University of Baghdad
College of Science
Department of Mathematics

First Class
First Course

Introduction to Computer System

Computer Basics 1

Information system: An interconnected environment for managing and processing data using a computer.

Parts of an information system:

- People
- Hardware: Devices that are required to store and execute (or run) the software
- Software: Collection of instructions that enables a user to interact with the computer
- Procedures
- Data

Computer Types:

1) Personal Computers

    - Desktop PC
        - System unit, separate monitor, keyboard, mouse
    - Notebook PC
        - Fold-up design, built-in keyboard and screen
    - Tablet PC
        - Slate design, touch screen, no separate keyboard, mouse, or monitor
    - Smartphone
        - Hand-held, phone and computer, touch screen

2) Multi-User Computers

- Server
    - Serves and supports a network
    - Provides centrally accessible storage space
    - Shares printers
    - Does not directly provide processing power to clients
- Mainframe
    - Collects large amounts of business data
    - Provides processing support to terminals
- Supercomputer
    - Largest computer available
    - Universities, research, government

Software Types:

1) Basic Input Output System (BIOS)

- Read-only chip on motherboard
- Startup instructions for computer

2) Operating System (OS)

- User interface
- Runs application
- File storage
- Communication with hardware

3) Utilities

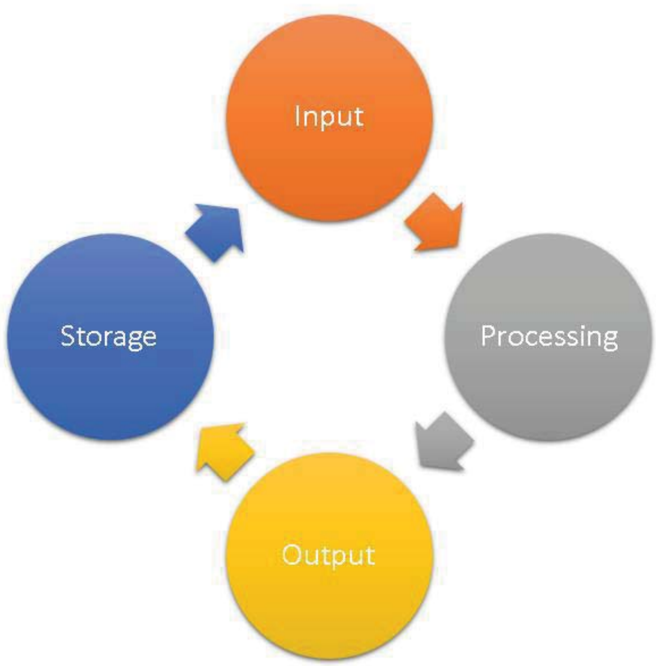
- Error correction, optimization, protection

4) Application Software

- Productivity tools, graphics, games, multimedia

Desktop Components:

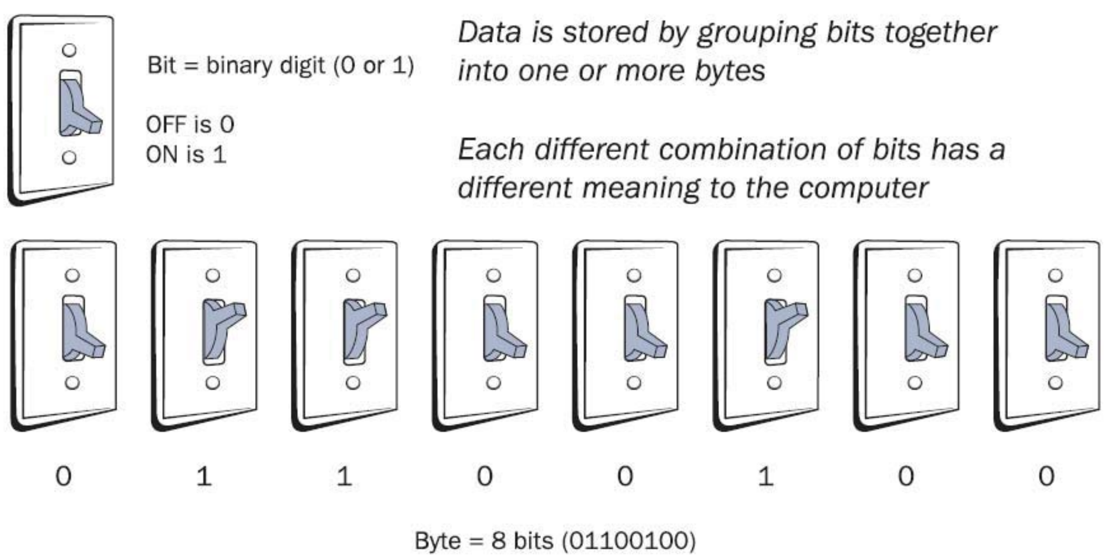
Information Processing Cycle:



Information Processing:

1) Input
   - Keyboard, Mouse, digital cameras, scanners, bar code readers
2) Processing
   - Motherboard, CPU, memory, buses, chipset
3) Output
   - Monitor, printer
4) Storage
   - Hard drive, USB flash drive, CD, DVD

How Data is Represented:



Common Quantities of Bytes:

1024 bytes	=	1 KB	KB	=	Kilobyte
1024 KB	=	1 MB	MB	=	Megabyte
1024 MB	=	1 GB	GB	=	Gigabyte
1024 GB	=	1 TB	TB	=	Terabyte
1024 TB	=	1 PB	PB	=	Petabyte

Name ______________________________ Date ______________

Binary → Decimal Conversion

For each '1' digit in the binary number, add the number at the top of the column.

Binary *128*	*64*	*32*	*16*	*8*	*4*	*2*	*1*	Decimal
	1	1	0	1	0	1	0	**106**
						64 + 32 + 8 + 2 = 106		
					1	0	0	
				1	0	1	0	
		0	1	0	1	0	1	
				1	1	1	1	
			0	0	1	1	0	
		1	0	1	0	1	0	
			1	1	1	1	0	
	1	0	1	0	0	1	0	
	1	1	1	0	0	0	1	
1	0	0	0	1	0	0	0	
		1	1	0	1	0	0	
1	1	0	0	0	0	1	1	
	1	1	0	1	1	1	0	
1	1	1	1	1	1	1	1	

Name ______________________________ Date ______________

Decimal → Binary Conversion

How to convert a decimal number (between 0 and 255) into binary.

