



GOVERNMENT POLYTECHNIC MAYURBHANJ

LECTURE NOTES
ON
COMPUTER APPLICATION

PREPARED BY

HEMANGINI DALEI

LECTURER IN CSE

Introduction to Computer :-

Computer is an electronic data processing device, that takes data as input, processes it and gives meaningful information as ~~input~~ output.

→ Computer comes from the Greek word "compute", which means "to calculate".

→ Full form of Computer

C - ~~Computer~~ Commonly
O - Operating
M - Machine
P - Purposefully
U - Used for
T - Technical and
E - Educational
R - Research

→ The first mechanical computer was developed by Charles Babbage in the year 1822. Thus Charles Babbage is known as Father of computer.

→ Difference between Man and Computer

Man

- 1) Man is a living being.
- 2) Man has intelligence power.
- 3) Man can feel about a thing.
- 4) Man can understand many languages.
- 5) After doing work man gets tired/bored.
- 6) Man works slowly.

Computer

- 1) computer is a non-living being.
- 2) computer doesn't have intelligence power.
- 3) computer doesn't have feelings.
- 4) computer can understand only machine language.
- 5) computer never gets tired/bored.
- 6) computer works very fast.

Strengths of Computer

1) Speed of Computer :-

Computer works very fast. It can perform millions of instructions in a minute, which may take a day or some days if performed manually.

2) Accuracy :-

Computer perform all the calculations and computations correctly.

3) High Storage capacity :

Computer can store large amount of information.

4) Reliability :-

Computer is a reliable machine.

5) Versatility :-

Computer has the ability to perform different types of jobs.

6) Diligence :-

Computer can work for many hours continuously without taking rest and without decreasing its speed, accuracy and efficiency.

7) Multitasking :-

Computer can perform more than one task at a single time.

Weaknesses of Computer

1) Lack of IQ -

Computer is a dumb machine with zero IQ.

2) Lack of feelings :-

Computer doesn't have any feelings.

3) Needs Electric power :

Computer needs electric power to run. Without electric power computer is useless.

4) It can ~~can~~ understand only machine language.

5) Lack of decision making power :-

Computer can not take decisions. It has no assumption power. It works as per the instructions given by the user. If user gives wrong input, it will produce wrong output.

Differences between Data and Information

Data

1) Data is the raw facts and figures through which an object can't be clearly identified.

Ex:- Name of a person, place, price of an item, age of a person etc.

2) Data has no meaning

3) Data does not help in decision making.

4) Data doesn't help in business related problem solving.

5) Data doesn't give clear idea to solve a problem.

Information

1) Information is the processed data through which an object can be clearly identified.

Ex:- Salary slip of employee, marksheet of a student etc.

2) Information is meaningful

3) Information helps in decision making.

4) Information helps in business related problem solving.

5) Information gives a clear idea to solve a problem.

Evolution of Computer :

(3)

Many centuries ago when man started counting the numbers, he was using sticks, fingers, bones, stones etc. for counting.

- Development of modern day computer was the result of advances in technologies and man's need to quantify.
- Abacus was the first counting machine developed in china more than 3000 years ago. The name Abacus was obtained from a Greek word Abax which means slab. This machine was used for arithmetic operations.
- In the early 1600, John Napier, a Scottish Mathematician, invented a calculating device called Napier's bones. This device was mainly developed for performing multiplication and division.
- In 1645 a device known as Pascaline was invented by French Mathematician Blaise Pascal. This machine was used for addition and subtraction of whole numbers. This device was operated by dialing a set of wheels.
- In 1671 Leibniz improved on Pascal's adding machine and invented the Leibniz's calculator.
- The punched cards were invented for storing and retrieving data.
- An American statistician Herman Hollerith developed a machine called Hollerith machine that could store data coded in the form of punched holes.
- In 1804, J.M. Jacquard, a French inventor, devised a Loom which used punched cards to direct the weaving pattern.
- In 1822 Charles Babbage invented a difference engine. He was the first person to design a fully programmable machine computer. Again he upgraded this to design an Analytical engine, which could store program instructions. Therefore Charles Babbage is ~~was~~ known as 'Father of Computer'.

→ Analog computers were widely used in the first half of 20th century for a number of scientific needs. Then digital computers were invented & widely used.

→ The first pure electronic computer was ABC and was invented by J.V. Atanasoff and C. Berry.

→ Among more important machines, American ENIAC (Electronic Numerical Integrator and Calculator) was prominent. It was a general purpose machine, but has an inflexible architecture.

→ In 1945 John von Neumann, gave the idea of sharing internal memory for storing both data and instructions, and was adopted in every computer organization. On this principle Universal Automatic Computer (UNIVAC) was invented.

Generation of Computer: -

→ Generations of computer is mean to major technological development that fundamentally changed the way computers operate, resulting in increasingly smaller, cheaper, more powerful, more efficient and reliable devices.

→ There are five generations of computers.

(1) First Generation Computer (1940-1956)

(i) The main component of 1st generation computer is vacuum tube.

(ii) Magnetic drums were used for memory.

(iii) It was unreliable and consumed a lot of electricity.

(iv) Supported machine language only and could solve only one problem at a time.

(v) Vacuum tubes were like electric bulbs and made up of glass, produced a lot of heat. So A.C was required.

(vi) The size of ~~this~~ 1st generation computer was very huge. These not portable.

(vii) It was very costly. and not portable.

(viii) Input was based on punched cards, paper tape and output was displayed on printouts.

Ex: - ENIAC, UNIVAC, EDVAC, EDSAC, IBM-701, IBM-650.

(2) Second Generation Computer (1956-1963) (4)

- (i) In this generation transistors were used, that were cheaper, consumed less power and more compact in size.
- (ii) These were smaller in size, reliable, generated less heat, faster ^{in processing} in comparison to 1st generation computer.
- (iii) These were still costly.
- (iv) A.C was required.
- (v) Supported both machine language and assembly language.
- (vi) Punched cards were used for inputs and printouts for outputs.
- (vii) These were the first computers to store their instructions in memory, which moved from magnetic drum to magnetic tape technology.

Ex: - IBM 1620, IBM 7094, CDC 6600, Burroughs 5000, Honeywell Well 800, GE 65, RCA Spectra 70.

(3) Third Generation Computer (1964-1971)

- (i) Integrated circuits (ICs) were used as main component.
- (ii) These computers were more reliable, smaller in size, generated less heat, faster processing and less maintenance required in comparison to the previous two.
- (iii) These generation computers supported high-level language.
- (iv) ~~More~~ Keyboards and monitors were used for input and output and interfaced with an operating system, which allowed the device to run many different applications at one time with a central program that monitored the memory.

(2)
Ex: - IBM - 360, PDP (personal data processor), IBM - 370/168.

(4) Fourth Generation Computer (1971-1990)

- (i) This generation computers were using VLSI (very large scale integrated) circuits as main components.
- (ii) These were more powerful, ~~compact~~^{small} in size, reliable and affordable. As a result, it gave rise to personal computer (PC) revolution.
- (iii) Pipeline processing was used.
- (iv) ~~No A.C. was req.~~ Generated less heat, thus no A.C. was required.
- (v) Concept of internet was introduced.
- (vi) Microprocessors were used.
Ex - DEC 10, STAR 1000, PDP 11, CRAY-1, CRAY-X-MP

(5) Fifth Generation Computer (1990-present)

- (i) The main component was VLSI (ultra large scale integrated) circuit.
- (ii) Artificial Intelligence concept was used, that used in applications like voice recognition.
- (iii) All high level languages were used in this generation.
- (iv) Used parallel processing, superconductors, Quantum computation, molecular and nanotechnology.
- (v) Use more user-friendly interfaces with multimedia features.
- (vi) Cheaper ~~in price~~ more cheaper.
Ex - Desktop, Laptop, Notebook, Ultrabook etc.

Classification of computers

(3)

- Computers are of different types based on their data processing abilities. They are classified according to purpose, data handling and processing capability, size etc.
- According to purpose, computers are either general purpose or specific purpose. General purpose computers are designed to perform a range of tasks. They have the ability to store a large number of programs, but lack in speed and efficiency.
- Specific purpose computers are designed to handle a specific problem or a specific task.
- According to data handling, computers are classified into analog, digital and hybrid.
- Analog computers: work on analog data, such as variation in temperature, pressure, speed, voltage etc. These computers do not deal directly with the numbers. They measure continuous physical magnitude. Application: (i) Engg. (ii) scientific work (iii) to control process in chemical industries.
- Digital computers work on digital data. They process data into a digital value (in 0s and 1s). They give the results with a more accuracy and at a faster speed.
- Hybrid computers are having features, common to both analog computers and digital computers. Hybrid computers measure the physical quantities such as temperature, pressure, voltage etc and convert them into digital data. These data are then processed by computer by using its digital data processing capabilities.
- According to processing capabilities computers are classified into
 - (i) microcomputer / personal computer.
 - (ii) Mini computer
 - (iii) Mainframe computer
 - (iv) Super computer

(i) Micro Computer / Personal Computer

→ Micro computers are the smallest and least expensive, popularly known as PC.

Features :-

- (i) These computers are portable.
- (ii) They ~~are~~ require minimum power.
- (iii) Processing power is appropriate for handling most of the tasks.
- (iv) Support different types of operating system and application software.
- (v) These are affordable.

Application :-

In business these are used for word processing, accounting, desktop publication, running spreadsheet and database management application.

→ At home these computers are used for playing games, entertainment etc.

Specification :-

processor - 18086/18088 microprocessor

memory - 640 KB of RAM

Numerical processor - 18087

System bus - 8 bit data bus & 16 bit address bus

clock speed - 8 MHz

Ex:- Desktop computers, laptop, palmtop, digital library, Notebook

(ii) Mini Computer : These computers are mostly preferred by small business, colleges etc.

Features :-

- (i) Fairly large primary memory.
- (ii) Medium scale processing capability.
- (iii) Can accept upto 500 terminals on LAN.

(iv) Affordable

(v) Used for medium or large volume data processing activity.

Application :-

Mainly used in the field of engineering and scientific organizations, Educational institutes, Universities, small/medium business organizations.

Specification

Processing speed - 10 to 30 MIPS

Word length - 32 bits

Ex:- Inspiron 1012, Apple iPhone,

I/O device - Wide range of I/O devices can be connected

Internal storage - 64 MB to 512 MB

(iii) Mainframe Computer :-

(6)

Features :- (i) Large primary memory.

(ii) Can process millions of instructions per second.

(iii) More number of peripheral devices can be attached.

(iv) High data communication capability i.e. ability to connect thousands of terminals to LAN.

(v) Wide variety of memory size

(vi) Ability to handle large computer application.

Applications :- Space research, university connectivity, wide area network (WAN) implementation etc.

Specification :-

processing speed - 30 to 100 MIPS

word length - More than 64 bits

I/O device - wide range of peripheral devices can be connected.

Internal storage - More than 1 GB

Ex:- IBM ZSeries, System Z9 & System Z10 servers

(iv) Super Computer :-

Feature :-

(i) Fastest processing speed

(ii) Most powerful computer among all other computers.

(iii) Very expensive.

(iv) Employed for specialized applications

(v) Require immense amount of mathematical calculations.

(vi) Designed for processing of floating point instructions.

(vii) ~~Parallel~~ It uses the parallel processing technique.

(viii) speed of processing is measured in GFLOPS

(i.e. Giga floating point operations per second)

(ix) These computers need their own OS and programming language.

Application :- Used for weather forecasting, animated graphics, fluid dynamic calculations, nuclear energy research, petroleum exploration etc.

