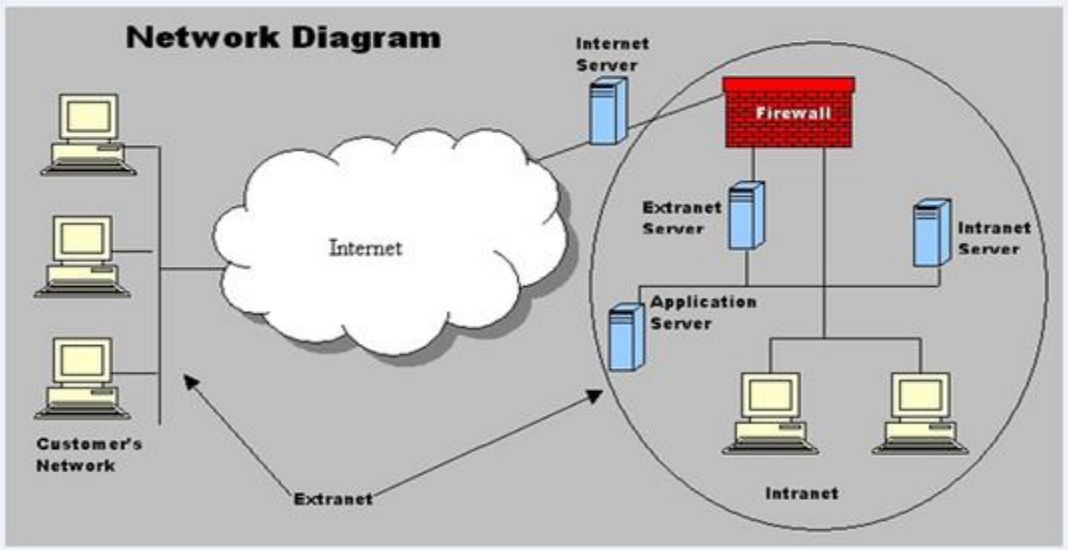
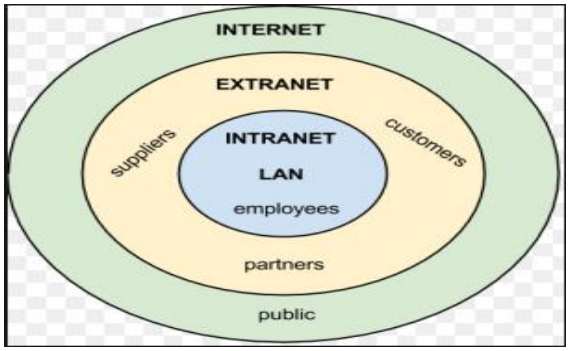


Figure 17: Internet, Intranet, Extranet



Internet Protocol (IP) Addresses

IP address definition

IP is the abbreviation of **Internet Protocol**. In fact, it is a protocol by which data is sent from one computer to another on the Internet.

An Internet Protocol address (**IP address**) is a logical numeric address that is assigned to every single computer, printer, switch, router or any other device that is part of a TCP/IP-based network. An IP address is a logical address that is used to uniquely identify every node or network connected to the Internet.

It is not possible to have multiple computers with the same IP address in the same network. We can compare the IP address to a phone number: It is not possible to have several phones with the same number, otherwise, it will be a complete mess when you want to call someone. It is the same in a network; computers could not communicate with each other if several computers among them have the same IP address. The IP address is linked to the computer's network card (wired or wireless). A computer with multiple network cards can have multiple IP addresses.

The Figure 19 shows an ADSL modem/router that distributes IP addresses to all the network devices that connect to it (wired or wireless). All the assigned IP addresses are different.

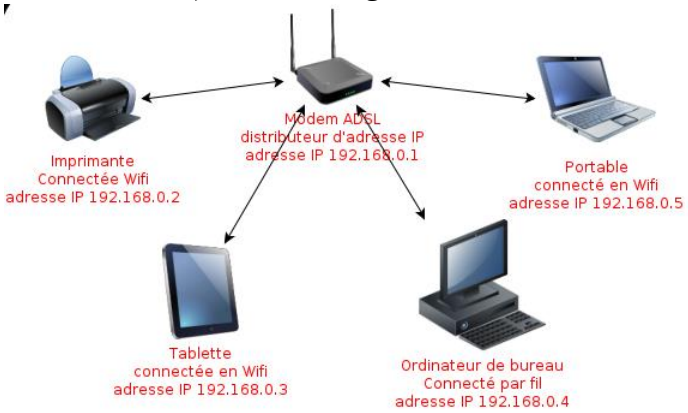


Figure 18: Different IP addresses in a Network

IPv4 address format

Internet Protocol version 4 (IPv4) defines an IP address as a 4 bytes (32 binary bit) number. The bytes range from 0 to 255 each and are separated each by a dot. The standard format of an IPv4 is:

w.x.y.z ***such that w, x, y, and z vary from 0 to 255***

Every computer is attributed one IP address at a time but possibly many IP addresses at different moments coded on 32 bits.

The originality of the IPv4 addressing format lies in the association of the identification of the network with the identification of the host.

An **IPv4 address** is divided into two parts:

1) The left part of the numbers refers to the network ID (netID).
2) The right numbers designate the id of the computer (host) on the network (hostID). They identify a post on this network.

IPv4 address	Network ID	Host ID

The network part (netID) is common to all the hosts that belong to the same network and the host part is unique within the same network.

