What is Computer?

1. A computer is a programmable machine that receives input, stores and manipulates data, and provides output in a useful format.

2. Computer is a machine for performing calculations automatically

3. Computer is a electronic machine capable of performing calculations and other manipulations of various types of data, under the control of a stored set of instructions. The machine itself is the hardware; the instructions are the program or software.

What is Computer Hardware?

Computer hardware refers to the physical parts of a computer and related devices. Internal hardware devices include motherboards, hard drives, and RAM. External hardware devices include monitors, keyboards, mice, printers, and scanners. The internal hardware parts of a computer are often referred to as components, while external hardware devices are usually called peripherals. Together, they all fall under the category of computer hardware.

What is Software?

A computer cannot do anything on its own. It must be instructed to do a desired job. Hence, it is necessary to specify a sequence of instruction, which a computer must perform to solve a problem. Such a sequence of instructions, written in a language, which can be understood by a computer, is called a computer program.

The term software refers to the set of computer programs, procedures, and associated, documents (flowcharts, manuals, etc.) which describe the programs, and how they are to be used.

A software package is a group of programs, which solve a specific problem or perform a specific type of job. For example, a word-processing package may contain programs for text editing, text formatting, drawing graphics, spelling checking etc.

Relationship between Hardware and Software:

In order for a computer to produce useful output, its hardware and software must work together. Nothing useful be done with the computer hardware on its own, and software cannot be utilized without supporting hardware. Both hardware and software are necessary for a computer to do useful job. Both are complementary to each other.

Types of Software:

Although the range of software available today is vast and varied, most software can be divided into two major categories:

1. System software, and

2. Application software

System Software:

System software is a set of one or more programs, designed to control the operation and extend the processing capability of a computer system. In general, a computer's system software performs one or more of the following functions:

1. Supports the development of other application software
2. Supports the execution of other application software
3. Monitors the effective use of various hardware resources
4. Communicates with and controls the operations of peripherals devices

Some of the most commonly known types of System Software are:

1. Operating Systems. Every computer has operating system software, which takes care of the effective and efficient utilization of the hardware and software components of the computer system.

2. Programming Language Translators. Programming language translator is system software which transforms the instruction prepared by programmer in a programming language, into a form which can be interpreted and executed by a computer system.

3. Communication Software. In a network environment (where multiple computer are interconnected together by communication network), communication software enables transfer of data and programs from computer system to another.

4. Utility Programs. Utility programs (also known as utilities) are a set of programs, which help user in system maintenance tasks, and in performing tasks of routine nature.

Application Software:

Application software is a set of one more programs, designed to solve a specific problem, or do a specific task For example, application software for payroll processing produces pay slips as the major output, and an application software for processing examination results produces mark sheets as the major output along with some other statistical reports.

The programs included in an application software package are called application programs, and the programmers who prepare application software are referred to as application programmers.

Word-processing Software:

A word processing software enables us to make use of a computer system for creating, editing, viewing, formatting, storing, retrieving and printing documents.

Spreadsheet Software:

A spreadsheet software is a numeric data analysis tool, which allows us to create a kind of computerized ledger. A manual ledger is a book having rows and

columns, which accountants use for keeping a record of financial transactions, and for preparing financial statements.

Database Software:

A database is a collection of related data stored and treated as a unit for information retrieval purposes. A database software is a set of one or more programs, which enable us to crate a database, maintain it (add, delete and update its records), organize its data in desired fashion, and to selectively retrieve useful information from it.

Graphics software:

Graphic software enables us to use a computer system for creating, editing viewing, storing, retrieving and printing designs, drawing, pictures, graphs and anything else that can be drawn in the traditional manner.

Personal assistance software:

A personal assistance software allows us to use personal computers for storing and retrieving our personal information, and planning and managing our schedules, contacts, financial and inventory of important items.