Decimal

Binary

128	64	32	16	8	4	2	1

if value ≥ 128
  write '1' in the '128' position
  subtract 128 from value
else
  write '0' in the '128' position

if value ≥ 64
  write '1' in the '64' position
  subtract 64 from value
else
  write '0' in the '64' position

if value ≥ 32
  write '1' in the '32' position
  subtract 32 from value
else
  write '0' in the '32' position

if value ≥ 16
  write '1' in the '16' position
  subtract 16 from value
else
  write '0' in the '16' position

if value ≥ 8
  write '1' in the '8' position
  subtract 8 from value
else
  write '0' in the '8' position

if value ≥ 4
  write '1' in the '4' position
  subtract 4 from value
else
  write '0' in the '4' position

if value ≥ 2
  write '1' in the '2' position
  subtract 2 from value
else
  write '0' in the '2' position

if value = 1
  write '1' in the '1' position
else
  write '0' in the '1' position

The System Unit 2

Inside the Personal Computer:

1) **Power Supply**: All electricity enters your PC through this shielded metal box.

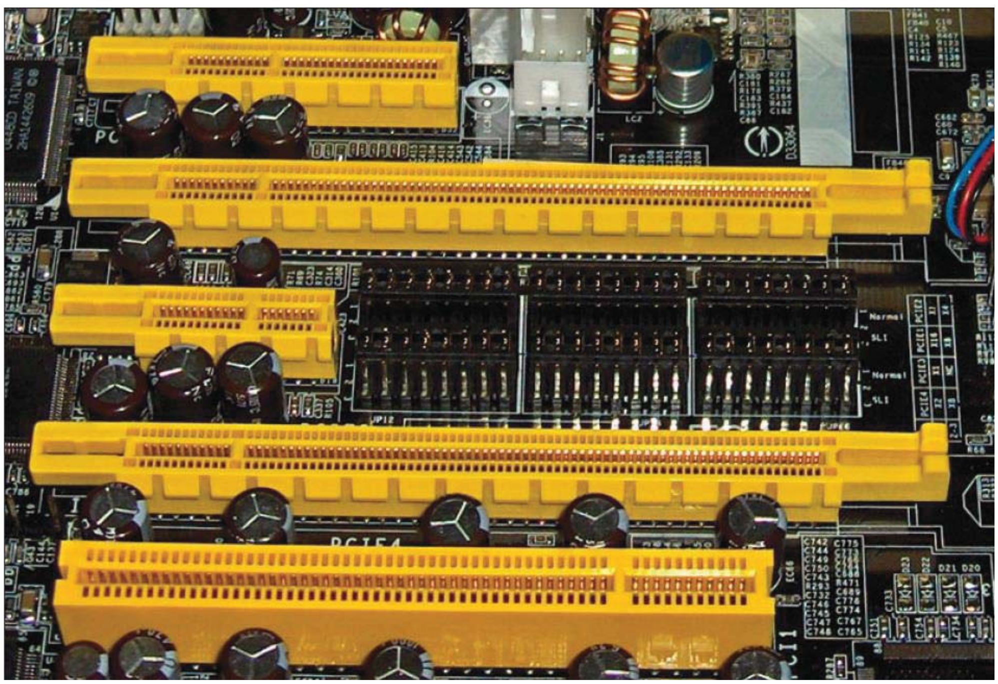
2) **CD-ROM/DVDROM Drive**: CD and DVD drives use a laser beam to read data from a spiral of indentation and flat areas on a layer of metallic film.

3) **Hard Drive:** This is the main repository-in the form of magnetic recordings on hard, thin platters of your programs and the documents on which you work.

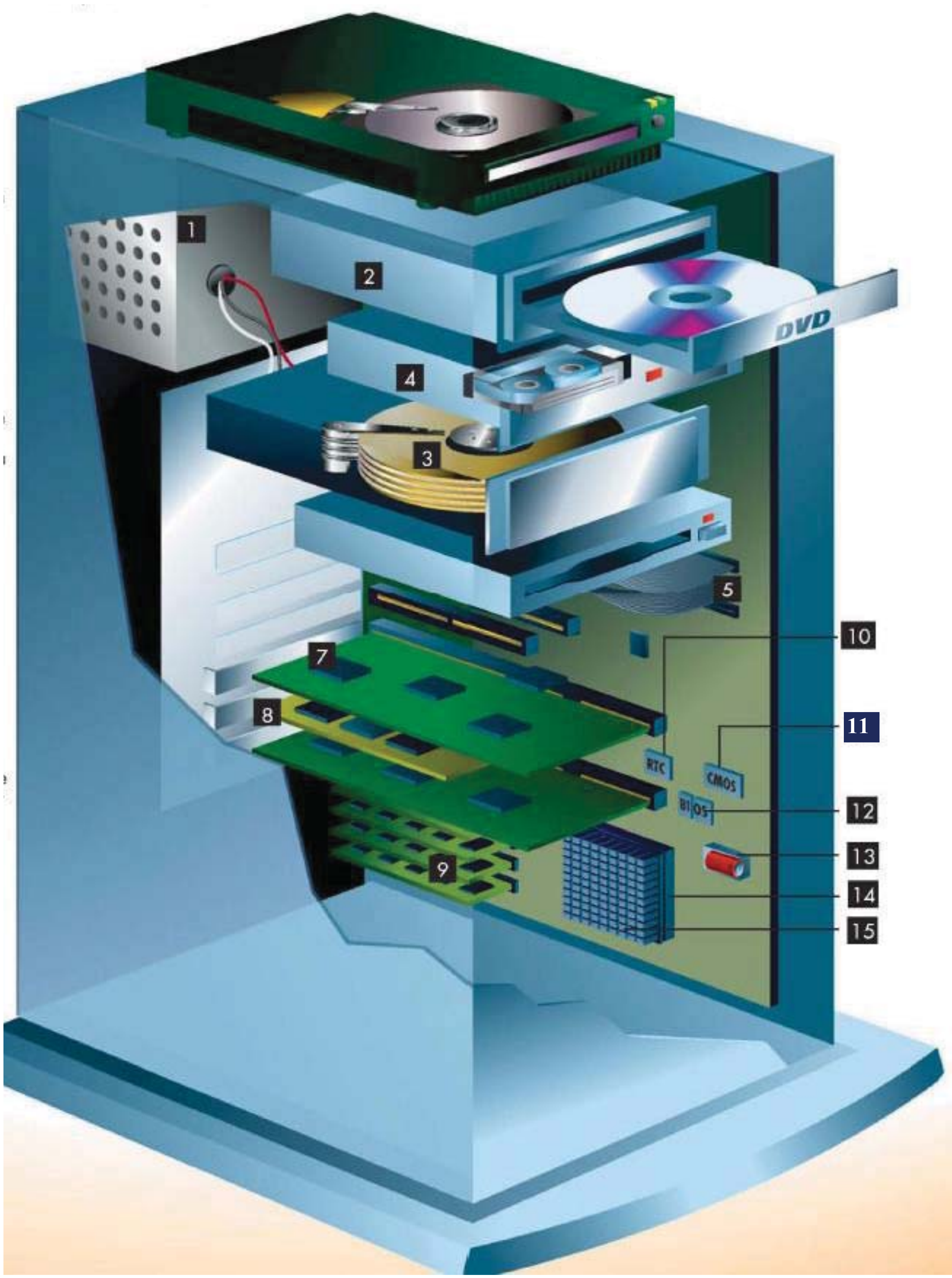
4) **Floppy Drive**: Here you insert a 3.5-inch floppy disk. Most floppy disks hold 1.44 MB of data.