Let us take an example of IPv4 address to identify the different parts in this address:

Complete Address	192.168.1.1
Network mask	255.255.255.0
Network part	192.168.1.
Host part	.1
Network address	192.168.1.0
Broadcast address	192.168.1.255

Note: The **multicast address** also called a group address is a single IP address for a set of hosts that are joined in a multicasting group. Each network has a particular address known as **broadcast address**. The broadcast address is a network addresses at which all hosts connected to a multiple-access communication network are enabled to receive datagrams. A message sent to a broadcast address may be received by all network-attached hosts.

Network mask

The network mask is used to separate between the network part and the host part of an IP address

Class	Network mask
Class A	255.0.0.0
Class B	255.255.0.0
Class C	255.255.255.0

IP address classes

With an IPv4 addresses, there are five classes of available IP ranges. Each class is identified by a letter from A to E. An IPv4 address class is a categorical division of the IPv4 addresses. Separate IP classes are used for different types of networks. Some classes are used for public Internet-accessible IPs, those networks behind a router (as in classes A, B and C) and some classes are reserved and used for multicasting of identical data to all computers on a network or for research (as in classes D, E). All these classes were defined to optimize the routing of packets between the different networks. They correspond to grouping into networks of the same size. The networks of the same class have the same maximum number of hosts.

Every class uses a number of bytes to identify the network or the hosts connected to the network:

- An IP address in class A has a netID part made only of one byte. The first byte of this class starts always with the bit 0.
- An IP address in class B has a netID part made only of two bytes. The first byte of this class starts always with the bits 10.
- An IP address in class C has a netID part made only of three bytes. The first byte of this class starts always with the bit 110.
- The IP addresses of D and E correspond to particular IP addresses.

In order to identify to which class and IP address belongs, we must examine the first bits of the address.

Class A : w = 1 to 126 (id network : w, host : x.y.z)
Class B : w = 128 to 191 (id network : w.x, host : y.z)
Class C : w = 192 to 223 (id network : w.x.y, host : z)
Class D : w = 224 to 239
Class E : w = 240 to 255

The addresses of class D are used for the multicast communications which means that communication is among a host groups. The first byte starts with the sequence of bits 1110. The first byte of class E addresses always starts with the sequence 1111.

EXERCISES

Exercise 1: class Address

a) Find the minimum and the maximum addresses of all the classes *A* and *C*.
b) How many networks and host per network can be found in these classes?

Exercise 2: class Address

Determine to what class belongs each of the IP addresses, the number of hosts per network, and the network address:

a) 222.1.1.1
b) 113.133.255.2
c) 240.5.67.255
d) 255.255.255.255
e) 132.0.0.255
f) 224.1.2.3

Exercise 3: class address

3.1) write the decimal notation of the following IP addresses:

a) 10001011.00001111.00100011.00011010
b) 11000000.10101000.00000000.00000011
c) 01111110.11111111.11111111.00000001
d) 00000001.11111111.00000001.00000010

3.2) determine the classes and the default network mask for these IP addresses.

Exercise 4:
For every IP address, 200.67.80.45, 50.98.78.67, and 130.89.67.45. Indicate the class, the network address, and the IP addresses that can be assigned to the hosts of this network.

Exercise 5:

Give the class of the following addresses: 204.160.241.93 (IP address of www.javasoft.com), 138.96.32.3 (www.inria.fr), 18.181.0.31 (www.mit.edu)and 226.192.60.40? How many hosts can be used for every address?

Chapter VII: Internet

Definition

The Internet is a global network connecting millions of computers. It is a massive network of networks (internetwork) in which any computer can communicate with any other computer as long as they are both connected to the Internet and that transmit data using the standard Internet Protocol (IP). Most Internet connections come via some sort of cable such as landline telephone lines (DSL), cable TV lines (cable), and fiber-optic lines (fiber) or via satellite that relies on a physical dish.

The Internet is a global system of interconnected governmental, academic, corporate, public, and private computer networks.

Internet History

The Internet started in the 1960s as a way for government researchers to share information. Computers in the '60s were large and immobile and in order to make use of information stored in any one computer, one had to either travel to the site of the computer or have magnetic computer tapes sent through the conventional postal system.

Another catalyst in the formation of the Internet was the heating up of the Cold War. The Soviet Union's launch of the Sputnik satellite spurred the U.S. Defense Department to consider ways information could still be disseminated even after a nuclear attack. This eventually led to the formation of the **ARPANET** (Advanced Research Projects Agency Network), the network that ultimately evolved into what we now know as the Internet. ARPANET was a great success but membership was limited to certain academic and research organizations who had contracts with the Defense Department. In response to this, other networks were created to provide information sharing.

January 1, 1983 is considered the official birthday of the Internet. Prior to this, the various computer networks did not have a standard way to communicate with each other. A new communications protocol was established called Transfer Control Protocol/Internetwork Protocol (TCP/IP). This allowed different kinds of computers on different networks to "talk" to each other. ARPANET and the Defense Data Network officially changed to the TCP/IP standard on January 1, 1983, hence the birth of the Internet. All networks could now be connected by a universal language.