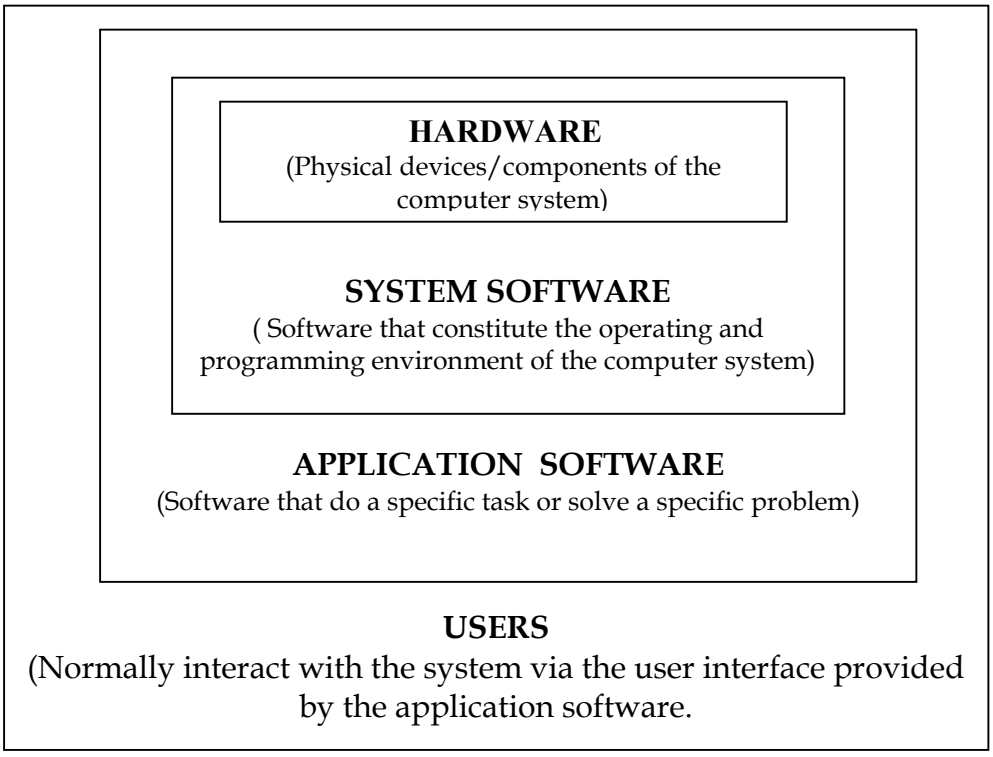
Education software:

Education software allows a computer system to be used as a teaching and learning tool. A few examples of such applications are those that teach young children to do (a) mathematics (b) recognize alphabets and (c) read whole words and sentences.

Entertainment software:

Entertainment software allows a computer system to be used as an entertainment tool. A good example of such an application is computers video games.

Logical System Architecture



(Relationship among the hardware, system software,
Application software and user of computer system)

Software Development Steps:

Developing software and putting it to use is a complex process, which involves the following steps:

- Analyzing the problem at hand, and planning the program(s) to solve the problem.
- Coding the program(s)
- Testing, debugging and documenting the program(s)
- Implementing the program(s)
- Evaluating and maintaining the program(s)

Firmware:

Computer software in conventional systems is supplied on storage media like CDs, floppies, tapes, disks, etc. However, with the advancement in technology, and the reduction in hardware cost, today, software is also being made available by many computer manufacturers on read-only memory (ROM) chips. These ROM chips can be easily plugged into the computer system, and they form a part of the hardware. Such programs, which are made available on hardware, are known as firmware.

Difference of Hardware and Software:

Hardware	Software
Hardware refers to the physical components of the computer that run the software.	Software refers to things that are used by the hardware, such as programs that you install on your computer including games, word processing programs, spreadsheet programs, graphic design programs and the like.
Hardware includes the physical components, such as the motherboard, chips, memory, and hard drives,	Software includes the programs that run on the hardware.
Hardware starts functioning once software is loaded.	To deliver its set of instructions, Software is installed on hardware.
Hardware is a physical device something that you're able to touch and see. For example, the computer monitor you're viewing this text on or the mouse you're using to navigate is considered computer hardware.	Software is code and instructions that tell a computer and/or hardware how to operate. This code can be viewed and executed using a computer or other hardware device.
Hardware includes every computer-related object that you can physically	Software includes every computer-related program that you cannot feel

touch and handle like disks, screens, keyboards, printers, chips, wires, central processing unit, floppies, USB ports, pen drives etc.	with the physical senses for example, operating system, an anti-virus program, the web browser etc.