5) **Disk Controllers**: The motherboards of most new PCs have two types of connections for passing data and intrusions to disk drives. The older IDE controller is used for floppy and optical drives, which are inherently slower that the controllers' ability to pass signals to the drives via flat, wide ribbons containing 40-80 wires. The newer Serial-ATA (SATA) connectors are reserved for hard drives, which take better advantage of the speed with which SATA passes information along a slim four-wire cable.

6) **Expansion Slot**: like disk controllers, expansion slots used to integrate new circuit boards into the motherboard.



7) **Video Card**: Translates image information into the varying electrical currents needed to display an image on the monitor.


1
2
DVD
4
3
5
10
7
11
8
RTC
CMOS
BIOS
12
13
9
14
15

8) **Sound Card**: Contains the circuitry for recording and reproducing multimedia sound.

9) **RAM**: Random Access Memory is a collection of microchips aligned on small circuit boards. RAM is where the computer stores programs and data while it uses them. When the computer is turned off, the contents of RAM are lost.

10) **Real-Time Clock**: A vibrating crystal in this component is the drummer that sets the pace and synchronizes the work of all the other components.

11) **CMOS**: This is a special type of memory chip that uses a small battery to retain information about your PC's hardware configuration even while the computer is turned off.

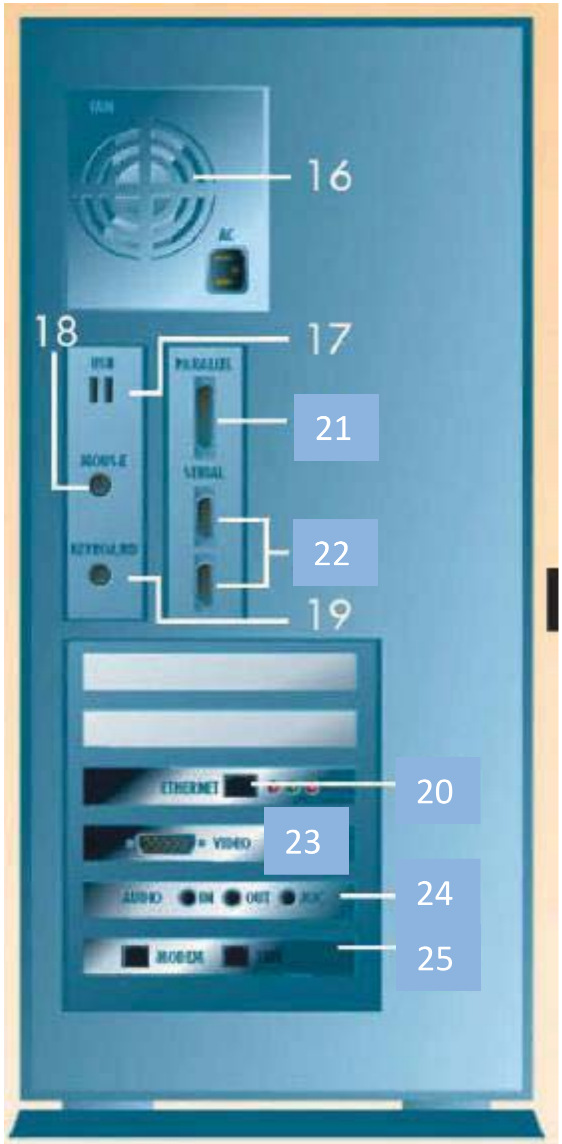
12) **BIOS**: If the microprocessor is your PC's brains, this is the heart. It is one or two chips that define the personality, or individuality, of your computer. The BIOS (Basic Input/Output System) knows the details of how your PC was put together and serves as an intermediary between the operating software running your computer and the various hardware components.

13) **CMOS Battery**: Rarely needs changing.

14) **Microprocessor**: Often called the brains of a computer, the microprocessor or central processing unit (CPU) is a tight, complex collection of transistors arranged so that they can be used to manipulate data.

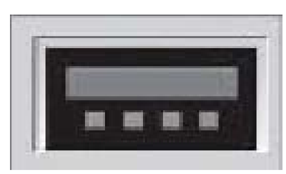
15) **Heat Sink**: Because microprocessors produce so much heat, a heat sink is used to dissipate the heat so that internal components of the chip don't melt.

The back side:



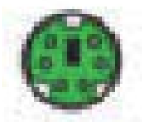
16) **Fan**: A fan built into the power supply draws cool air over the heat-critical components inside the case.

17) **USB Ports**: Universal serial bus ports are a solution to PCs' lack of interrupts and other system resources to let software connect directly to peripherals. USBs can connect keyboards, input devices (mice, trackballs, etc.), flash memory drives, printers, and other devices without encountering resource conflicts.

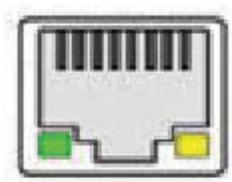


18) **Mouse Port**: Also called a PS2 port.

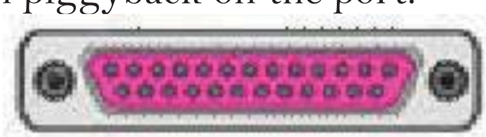
19) **Keyboard Port**: Looks like PS2.



20) **Network Connector**: The network connecter allows you to connect your PC to a local area network (LAN) or a broadband cable or DSl modem for high-speed Internet access.



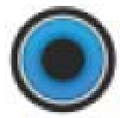
21) **Parallel Port**: Although falling into disuse, when the parallel port is used, it's most often to connect a printer, but some drives and other peripherals can piggyback on the port.



22) **Serial Ports**: Some PCs still have one or two serial ports, but they are all but obsolete because of the USB port.

23) **Video Card Connection:** to connect Monitor or Data show.

24) **Sound Card Connections**: External jacks on the sound card or motherboard enable you to attach a microphone, speakers, or an external sound source.