Ex:-

Cray-1 :- World's 1st super computer

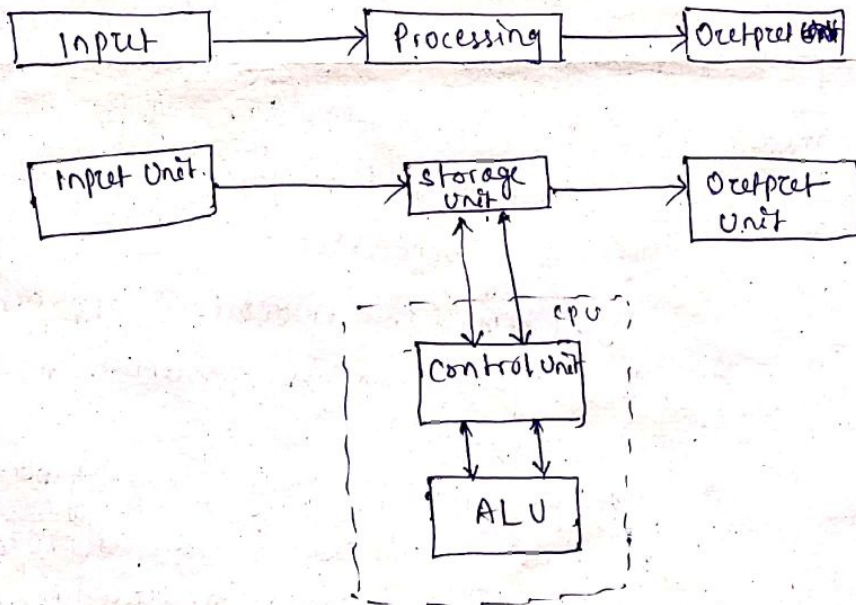
Param :- India's 1st super computer

Anuram - A series of super computer designed and developed by Bhabha Atomic Research Center (BARC).

Principles of Von Neumann's Architecture

- There are two main points of Von Neumann's stored principle.
- 1) Binary codes should be used for the representation of data and instructions, in a manner, which would make no distinction between them to be stored together by sharing the same storage space within the computer.
 - 2) The computer should be able to process both data and instruction by taking them from the same primary memory. This includes the modification of programs.

Basic Organisation of Computer



→ Functionally, a digital computer can be divided into 4 units.
i.e.

- (i) Input Unit
- (ii) Central Processing Unit
- (iii) Memory Unit
- (iv) Output Unit

Input Unit

→ In order to solve any problem, computers need to receive data and instructions. ~~into the computer~~ Therefore we need to input data and instructions into the computer. The input unit creates a link between the user and the computer. The input unit consists of one or more input devices. These devices translate the information into a form understandable by the computer.

→ All input devices perform the following functions.

- (i) Accept the data and instructions from the outside world.
- (ii) Convert it into a form that the computer can understand.
- (iii) Supply the converted data to the computer system for further processing.

→ The commonly used input devices are as follows:

1) Keyboard: - It is the most common and very popular input device which helps to input data to the computer. The layout of the keyboard is like a traditional type writer. There are also some additional keys. A keyboard contains four types of keys.

- (i) Alphabet Keys - consist of A-Z, a-z.
- (ii) Numeric Keys - consist of 0 to 9.
- (iii) Function Keys - consist of keys F1 to F12.
- (iv) Control Keys - consist of ctrl, del, alt, shift etc.

→ Keyboards are of two sizes 84 keys or 104/102 keys.

But nowadays keyboards with 104 or 108 keys are available.

→ Some special purpose keys are available in keyboard like: Enter, CapsLock, NumLock, SpaceBar, Tab & PrintScreen.

2) Mouse: - Mouse is the most popular ~~pointing~~ pointing device. It takes full advantage of graphical user interface. → Mouse is designed in such a way that, it can comfortably fit under the palm of the hand. Mouse is used to control the movement of the pointer on the screen and to make selections from the screen.

→ It has 3 to 4 buttons. Some also have a small wheel.

The bottom of the mouse is flat and contains a multi-directional mechanism, and usually a small ball.

→ Now-a-days optical mice are becoming popular with no ball at the bottom, rather it functions with a LED and a sensing mechanism to detect a location on the screen.

Functions of mouse! -

1) Single click - To select an item on screen.

2) Double click - To open an application.

3) Right click - To view some special operation to perform.

4) Scroll - To move the mouse pointer from top to bottom

and left to right etc.

3) Joystick! - Joystick is also a pointing device, which is used to move the cursor position on a monitor screen.

→ It is a stick having a spherical ball at its both lower and upper ends. The lower spherical ball moves in a socket.

→ The joystick can move in all four directions.

→ It is used in Computer Aided Designing (CAD) & playing computer games.

4) Light Pen! - It is a pointing device similar to a pen. It is used to select displayed menu items or draw pictures on the monitor screen.

5) Track Ball! - Track ball is an input device that is mostly used in notebook or laptop computer, instead of a mouse.

This is a ball which is half inserted and by moving fingers on the ball, the pointer can be moved. It requires less space than a mouse.

6) Scanner : - Scanner is an input device that captures images from the source which are then converted into a digital form that can be stored on the disk or printed by the printer. Generally a scanner is used to convert a hardcopy to a softcopy.

7) Digitizer : - It is an input device which converts analog information into digital form. Digitizer can convert a signal from a television or camera into a series of numbers that could be stored in a computer. It is used for fine work of drawing and image manipulation.

8) Microphone : - It is used to input sound that is then stored in a digital form.

9) Magnetic Ink Character Reader (MICR)

MICR is an input device, generally used in banks. The bank's code number and cheque number are printed on the cheques with a special type of ink that contains particles of magnetic material that are machine readable. This reading process is called Magnetic Ink Character Recognition (MICR). The main advantage of MICR is that, it is fast and less error prone.

10) Optical Character Reader (OCR)

→ OCR is an input device used to read a printed text.
→ OCR scans the text optically, character by character, converts them into a machine readable code and stores the text on the system memory.

11) Bar Code Reader (BCR)

→ BCR is a device used for reading bar coded data (data in the form of light and dark lines). Bar coded data is generally used in labelling goods, numbering the books, etc.

12) Optical Mark Reader (OMR)

OMR is a special type of optical scanner used to recognize the type of mark made by pen or pencil. It is used where one out of a few alternatives is to be selected and marked.

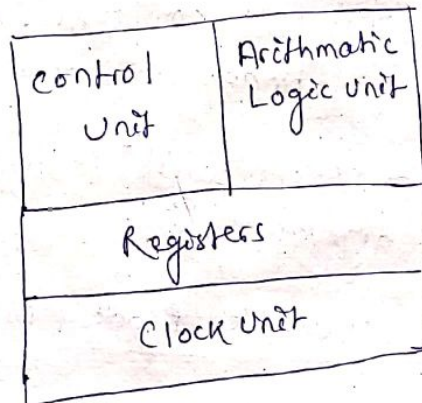
→ It is specially used for checking the answer sheets of examinations having multiple choice questions.

Central Processing Unit (CPU)

→ CPU is the main component of computer. It is responsible for all events inside the computer. The CPU is the brain of computer.

→ It performs the following functions:

- (i) It performs all calculations.
- (ii) It takes all decisions.
- (iii) It controls all units of the computer.



→ CPU performs consists of different components, such as Control Unit (CU), Arithmetic and Logic Unit (ALU), registers, clock unit etc.

Control Unit: (i) CU is responsible for coordinating and managing all the units of computer.

(ii) It is responsible for controlling the transfer of data and instructions among other units of a computer.

(iii) It obtains the instructions from memory, interprets them, and directs the operation of the computer.

(iv) It communicates with input/output devices for transfer of data or results from storage.

(v) It does not process or store data.

ALU :- This is the unit where actual computation takes place. It consists of two subsections.

- (i) Arithmetic section: which is responsible for all arithmetic operations like addition, subtraction, multiplication and division. All complex operations are done by making repetitive

use of the above operations.

(ii) Logic section :- This section is responsible for logical operations like comparing, selecting, matching and merging of data.

Output Unit :-

Output unit consists of a number of output devices.

→ Output device is any peripheral device that converts machine readable information into user/people readable form. The output information may be displayed on screen, printed on paper or output as sound.

→ There are mainly two types of output devices: i.e. softcopy output device and hardcopy output device.

→ The softcopy device allows user only to view the result.

Ex:- Visual Display Unit (VDU) : like monitor.

→ The hardcopy device allows the user to take an impression of the output on paper.

Ex:- Printer, plotters.

→ Other example of output device is speaker.

Monitors :-

→ Monitors are commonly known as Visual Display Units (VDU). It forms images from tiny dots, called pixels that are arranged in a rectangular form. If the number of pixels more, more is the resolution/clarity in picture.

→ There are two kinds of monitors.

(i) Cathode - Ray Tube (CRT)

(ii) Flat Panel Display

CRT monitor :-

The CRT monitor display is made up of small picture elements called pixels. The smaller the pixels, the better the image clarity or resolution. It takes more than one pixel to form a whole character. Most screens are capable of displaying 80 characters of data horizontally and 25 lines vertically.

Disadvantage

- (i) Large in size
- (ii) High power consumption

Flat Panel Display Monitor :

→ It has reduced volume, weight and power requirement in comparison to CRT monitor.

→ It can be hanged on wall easily or wearred on wrist.

Ex: - calculators, video games, ~~more~~ laptop computer etc.

→ It is of two types:

(i) Emissive display: These are the devices that convert electrical energy into light. Ex: -

Ex: - LED (Light-Emitting Diodes)

(ii) Non-Emissive display: These are using optical effects to convert sunlight or light from some other source into graphics pattern.

Ex: - LCD (Liquid crystal-Display)

→ There is a special type of VDU called touch screen monitor, that works as both input device and output device.

Ex: - ATM ~~touch~~ touch screen monitor.

→ TFT (Thin Film Transistor) is a display monitor that is used with laptop because they use less power and take up less space.

Printer :-

Printer is an output device, which is used to print information on paper. There are two types of printers.

(i) Impact Printer

(ii) Non-impact Printer

Impact Printer :

Impact prints print the characters by striking them on the ribbon, which is then pressed on the paper.

Characteristic :-

- (i) Very low consumable cost
- (ii) very noisy
- (iii) Useful for bulk printing due to low cost
- (iv) There is physical contact with the paper to produce an image.

→ Impact printers are of two types.

- 1) Character Printers
- 2) Line printer

Character Printer :- Character prints print one character at a time. These are of two types.

↳ Dot Matrix Printer :-

These printers are popular because of their ease of printing and economical price.

→ These printers print the characters using one or two columns of tiny dots on a print head. The dot hammer moving serially across the paper strikes an inked-ribbon and creates images on paper.

→ The print head of this printer consists of a matrix of pins of size (5x7, 7x9, 9x7, 9x9) or, with print head that have 7, 9, 18 or 24 pins.

Advantage

- 1) Available in low cost
- 2) Widely used
- 3) Other language characters can be printed.

Disadvantage

- 1) Speed is very slow
- 2) Printing quality is poor.

2) Daisy Wheel Printer :-

→ This printer operate just like a typewriter.

A hammer strikes a wheel with petals, ~~but~~ each petal containing a letter form at its tip. The letter form strikes a ribbon of ink, depositing the ink on the page and thus printing a character. By rotating the daisy wheel different characters are selected for printing.

Advantage

- 1) More reliable than Dot matrix printer
- 2) Better quality printing
- 3) Fonts of characters can be easily changed

Disadvantage

- 1) Slower than DMP
- 2) Noisy
- 3) More expensive than DMP

Line Printer :-

Line Printers are the printers which print one line at a time. These are of two types.

(1) Drum Printer :-

→ This printer is like a drum in shape. Hence it is called drum printer. The surface of the drum is divided into a number of tracks. Total tracks are equal to the size of the paper. i.e. for a paper width of 132 characters, drum will have 132 tracks. A character set is embossed on the track. Different character set are available in market are 48 characters, 64 & 96 characters set. One rotation of drum prints one line. They can print 300 to 2000 lines per minute.

Advantage

- Very high speed

Disadvantage

- 1) Very expensive
- 2) Character fonts cannot be changed.

2) Chain Printer : -

The chain printer uses a printing mechanism that uses character typefaces linked together on a chain. The chain spins horizontally around a set of hammers aligned with each position. When the required character is in front of the selected print-position, hammer in that position hits the paper ~~on to~~ into the ribbon against the character on the chain. It can print 3000 lines per minute.