Basic Computer Organization

All computer systems perform the following five basic operations, for converting raw input data into information, which is useful to their users:

1. **Inputting:** The process of entering data and instructions into the computer system.

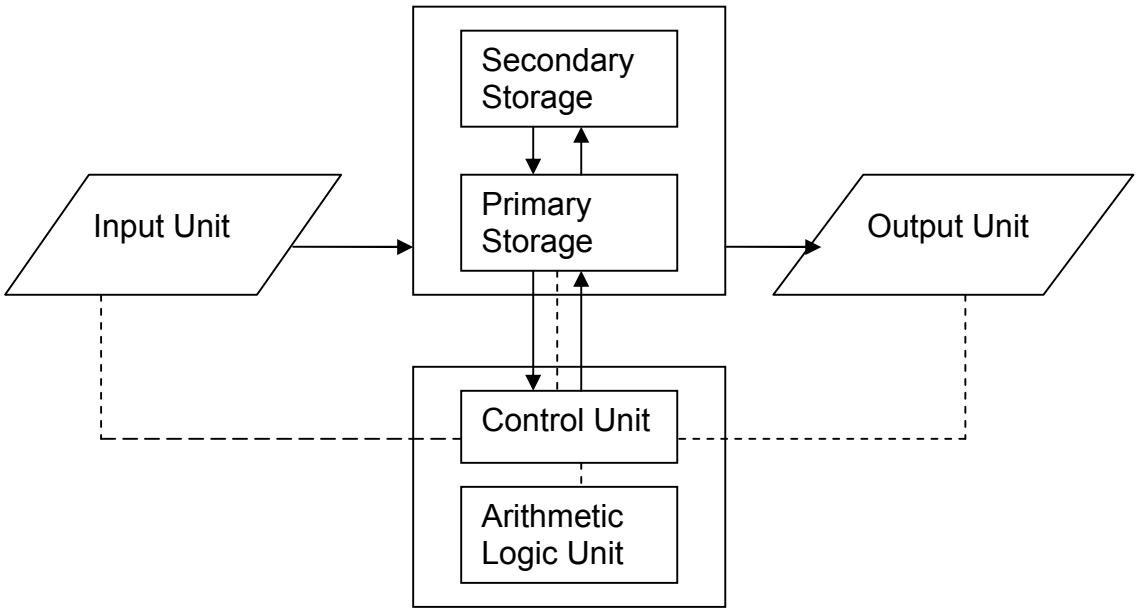
2. **Storing:** Saving data instructions to make them readily available for initial or additional processing, as and when required.

3. **Processing: P**erforming arithmetic operations (add, subtract, divide, etc.), or logical operations (comparisons like equal to, less than, grater than, etc.) on data, to convert them into useful information.

4. **Outputting:** The process of producing useful information or results for the user, such as a printed report or visual display.

5. **Controlling:** Directing the manner and sequence in which all of the above operations are performed.

A **block diagram of the basic computer organization** is as shown below figure. In this figure, the solid lines indicate the flow of instruction and data, and the dotted lines represent the control exercised by the control unit.



Central Processing Unit (CPU)
Basic Organization of a Computer System

It displays the five major building blocks (functional units) of a digital computer system. These five units correspond to the basic operations, performed by all computer systems.

INPUT UNIT:

Data and instructions must enter the computer system, before any computation can be performed on the supplied data. This task is performed by the input unit, which links the external environment with the computer system. Data and instructions enter input units in forms, which depend upon the particular device used. Input interfaces are designed to match the unique physical or electrical characteristics of input devices, to the requirements of the computer system.

The following functions are performed by an input unit:

1. It accept (or reads) the instructions and data from the outside world.
2. It converts these instruction and data in computer acceptable form.
3. It supplies the converted instructions and data to the computer system for further processing.