Main Internet Services

E-mail

When you send and receive email, you use can email client that can be Web-based you check it through your Web browser (Hotmail, Gmail, or Yahoo Mail) or an application on your computer (Outlook or Thunderbird). First, the sender composes a message using the email client on their computer. When the user sends the message, the email text and attachments are uploaded to SMTP (Simple Mail Transfer Protocol) server as outgoing mail. The SMTP communicates with the DNS to find the recipient's email server, if found, the message and attachments are transferred to. To find out if he has a new email, the user must connect to the email server and check his mailbox or the desktop email client will check for new email. Finally, one these two protocols POP3 (Post Office Protocol) or IMAP4 (Internet Mail Access Protocol) is used to transfer email from the recipient's email server to the client.

Internet Forums

An Internet forum also called discussion board or discussion group is a Website that provides an online exchange of information between people about a particular topic. It provides a venue for questions and answers and may be monitored to keep the content appropriate. Forums can be entirely anonymous or require registration with username and passwords. Messages may be displayed in chronological order of posting or in question-answer order where all related answers are displayed under the question. Quora and GameFAQs are examples of Internet forums.

Blogs

A blog is Website where someone a blogger regularly records their thoughts or experience or talks about a subject via blog posts. It is a hierarchy of text, images, media objects and data, arranged chronologically, that can be viewed in an HTML browser. Blogs and blog posts can be shared on social networks (Twitter, Facebook, Google+) and people can leave comments under the blog posts in order to start meaningful conversations.

Online Chat

Online chat may refer to any kind of communication over the Internet that offers a real-time transmission of text messages from sender to receiver. Chat messages are generally short in order to enable participants to respond quickly. Online chat may address point-to-point communications as well as multicast communications from one sender to many receivers. Online chat uses tools such as instant messengers, Internet Relay Chat (IRC), talkers, and possibly a Multi-User Dimension (MUD).Online chat is one the most popular services on the Internet.

Web browser

A Web browser is a software application for accessing information on the World Wide Web (WWW). Each individual Web page, image, and video on a Web server is identified by a distinct URL, enabling browsers to retrieve and display them on the user's device. Once a Web page has been retrieved, the browser's rendering engine displays it on the user's device. Virtually all URLs on the Web start with either http: or https: which means the browser will retrieve them with the Hypertext Transfer Protocol.

The most popular Web browsers are Google Chrome, Mozilla Firefox, Safari, Internet Explorer, and Edge.

Search engine

A Web search engine is a software system that is designed to search for information on the World Wide Web. A user enters keywords or key phrases into a search engine and receives a line of results referred to as search engine result pages (SERPs). The information may be a mix of Web pages, images, videos, and other online data.

The search engines are based on "robots", also called bots, spiders, crawlers or agents who browse sites on the Web at regular intervals and automatically to discover new addresses (URL). They follow the hypertext links that link the pages to each other, one after the other. Each identified page is then indexed in a database, accessible by Internet users using keywords. When a user enters a query into a search engine it is a few keywords. The index already has the names of the sites containing the keywords, and these are instantly obtained from the index.

Some Websites use their own search engine to let their visitors to search their content. Some search engines such as Google Desktop, Exalead Desktop, and Copernic Desktop are software installed on a personal computer. They are called desktop search engines that combine search your computer files and search Websites. Currently, Google and Microsoft's Bing control the vast majority of the market.

Virus and Trojan: Understand and protect

Virus

A computer virus is a small program that attaches itself to a program or executable file so it can spread from computer to another, leaving infections as it travels. Viruses can damage programs, delete files or software, or reformat the hard disk. Viruses are spread through human actions such as running infected programs, sharing emails with viruses as attachments in the email, or via CDs and flash memories.

Trojan horse

Trojan horse is not a virus, it is a destructive program that looks as a genuine application. Unlike viruses, Trojan horses do not replicate themselves but they can be just as destructive. Trojans contain malicious code, that, when triggered, cause loss, blocking, modification, copying, or theft of data or disrupting the computer performance. They also open backdoor entry to your computer which gives malicious users/programs access to your system, allowing confidential and personal information to be theft.

Worms

Computer worms are similar to viruses in that they replicate functional copies of themselves and can cause the same type of damage. In contrast to viruses, worms are standalone software and do not require

a host program or human help to propagate. Advanced worms leverage encryption, wipers, and ransomware technologies to harm their targets.

Spyware

Spyware is a kind of malware that is designed to collect information and data on users and observe their activity without users' knowledge. Spyware can capture information like Web browsing habits, e-mail messages, usernames and passwords, and credit card information. If left unchecked, the software can transmit this data to another person's computer over the Internet.

Port

A port for a computer connected to a network is a logical construct that identifies a type of network service. The port is always associated with an IP address of a host and the protocol type of the communication (example: http://www.example.com:8080/path). Ports are identified for each protocol by 16-bit unsigned numbers (1 to 65,536) called as the port number. Some port numbers are reserved to identify specific services such as 80 (HTTP) used in the Web, 20 to download files from FTP servers, and 25 and 110 to respectively send (SMTP) and receive (POP3) emails.

Hacker

A code hacker refers initially to a technically skilled programmer who tries to find hidden bugs in programs. The term now means a person who driven by passion seeks to find the flaws of a system (and who is therefore someone very respectable) but also the one who attempts to break into a system (a computer) illegally or without the owner authorized access. By extension, "hacking" refers to the discipline of searching for the flaws in a system.

Antivirus

Antivirus software is a utility program designed to scan, prevent, detect and remove malware infection on individual computing devices, networks, and IT systems. The antivirus must keep an updated database of virus types. Examples of common antivirus programs include AVG, Norton, Avast, Avira, Kaspersky, McAfee, and Windows Defender.