Advantage

- 1) very high speed
- 2) Reliable
- 3) character fonts can easily be changed.
- 4) Different languages can be used with the same printer.

Disadvantage

- 1) It is very expensive
- 2) very noisy

Non-Impact Printer : -

Non-impact printers are those printers, which doesn't have a physical contact between the print head and paper during printing.

Characteristics

- (i) These printers are faster than impact printers.
 - (ii) These are not noisy.
 - (iii) Printing quality is very good.
 - (iv) They support many fonts and different character size.
 - (v) They print a whole page at a time.
- These printers are of two types -

1) Laser printer : Laser printer is a page printer, that uses laser beam light source to create image on a photographic drum. Then the images on the drum are treated with magnetically charged toner and then are transferred onto a paper.

→ This printer is capable of printing from 3 to 20 pages per minute.

Advantage

- 1) Very high speed
- 2) very high quality printing having 300 to 1200 dots per inch.
- 3) Good graphics quality.
- 4) Supports many fonts and different character size.

Disadvantage

- 1) Expensive

2) Inkjet Printer :-

Inkjet printers are based on the non-impact printers, which print page by page. The print head contains tiny nozzles through which different colored inks can be sprayed onto the paper to form characters or graphics images. It is capable of printing 2 to 6 pages per minute (ppm).

Advantage

- 1) High quality printing
- 2) Capability of printing variety of color printing at a relatively low cost.
- 3) More reliable.

Disadvantage

- 1) The cost of printing per page is high.
- 2) Slow as compared to laser printer.

Speaker :-

A speaker is an output device that connects to a computer to generate sound. It receives audio input from the computer's sound card and produces audio output in the form of sound waves.

Plotters :-

Plotter is an output device that draws images with ink pens. It is a graphics printer for making sophisticated graphs, charts, maps, three-dimensional graphics as well as high-quality colored documents. It can produce a larger printout than standard printer.

There are two types of plotters:

1) Drum plotter :- This type of plotter has a drum that rotates to produce plots. Pens in a drum plotter move across the paper while the drum is turning. A drum plotter is capable of plotting on a drawing sheet and is used mostly in commercial CAD applications.

2) Flatbed plotter :- This type of plotter has a bed, this is also called as table plotter. The plotter draws graphics on the paper placed on the bed. This plotter is usually used for producing large drawings.

⇒ Other examples are :- Projector, headphone etc.

Computer Memory and Classification of Memory

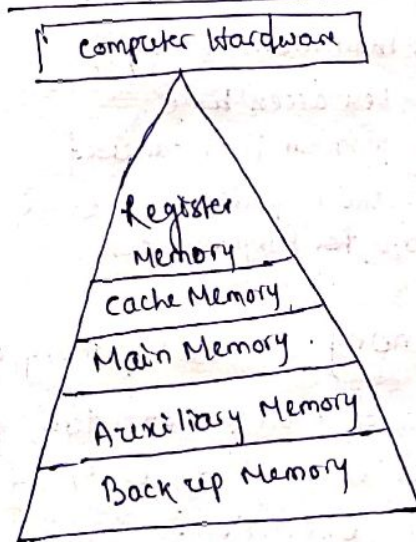
12

Computer memory is used to store data and programs, either permanently or temporarily depending on the type requirements and type of memory.

Properties of memory

- 1) It should have two stable states.
- 2) While it is in one of the two stable states, it should not consume any power or consume very little power.
- 3) It should be possible to switch between the two stable states an infinite number of times.
- 4) Information stored in a memory location should not decay with time.
- 5) Each memory cell should be low cost and occupy less physical space.
- 6) The time taken to read data or retrieve data from/to a memory location should be as low as possible.

Memory Hierarchy



→ The different classes of memory in memory hierarchy are used for different purposes. Those are:

- 1) Register Memory: - Register is integrated inside the CPU. It consists of a number of flipflops. It is a small capacity memory, used for storing data/instruction temporarily during the execution of the program/instruction.

7) The number of registers vary from computer to computer.
There are some special purpose register such as:

MAR - Memory Address Register

MDR - Memory data register

IR - Instruction Register

PC - Program counter

II - Instruction Interpreter

Accumulator -

→ Register is the fastest memory available in a computer.

2) Cache Memory :- Cache memory is a very high speed semiconductor memory. Cache memory is situated between the CPU and main memory. The purpose of this memory is to hold/store data and program which are most frequently used by the CPU. When the processor needs to read from or write to a location in main memory, it first checks whether a copy of that data is in cache. If it is, the processor immediately reads from or writes to the cache, which is much faster than reading or writing to main memory.

Advantage

- 1) It is faster than main memory.
- 2) It consumes less access time.
- 3) It stores the program that can be used within a short period of time.
- 4) It stores data for temporary use.

Disadvantage

- 1) It has limited capacity.
- 2) It is very expensive.

3) Main Memory :- Main memory or primary memory is used to store data and instruction during the execution of program.
→ It holds the binary data. It directly interacts with the CPU during execution. This memory is the semiconductor memory. It is usually volatile in nature, means data is lost in case power failure occurs. Data and instruction required to be processed resides in the main memory. A computer has two subcategory: (i) RAM (ii) ROM

RAM (Random Access Memory)

- 1) RAM is used to store data, program and program result temporarily.
- 2) RAM We can both read from and write to RAM.
- 3) RAM is volatile. i.e. data stored in RAM is lost if there is a power failure.
- 4) RAM contains the instruction, program or data needed by the processor for processing.
- 5) RAM is not reliable.
- 6) It can be changed or the size of RAM can be upgraded.
- 7) RAM are very expensive.
- 8) RAM is small, both in terms of physical size and the amount of data it can hold.
- 9) It is of two types: SRAM & DRAM.

SRAM

- 1) It stands for Static Random Access Memory.
- 2) The word static indicates that the memory retains its contents as long as power is being supplied. An no need to refresh the memory to maintain data.
- 3) It is a faster memory.
- 4) It is larger in size.
- 5) It is expensive.
- 6) It consumes more power.

ROM (Read Only Memory) (13)

- 1) ROM stores the information permanently during manufacture.
- 2) The informations in ROM can only be read.
- 3) ROM is non-volatile.
- 4) ROM contains the instructions required to start a computer, referred to as bootstrap.
- 5) ROM is reliable than RAM.
- 6) It cannot be accidentally changed.
- 7) ROMs are cheaper than RAM.
- 8) ROM is larger than RAM.
- 9) It is of 4 types.
PROM, EPROM & EEPROM, UVBPRAM.

DRAM

- 1) It stands for Dynamic Random Access memory.
- 2) It must be continually refreshed in order to maintain the data.
- 3) It is slower than SRAM.
- 4) It is smaller in size.
- 5) It is less expensive.
- 6) It consumes less power.

PROM : It stands for programmable ROM. It facilitates ~~both for~~ the user once to write data.

EPROM : - Erasable Programmable ROM. It facilitates both erasing and programming on ROM.

EEPROM : - Electrically Erasable Programmable ROM. It can be erased and reprogrammed about ten thousand times.

Both erasing and programming take about ~~by~~ ~~to~~ 10(ms).

In EEPROM any location can be selective erased and programmed.

→ Only one byte at a time can be erased. Instead of the entire chip.

UV EPROM : Ultraviolet erasable programmable ROM. It can be erased by exposing it to ultra-violet light for a duration upto 40 minutes.

y) Auxiliary Memory : -

Auxiliary memory or secondary memory is a memory where data and program are stored permanently. This type of memory holds data until they are physically erased, hence these are non-volatile in nature. CPU directly does not access these memories, instead, the contents of secondary memory are first transferred to the main memory, and then the CPU can access it.

→ There are some popular auxiliary memory given below.

(i) Floppy disk : - It is a magnetic storage medium mainly used for computers and laptops. The Floppy drive is a part of computer, that reads ^{from} and writes to the Floppy disk. It stores less data. It is available in two physical sizes i.e. $5\frac{1}{4}$ " and $3\frac{1}{2}$ ". The capacity of $5\frac{1}{4}$ " is 1.2 MB and capacity of $3\frac{1}{2}$ " is 1.44 MB. Each floppy disk has a circular cut at the centre of the disk, which is locked with the spindle motor to rotate the magnetic coated disk. Another hole on the jacket allows the contact between read/write head of the drive and disk for read/write operation.

Data are recorded on tracks and sectors present on a formatted magnetic disk. During read/write operation, the read/write head of the drive moves to the specific tracks and ~~sector~~ sector depending on the address of the data and performs the read/write operation.

Hard Disk :- Hard disk is permanently installed in the computer and can store huge amount of information. Now a days the capacity of hard disk comes in TB (Terabyte)

A hard disk is a flat, circular, oxide-coated disk which rotates continuously. The disk platter is divided into sectors and each sector is divided into tracks. Information is recorded into the tracks. An access arm in the disk drive positions a read/write head over the appropriate track to read and write data from and to the track. Once the correct section of the track is pointed by read/write head, successive bytes of information can be transferred to the memory at rates of several megabytes per second.

CD-ROM :- It is widely used auxiliary memory. It is based on optical mechanism for storing digital data. It consists of a plastic disk coated with reflected material. Bits are recorded on ponds and lands on this surface, by focussing a high power laser beam on the surface. There are mainly three types of CD-ROM.

- (i) In the first type the manufacturer records the data which can't be overwritten/erased.
- (ii) The second type allows for recording of data by the user only one time after which data become permanent on the disk.
- (iii) The third type allows the user to reuse the disk many times for data recording.

5) Back up Memory :- This memory is used by the user for keeping backup of important data for future reference. Magnetic tapes, CD-ROMs, pen drives, memory cards come under this memory. These are

→ A computer mainly comprises two categories of resources i.e. hardware and software.

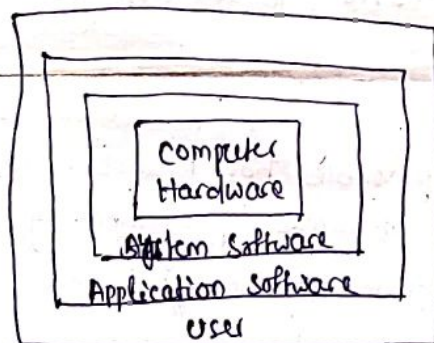
Hardware :- Hardware refers to all the physical components present in a computer, which we can see, touch and feel.

- Ex:-
- (i) Input devices
 - (ii) Output devices
 - (iii) CPU
 - (iv) Storage devices

Software :- Software refers to the set of programs, which make the hardware operational, and those we cannot see, touch and feel.

→ Software is of two types

- (i) System software
- (ii) Application software



(Logical structure of computer)

System software

- 1) It is closer to computer hardware.
- 2) It is difficult to write.
- 3) It is difficult to understand
- 4) It is difficult to manipulate
- 5) It is less interactive
- 6) It is smaller in size and takes less space in memory.
- 7) It is designed using low-level language.

Ex:- operating system, device driver, language processor, utility programs.

Application software

- 1) It is closer to user.
- (2) It is ~~difficult~~ easy to write.
- (3) It is easy to understand.
- (4) It is easy to manipulate.
- (5) It is more interactive
- (6) It is larger in size and takes more space in memory.
- (7) It is designed using high-level language.

Ex:- Audio s/w, video s/w, game s/w, graphics s/w, MS office package, web browsers, all high level programming languages.

Generally system software performs the following activities .

- 1) It provides a platform for installation and development of application software .
- 2) It monitors the effective utilization of various hardware resources .
- 3) It facilitates error free communication between the main computer system and attached peripheral devices .
- 4) It facilitates the execution of a program written in high level language .
- 5) It helps in carrying out some common system maintenance activities .

Operating system : - Operating system is a system software that acts as an interface between the computer and user . It is responsible for managing the various resources and also provides a platform to the user for loading application software :

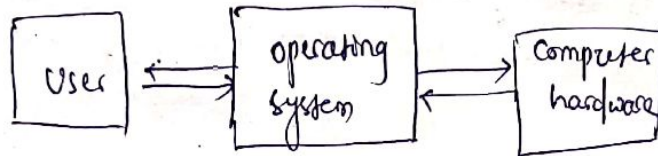
Language processor : - Language processors are the system software which are responsible for translating and interpreting the program written in some programming languages .
Ex) - compiler, interpreter, assembler etc .