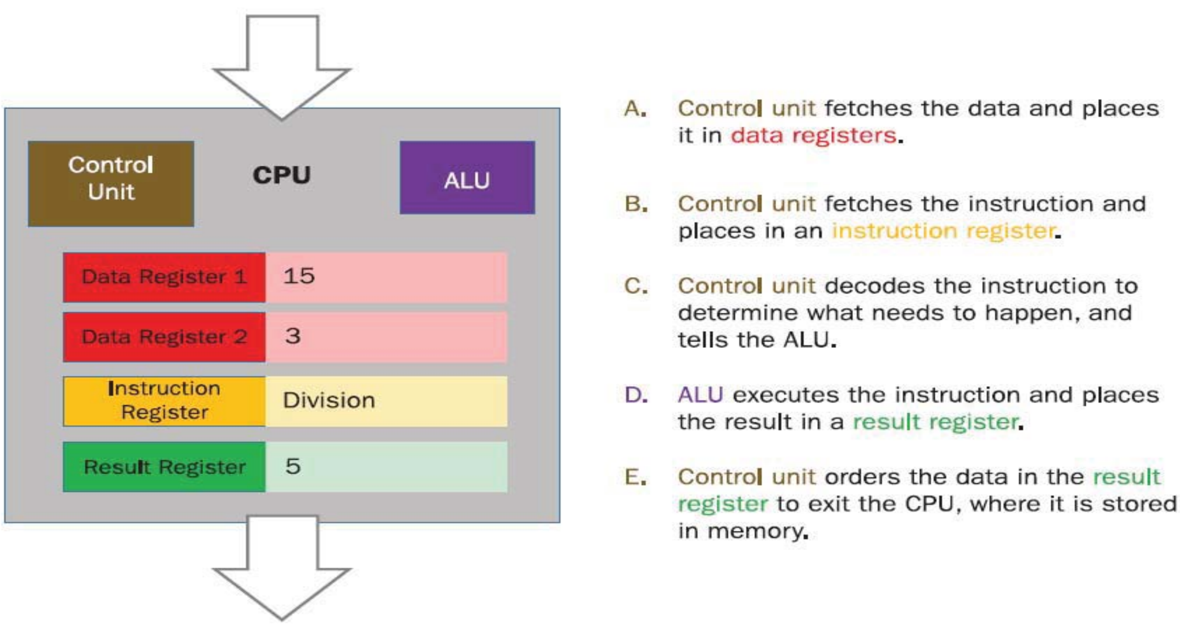


25) **Modem**: Connects your PC to a telephone line so that you can get to information services and the Internet. Modems also come as external devices that connect to serial port.

Central Processing Unit (CPU):

- **Control Unit**: Manages the flow of data through the CPU
- **Arithmetic Logic Unit (ALU)**: Does the actual processing
- **Registers**: Holding areas for data and instructions

Understanding CPUs:



Simple Computer Simulation (Activity) 3

Overview:

The purpose of this activity is to give the students a basic sense of how computers work by having them act out a simple computer simulation. Each student takes on the role of a different part of a simplified computer and they work in groups to run a simple program. The end result of this program is to draw a picture on a simulated computer display.

Description:

In this simulation of a (greatly simplified) computer, we consider a computer as being comprised of 3 major components:

- CPU (Central Processing Unit)
  This is the part that executes the program and tells the other components what they need to do.
- ALU/Memory (Arithmetic/Logic Unit & Memory)
  The ALU is the part of the computer that performs all the math and logic operations. The Memory keeps track of information so that it can be recalled later.
- Display
  The Display is the part that shows the results to the person using the computer.

We assign a student to each of these components and give them a simple program to run. The student acting as the CPU processes each

instruction in order and tells the ALU/Memory and the Display what to do.
While the program is being run, the Display should hide the image so that the CPU and the ALU/Memory have no idea what it being drawn on the screen.
Once the program is done, the Display shows the result to the other members of the group.

Note: that these component boundaries used in this activity are slightly different than what would be found in a real computer. For example, the ALU is actually part of the CPU, and the Memory is generally not bundled with the ALU. These divisions were chosen for this activity to help distribute the work done by each student while keeping the activity mostly accurate.

The roles for the students:

- CPU (Central Processing Unit)

  This student is given the program to run and is responsible for telling the other components (students) what they need to do.

- ALU/Memory (Arithmetic/Logic Unit & Memory)

  This student keeps track of the current values of x and y and performs any math operations requested by the CPU.

- Display

  This student responds to "plot" commands from the CPU by plotting the x,y values on the display grid.

Example:

This is a simple example with three students:
Student **A** is acting as the **ALU/Memory**
Student **C** is acting as the **CPU**
Student **D** is acting as the **Display**

Initial setup:

The group gets 3 worksheets: one each for the CPU, Display and ALU/Memory. Each student takes the sheet for their task and waits for student C (the CPU) to start.

Student D (the Display) needs to be arranged so that the other students cannot see the Display worksheet.

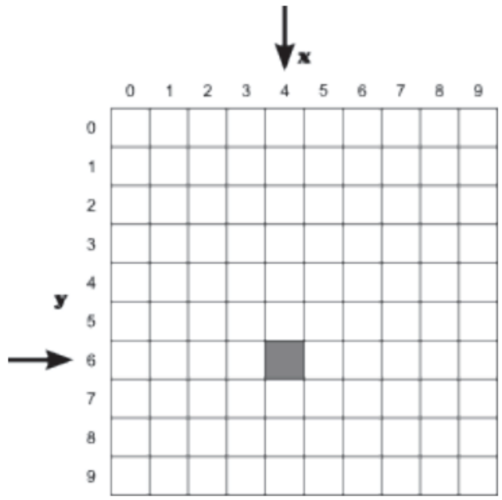
Student C starts by processing instructions (in order) from the program on the CPU sheet – giving the tasks to the other students as required.

For this example, assume that the selected program begins with the following instructions:

> Add 4 to x
> Add 6 to y
> Plot (x,y)
> Add 2 to x
> Subtract 3 from y
> Plot (x,y)

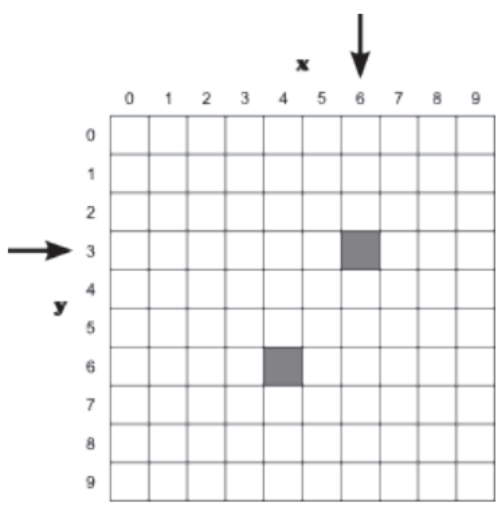
The group should handle these instructions as follows:

- C executes the 1st command by telling A to "Add 4 to x"
- A adds 4 to x and records the updated value in the x-column: 4
- C executes the 2nd command by telling A to "Add 6 to y"
- A adds 2 to y and records the updated value in the y-column: 6
- C executes the 3rd command by:
    - Requesting the current values of x and y from A
    - Telling D to plot the x,y values
- D plots the (x,y) values by:
    - Finding the column that corresponds to the x-value (4) and
    - Finding the row that corresponds to the y-value (6) and
    - Filling the square at the intersection.