Firewall

A firewall is software used to maintain the security of a private network. It monitors and controls incoming and outgoing network traffic based on predetermined security rules. Firewalls prevent unauthorized Internet users or illicit software from gaining access to private networks connected to the Internet especially intranets. A firewall is considered as the first line of defense in protecting private information. For greater security, data can be encrypted.

Social networks

A social network is a Website that allows people with similar interests, friends, and families to come together and share information, ideas, advices, photos and videos. It can be used for personal or business reasons. The most popular social networks include Facebook, Google+, Twitter and LinkedIn.

Hypertext Markup language (HTML)

Definition

HTML is the language used to create Web pages and Web applications. "**Hypertext**" refers to the hyperlinks that an HTML page may contain. "**Markup language**" refers to the way **tags** are used to define the page layout and elements within the page.

HTML is a standardized system for tagging text files to achieve text files to achieve font, color, graphic, and hyperlink effects on Web pages. It is the set of **markup symbols** or codes inserted in a file intended for display on a Web browser page. The markup tells the Web browser how to display a Web page's words and images for user. Each individual markup code is referred to as an **element** (but many people also refer to it as a tag). Some elements come in pairs that indicate when some display effect is to begin and when it is to end.

Web browsers receive HTML documents from a webserver or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. They provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by **tags**, written using angle brackets. Tags such as <img /> and <input /> introduce content into the page directly. Others such as ... surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML5 with **Cascading Style Sheets (CSS3)**, and **JavaScript 1.8.5**, it forms a triad of cornerstone technologies for the Web. HTML can embed programs written in a scripting language such as JavaScript which affect the behavior and content of web pages. Inclusion of CSS defines the look and layout of content.

Example of HTML used to define a basic Web page with a title and a single paragraph of text.

```
1  <!doctype html>
2  <html>
3  <head>
4      <title>My First Web Page</title>
5  </head>
6  <body>
7      <p>This is an example of a paragraph in HTML.</p>
8  </body>
9  </html>
```

The first line defines what type of contents the document contains. "<!doctype html>" means the page is written in HTML5. Properly formatted HTML pages should include <html>, <head>, and <body> tags, which are all included in the example above. The page title, metadata, and links to referenced files are placed between the <head> tags. The actual contents of the page go between the <body> tags, it is the visible part of the HTML page.

Note: To create a Web page, open a text editor such as WordPad, NotePad, Notepad++, or Dreamweaver. Open a new text file, write your HTML code and save it by specifying the location and the page name with an extension .html (example: MyPage.html). As a result, the Web page will be created at the specified location and will be viewed using any Web browser on your computer.

HTML Elements

All HTML documents must start with a document type declaration: <!DOCTYPE html>. The HTML document itslef begins with <html> and ends with </html>. The visible part of the HTML document is between <body> and </body>.

An HTML document is made of HTML elements. An **HTML element** usually consists of a **start tag** and **end tag**, with the **content** inserted in between: *<tagname> Content of the element…</tagname>*

The HTML element is everything from the start tag to the end tag.

Example: ***<h1> My First Heading </h1>***

An HTML element can contain HTML elements, they are called **nested elements**. The content of these elements can be a set of HTML elements.

Example: ***<head> <title> My Page </title> </head>***

Content of head element

Elements with no content are called **empty elements**, they do not have an end tag. **Example**: the ***
*** break row tag defines a line break.

All HTML element can have attributes. **HTML attributes** provide additional information about an element such as its width, height, title, id, address, background-color, style, and language. An attribute is always specified in the stat tag and comes in pairs like attribute-name = “value”.

Example:

******

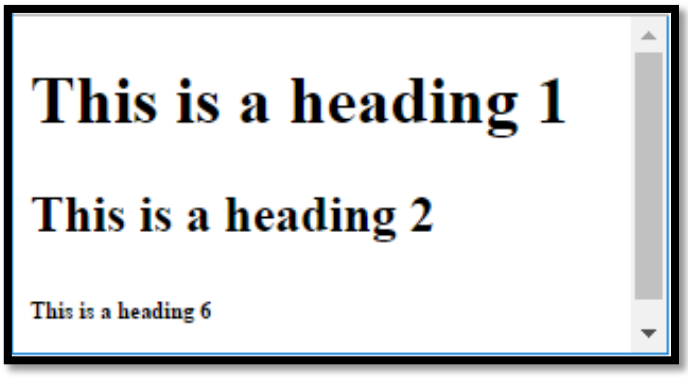
In this the example, src, width, height, and alt are the attributes of the img element.

HTML Headings

Heading are defined in HTML with the **<h1>, <h2>, <h3>, <h4>, <h5>**, and **<h6>** tags. <h1> defines the most important heading (ex: section) and <h6> defines the least important heading (ex: subsubsection). Headings are important, they show the structure of a document and search engines use them to index the structure and content of your Web pages.

Example:

```
<!doctype html>
<html>
<body>
    <h1> This is a heading 1</h1>
    <h2> This is a heading 2</h2>
    <h6> This is a heading 6</h6>
</body>
</html>
```



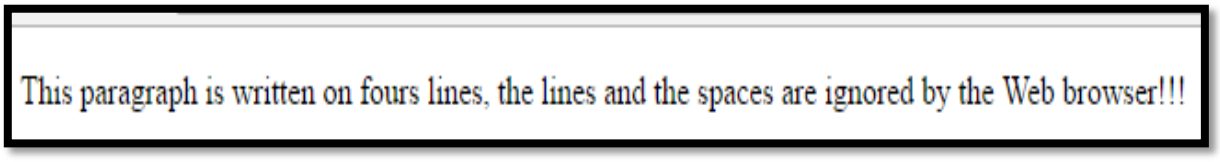
HTML Paragraphs

In HTML, we use the tag **<p>** to insert a paragraph in the Web page.