Device Driver : - Device drivers are the system software, which are used to establish an error free communication between device and computer .

Utility programs : - Utility programs are capable of directly interacting with computer hardware for various purpose . These are generally used for system maintenance activity , for example - Disk format , partition utility , disk backup utility , system performance monitors .

Overview of Operating System :-

→ Operating system is a system software that acts as an interface between computer hardware and user.



→ Operating system (OS), after being initially loaded onto the computer by a boot program, manages all other programs and resources in a computer. That is why operating system is also called resource manager.

Functions of Operating System :-

- (1) Resource Management
- (2) Processor Management
- (3) Memory Management
- (4) Device Management
- (5) Information management

Resource Management :- OS manages various resources of computer by

- i) Keeping track of all available resources.
- ii) Decides a policy, to allocate available resources to various requesting jobs.
- iii) Physically allocate the resources to the requesting jobs.
- iv) Finally de-allocate the resources.

Processor Management :-

→ OS keep track of all active processes and available processor slot, allocate the processor to various active processes based on some policy. Then it takes back the processor after completion of the process.

→ OS performs the following functions through different system calls.

- i) Create child process identical to parent process.
- ii) Terminate a process
- iii) Assign / change priority of a process
- iv) Suspend a process
- v) Block / allow a process
- vi) Delay a process.

Memory Management :-

- OS keeps track of available memory.
- It allocates the memory to different active processes based on some priority policy.
- Finally it takes back the memory from the processes when they are complete.

Device Management :-

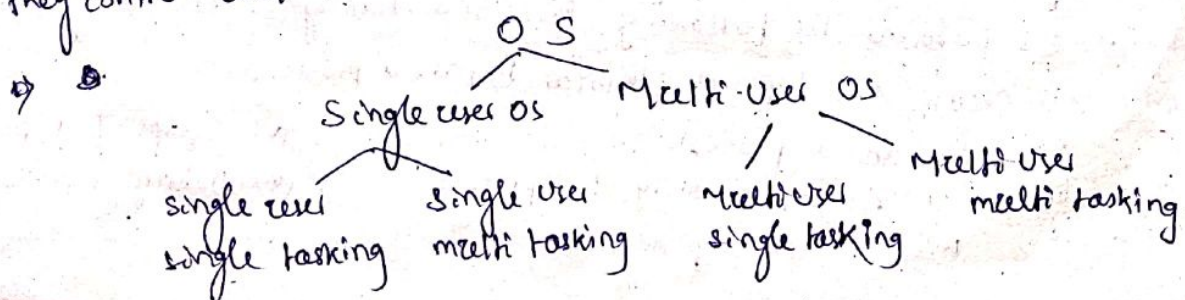
- OS keeps track of the I/O devices connected to the computer.
 - It decides a policy basing on which I/O devices will be assigned to different requesting processes.
 - Physically allocate the device to the processes.
 - Finally release the devices after completion of the job.
- * The method of assigning the I/O devices to processes is known as spooling.

Information Management :-

- OS does the following operations for information management.
 - Creation of file and directory
 - open/closing of file
 - Read/write data from/to a file
 - Read/write data from/to a buffer
 - Maintain the file states
 - Assign and maintain directory structure.
- * Buffer : It is a temporary storage area shared by hardware devices to store intermediate result/data during execution.

Types of Operating System :-

Operating system is categorised based on the types of computer they control and the sort of applications they support as follows.



Single User - Single tasking OS :- This OS is designed to manage the computer so that one user can effectively do one task at a time.

Ex:- Dos, Palm OS

Single User - Multi tasking OS :- This OS is designed to manage the computer so that one user can do more than one task at a time.

Ex:- Microsoft's Windows, Apple's MacOS :

Multi User - multi tasking OS :- This OS allows multiple users to perform multiple tasks at the same time.

Ex:- UNIX, Ubuntu, LINUX, IBM AS400, Windows 10 etc.

Typical classifications of OS

① Batch Operating System

In a batch operating system each user prepares his program and submits it to the computer. To speed up processing the operator collects the jobs or programs with similar needs and batches together and runs them through the computer as a group. The output is then sent back to the appropriate programmer.

Disadv:-

- 1) Lack of interaction between the user and the job.
- 2) CPU is often idle, because speed of I/O devices is slower than CPU.
- 3) Difficult to provide desired priority.

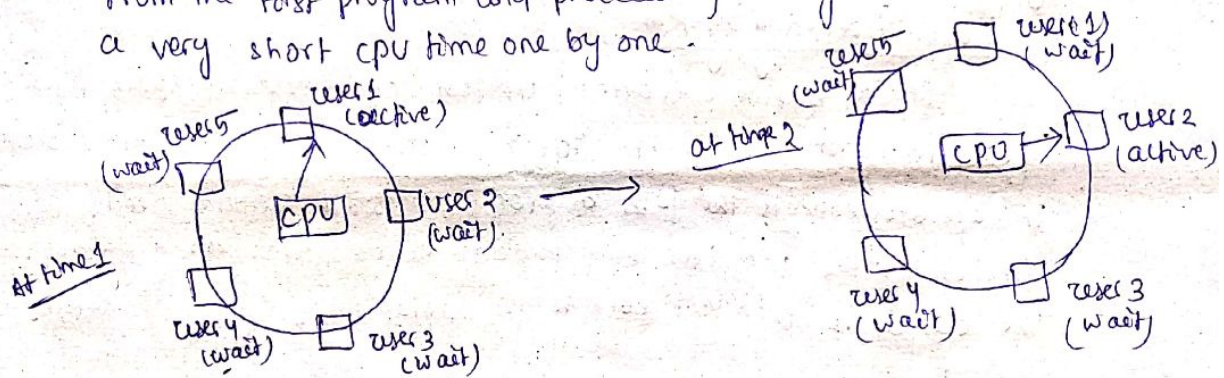
② Multi programming OS :-

→ The main objective of multi programming operating system is to have some programs running at all times, so as to maximize CPU utilization.

→ In multi programming OS, multiple programs are loaded into the computer's memory and execution begins with the first program and continues till an instruction waiting for an I/O device is reached. The context of the program is stored and the second program is given a chance to run. This process continues until all programs finished running. This system provides maximum job throughput within a time frame.

③ Time-sharing OS ! - The main objective of time-sharing OS is to minimize the CPU response time.

- In time sharing operating system each task is given some time to execute, so that all the tasks work smoothly. Each user gets time of CPU as they use single system.
- This OS supports multiple users to work at multiple terminals.
- The CPU time shared among multiple users sitting at various terminals ~~the time~~ is called time slice or time quantum.
- CPU time is equally distributed among the users, ~~on the~~ system.
- The basic idea behind this system is to allow all user's program to have a brief share of the CPU time. Each user program starting from the first program and proceeding through the last, is allocated a very short CPU time one by one.



Advantage ! -

- 1) Each task gets an equal opportunity.
- 2) Less chances of duplication of software.
- 3) CPU idle time can be reduced.

Disadvantage ! -

- 1) Reliability problem.
- 2) One must have to take care of security and integrity of user programs and data.
- 3) Data communication problem.

Ex - UNIX, VMS

(4)

Multi Processing Operating System :-

→ Multi processing OS works with two or more central processing unit (CPU) within a single computer system.

→ In this system all CPUs may be equal, or some may be reserved for special purposes.

Advantage :-

1) It enables several programs to run concurrently / simultaneously.

(a) ~~More than one~~

(a) It increases system throughput (means more ^{no. of} tasks completed at a particular time).

Disadvantage :-

1) Multiprocessing system is more complicated than single-processing system.

(a) System must allocate resources to competing processes in a reasonable manner.

Multitasking Operating System

→ In multitasking OS only one CPU is involved to perform the tasks.

→ But it supports to perform multiple tasks at the same time.

→ It switches from one program to another so quickly that it gives the appearance of executing all of the programs at the same time.

→ For example, at a particular time the computer can perform word processing; it can send data to a printer to print, it can send files to secondary storage device to ~~save~~ etc.

Advantage :-

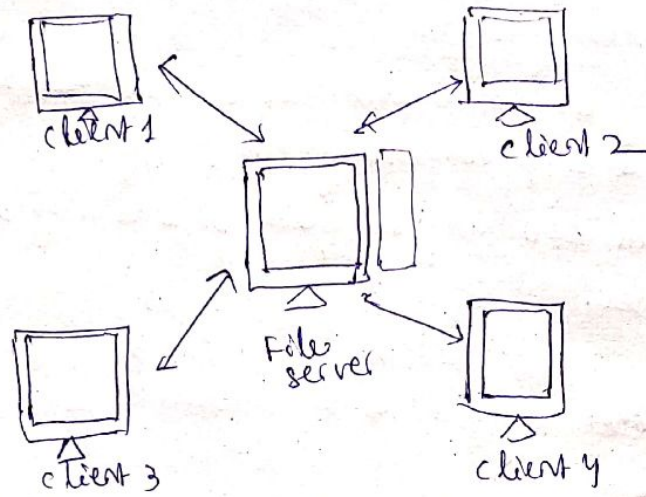
1) Coordination of various resources is more.

Network Operating System :-

→ This operating system has the capability to manage data, users, groups, security, applications and other networking functions by connecting computers and devices into a local-area network (LAN).

→ The main objective of networking operating system is resource sharing.

→ This OS work on client-server system.



Advantage :-

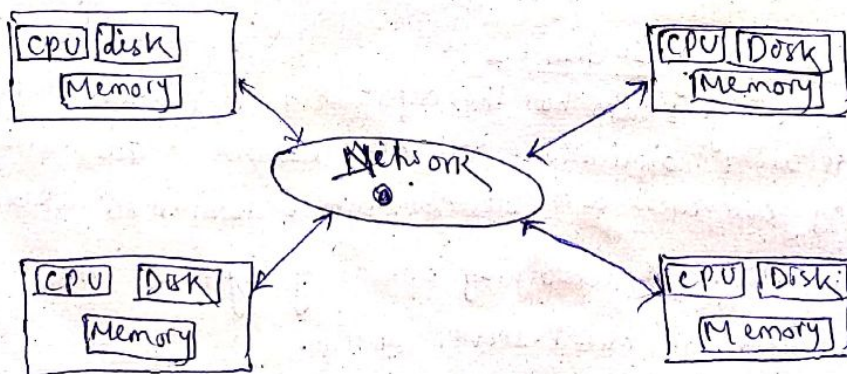
- 1) Highly stable centralized server
- 2) Security concerns are handled through servers.
- 3) Net technologies and hardware up-gradation are easily integrated to the system server.
- 4) Resources are shared among different clients.

Disadvantage

- 1) Servers are costly.
- 2) Maintenance and updates are required regularly.
- 3) Complex in nature.
- 4) A policy must be decided to allocate resources among users.

Distributed Operating System

- The main objective of distributed OS is load sharing.
- In this OS, computers communicate with each other using a shared communication network.
- ~~Independence~~ In this OS, the computers are located at various locations with their own memory unit and CPU. A large task is divided into a number of smaller tasks and distributed among the ~~com~~ users located at different locations. They perform their tasks independently and submit to the network.



Advantage :-

- 1) Failure of one device will not affect the entire system.
- 2) Load of one ^{system} can be shared among others, if that system is failed.
- 3) Computation speed is high.
- 4) Delay in data processing is reduced.

Disadvantage :-

- 1) Failure in main network will stop the entire communication.
- 2) This type of OS is very expensive.