The group continues executing the instructions:

- C executes the 4th command by telling A to "Add 2 to x"
- A adds 2 to x and records the updated value in the x-column: 6 (= 4 + 2)
- C executes the 5th command by telling A to "Subtract 3 from y"
- A subtracts 3 from y and records the updated value in the y-column: 3 (= 6 - 3)
- D plots the (x,y) values by:
    - Finding the column that corresponds to the x-value (6) and
    - Finding the row that corresponds to the y-value (3) and
    - Filling the square at the intersection.



...and so on with the remaining instructions.

Follow-up Discussion:

The purpose of the simulation exercise is to give the students a small taste of what it is that computers do.

The single-most important "take-away" from this exercise is the following:

1. The computer had no idea what was being drawn on the display - it was just mindlessly following the instructions in the program.
2. Computers do not "understand" what they are doing. Drawing a picture of a cat is the same (from the computer's perspective) as drawing a picture of a dog - it's just a series of instructions to execute.
3. Since computers are simply executing the instructions in the program, it is the programmer's responsibility to write the program correctly. If there is a mistake in the program, the computer will still go ahead and try to execute the program as written.

Other possible discussion topics:

- What happens if the CPU tells the display to plot an x or y value that is greater than 9?
    - It could be ignored (not drawn)
    - It could report an error
    - It could be drawn in some other part of the screen
        - On some systems, plotting a large x value would draw the pixel at the beginning of the next row.

- What happens if the CPU is faster than the display?
    - On a real computer system, the CPU generally doesn't wait for the display, so the display will either queue up the drawing commands (and seem sluggish as it struggles to catch up) or it will drop some of the commands (and the display will not be accurate).

- What happens if the display is faster than the CPU?
  - Then it sits around waiting for the next drawing command. This is preferable to the previous situation since the display will always be accurate.

- Note that the computer's display has the origin (x=0, y=0) in the upper left corner and y-values increase as they work down the page. Normal math plots have the origin in the lower left corner and the y-values increase as they work up the page.
  - Why are they different?
    - Traditional math plots work up from the bottom since this is consistent with how we see the world around us. We start from ground-level and measure how tall things are: trees, buildings, ...
    - Computer displays are modeled after how we write on a sheet of paper: we start at the top and move down to the next line when we finish a row.
    - (Of course, computer programs can be written that plot points in the traditional math way.)
  - Which method is better?
    - Ha! Trick question. They both work perfectly fine. You just need to pick one and be consistent.

Name ______________________________ Date ______________

CPU - Program 1

How Computers Work

Greetings CPU! Your job as the **Central Processing Unit** (CPU) is to execute the following program and tell the other components what they need to do:

- Math commands like "Add 5 to x" or "Subtract 3 from y" should be sent to the ALU.
- Drawing commands like "Plot(x,y)" should be sent to the Display.
  First, ask the Memory for the current values of x and y, then give them to the Display.

Be sure to check off each line of the program as you finish it (so that you don't lose track of where you are).

- ☐ Add 4 to x
- ☐ Add 2 to y
- ☐ Plot (x,y)
- ☐ Add 3 to x
- ☐ Add 2 to y
- ☐ Plot (x,y)
- ☐ Subtract 6 from x
- ☐ Plot (x,y)
- ☐ Add 5 to x
- ☐ Subtract 3 from y
- ☐ Plot (x,y)
- ☐ Subtract 1 from x
- ☐ Add 5 to y
- ☐ Plot (x,y)
- ☐ Subtract 3 from x
- ☐ Subtract 1 from y
- ☐ Plot (x,y)
- ☐ Subtract 4 from y
- ☐ Plot (x,y)
- ☐ Subtract 1 from x
- ☐ Add 1 to y
- ☐ Plot (x,y)
- ☐ Add 4 to x
- ☐ Subtract 1 from y
- ☐ Plot (x,y)
- ☐ Add 1 to x
- ☐ Add 4 to y
- ☐ Plot (x,y)
- ☐ Subtract 3 from x
- ☐ Add 1 to y
- ☐ Plot (x,y)
- ☐ Subtract 5 from y
- ☐ Plot (x,y)
- ☐ Add 4 to x
- ☐ Add 2 to y
- ☐ Plot (x,y)
- ☐ Subtract 6 from x
- ☐ Plot (x,y)
- ☐ Add 3 to x
- ☐ Add 4 to y
- ☐ Plot (x,y)
- ☐ Add 3 to x
- ☐ Subtract 5 from y
- ☐ Plot (x,y)

DONE!

Name ______________________________ Date ______________

ALU/Memory

How Computers Work

Greetings ALU/Memory! Your job as the **Arithmetic/Logic Unit** (ALU) is to perform simple addition and subtraction operations on **x** and **y** as requested by the CPU. In addition, your job as the computer **Memory** is to keep track of the current values of **x** and **y**, so that you can tell the CPU what they are when asked.

Use the table below to record the values. Initially, both **x** and **y** are set to 0.

x	y
0	0

Name ______________________________ Date ______________

Display

How Computers Work

Greetings Display! Your job is to wait until the CPU tells you to plot an (**x**, **y**) point on the screen. When you are asked to plot a point, find the column that corresponds to the **x**-value and the row that corresponds to the **y**-value and plot the intersection by filling in the square.

Don't show the image to your group partner(s) until the CPU has finished running the entire program.

x

y	0	1	2	3	4	5	6	7	8	9
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										

Name ______________________________ Date ______________

CPU - Program 2

How Computers Work

Greetings CPU! Your job as the **Central Processing Unit** (CPU) is to execute the following program and tell the other components what they need to do:

- Math commands like "Add 5 to x" or "Subtract 3 from y" should be sent to the ALU.
- Drawing commands like "Plot(x,y)" should be sent to the Display.
  First, ask the Memory for the current values of x and y, then give them to the Display.

Be sure to check off each line of the program as you finish it (so that you don't lose track of where you are).