Example:

```
<!doctype html>
<html>
<body>
  <p> This paragraph
      is    written on fours lines,
      the lines and the spaces are
      ignored by the Web browser!!!
  </p>
</body>
</html>
```

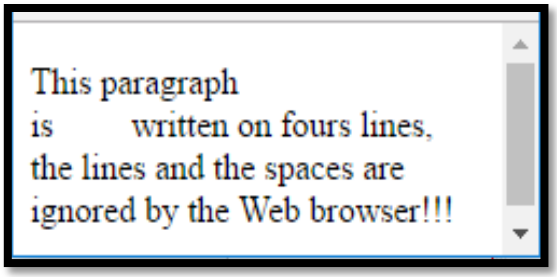
The result rendered by the browser is shown below:



A header can represent a title of a section, and a section can contain many paragraphs.

To have a **new line** in your text, use the HTML **
** elements without starting a new paragraph. To force an extra space to show, insert a **non-breaking space * ***

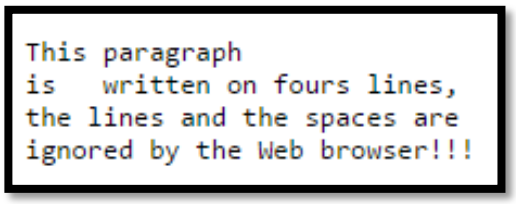
```
<!doctype html>
<html>
<body>
<p> This paragraph <br/>
    is         written on fours lines,<br/>
    the lines and the spaces are <br/>
    ignored by the Web browser!!!
  </p>
</body>
</html>
```



In addition, the HTML **<pre>** element can be used to preserve both spaces and line breaks. It defines a preformatted text. The text inside a <pre> element is displayed in a fixed-width font usually Courrier.

Example:

```
<pre>
This paragraph
is   written on fours lines,
the lines and the spaces are
ignored by the Web browser!!!
</pre>
```



HTML Links

Links in HTML are hyperlinks. By a clicking on a link, users will be directed to another document specified in that link. In HTML, a link is defined with the **<a>** tag.

Examples:

Link text

<a href=”http://www.ul.edu.lb” **target**=”_blank”>Lebanese University**** *external link*

** Go to next Page *internal link to Website*

To jump to specific parts of a long Web page, we can use HTML bookmarks. Write these two elements in the same page: the first one create a bookmark, and the 2nd creates a link.

```
<h1 id=”first”> First Section</h1>
<a  href=”#first”> Go back to the first Section </a>
```

Within page

****<img src**=”AnImage.png” **alt**=”Page B not found”>****

The ***href*** attribute specifies the destination address of the link, the ***link text*** is the visible part of the link, and the ***target*** attribute specifies where to open the linked document. The possible values for the target are _blank, _self, _parent, _top, framename. Sometimes, instead of text, we use image as link (see the last example about links above). This is done by replacing text with the <img> tag. The link text color changes if the link is visited, unvisited, or active.

HTML Images

An image is defined with the **<img>** tag. It main attribute is src that specifies the Web address of the image called Uniform Resource Locator (URL).

<**img** **src**=”myimage.png” **alt**=”An image” **style**=”width:300px;height:300px;”/>

If the image is in the same folder of the Web page, typing the page name with its extension in the src attribute is enough to be found by the Web browser. However, if the image is in a different folder than the Web page, then the folder name must be included in the src attribute of the image.

<**img** **src**=”C:/Users/AnImage.gif” **alt**=”An image”**style**=”width:300px;height:300px;”/>

It is common when designing Web sites to store images in a sub-folder of the folder that stores the Web pages.

<**img** **src**=”/Images/Image.jpg” **alt**=”An image”**style**=”width:300px;height:300px;”/>

In order to insert an image stored on image servers or from any Web address in the world, a full path to that image must specified in the src attribute.

<**img** s**rc**=”http://www.dgreetings.com/fragrance-flowers/images/a-rose-flower.jpg **alt**=”A rose flower” **style**=”width:300px;height:300px;”/>

HTML Tables

The **<table>** tag is used to define an HTML table. The **<tr>** tag meaning table row defines a row in the table and the **<th>** tag meaning table header defines a table header. By default, table headings are bold and centered. A table data or cell is defined with the **<td>** tag meaning table data, they can contain all sorts of THML elements such as text, images, lists, links, other tables, etc.