Real-time Operating System :-

- This OS serves the real-time systems. The time ~~req~~ interval required to process and responds to inputs is very small. This time interval is called response time.
- In real-time OS, the total correctness of an operation depends not only upon logical correctness, but also upon the time in which it is performed. Such applications include embedded systems, industrial robots, spacecraft, industrial control and scientific research equipment.
- Real-time systems are used when there are time requirements are very strict.
- There are two types of real-time OS.
 - ① Hard-real time system :- These OSs are meant for the applications where time constraints are very strict and ~~the~~ even the shortest possible delay is not acceptable. These systems are built for saving life like automatic parachutes or air bags which are required to be readily available in case of any accident. Virtual memory is not found in these systems.
 - ② Soft Real time system :- These OSs are for applications where for time-constraint is less strict.

Advantages ! -

- 1) Maximum utilization of devices and system. Thus more output from all the resources.
- 2) Time assigned for shifting tasks in these systems are very less.
- 3) Focus on running applications and less importance to applications which are in queue.
- 4) These types of systems are error free.
- 5) Memory allocation is best managed in these type of systems.

Disadvantage ! -

- 1) Very few tasks run at the same time and their concentration is very less to avoid error.
- 2) System resources of these systems are not so good, and are very expensive.
- 3) The algorithms are very complex and difficult for the designer to work on.

Examples ! -

Scientific experiments, medical imaging systems, industrial control systems, weapon systems, robots, missile systems etc.

Disk Operating System, - (DOS)

(2)

- Disk Operating System was the first widely-installed operating system for personal computer.
- The first personal computer version of DOS was PC-DOS developed for IBM by Bill Gates and his new Microsoft Corporation.
- DOS translates the user commands in the format, that the computer can understand and instruct the computer to work accordingly. It also translates the result and any error message in the format the user can understand.
- The main functions of ~~DO~~ DOS are to manage disk files, allocate system resources according to their requirement.

Features of DOS, -

- 1) DOS is a single user operating system.
- 2) DOS provide hierarchical file system.
- 3) It supports many application packages.
- 4) It is a command based operating system. This means user has to type commands to perform a task.
- 5) DOS commands are divided into two categories:
 - (i) Internal commands
 - (ii) External commands.
- 6) The code for internal commands are included in the COMMAND.COM file, whereas each external command has an individual file to contain the code.
- 7) DOS requires less memory space for installation and running.
- 8) All internal commands are processed by the command interpreter which has 3 portions, i.e. (i) start-up portion, (ii) resident portion (iii) transient portion.

DOS Internal commands

Internal commands of DOS do not reside on disk. When this command is given by the user, the computer refers to the command.com file for its execution.

Commands

- ① md → This command is used to make a new directory.
Ex: - c:\>md sahil ←
- ② cd → This command is used to change the current directory.
Ex: - c:\>cd sahil ←
 c:\sahil>
- ③ rd → This command is used to remove/delete the directory.
Ex: - c:\>rd sahil ←
- ④ cls → This command is used to clear the display screen.
Ex: - c:\>cls ←
- ⑤ ren → This command is used to rename a file/directory.
Syntax: ren oldname newname
Ex: - ren f1 f2 ←
- ⑥ copy con → This command is used to create a new file and allow the user to write the contents.
Ex: - copy con f3 ←
 File name
- 7) type → This command is used to display the content of the file.
Ex: - c:\>type f3 ←
- 8) Date → This command is used to display the current date.
Ex: c:\>Date ←
- 9) Time → This is used to display the current time.
Ex: - c:\>time ←
- 10) ver → This is used to display current version of DOS.
Ex: - c:\>ver
- 11) vol → This is used to display the disk volume.
- 12) Dir → This is used to display ~~current~~ directory contents.
- 13) Path → To display or set a search path for executable files.
- 14) Copy - To copy one or more files to another location.
Ex: - copy file1 file2 ←
- 15) Exit → To quit from COMMAND.COM program.

DOS External Commands

(7)

External commands of DOS reside in separate files on the hard disk and have an extension of .COM, .EXE, .BAT etc. Some external commands are as follows:

(1) `diskcopy.com` → It copies the content of one floppy disk to another.

(2) `attrib.exe` → It sets the read-write-execute attribute of a file.

(3) `Tree.com` → It displays the entire file structure in a tree manner.

(4) `Backup.exe` → It creates a backup file from hard disk to floppy disk.

(5) `more.com` → It displays the contents screen by screen.

(6) `restore.com` → Restore the file from the backup disk.

(7) `Format.com` → Formats a disk for use with MS-DOS.

(8) `Print.exe` → Used to print a given file.

(9) `Find.exe` → Used to search for a specific text in a file.

(10) `Start.exe` → It runs a window program or an MS-DOS program.

(11) `Sort.exe` → It sorts inputs and writes results to the screen, a file or another device.

UNIX Operating System

UNIX operating system was developed in 1969 by a group of AT&T employees of Bell Labs including

Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna.

→ It is widely used in both servers and workstations.

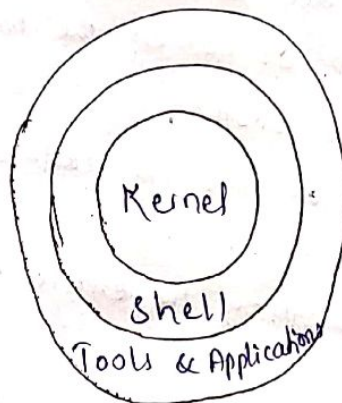
→ It is a very powerful and flexible operating system.

→ UNIX is functionally organized in three levels.

(i) Kernel: Which schedules tasks and manages memory.

(ii) Shell: Which converts and interprets user's commands calls programs from memory and executes them.

iii) Tools and applications: Which offer additional functionalities to the operating system.



(Three level of UNIX system)

Kernel: - Kernel is the heart of operating system. It controls the hardware and manages parts of the system on and off at the programmer's commands. If the user asks for the list of file (ls) in a directory, the kernel tells the computer to read all the files in that directory from the disk & display to the user.

Shell: - Shell is a Command Line Interpreter (CLI) that interprets the user commands.

→ It provides the functionality of "pipes" by which a number of commands can be linked together, permitting the output of one program become input to the other program.

→ There are several types of shell used such as Bourne shell, C shell and motd driven shell.

Tools and Applications: - There are hundreds of tools available to UNIX users. Tools are grouped into categories for certain functions, such as word processing, business applications or programming.

UNIX Features

- 1) Portability ! - UNIX can be used from one brand of computer to another with a minimum code changes as it is written in C language.
- 2) Machine Independent ! - UNIX is machine independent. It hides the architecture details from the user making it easier to write application program that will run in any machine.
- 3) Multi Users ! - UNIX is a multiuser operating system, which supports multiple users to operate simultaneously that allows sharing of system resources among users.
- 4) Hierarchical file system ! - UNIX has a hierarchical file system to store information, for easy maintenance of files.
- 5) UNIX shell ! - UNIX provides an editor and command line to the users for writing command scripts.
- 6) Utility Programs ! - UNIX has got over 200 utility programs for various functions.
- 7) Background processing ! - In UNIX user has the facility to submit a task for processing and switch to another task, and the submitted task is processed as background task to utilize system time effectively.
- 8) Software development tool ! - UNIX offers variety of software development tools through which user can develop different application softwares.
- 9) Pipes and filters ! - Pipes and commands enables the user to give multiple commands in a single command.
- 10) Security ! - Security level of UNIX is very high, which doesn't allow any user to enter into the core part. This is why, UNIX OS is not affected by any virus program.

UNIX Commands

- 1) mkdir → To create a new directory
- 2) cd → To change directory
- 3) rmdir → To remove / delete the directory
- 4) cp → To copy a file
- 5) lp → To print a file
- 6) date → To display current date.

- 7) who → To display the current user
- 8) cat → To create and display content of a file.
- 9) chmod → To change the read-write-execution permission.
- 10) vi → Involves vi editor to edit the files.
- 11) ls → To display the contents of a directory.
- 12) pwd → To display current working directory.

Windows Operating System :-

Microsoft Windows is a series of software operating systems and graphical user interfaces produced by Microsoft. It is the most widely used operating system on a personal computer. The first version of Microsoft Windows is Windows 1.0, released on 20 November 1985.

Versions of Windows

Windows 1.0 → Windows 2.0 → Windows 3.0 → Windows 3.1,
 → Windows 95 → Windows 98 → Windows 2000 → Windows XP
 → Windows 2007 → Windows 2016 → Windows 2017 → Windows 7
 → Windows 8 → Windows 10

Features :-

- 1) GUI :- Windows provides a graphical user interface (GUI) to the user, for interacting with the computer. It provides the various commands in the form of menu and icons. It is operated using a mouse.
- 2) Desktop :- Desktop is a display area that represents the kinds of objects. ~~one might~~ it provides the user with all facilities for using the computer.
- 3) Icon :- It is a small graphical symbol which represents a particular executable program. Icons are kept on the desktop or inside an opened window.

4) Folder :- Folder is a collection files. We can store different files in a folder.

5) Frame :- frame is the boundary of a window. When the mouse is kept over the frame, it's cursor changes to double headed arrow, and allows the user to resize the window.

6) Taskbar :- It is normally present at the bottom of the screen and contains the active application program.

7) Title bar :- ~~Each~~ It is present at the top of the screen. It contains the title of the application file along with the application opened.

8) Menu bar :- Menu bar present just below the title bar and contains pull down menu item. Each pull down menu contains the commands, related to various tasks and applications.

9) Scroll bar :- Window contains a vertical / horizontal scroll bar if the normal window size is larger than opened window. The button present on the scroll bar can be moved from top to bottom or left to right for displaying the full content of the window.

10) Tool bar :- It is generally displayed just below the menu bar, ~~can~~ and contains button corresponding to various tools for executing various tasks.

11) Minimize, maximize, close button :- These buttons are present at the top right corner of the window.

→ By clicking on the minimize button, the application window is closed, but application keeps on running.

→ By clicking on the maximize button, the application window occupies, the full screen.

→ By clicking on close, the application is closed.

12) Dialog box :- Dialog box, provide the user an interface to enter text for ~~diff~~ setting different parameters.

- 13) Accessory: - It contains frequently needed tools and utilities, ~~It contains~~ such as system tools, entertainment tools, calculator, paint, Notepad, Wordpad etc.
- 14) Control Panel: - It contains various options for doing system maintenance adding ~~tools~~ removing program, performing system setting etc.
- 15) My Computer: - It contains the entire secondary storage device. By using this, the user can view the folders and files existing on different drives, free storage space etc.
- 16) Recycle bin: - It holds all the deleted files. If we make the recycle bin empty, then the files are permanent-ly deleted from the storage device, or we can restore them.
- 17) Windows Explorer: - It is used to view all the files and folders on the storage device. We can also delete, copy files, move files from one folder to another, through windows explorer.

Comparison between Windows & DOS

- | <u>Windows</u> | <u>DOS</u> |
|--|--|
| 1) It is graphical user interface. | 1) It is command user interface. |
| 2) It is a powerful and fast OS. | 2) It is a less powerful & slow OS. |
| 3) It provides in-built commands in various applications and accessories which can be used on a mouse click. | 3) All the commands have to be typed at the DOS prompt which is a difficult job. |
| 4) There is less chance of error. | 4) There is always a chance of error, as the user has to type the command at DOS prompt. |
| 5) Virtually all DOS commands are available in Windows. | 5) All the commands available in Windows are not available in DOS. |
| 6) It is a multitasking operating system. | 6) It is a single tasking operating system. |
| 7) It supports graphics. | 7) It support does not support graphics. |
| 8) Windows is a very costly software. | 8) DOS is less costly. |

Comparison between UNIX and DOS

(10)

UNIX

- 1) UNIX is a powerful operating system.
- 2) It is a multitasking, multiprogramming operating system.
- 3) It supports visual display and graphics.
- 4) It supports networking of computers.
- 5) UNIX supports login concepts, and therefore provides high security level.
- 6) UNIX is very friendly for writing programs.
- 7) UNIX provides the features to host web pages.
- 8) It is a costly software.

DOS

- 1) DOS is a less powerful operating system.
- 2) It is a single user, single-tasking operating system.
- 3) It doesn't support visual display and graphics.
- 4) It doesn't support networking of computers.
- 5) It does not provide security level.
- 6) It is not user friendly, for writing programs.
- 7) It doesn't provide features to host web pages.
- 8) It is less costly software.