- ☐ Add 5 to x
- ☐ Add 3 to y
- ☐ Plot (x,y)
- ☐ Add 1 to x
- ☐ Add 3 to y
- ☐ Plot (x,y)
- ☐ Subtract 3 from x
- ☐ Subtract 1 from y
- ☐ Plot (x,y)
- ☐ Subtract 2 from x
- ☐ Subtract 2 from y
- ☐ Plot (x,y)
- ☐ Add 4 to x
- ☐ Subtract 2 from y
- ☐ Plot (x,y)
- ☐ Add 2 to x
- ☐ Add 2 to y
- ☐ Plot (x,y)
- ☐ Subtract 3 from x
- ☐ Add 4 to y
- ☐ Plot (x,y)
- ☐ Add 1 to x
- ☐ Plot (x,y)
- ☐ Subtract 2 from x
- ☐ Plot (x,y)
- ☐ Subtract 2 from x
- ☐ Subtract 2 from y
- ☐ Plot (x,y)
- ☐ Subtract 1 from y
- ☐ Plot (x,y)
- ☐ Add 6 to x
- ☐ Plot (x,y)
- ☐ Add 1 to y
- ☐ Plot (x,y)
- ☐ Subtract 2 from x
- ☐ Plot (x,y)
- ☐ Subtract 3 from x
- ☐ Add 1 to y
- ☐ Plot (x,y)
- ☐ Subtract 4 from y
- ☐ Plot (x,y)
- ☐ Add 4 to x
- ☐ Plot (x,y)
- ☐ Subtract 3 from x
- ☐ Add 1 to y
- ☐ Plot (x,y)
- ☐ Subtract 2 from y
- ☐ Plot (x,y)
- ☐ Add 1 to x
- ☐ Plot (x,y)
- ☐ Add 4 to y
- ☐ Plot (x,y)

DONE!

Name ______________________________________ Date _______________

ALU/Memory

How Computers Work

Greetings ALU/Memory! Your job as the **Arithmetic/Logic Unit** (ALU) is to perform simple addition and subtraction operations on **x** and **y** as requested by the CPU. In addition, your job as the computer **Memory** is to keep track of the current values of **x** and **y**, so that you can tell the CPU what they are when asked.

Use the table below to record the values. Initially, both **x** and **y** are set to 0.

x	y
0	0

Name ____ Date ________________

Display

How Computers Work

Greetings Display! Your job is to wait until the CPU tells you to plot an (**x**, **y**) point on the screen. When you are asked to plot a point, find the column that corresponds to the **x**-value and the row that corresponds to the **y**-value and plot the intersection by filling in the square.

Don't show the image to your group partner(s) until the CPU has finished running the entire program.

x

y	0	1	2	3	4	5	6	7	8	9
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										

How a PC Comes Alive

4

Personal computer can't do anything useful unless it's running an **operating system**-a basic type of software, such as Microsoft Windows that acts as a supervisor for all the applications, games, or other programs you use.

The operating system sets the rules for using memory, drives, and other parts of the computer. But before a PC can run an operating system, it needs some way to load the operating system from disk to **random access memory (RAM)**. That what we called it bootstrap or simply boot.

The boot operation has two functions:

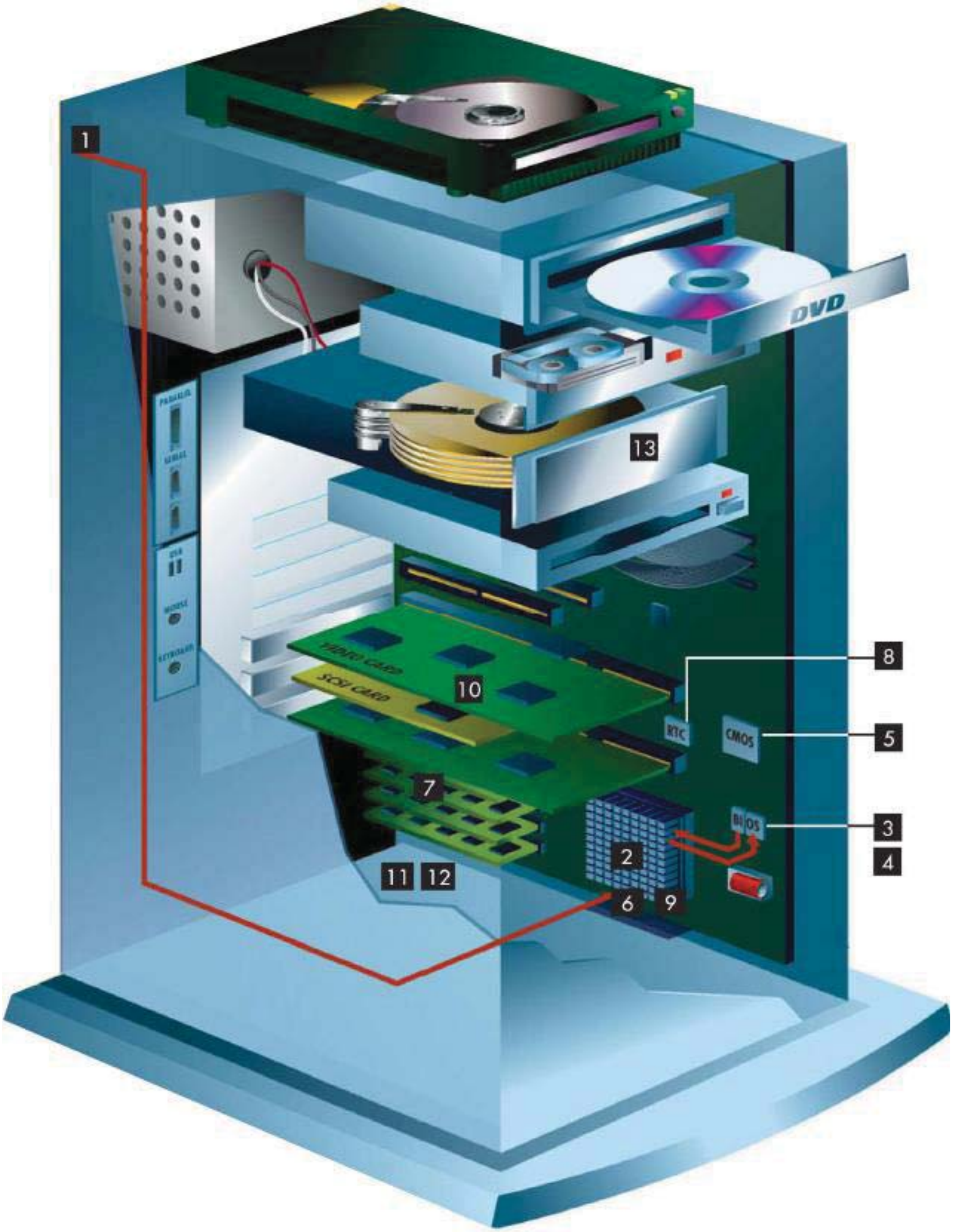
1) Run a **POST**, or **power-on self-test**
2) Search drives for an operating system.

When these functions are complete, the boot operation launches the process of reading the system files and copying them to random access memory.