Example 1: A table with specified border

```
<!doctype html>
<html>
<body>
<table border="1">
<tr>
  <th> Column Title1</th>
  <th> Column Title2</th>
</tr>
<tr>
  <td>Row 1, Column 1</td>
  <td>Row 1, Column 2</td>
</tr>
<tr>
  <td>Row 2, Column 1</td>
  <td>Row 2, Column 2</td>
</tr>
</table>
</body>
</html>
```

Column Title1	Column Title2
Row 1, Column 1	Row 1, Column 2
Row 2, Column 1	Row 2, Column 2

Example 2: A table without a border

```
<table>
<tr>
  <th>Name</th>
  <th>Salary</th>
</tr>
<tr>
  <td>Spider Man</td>
  <td>$5000,000</td>
</tr>
<tr>
  <td>cristiano Ronaldo</td>
  <td>21 million &#8364;</td>
</tr>
```

Name	Salary
Spider Man	$5000,000
cristiano Ronaldo	21 million €

Example 3: A table with a cell that contains a table, caption, border, and width

```
<table border="2px" width="20%">
<caption> This table contains a table</caption>
<tr>
  <th>First row</th>
  <td>
    <table border="2" align="center">
      <tr><td>1</td><td>2</td></tr>
      <tr><td>3</td><td>4</td></tr>
    </table>
  </td>
  <td> B</td>
</tr>
<tr>
  <th>Second row</th>
  <td>C</td>
  <td> D</td>
</tr>
<tr>
</table>
```

This table contains a table

First row	1 2 3 4	B
Second row	C	D

Lists

There are 3 types of list that can be defined in HTML: (1) unordered, (2) ordered, and (3) description lists.

An **unordered HTML list** stats with the <ul> tag. Each **l**ist **i**tem starts with the <li> tag. By default, the list items will be marked with bullets. We can change the style of the list item marker using the CSS list-style-type property that takes one of these values: disc, circle, square, and non.

```
<ul>
  <li> First item </li>
  <li> Second item </li>
  <li> Third item </li>
</ul>
```

- First item
- Second item
- Third item

```
<ul style="list-style-type:circle">
  <li> Blue </li>
  <li> White </li>
  <li> Red </li>
</ul>
```

- Blue
- White
- Red

An **o**rdered **l**ist starts with the <ol> tag and each list item starts with the <li> tag. The list items will be marked by numbers by default.

```
<ol>
   <li> First item </li>
   <li> Second item </li>
   <li> Third item </li>
 </ol>
```

1. First item
2. Second item
3. Third item

We can set the type of the list item marker in an ordered list using the the **type** attribute of the <ol> tag. The possible values are type =”1”, type=”A”, type=”a”, type=”I”, and type=”i". By default, the list items will be numbered with numbers. We can also specify from where the numbering begins using the **start** attribute. The **value** attribute changes the natural numbering of an item.

```
<ol start="3" type="I" >
  <li> First item </li>
  <li> Second item </li>
  <li value="6"> Third item </li>
 </ol>
```

III. First item
IV. Second item
VI. Third item

Lists can be nested i.e. a list can contain other lists.

```
<ol>
  <li> Item 1 </li>
  <li> Item 2
       <ul type="disc"> <li> A </li>
            <li> B </li>
            <li> C </li>
       </ul>
   </li>
   <li> Item 3 </li>
</ol>
```

1. Item 1
2. Item 2
   - A
   - B
   - C
3. Item 3

HTML Formatting Elements

HTML also defines special elements for defining text with special meaning. HTML uses elements like and for formatting output, like **bold** or *italic* text. Formatting elements were designed to display special types of text:

- **** : Bold text some text
- **** : Important text some text
- **** : Italic text some text
- **** : Emphasized text some text
- **<mark>** : Marked text <mark> some text </mark>
- **<small>** : Small text <small> some text </small>
- **<del>** : Deleted text <del> some text </del>
- **<ins>** : Inserted text <ins> some text </ins>
- **<sub>** : Subscript text _{some text}
- **<sup>** : Superscript text ^{some text}

Red = FF0000
Green = 00FF00
Blue = 0000FF
Cyan = 00FFFF
Magenta = FF00FF
Yellow = FFFF00

- **<hr/>** : is used to add an horizontal row in the Web page
- **<!-- -->**: is used to add a comment
- ** **: align to the center some text not supported in HTML5
- ****: specifies the font size, and color of a text **** some text not supported in HTML5. Instead of specifying directly the color, you can use the triplet color Red Green Blue (RGB) in hexadecimal. Example: some text , the result is the text in White.

```
  Some HTML elements were used to make a text in bold and italic.
<p align="left"><b><i>This sentence is written in bold and italic.</i></b>
<br/><hr/>
This is the end of the paragraph.</p>
<p> This sentence contains <u>underlined words</u>,
<strike> strikethrough</strike>, <tt> monospace</tt>,
<sup>superscript</sup>, <sub> subscript</sub>.</p>
<p align="center"> I want to drink <del>wine</del> <ins>cola</del> </p>
<p> The following words use <big> big </big> and <small>small words </small></p>
```

Some HTML elements were used to make a text in bold and italic.

This sentence is written in bold and italic.

This is the end of the paragraph.

This sentence contains underlined words, ~~strikethrough~~, monospace, superscript, subscript.

I want to drink ~~wine~~ cola

The following words use big and small words

EXERCISES

Exercise 1 (Headers & Paragraphs)

Add the necessary headings and paragraphs to the text to obtain the Web page shown below:

```
<!DOCTYPE html>
<html>
 <head> <title> Exercise1 </title> </head>
 <body>
```

UEFA Champions League

The Champions League is an annual continental club football competition and contested by top-division European clubs.

Healthy & Diet Recipe

A breakfast made of Broccoli & Feta Omelet with toast leaves you feeling satisfied yet energized.

Broccoli benefit

The broccoli provides filling fiber.

Omelet benefit

The protein-loaded eggs curb appetite.

Exercise 2 (Headers)
Add a heading to your document indicating that this is "Bart Simpson's catchphrase" and four other headings with the text "I didn't do it!". Start these headings with the least important heading to more important ones as shown below:

Bart Simpson's catchphrase

I didn't do it!

I didn't do it!