Programming Languages :-

Computer programming language fall on the following categories

- 1) Machine level language
- 2) Assembly language
- 3) High level language

Machine level language :-

Computer is a digital device, and can understand only binary data. Therefore instructions given to computer as a stream of binary data i.e 0 and 1. The language that computer can understand is binary language or machine language. It is a low level language. One need to have a detail knowledge of the internal hardware architecture of the computer for writing the program.

Advantage

- 1) A machine level program doesn't require an interpreter for execution.
- 2) A machine level program is executed very fast.

Disadvantage :-

- 1) A machine level program is machine dependent. i.e. a program written for a particular machine can't be executed by other computer.
- 2) The programmer need to have an idea about the computer hardware, hence it is very difficult to write this program.
- 3) There is more chance of error.
- 4) A machine level program is difficult to understand and modify.

Assembly Language :-

- Instructions in assembly language program is written using mnemonic codes. Memory address location is addressed by using alpha numeric variables, which helps the programmer to remember the location address easily. It provides an additional instruction known as pseudo instruction for various activities.
- In assembly language program each instruction consists of two parts i.e. opcode and operand.

OPCODE (operation code)	OPERAND (address/operand)
----------------------------	------------------------------

- A program written in assembly language is translated using an ~~asm~~ assembler, which translates the mnemonic codes and convert them into machine readable form.

Advantage :-

- 1) Assembly language program is easier to understand.
- 2) It is easier to locate error and debugging ~~in~~ an assembly language program.
- 3) It is easy to find out mention the address operand, in the form of alphanumeric variables/ labels.

Disadvantage

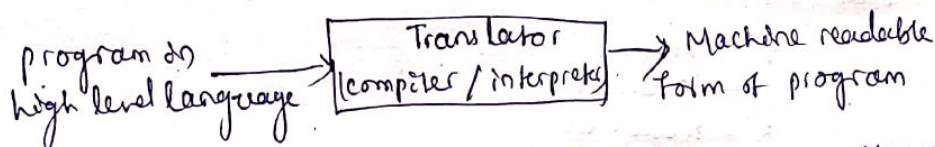
- 1) Assembly language program is machine dependent.
- 2) A knowledge of hardware architecture is required to write assembly language program.
- 3) It is restricted to low level programmers only.

High level language :- (HLL)

(11)

A high level language is an English like language.

- 1) It is user friendly and provides the user with facility and instruction set, so that user can encode an algorithm easily.
- 2) As computer can understand only binary code, a translator is required to translate the high level language to binary format. That translator is known as compiler or interpreter.



- 1) Each high level language has got a separate compiler or interpreter which is capable of interpreting various syntax and semantics of the program.

Advantage

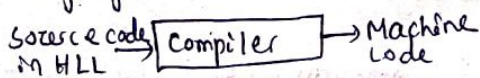
- 1) It is machine independent.
- 2) It is easier to learn and use.
- 3) It generally gives minimum number of errors.
- 4) Documentation of high level language is better.
- 5) It is easy to modify and maintain the program written in high level language.

Disadvantage :-

- 1) A translator is required to translate the ^{program in} high level language to machine level.
- 2) Thus a price in computer time is paid.
- 3) The object code generated by the translator might be inefficient.

Compiler

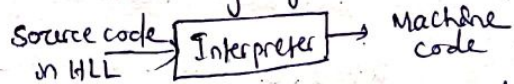
- 1) Compiler translates HLL to machine language.



- 2) It translates the entire program at a time.
- 3) It requires more memory space.
- 4) If error is detected nothing is executed.
- 5) Execution time is less.
- 6) Neither source program nor compiler required for execution.
- 7) It is costly.

Interpreter

- 1) Interpreter also translates HLL to machine language.



- 2) It translates the program line by line.
- 3) It requires less memory space.
- 4) If error is detected stops but previous statements are already executed.
- 5) Execution time is more.
- 6) Source program along with interpreter is required for execution.
- 7) It is less costly.

Computer Virus :-

Computer virus is a malicious software written intentionally to enter a computer without the user's permission or knowledge with an ability to replicate itself by spreading to harm computer resources.

Full form of VIRUS

V - Vital
I - Information
R - Resources
U - Under
S - Seize

→ Viruses increase the chances of spreading to other computers by infecting files on a network file system. Computer viruses are most easily spread by attachments in e-mail messages or by instant messaging or by downloading from the internet.

Types of Virus :-

Viruses are classified based on their origin, techniques, types of files they infect, where they hide, kind of damages they cause, the type of operating system or platform they attack. The common types of viruses are:

1) Boot Sector Virus :- A boot sector virus infects the master boot record in the hard disk. They replace the boot record program which is responsible for loading of operating system in memory, by copying it else where in the disk or by overwriting it.

Ex:- From, Disk Killer, Stone virus, Michael Angelo.

→ Boot sector virus is the most dangerous kind of virus, and are also known as system virus.

2) Program Virus :- The program virus infects the executable files with extension .exe, .com, .bin, .drv, .olx, .sys etc.

The viruses are loaded into memory during execution of the files, along with them. The virus program remain active in the memory and multiply itself making the memory full.

Ex:- Sunday, cascade.

3) Macro Virus :- This type of virus infects the macros within a document or template. When a user opens a word document or processing or spread sheet document, the macro virus is activated and infects the normal template. This virus spreads from one computer to another through the infected document file.

Ex:- DMV, Nucleus, word concept.

4) Multipartite Virus :- This virus is the hybrid of boot sector virus and program virus. This virus first infects the program files and when the infected program is executed, then virus ~~first~~ infects the boot record. When the system is booted next time, the virus from the boot record loads itself to the memory and infects other programs.

Ex:- Invader, Flip, Tequila

5) Polymerphic Virus :- This virus is capable of encrypting its code in different manner, so that each appears different in each infection. These viruses are difficult to detect.

Ex:- Cascade, Evil, Piled, Virus 101.

6) Stealth Virus :- This virus use certain ~~very~~ techniques to avoid detection. They usually direct the disk head to read a wrong sector instead of the one in which they reside, or they change reading of the infected file's size shown in the directory list.

Ex:- Joshi, whale, Frodo.

Virus Life Cycle

Each virus goes through a life cycle comprising of the following phases.

- 1) Virus creation
- 2) Virus infection and replication
- 3) Virus activation
- 4) Virus detection
- 5) Virus eradication

Virus creation :- A computer virus is created by a programmer who writes a program with an intention to harm the computer, which later spread as computer virus.

Virus Infection and replication :- After creation, the virus infects certain computers by many ways and replicates itself in the hard disk or memory of the computer. It is a continuous process through which the computer virus copies itself from one computer to other.

Virus activation :- Once a virus infects a computer, it gets activated automatically. Depending on the nature of virus, it activates itself on occurrence of a particular event, such as certain date, user executes a particular file, user opens an infected e-mail etc. During this phase the virus makes all damages to the computer.

Virus detection :- Virus is detected by some anti-virus program. An expert user can detect the presence of a virus more quickly in comparison to a novice user.

Virus eradication :- Virus is eradicated through an appropriate antivirus software. It is very essential to update the anti-virus software at a regular interval. User can install anti-virus program which prevents the entry of virus into the computer.

Symptoms of Virus Infection :-

- 1) Computer is giving problem during booting or takes a lot of time for booting.
- 2) Computer is resetting automatically.
- 3) Computer is hanging when the user tries to execute a particular program.
- 4) Computer displays some unusual figures/signs on the screen.
- 5) Computer is performing some operations automatically, though user has not given command for that operation.
- 6) Computer is giving some message such as "Insufficient memory" or "Disk full" etc.
- 7) If hard disk is accessed usually many times.

Cause of Virus Infection

~~Any computer virus can be affected or infected~~ infect.

Any computer can be affected or infected by virus in either of the following ways.

- 1) By inserting a virus infected CD or pendrive to the computer.
- 2) Through a LAN setup where one computer belonging to the network has been affected.
- 3) Through internet and e-mail.
- 4) Through installation of illegal/pirated software specially games.

Prevention of Virus Infection

- 1) Don't allow outside CD / floppy or pendrive without proper scanning.
- 2) Always install and update a suitable anti-virus software from an authorized source.
- 3) protect the system by setting the antivirus software and the fire wall to the auto protection mode.
- 4) Don't visit the websites which are not reputed.
- 5) Always protect your computer from unauthorized use by setting password.
- 6) Don't open known emails received in your mailbox.
- 7) check the size of the executable files at a regular interval.

Examples of some common viruses

Brain, Friday 13, Happy Birthday, April 1st, Jerusalem & Israel, die hard-2, liberty, from, monkey, slow, black monday, I love U etc.

Examples of anti-virus software

Anti-Virus! - Anti-virus is a software program used to detect and prevent from viruses. The functions of anti-virus are:-

- 1) It scans specific files or directories.
- 2) Allows to schedule scan to automatically run.
- 3) Allow to initiate a scan of a specific file, CD or a Hard drive at any time.
- 4) It removes any malicious code detected.
- 5) show the health of the computer.

Ex: - Norton, McAfee, Dr. Solomon, e-trust, PC-clean, NOD-32, Quickheal

Computer Network :-

Computer network is defined as a group of computers connected in some ways so as to be able to exchange data.

→ The main objective of network is data communication.

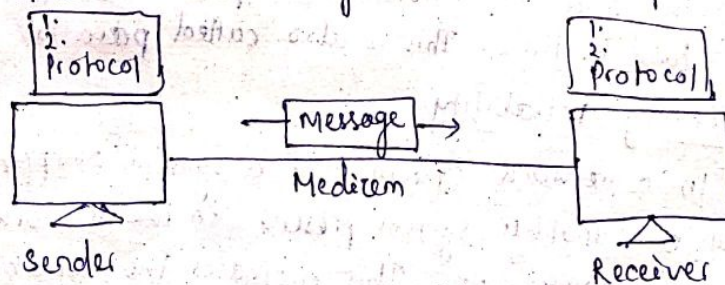
Data Communication :- Data communication is the exchange of data between two devices through some transmission medium such as a wire cable. For data communication to occur, the communicating devices must be a part of a communication system made up of a combination of hardware (physical component) and software (programs).

Characteristics of data communication :-

- 1) Delivery :- The system must deliver data to the correct destination. That means data must be received by the intended device or user.
- 2) Accuracy :- The system must deliver the data accurately. Data that have been delivered wrong are retransmissible.
- 3) Timeliness :- The system must deliver data in a timely manner. Data delivered late are useless.

Components of data communication :-

A data communication system has five components.



- 1) Message :- Message is the information (data) to be communicated. It consists of text, numbers, pictures, sound or video, or any combination of these.
- 2) Sender :- Sender is the device sends the data message. It can be a computer, workstation, telephone handset, video camera etc.

3) Receiver :- Receiver is the device that receives the message. It can be a computer, workstation, telephone handset, television etc.

4) Medium :- Transmission medium is the physical path by which a message travels from sender to receiver. It could be a twisted-pair cable, coaxial cable, fiberoptic cable or radio waves etc.

5) Protocol :- A protocol is a set of rules that governs data communications. It represents an agreement between communicating devices.

Goals of Network :-

Every network has the following goals.

1) Resource sharing :-

The major goal of any network system is resource sharing. In a computer network, it is possible to share the various types of resources, in terms of computer hardware, software and peripheral devices between the individual system. This breaks the geographical barrier of resource usage.

2) Job sharing :-

When computers are connected to a network, a large task can be divided into small modules and distributed to various computers present in a network. This technique reduces the load on individual computer, but finishes the whole task in time. This is also called parallel processing.

3) Achieving reliability :-

In a network system, if a system (computer) fails, we can use another system present in the network to attend the ongoing job. This increases the reliability of the system.

Protocol:-

A protocol is a set of rules that governs the communications between computers on a network. These rules include guidelines that regulate the network characteristics, such as: access modes, allowed physical topologies, types of cabling and speed of data transfer.