How the Power-On Self-Test Gets Your PC Started:

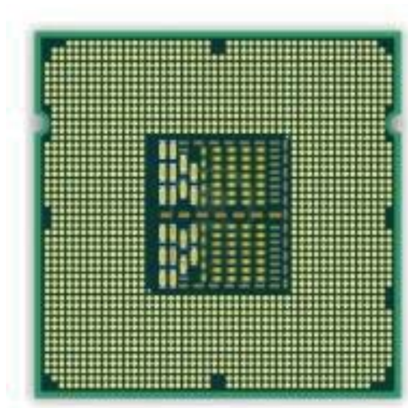
1

- When you turn the power on your PC, electricity begins to warm up the parts that will be used.
- A narrow stream of electricity follows a small channel to the microprocessor (CPU).


1
DVD
13
8
RTC
CMOS
5
VIDEO CARD
SCSI CARD
10
7
2
3
4
11
12
6
9

2

- The channel of electricity arrives at the CPU, which acts as the "brain and boss" of the system.
- The electrical signal gets rid of old data from the internal memory chip.
- The signal then places F000 (a specific Hexa-decimal) into the CPU's digital notepad called the **program counter**.



3

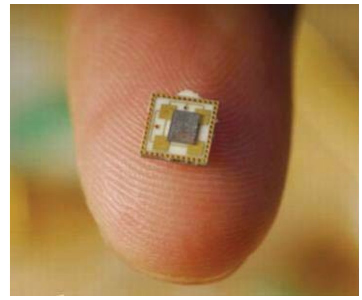
- The number in the program counter goes on to tell the CPU the memory address of the first instruction, which is found in the flash memory chip the motherboard.
- The Flash Memory Chip holds a few small programs and settings that determine how your computer will work.
- Together they are the BIOS, or the computer's **Basic Input- Output System BIOS** Chip.

4

- The BIOS triggers the **Power-On Self-Test (POST)** to check that the computer is functioning properly.
- Your computer screen is still black, but you now begin to see flashing LED lights, and hear noises.

5

- The BIOS then checks a 64 byte piece of RAM on a **complementary metal oxide semiconductor (CMOS)** chip that is always on.
- The CMOS has memory of what is installed on your computer.



- As the BIOS continues the POST it checks the received information CMOS Chip.

6

- The BIOS tells the CPU to read a code found at various locations and then compares that code with what is stored in the BIOS.
- They basically make sure that they're working right. Is Everything Working Right?

7

- BIOS loads into memory the **device drives** and **interrupt handlers** from hardware in the system (keyboard, mouse, hard drive, floppy drive).

- Device drive translates the code as need for the CPU to understand it.
- The interrupt handlers are responsible for bringing CPU's attention to the code waiting for the microprocessor.

8

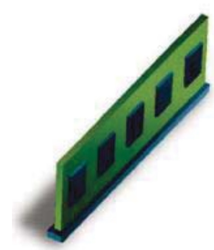
- CPU checks the system's Real Time Clock, which is in charge of pacing signals.

9 • The BIOS tells the CPU to send signals to the **system bus** (which connects components on the motherboard) to make sure everything is functioning. The bus includes the electrical circuits printed on and into the motherboard, connecting all the components with each other.

10 • The POST tests the memory on the display adapter and video signals that control the display.
• The BIOS code becomes part of the system's overall BIOS an memory.
• Your computer will finally display something.

11 • The BIOS checks to see if its in a "**cold boot**" (if the computer was off) or a "**warm boot**"/"**reboot**" by checking the memory address.
• If it is a reboot (if it finds the number 1234) it skips the rest of POST.

12 • If it is in a cold boot, the BIOS tests the RAM to make sure it is working right.
• These tests write data to each chip, then compare that data to the previous data
• On some computers you will now see an account of the computer’s memory that has been checked.



13 • The POST signals the floppy, optical and hard disk drives to determine what drives are available for use.
• POST ends.

The BIOS transfers the control of the PC to the operating system on the hard disk, a process called the boot. ➔ **POST is complete**

Operating Systems 5

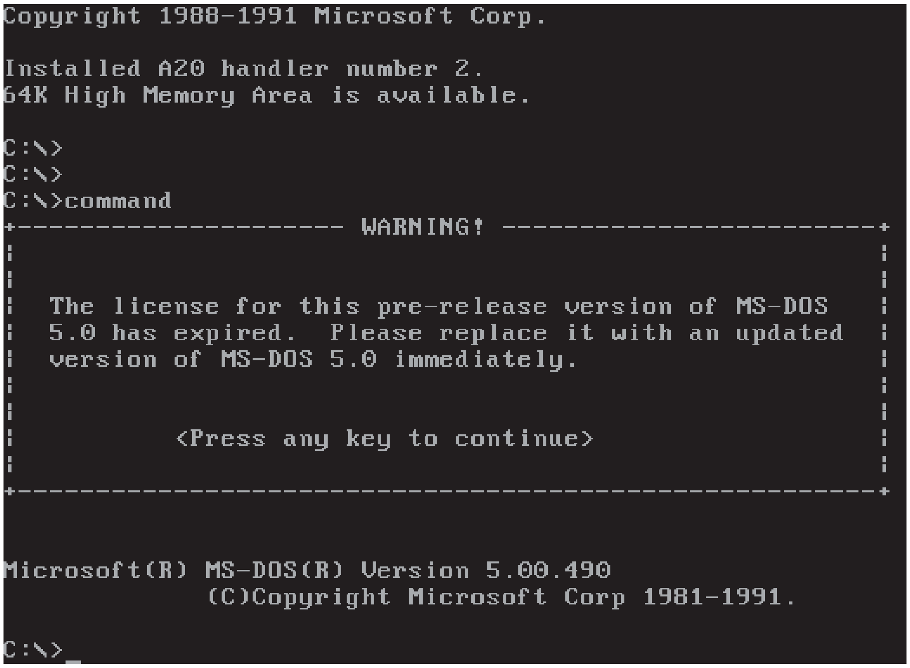
Operating System (OS): is software on the hard drive that enables the computer hardware to communicate and operate with the computer software.
Without a computer operating system, a computer and software programs would be useless.

What does operation system do?

- Provides the user interface.
- Communicates with the hardware.
- Runs applications.
- Manages files.

Example of operating systems:

1) <u>**MS-DOS (Microsoft Disk Operating System)**</u>
   - Command-line interface (CLI).
   - Single-Tasking system.
   - Started in 1981 by Microsoft for the IBM-Compatible computers.
   - Today, MS-DOS is no longer used.