I didn't do it!

I didn't do it!

Exercise 3 (Paragraph, Break Row, & Space)
Give the output of the following HTML document when viewed by a Web browser.

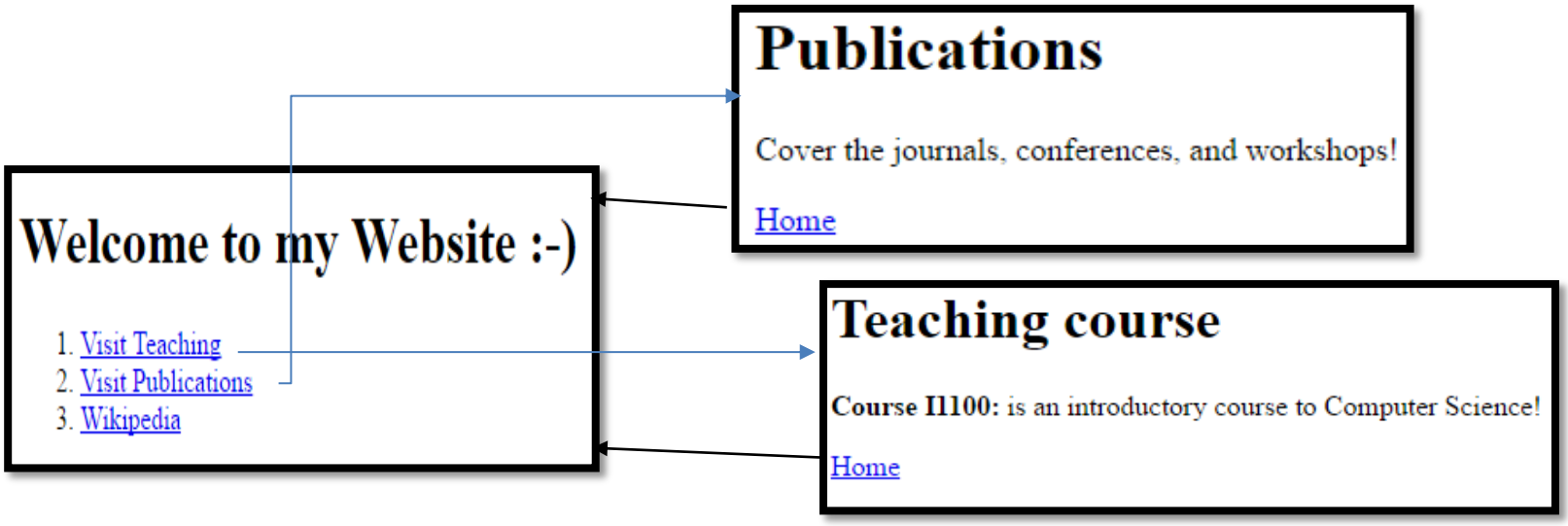
```
<!DOCTYPE html>
<html>
 <head> <title> Exercise3 </title> </head>
 <body>
 <p> <i><b>WhatsApp Messenger</b></i> is free instant messaging
application for smartphones.<br/> It uses the Internet
 to make voice calls.<p> Many types of documents can be sent
using Whatsapp.</p>
 It was acquired by Facebook in 2014 for ~US$19.3 billion.
 </p>

 <p> WhatsApp had a user above of over one billion.<br/> It is
the most popular    
 messaging application at the time.
 </p>
 </body>
</html>
```

Exercise 4 (Links and List)

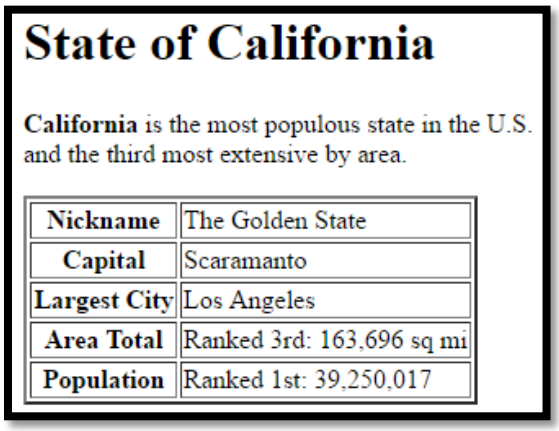
Add the appropriate **links** to connect properly the three Web pages named DoctorPage, TeachingPage, and PublicationPage given below. The Doctor page contains a **list** of three links, the first one referring

to the TeachingPage, the second referring to the PublicationPage, and the last one to Wikipedia (https://en.wikipedia.org/). The Home link in both the TeachingPage and PublicationPage sends the user back to the Doctor's page. We assume that all the Web pages are stored in the same folder. Add the right attribute and its value to the links in the DoctorPage to open them in new windows.



Exercise 5 (Table)

Use the <table> tag and the border attribute, get your table to look like the following table:



State of California

California is the most populous state in the U.S. and the third most extensive by area.

Nickname	The Golden State
Capital	Scaramanto
Largest City	Los Angeles
Area Total	Ranked 3rd: 163,696 sq mi
Population	Ranked 1st: 39,250,017

Exercise 6 (Table)

Use the <table> tag to obtain the following table. The table border is 1pixel, width is 50% of the page width, the logo image is a link that sends to the official Website of the Lebanese University (http://www.ul.edu.lb).