Some examples of input devices are:

1. Keyboard
2. Point and Draw devices
    a) Mouse
    b) Trackball
    c) Joystick
    d) Electronic pen
    e) Touch Screen

3. Data scanning devices
    a) Image Scanner
        Flatbed Scanner
        Hand held scanner
    b) Optical character Recognition (OCR)
    c) Optical Mark Reader (OMR)
    d) Bar Code Reader
    e) Magnetic Ink Character Reader (MICR)

4. Digitizer
5. Electronic cards based devices
6. Voice recognition devices
7. Vision based devices

OUTPUT UNIT:

The job an output unit is just the reverse of that of an input unit. It supplies the information obtained from data processing, to the outside world. Hence it links the computer with the external environment. As computer work with binary code, the

results produced are also in the binary form. Hence before supplying results to the outside world, they must be converted to human acceptable form. This task is accomplished by units called output interfaces. Output interfaces are designed to match the unique physical or electrical characteristics of output devices to the requirements of the external environment.

The following function is performed by an output unit:

1. It accepts the results produced by the computer, which are in coded from, and hence, cannot be easily understood by us.
2. It converts these coded results to human acceptable (readable) form.
3. It supplies the converted results to the outside world.

Some examples of output devices are:

1. Monitors
2. Printers
    a) Dot-Matrix printers
    b) Inkjet printers
    c) Drum printers
    d) Chain/Band printers
    e) Laser printers
3. Plotters
    a) Drum plotter
    b) Flatbed plotter
4. Screen image projector
5. Voice response systems

STORAGE UNIT:

The data and instructions, which are entered into the computer system through input units, have to be stored inside the computer, before the actual processing starts. Similarly, the results produced by the computer after processing, must also be kept somewhere inside the computer system, before being passed on to the output units.

The specific function of the storage unit is to hold (store):

1. The data and instructions required for processing (received from input devices).
2. Intermediate results of processing.
3. Final results of processing, before these results are released to an output device.

The storage unit of all computers is comprised of the following two types of storage:

1. Primary storage:

The primary storage, also known as main memory, is used to hold pieces of program instructions and data, processing, of the job(s), which the computer system is currently working on. These pieces of information are represented electronically in the main memory chip’s circuitry, and while it remains in the main memory, the

central processing unit can access it directly at a very fast speed. However, the primary storage can hold information only while the computer system is on. As soon as the computer system is switched off or reset, the information held in the primary storage disappears. Moreover, the primary storage normally has limited storage capacity, because it is very expensive. The primary storage of modern computer system is made up of semiconductor devices.

RAM:

Random access memory or *RAM* most commonly refers to computer chips that temporarily store dynamic data to enhance computer performance. By storing frequently used or active files in random access memory, the computer can access the data faster than if it to retrieve it from the far-larger hard drive. Random access memory is also used in printers and other devices.

Masked ROM:

In this type of ROM bits are stored permanently by marking and metallization process. This is done by manufacturers. This type of ROM can be programmed only one-by the manufacture.

PROM:

Data are written into a ROM at the time of manufacture. However, a programmable ROM (PROM) allows the data to be loaded by the user, by connecting a fuse between the emitter and the bit-line. PROMs provide flexible and economical storage for fixed programs and data, where high production volumes are involved.

EPROM:

The Erasable PROM chip allows the stored data to erase and new data can be reprogrammed. It provides more flexibility during the development phase of digital system. The contents of EPROM cells can be erased by increasing the discharge rate of the storage capacitors.

EEPROM:

In an electricity erasable PROM, the contents of cells can be erased by the application of a high voltage. Advantages with EEPROMs are: it need not be physically removed for reprogramming and the process can be made selective since electrical erasure is used.

Cache Memory:

A **CPU cache** is a cache used by the central processing unit of a computer to reduce the average time to access memory. The cache is a smaller, faster memory which stores copies of the data from the most frequently used main memory locations.

Cache memory is random access memory (RAM) that a computer microprocessor can access more quickly than it can access regular RAM. As the microprocessor processes data, it looks first in the cache memory and if it finds the data there (from

a previous reading of data), it does not have to do the more time-consuming reading of data from larger memory.

Primary storage of a computer system has the following limitations:

Limited capacity:

It is often necessary to stare many millions, sometime billions, and even trillions, of bytes of data in a computer. Unfortunately, the storage capacity of the primary storage of today's computer is not sufficient to store the large volume of data, which needs to be handled by most data processing centers.