MS-DOS interface

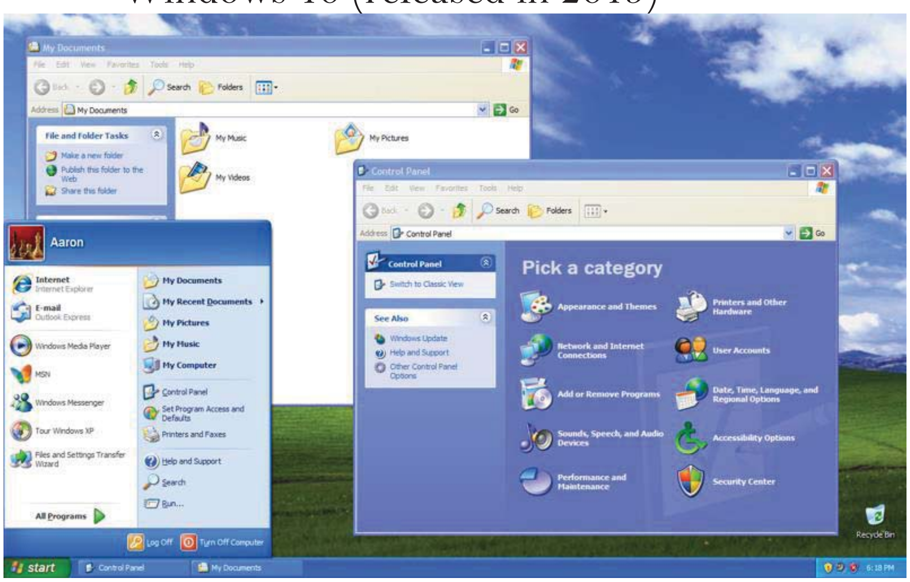
2) <u>Mac OS (Apple Macintosh operating system)</u>

- Graphical User Interface (GUI), It is the first GUI system ever.
- Multi-Tasking system.
- Started in 1984 by Apple for the Macintosh computers.
- Apple introduced macOS X 10.10; named Yosemite in 2014.

MacOS interface

3) <u>Microsoft Windows</u>

- Graphical User Interface (GUI).
- Multi-Tasking system.
- Started in 1985 by Microsoft for the IBM-Compatible computers.
- Microsoft released the latest version Windows 10 in 2015.
- The most famous versions of this system are:
  - Windows 3.1 (released in 1992)
  - Windows 3.11 (released in 1993)
  - Windows 95 (released in 1995)
  - Windows 98 (released in 1998)
  - Windows ME (released in 2000)
  - Windows XP (released in 2001)
  - Windows Vista (released in 2006)
  - Windows 7 (released in 2007)
  - Windows 8 (released in 2012)
  - Windows 10 (released in 2015)



Windows XP interface

4) <u>Linux</u>

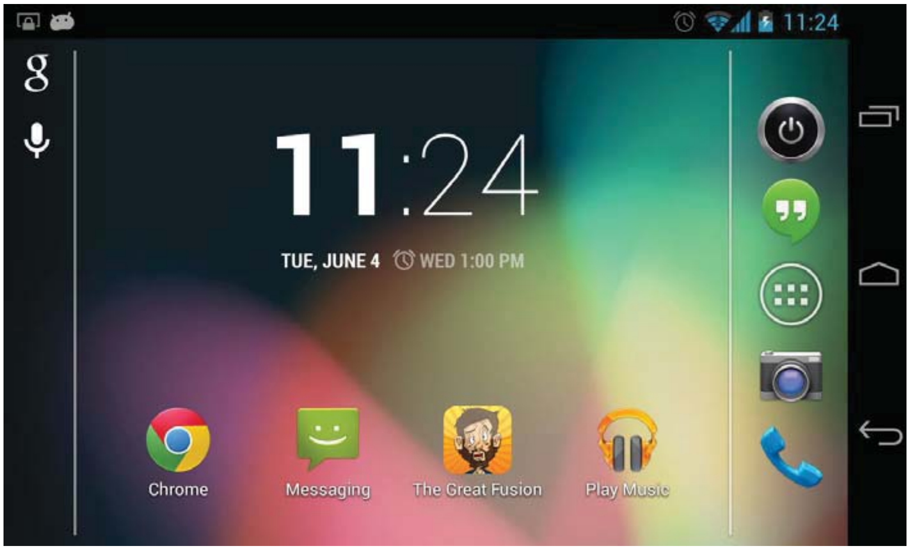
- Graphical User Interface (GUI).
- Multi-Tasking system.

- It is a free and open-source operating system.
- Linux is introduced by Linus Torvalds, a student in Finland in 1991. And it is based on UNIX operating system which released in1969 (the same year of Linus' birth).

Linux interface

5) <u>Android OS</u>

- Graphical User Interface (GUI).
- Multi-Tasking system.
- Originally founded by Andy Rubin in 2003 and later acquired by Google in 2005.
- Android is a free Linux based platform, redeveloped by Google released in 2007 for mobile platforms.
- The most famous versions of this system are:
  - Android (Cupcake) released in 2009
  - Android (Ice Cream Sandwich) released in 2011
  - Android (KitKat) released in 2013
  - Android (Oreo) released in 2017



Android interface

6) <u>iOS (iPhone OS)</u>

- Graphical User Interface (GUI)
- Multi-Tasking system.
- Runs on Apple iPhone, iPad, and iPod.
- Released by Steve Jobs in 2007.
- The latest version iOS 10 released in 2017.

iOS interface

Device Drivers: it is a software that

- Translate the OS requests into the language of the device.
- Translates messages from the device back to the OS.
- Published by the device manufacturer for a specific operating system and device model.

File Extensions and Types

$$\underbrace{\textbf{xxxxxxxx}}_{\text{File name}}\ \textbf{.}\ \underbrace{\textbf{xxx}}_{\text{Extension}}$$

- Extension: characters after the period in a file name that indicate the file's type
- Extensions tell the OS which application to use to open a data file
- Extensions tell the OS which files are executable (.exe or .com)

Extension	*File Type*	*Associated Application*
txt	Text	Notepad, WordPad, Microsoft Word
gif, png, jpg, tif	Photo or graphic	Paint, Photoshop, or almost any other photo editing program
doc, docx, docm	Word processing document	Microsoft Word, some other word processing programs also support
rtf	Word processing document	WordPad, Microsoft Word, or almost any word processing program
xls, xlsx, slxm	Spreadsheet	Microsoft Excel
ppt, pptx, pptm	Presentation	Microsoft PowerPoint
mdb, accdb	Database	Microsoft Access
pdf	Portable document format (platform-independent formatted document)	Adobe Reader, Adobe Acrobat, limited support in Microsoft Word
xps	XML document format (Microsoft-specific platform-independent formatted document)	XPS Viewer, Windows 7, Windows 8, limited support in Microsoft Word
exe, com, bat	Executable program files	n/a
dll, ini, dat	Helper files for programs and for Windows itself	n/a
zip	Compressed archive file	Windows Explorer, or a third-party program such as WinZip