Page Title	
Navigation bar • Link1 • Link2	**Page Content** GOES HERE ... 
Footer: Last updated 1/17/2017 - 3:30AM	

Exercise 7 (List)

Use two of the HTML list types to produce the following Web page. Add two horizontal rows, one between the list and another one at the end.

Top 3 best-selling cars in the world for 2015

1. **Toyota Corolla**
2. Volkswagen Golf
3. Ford F-Series

Information about the sellinngs

Toyotal Corolla
The Toyota Corolla did something no vehicle has ever done in history.
Selling more than 1.3 million units worldwide (1,339,024).

Volkswagen Golf
Also rising one rank to second place is the Volkswagen Golf, ending 2015 on a high note despite the diesel scandal.
In total, 1,041,279 new Golf compacts were sold worldwide.

Ford F-Series
As the top-selling vehicle in the U.S. for the past 40 years, the Ford F-Series rose to third place from fourth in 2015.
Global sales of the pickup rang in at 920,172 units.

Computer Science and Society

Computer oriented crimes (cybercrimes) are offenses that are committed against computer data or a computer system generally connected to a network. Their motive is to intentionally harm the reputation of the victim or cause harm or loss to the victim directly or indirectly.

Crimes that use computer devices or networks include false information, fraud and identity theft, hacking, and data and system interferences.

Penalties for computer-related crimes can range from a fine and a short period of jail time to felony and carry many years in prison.

False information

False information involves the alteration of the truth. In computer science, the truth is not a written act but is found in the computer data. These data must be of legal significance that they may serve as a basis for the exercise of a right or action, to establish or prove a right.

In practice, the false information can occur by introducing, modifying or deleting data in a computer system or modifying by any technological means the possible use of the data in a computer system.

The false information offense can also have a particular moral element. It could be made by someone who acted with conscience to alter the truth with a particular intention. This intention may be fraudulent, the intention to procure or procure for others an illegal advantage or the intention with purpose of harming others.

The penalties for this crime vary from imprisonment from six months to five years and a five of twenty-six to one hundred thousand euros or one of these penalties. In case of another false information offense within five years of a previous conviction, the penalties mentioned previously are doubled. Finally, the attempt of false computing is also punishable by imprisonment of six months to three years and a fine of twenty-six to fifty thousand euros or one of these penalties. The legislator also criminalized the use of false information. Thus, anyone who makes use of illegally obtained data, knowing that they are false, is punished as if he were the author of this false information.

Skimming is an example of false information offense. A type of fraud in which the numbers on a credit card are recorded and then transferred to a duplicate card. This is done without the knowledge of the original credit card holder.

Internet fraud and identity theft

Internet fraud is a type of fraud that makes use of the Internet. The term “Internet fraud” refers generally to any type of fraud scheme that uses one or more components of the Internet such as chat rooms, e-mail,

message boards or Web sites to present fraudulent solicitations to prospective victims, to conduct fraudulent transactions, or to transmit the proceeds of fraud to financial institutions.

The majority of Internet frauds are related to online auction and retail sites that purport to offer high-values items. The victims send their money for the promised items without receiving them or getting items less valuable (counterfeit or altered goods). Other type of Internet fraud is the one that advertises business opportunities that allow individuals to earn thousands of dollars a month in "work-at-home" ventures and asks them to pay several hundred dollars without delivering any serious materials.

Identity theft and identity fraud are terms used to refer to all types of crime in which someone wrongfully obtains and uses another person's personal data in some way that involves fraud or deception, typically for economic gain. Personal data like bank account number or credit card number, telephone calling card number, and other valuable identifying data can be used if they fall into the wrong hands to personally profit at other's expense.

The identity theft can occur through phishing which is the act of sending an email to a user safely claiming to be an established legitimate enterprise in attempt to scam the user into surrendering private information that will be used for identity theft. The email directs the user to visit a Web site where they are asked to update personal information (passwords, credit card, social security, etc.) and the Web site steals this information.

Hacking

Unauthorized access to a computer system, or hacking is an offense within the penal code. This practice refers to breaking into someone's computer system without their knowledge, without having the required authorization. A distinction must be made between external hacking and internal hacking. External hacking is done by someone outside the hacked computer system. This behavior is considered as an offense when the author knowing that he is not authorized, accesses a computer system. A simple access without causing any damage is considered as an offense. External hacking incurs a prison sentence of three months to one year and/or a fine of twenty-six to twenty-five thousand euros. In case of fraudulent intent, the prison term is six months to two years.

Hacking is internal when it is committed by someone who has access to the computer system but has exceeded the limits of their authorization. The legislator considers that a hacker acted with a fraudulent intention or with aim to harm, his behavior should be incriminated. This hacker may be sentenced to imprisonment for six months to two years and/ or to pay a fine of twenty-five thousand euros. One of the hacking aim is to steal data by copying them on a digital medium. This practice is often encountered in industrial espionage. On the other hand, using a hacked system is an aggravating circumstance. In the last two cases, penalties are imprisonment of one to three years and/or a fine of twenty-six to fifty thousand euros. The legislator also considers the hacking attempt as offense and it is punished with same penalties as the hacking itself.

Data and system interferences

Data interference: is the intentional or reckless alteration, damaging, deletion or deterioration of computer data, electronic document, or electronic data message, without right, including the introduction or transmission of viruses.

System interference: The intentional alteration or reckless hindering or interference with the functioning of a computer or computer network by inputting, transmitting, damaging, deleting, deteriorating, altering or suppressing computer data or program, electronic document, or electronic data message, without right or authority, including the introduction or transmission of viruses.