Types of Network Protocols:-

1) TCP: TCP stands for Transmission Control Protocol.

TCP is a popular communication protocol which is used for communicating over a network. It divides any message into a series of packets that are sent from source to destination and there it get reassembled at the destination.

2) IP: IP stands for Internet Protocol. It is responsible for

(i) IP addressing. The IP addresses in packets help in routing them through different nodes in a network until it reaches the destination system.

(ii) Host-to-host communication

(iii) Packet formatting

(iv) Fragmentation

(3) UDP: UDP stands for User Datagram Protocol.

(4) POP3: POP stands for Post Office Protocol. POP3 is designed for receiving ~~E-mail me~~ incoming E-mail messages from remote servers.

(5) SMTP:- SMTP stands for Simple Mail Transfer protocol. SMTP is used to send and distribute outgoing E-mail messages on internet.

(6) FTP: FTP stands for File Transfer protocol. FTP allows users to transfer files from one system to another. Types of files may include program files, multimedia files, text files, and documents etc.

(7) HTTP: HTTP stands for Hyper Text Transfer Protocol.

HTTP is used for sending and receiving webpages over internet. HTML tags are used for creating links. These links may be in any form like text or images. It is designed on client-server principles which allow a client system to establish a connection with the server system for making a request. The server acknowledges the request initiated by the client and respond accordingly.

(8) DHCP: DHCP stands for Dynamic Host Configuration Protocol. It allows the computer to connect to other computers automatically.

(9) Telnet: - Telnet is used for connecting one system with another, which is called remote login. The computer which requests for the connection is called local computer and the computer which accepts the connection is called remote computer.

(10) Gopher: - Gopher is used for searching and retrieving or displaying documents from remote sites. It is possible to start an online connection with other computer through gopher.

(11) ARP: ARP stands for Address Resolution Protocol. It is used for mapping an internet protocol address to physical machine address.

imp Data Transmission Mode

The way in which data is transmitted from one place to another is called data transmission mode, also called data communication mode. It indicates the direction of flow of information. There are three types of transmission modes. Those are as follows:

(1) Simplex Mode :-

- In simplex mode, data can flow in only one direction.
- In this mode, a sender can only send data and cannot receive it. Similarly, a receiver can only receive data but cannot send it.
- In simplex mode, it is not possible to confirm successful transmission of data. It is also not possible to request the sender to re-transmit information.

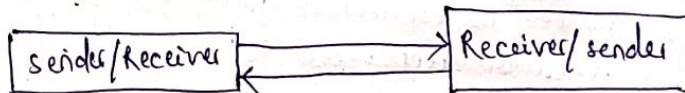


Ex: ① Data sent from computer to printer.

② ~~Radio~~ T.V and remote control communication.

(2) Half-duplex Mode :-

- In half-duplex mode, data can flow in both directions but only in one direction at a time.
- In this mode, data is sent and received alternatively. It is like a one-lane bridge where two-way traffic must give way in order to cross the other.



Ex: - Internet browsing, in which user sends a request to a web server for a web page. Web server receives the request and sends data of the requested page.

(3) Full-duplex Mode :-

- In full-duplex mode, data can flow in both directions at the same time.
- It is the fastest directional mode of data communication.
- In this mode both the sender and receiver can send and receive the message at the same time.



Ex: ① Telephone communication system, where two persons can talk at the same time.

② Automobile traffic on a two-lane road, where the vehicles can move in both directions at the same time.

imp Network Topologies

A network topology refers to the design or the structure of the network.

→ It describes the schematic description of a network arrangement and configuration of cables, computers and other peripherals, connecting various nodes through lines of connection.

There are networks are distinguished based on their size (in terms of the number of machines), their data transfer speed, and their reach.

→ There are many types of network topology.

1) Bus topology.

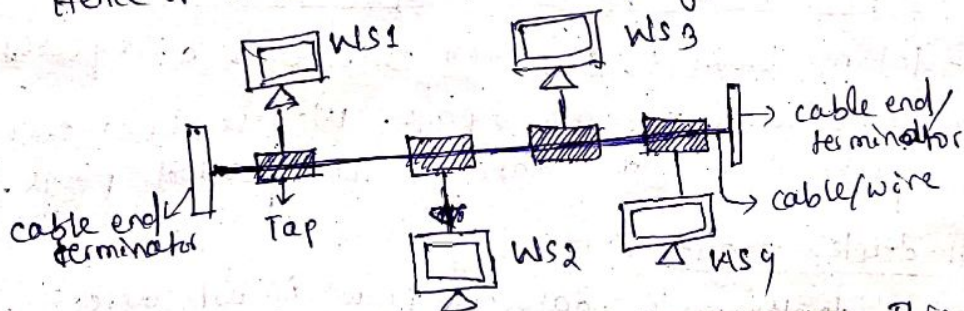
2) Star topology

3) Ring topology

4) Tree topology

5) Mesh topology.

Bus topology :- In this topology every computer or node of network device is connected along a single wire and cable. Hence it is called linear bus topology.



→ The cable has terminator at both end point. This terminator removes the undelivered data.

→ In this topology, a device or workstation (WS1, WS2, WS3, ...) wanting to communicate with another device on the network sends a broadcast message onto the wire or cable, that all other devices see, but only the intended recipient actually accepts and processes the message.

Advantage! -

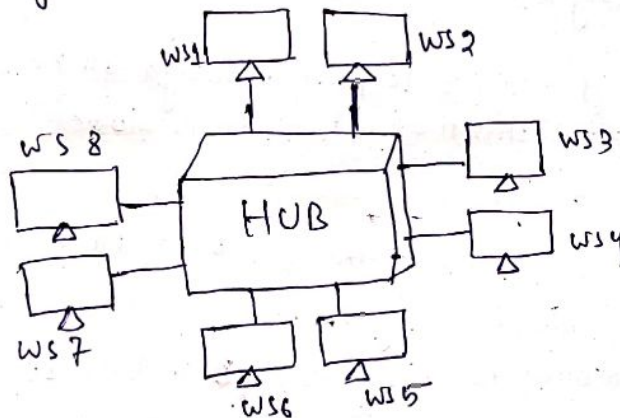
- 1) It is cheap in cost and used in small networks.
- 2) It requires less cable
- 3) Failure of one doesn't affect the other network.
- 4) It is easy to understand and install.

Disadvantage! -

- 1) The entire network shut down if there is a failure in main cable.
- 2) It transmits data only in one direction.
- 3) Terminators are required at both ends of the backbone cable.
- 3) cable has a limited length.

Star Topology! -

- In this topology all the computer or workstations are connected to a central node called as a Hub or switch.
- The hub manages and controls all functions of the network.
- Every node has its own dedicated connection to the hub.

Advantage! -

- 1) No disruptions to the network when connecting or removing devices.
- 2) Failure of one node does not affect the entire network system.
- 3) It is easy to install and to detect failure.

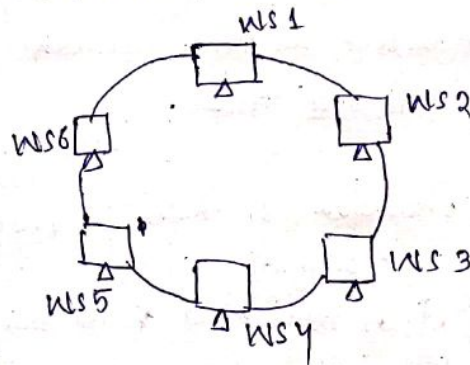
Disadvantage! -

- 1) If the connection fails, the nodes attached are disabled.
- 2) It is more expensive than bus topology.

Ring Topology! -

- In this topology all the computers are interconnected in a ring like structure and each node has a direct point to point connection with each neighbouring node.
- In this topology, every device has exactly two neighbors for communication.
- All messages travel through the ring in the same direction (either "clockwise" or counterclockwise)

→ The data will pass through all the nodes and only accepted by intended destination node.



Advantage

- 1) There is no need for network server to control the connecting between workstations.
- 2) Transmitting network is not affected by adding more nodes.
- 3) It is cheap to install.

Disadvantage :-

- 1) A failure in any node breaks the loop and takes down the entire network.
- 2) Each packet of data has to pass through all the computers between source and destination. Hence it is slower.

Tree Topology :-

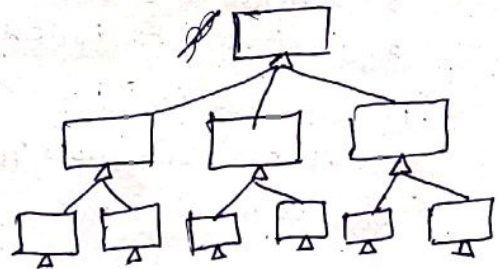
- Tree topology forms a hierarchical structure which has a central root node at the top of the hierarchy.
- A tree topology combines the characteristics of linear bus and star topologies.
- A tree topology starts with the root node and expands as a tree structure creating child node.

Advantage :-

- 1) Expansion of nodes is easy.
- 2) It is easily managed and maintained and error detection is done easily.

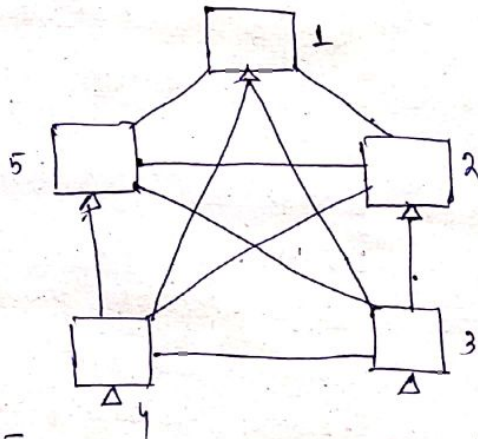
Disadvantage :-

- 1) If the root node fails, the entire network goes down.
- 2) It needs more cables.
- 3) It is expensive.



Mesh Topology :-

- In a mesh topology each computer system directly links with every other computer system.
- Generally mesh topology overcomes the disadvantage of star topology.



Total no. of connection C

$$C = \frac{n(n-1)}{2}$$

where n is the number of nodes.

Advantage :-

- 1) On a LAN, all computer system are connected with each other without using central hub.
- 2) Communication speed is faster.
- 3) Easy to remove terminals.
- 4) Security is maintained.
- 5) Damage of a node will not disturb the total network.

Disadvantage :-

- 1) ~~No~~ No. of cables required is more.
- 2) Complex structure

Types of Network :-

Networks are distinguished based on their size (in terms of the number of machines), their data transfer speed, and their reach.

→ There are three types of network.

- 1) LAN (Local Area Network)
- 2) MAN (Metropolitan Area Network)
- 3) WAN (Wide Area Network)

LAN :-

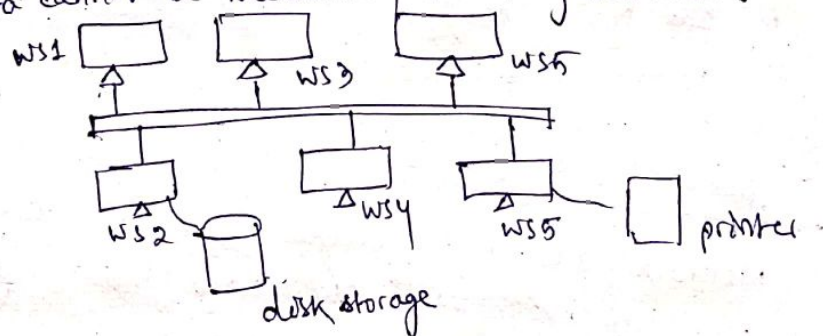
- LAN stands for Local Area Network.
- It is a group of computers which all belong to the same organization, within a small geographic area, ~~using~~ and often ~~the~~ using the same technology.
- This is the simplest type of network.
- The data transfer speed over a LAN can reach up to 10 Mbps (for an Ethernet network) and 1 Gbps (for FDDI or Gigabit Ethernet).
- A LAN can reach as many as 100 or even 1000 users.
- The range of LAN is ^{upto} 2.5 km.
- LAN provides two different operating modes.
 - (i) peer-to-peer - network in which communication is carried out from one computer to another without a central computer.
 - (ii) client-server network in which a central computer provides network services to users.