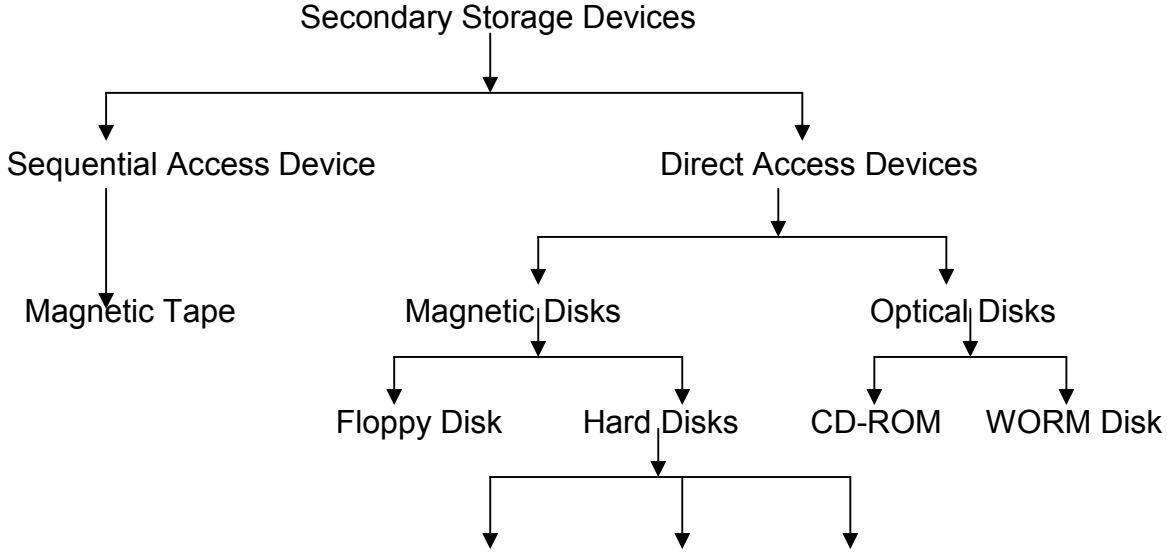
Volatile:

The primary storage is volatile, and the data stored in it is lost, when the electric power is turned off or interrupted. However, computer systems need to store data on a permanent basis for several months, or even several years.

As a result, additional memory, called auxiliary or secondary storage, is used with most computers. This section of computer's memory is non-volatile, and has lower cost per bit store, but it generally operating speed far slower than of the primary storage. This section of the memory is used to store volume of data on a permanent basis, which can be partially transferred to primary storage, as and when for processing.

2. Secondary storage:

The secondary storage, also known as auxiliary storage, is used to take care of the limitations of the primary storage. That is, it is used to supplement the limited storage capacity and the volatile characteristic of primary storage. This is because secondary storage is much cheaper than primary storage, and it can retain information even when the computer system is switched off or reset. The secondary storage is normally used to hold the program instructions, data, and information of those jobs, on which the computer system is not working on currently, but needs to hold them for processing later. The most commonly used secondary storage medium is the magnetic disk.



Zip Disk Disk Pack Winchester Disk

ARITHMETIC LOGIC UNIT:

The arithmetic logic unit (ALU) of a computer system is the place, where the actual execution of the instructions takes place, during the processing operation. To be more precise, calculations are performed, and all comparisons (decisions) are made in the ALU. The data and instructions, stored in the primary storage before processing, are transferred as and when needed to the ALU, where processing takes place. No processing is done in the primary storage unit. Intermediate results generated in the ALU are temporarily transferred back to the primary storage, until needed later. Hence, data may move from primary storage to ALU, and back again to storage, many times, before the processing is over.

The type and number of arithmetic and logic operations, which a computer can perform, is determined by the engineering design of the ALU. Almost all ALUs are designed to perform the four basic arithmetic operations (add, subtract, multiply and divide), and logic operations or comparisons, such as less than, equal to, and greater than.

Instruction Set: Every CPU has built in ability to execute a set of machine instructions, called its instruction set. The machine language designed for a processor (CPU), is based on the list of instructions supported by the CPU in its instruction set.

Registers: As the instructions are interpreted and executed by the CPU, there is a movement of information between the various units of the computer system. In order to handle this process satisfactorily, and to speed up the rate of information transfer, the computer uses a number of special memory units, called registers.

1. **Memory Address Register (MAR)**: It holds the address of the active memory location. It is loaded from the program control register, when an instruction is read from memory.
2. **Memory Buffer Register (MBR)**: It holds the contents of the memory word read from, or written in, memory. An instruction word placed in this register is transferred to the instruction register.
3. **Program Control Register (PC)**: It holds the address of the next instruction to be executed.
4. **Accumulator Register (A)**: It holds the data to be operated upon, the intermediate results, and the results of processing. It is used during the execution of most instructions.
5. **Instruction Register (IR)**: It holds the current instruction, which is being executed.
6. **Input/Output Register (I/O)**: It is used to communicate with the input/output devices.

CONTROL UNIT

How does the input device know that it is time for to feed data into the storage unit? How does the ALU know, what should be done with the data once they are received? Moreover, how is it that only the final results are sent to the output device,

and not the intermediate result? All this is possible due to the control unit acts as a central nervous system, for the other components of the computer system.

CENTRAL PROCESSING UNIT

The control unit and the arithmetic logic unit of a computer system are jointly known as the Central Processing Unit (CPU). The CPU is the brain of the computer system. In a human body, all major decisions are taken by the brain and the other parts of the body function as directed by the brain. Similarly, in a computer system, all major calculation and comparisons are made inside the CPU, and the CPU is responsible for activating and controlling the operation of other units of the computer system.

Generation of Computer Languages

First generation

Before computers were even built, Ada Lovelace (1815-1852) made the first notes on how to control a mechanical computer. She is considered to be the first programmer.

This *analytical computer* was designed by Charles Babbage (1791-1871) and was not implemented until 1991. He also designed a *printer*, implemented in 2000, an astonishingly complex device for the 19th century. His computer led to more complex designs.

Alan Turing (1912-1954) designed a theoretical computer which allows us to explain the functions of a modern CPU inside a computer. Despite its simplicity, a *Turing Machine* [T1] can be adapted to simulate the logic of any computer algorithm.

Code of the first generation programming language [G1] was entered through **physical switches** on the computer and involved commands to move *data bits* to and from *registers*, compute on these and more. A sequence of such commands is known as *code*. Specifically, code that a machine can read and understand according to its logical design is called **machine code** [M1], which is in contrast to human readable code. The first generation of programming languages is all about machine code. Such code give fast executions but is tedious to write and is brittle [C1] to changes.

Second generation

Improves on first generation by providing code as human readable **source code** with **logically structures**. The source code must be assembled into machine code before it can be executed by a CPU. This assembly step is done by an *assembler*.

Such languages are sometimes still used for *kernels* and *device drivers*, i.e. the core of the operating system and for specific machine parts. More often, such languages are used in areas of intense processing, like *graphics programming*, when the code needs to be **optimized for performance**.

2GL source code is **not portable** across processors or processing environments.

Third generation

Improves on second generation by making a language **architecture independent**, i.e. no longer tailored to the processor or environment, which requires a *compiler* to make machine code for the CPU. Some, like Java, make use of an intermediate code which is run by a *virtual machine*, reaching further architecture independence. In such a case, only the virtual machine needs to be run by something that is architecture specific.

Another aspect of third generation [G3] is that they are **more programmer-friendly** with features like good support for*aggregate data types* and *expressing concepts* in a way that favors the programmer, not the computer (e.g. no longer needing to state the length of multi-character (string) literals in Fortran).

High level language is a synonym for third-generation programming language. The computer take care of non-essential details, not the programmer.

Early language examples: Algol, Cobol (business), Fortran (scientific)
Structured language examples: C
Object oriented examples: C++, Delphi, C#, Java, Python
Functional examples: Scala

This generation support **structured programming** [S1]. While later developments include object-**oriented programming** [O1] which dominates today and **functional programming** [F1], some groups have advocated the concept of 4GL and 5GL to mark the distinct difference these other languages have in comparison with 3GL. It remains to see what classifications will win the hearts of the historians in the next 100 years.

Fourth generation

Improves on 3GL and their development methods with **higher abstraction** and **statement power**, to reduce errors and increase development speed by reducing programming effort.

A 4GL is *designed with a specific purpose in mind.* This offers a **high potential**. 4GL are not always successful, sometimes resulting in inelegant and un-maintainable code. However, given the right problem, the use of an appropriate 4GL can be spectacularly successful. (This refers to an example with 8 times productivity gains over Cobol). [G4]

In some senses, *software engineering* supports 3GL, while 4GL are more oriented towards *problem solving* and *systems engineering.*

Some argues that 4GL is a subset of the so-called DSL or domain-specific languages. [D1]

Examples: Oracle Forms, PL/SQL, Revolution language, SAS, SPSS, SQL

Fifth generation

Improves on the previous generations by skipping algorithm writing and instead

provide **constraints**.

While 4GL are designed to build specific programs, 5GL are designed to make the computer solve a given problem without the programmer. This way, the programmer only needs to worry about what problems need to be solved and what conditions need to be met, without worrying about how to implement a routine or algorithm to solve them. [G5]

Examples: Prolog, OPS5, Mercury

On a critical view, Prolog kind of systems have a *high performance penalty*. With better implementation techniques, this penalty have been shown to be 25%-50% of conventional programming languages.

Operating System:

An operating system (often referred to as OS) is an integrated set of programs that controls the resources (the CPU, memory, I/O devices etc.) of a computer system and provides its users with an interface or virtual machine that is more convenient to use than the bare machine. According to this definition, the two primary objectives of an operating system are:

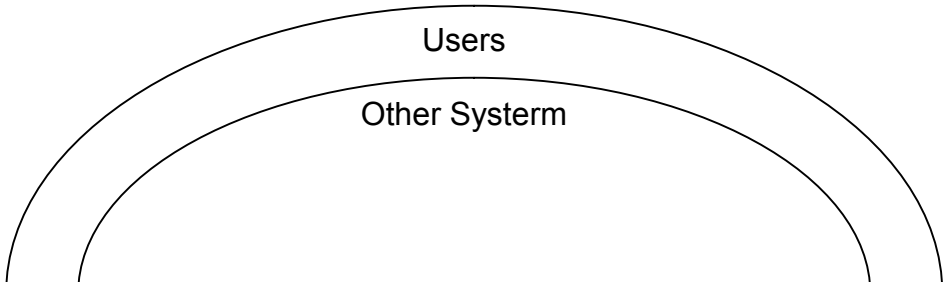
1. Making a computer system convenient to use:

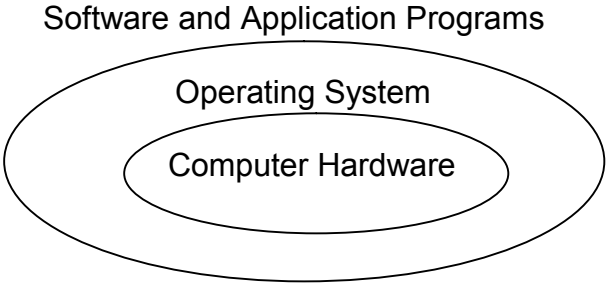
An operating system is a layer of software on top of the bare hardware of a computer system, which manages all parts of the system, and present to the user with an interface or virtual machine, which is easier to programmer and use. That is the operating system hides the details of the hardware resources from the programmer and provides the programmer with a convenient interface for using the computer system. It act as an intermediary between the hardware and its user, providing a high-level interface to low-level hardware resources, and making it easier for the programmer and other user to access to use those resources.

The logical architecture of computer system is shown in figure. The hardware resources are surrounded by the operating system layer, which in turn is surrounded by a layer of other system software (such as compiler, editors, utilities, etc.) and a set of application program (such as commercial data processing applications, scientific and engineering applications, entertainment and educational applications, etc.). Finally, the end user views the computer system in terms of user interfaces provided by the application programs.

2. Managing the resources of a computer system:

The second important objective of an operating system is to manage the various resources of the computer system. This involves performing such tasks as keeping track of who is using which resource, granting resource requests, accounting for resource usage etc.





Logical architecture of a Computer System

Main Functions of an Operating system:

1. Process Management.
The process management module of an operating system takes care of the creation and deletion of processes, scheduling of various resources to the different process requesting them, and providing mechanisms for synchronization and communication among processes.

2. Memory Management.
The memory management module of an operating system takes care of the allocation and deallocation of memory space to the various programs in need of this resource

3. File Management.
The file management module of an operating system takes care of file related activities such as organization, storing, retrieval, naming, sharing and protection of files.

4. Security Management.
The security module of an operating system protects the resources and information of computer system against destruction and unauthorized access.

5. Command Interpretation.
The command interpretation module of an operating system takes care of interpreting user commands, and directing the system resources to handle the requests with this mode of interaction with the system, the user is usually not too concerned with the hardware details of the system.

Process Management:

A process (also called job) is a program in execution. The main objective of the process management module of an operating system is to manage the process submitted to the system in a manner to minimize the idle time of the various processors (CPU, I/O processors, etc.) of the computer system.

Some of the mechanisms commonly used in modern operating systems to achieve this objective are:

- Multiprogramming
- Multitasking
- Multiprocessing
- Time-sharing

Some popular Operating Systems:

UNIX: UNIX is a multi-user, time-sharing operating system.

MS-DOS: MS-DOS stands for Microsoft Disk Operating System. It is a single-user operating system.

Microsoft Windows: Microsoft Windows operating system was developed by Microsoft to overcome the limitations of its own MS-DOS operating system. Its native interface is a GUI. Hence, it is easier for a new user to learn and use the system.

Microsoft Windows NT: Microsoft Windows NT is a multi-user, timesharing operating system developed by Microsoft. It was designed to have Unix-like features so that it can be used for powerful workstations, networks and database servers.

Linux: Linux is an open-source operating system enhanced and backed by thousands of programmers worldwide. It is a multi-tasking operating system, which was originally designed to be used on personal computers.

Windows XP command line

ATTRIB	Change file attributes
CD	Change Directory - move to a specific Folder
CHKDSK	Check Disk - check and repair disk problems
CLS	Clear the screen
CMD	Start a new CMD shell
COLOR	Change colors of the CMD window
COMP	Compare the contents of two files or sets of files
COMPACT	Compress files or folders on an NTFS partition
COMPRESS	Compress individual files on an NTFS partition
CON2PRT	Connect or disconnect a Printer
CONVERT	Convert a FAT drive to NTFS.
COPY	Copy one or more files to another location•
DATE	Display or set the date
DEL	Delete one or more files
DELTREE	Delete a folder and all subfolders
DIR	Display a list of files and folders
DISKCOMP	Compare the contents of two floppy disks
DISKCOPY	Copy the contents of one floppy disk to another
DOSKEY	Edit command line, recall commands, and create macros
ECHO	Display message on screen
ERASE	Delete one or more files
EXIT	Quit the current script/routine and set an errorlevel
FC	Compare two files

FIND	Search for a text string in a file
FINDSTR	Search for strings in files
FORMAT	Format a disk
FREEDISK	Check free disk space (in bytes)
LABEL	Edit a disk label
MD	Create new folders
MORE	Display output, one screen at a time
MOVE	Move files from one folder to another
MSG	Send a message
MV	Copy in-use files
PATH	Display or set a search path for executable files
PING	Test a network connection
PRINT	Print a text file
PROMPT	Change the command prompt
PsPasswd	Change account password
RECOVER	Recover a damaged file from a defective disk.
REGINI	Change Registry Permissions
REM	Record comments (remarks) in a batch file
REN	Rename a file or files
REPLACE	Replace or update one file with another
RD	Delete folder(s)
SHUTDOWN	Shutdown the computer
SORT	Sort input
START	Start a program or command in a separate window
TIME	Display or set the system time
TREE	Graphical display of folder structure
TYPE	Display the contents of a text file
VER	Display version information
VERIFY	Verify that files have been saved
VOL	Display a disk label
XCOPY	Copy files and folders