Advantage :-

- 1) High speed data exchange.
- 2) Resource sharing.

Disadvantage :-

- 1) server failure develops the failure of the network.
- 2) Data cannot be transmitted over long distances.



MAN :-

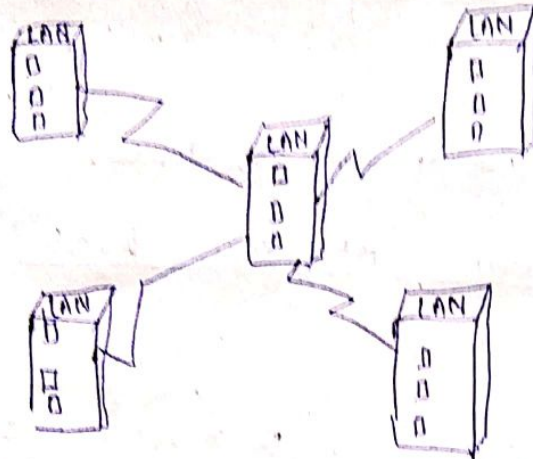
- MAN stands for Metropolitan Area Network.
- It is a type of communication network which covers more distance than LAN. i.e. up to 50 kms.
- A MAN connects multiple geographically nearby LANs to one another at high speeds.
- It generally covers a city.
- It might be owned and operated by a single organization.

Advantage:-

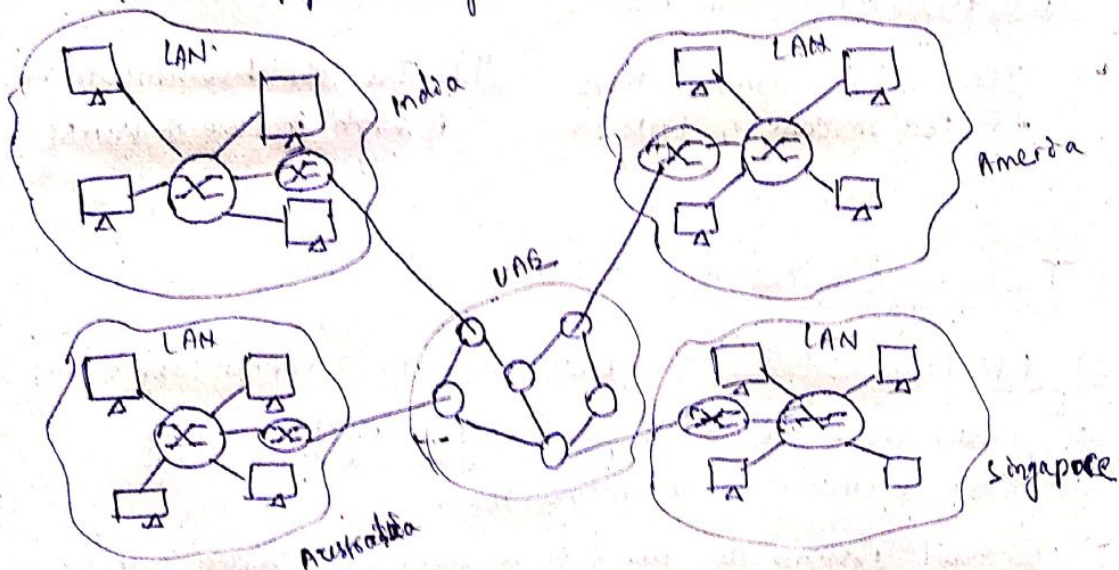
- 1) Resources can be shared between the branch offices in a city.
- 2) It uses high speed transmission medium, like fibre optics.

Disadvantage:-

- 1) connection of more computers within a city makes the system slow.

WAN:-

- > WAN stands for Wide Area Network.
- > It is a type of network used to connect large geographical areas for transmission of voice data, images and video data.
- > WAN is used to connect various countries, continent or the whole world.
- > The most popular example of WAN is Internet.

Advantage

- 1) It covers long distances than LAN & MAN.
- 2) It may transmit data over wireless link such as satellite.
- 3) Rate of data transmission is high.

- 4) Data is centralized.
- 5) A lots of applications such as whatsapp, facebook, messenger are used to exchange messages.
- 6) Distribute workload and decreases travel charges.

Disadvantage

- 1) Security problem as many technologies combined with each other.
- 2) Setup cost is high.
- 3) Server down and disconnection issue.

⇒ Difference between LAN and WAN

LAN

- 1) A LAN is limited to a small geographic location.
- 2) The rate of data transmission is generally higher.
- 3) A LAN is established by wired media. But now a days wireless LAN is also implemented in some cases.
- 4) The cost of communication is less in case of LAN.

WAN

- 1) A WAN is spread over a very large geographic location.
- 2) The rate of data transmission is lower.
- 3) A WAN is always established by using a wireless media.
- 4) The cost of communication is high in case of WAN.

Internet Services :-

- (1) Electronic Mail :- Electronic mail, often abbreviated as email, is a method of exchanging digital messages, designed primarily for human use.
- ⇒ E-mail systems are based on a store-and-forward model in which e-mail server accept, forward, deliver and store messages for users.

- An Electronic mail message consists of two components. (7)
- (i) Message header
 - (ii) Message body.
- The header is separated from the body by a blank line.

Advantages of E-mail

1. Cost Effective :- The message sent through e-mail costs very low. It is very cheaper than courier or fax or telegram.
2. High speed :- E-mail can be sent and received very fast and almost instantaneously.
3. Easy to Use :- It is very easy to use E-mail tool on the internet. The user just have to type the message and send it to the receiver by a mouse click. The user need not go to the post office, buy the envelope and write the message on paper.
4. Time saving :- User can send multiple copies of the message to the same recipient or to various other recipient at the click of a mouse within a small interval of time.
5. Wastage reduction :- A lot of paper work is saved which also reduces file maintenance.
6. Record Maintenance :- As all the messages are stored ~~and~~ in the form of files in computer, records are maintained for sent and received e-mail messages.
7. Message Storing :- If for any reason, the recipient is away, the e-mail message waits him until he received and reads the message.

2) File Transfer Protocol (FTP)

- FTP is the internet file transfer protocol/program.
- It is used to transfer files over the internet.
- Using FTP user can get files to his personal computer connected to internet from other internet machine.

- Similarly user can send or upload files at some sites, by giving email-address and password.
- To use FTP directly, the user site must be ~~wired~~ connected with Internet.
- User can also use FTP by email via servers, but it is slower.

3) World Wide Web (WWW) :-

- WWW is an Internet standard for distributed hypertext. This means that WWW documents can have links to other documents which can be any where on the Internet.
- WWW was invented by Tim-Berners-Lee in 1989.
- WWW is commonly known as Web, is an information system where documents and other web resources are identified by uniform resource locators (URL), which may be interlinked by hypertext, and are accessible over the Internet.
- WWW uses the client-server model and HTTP protocol for interaction between computers ~~and~~ on Internet. The computer which uses HTTP protocol, called web server and the computer which accesses the server is called web client.

4) Chatting :-

- Chatting is a 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 other participants to respond quickly.
- Chatting is similar to a spoken conversation between the users in terms of sending and receiving text messages.
- Online chat may address point-to-point communications as well as multicast communications from one sender to many receivers and voice and video chat or web conferencing service.

5) Internet (web) Conferencing

Web conferencing allows users to carry on business meetings and seminars, make presentations, conduct demonstrations, provide online education and offer direct customer support.

- It is a live, virtual connection between two or more people residing in separate locations for the purpose of communication.
- It provides transmission of full-motion video images and high quality audio between multiple locations.

6) Electronic News Paper :-

An electronic news paper is a self-contained, reusable, and refreshable version of a traditional newspaper that acquires and holds informations electronically.

- The information to be displayed will be downloaded through a wireless internet connection.
- Xerox's Palo Alto Research Center (PARC) is working on a newspaper that would consists of a single sheet of their e-paper (called Gyrodex) while Lucent, in partnership with a company call E-Ink is working on a multipage device.

7) Online Shopping :-

Online shopping involves purchasing products or services over the internet.

- Online shopping is done through an on-line shop, e-shop, e-store, virtual store, webshop, internet shop or online store.
- All the products in online store are described through text, with photos and with multimedia files. Many online shops provide links for extra information about their products.
- They often make available, safety procedures, instructions manufacture specifications and demonstrations. Some provide advice or how-to guide.
- A user with internet connection on his computer or mobile can search products and purchase products and can rate the products.

FILE MANAGEMENT & DATA PROCESSING

Chapter-4

①

- In DOS data are organized into files and directories.
- In GUI based operating system like windows, data are organized into files and folders during storage in computer memory.
 - The raw data in binary format is either byte or kilobyte (KB) or Megabyte (MB) or Gigabyte (GB).
 - A byte is the smallest unit of information. It is used to measure the size of the documents.

Memory Units

1 byte

1 Kilobyte (KB)

1 Megabyte (MB)

1 Gigabyte (GB)

1 Terabyte (TB)

Size

8 bits

1024 bytes = 2^{10} bytes

1024 KB = 1024×1024 bytes
= $2^{10} \times 2^{10} = 2^{20}$ bytes

1024 MB = $1024 \times 1024 \times 1024$ bytes
= $2^{10} \times 2^{10} \times 2^{10} = 2^{30}$ bytes

1024 GB = $1024 \times 1024 \times 1024 \times 1024$ bytes
= $2^{10} \times 2^{10} \times 2^{10} \times 2^{10} = 2^{40}$ bytes

Files :-

- Files are the most basic unit of data, used to store programs, image, video, song and documents on disk.
- User can create, save, open, move, delete files. It is also possible to move files from one folder to another.
 - User can also download files from network or Internet.
 - Type of a file is known by its icon or by its extension.

Types of Files :-

There are different types of files depending on the type of information they contain. There are image files, program files, text files, music files etc. To distinguish the type of information that files contain, they are assigned extensions.

File Extension :- Files are identified by 'extension' at the end of their name.

For example (i) ABC.JPG is a JPEG image file.

(ii) XYZ.doc is a Microsoft Word document file.

(iii) PQR.EXE is an executable application in Windows.

Names of Files :-

Name of a file can be upto 255 characters. It can contain letters, numbers, blank spaces and special characters (like dashes, underlines etc.).

→ A filename should not contain " , ? , \ , > , < & ! symbols.

Folder :-

→ Folder is a collection of one or more files, or it can be empty with just a name.

→ Folders provide a method for organizing files.

→ Folders can also store other folders called subfolders.

→ In earlier operating system before Windows, folders were called as directories.

→ Folders are given a name just like files.

Difference between File and Folder

<u>File</u>	<u>Folder</u>
1) Files store data, like text, music, video etc.	1) Folder stores files and other folders.
2) Files take space in hard drive ranging from bytes to kilobytes or to gigabytes etc.	2) Folders usually take no space in hard drive.
3) Files are smaller in size.	3) Folders are bigger in size as they hold many files.

File Access Methods:-

(2)

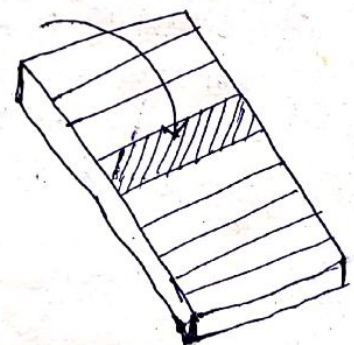
An access method defines the technique that is used to store and retrieve data. It is a function of a mainframe operating system that enables access to data on disk, tape or other external devices.

- Access methods have their own dataset structure to organize data, system-provided programs to define data sets, and utility programs to process datasets.
- In computing access method is a program or a hardware mechanism that moves data between the computer and hard disk or a display terminal.
- There are two types of access methods.
 - (i) Random access method
 - (ii) sequential access method.

Random Access Method (Direct access) :-

Key Features

- A file is made up of fixed-length logical records that allow programs to read and write records in no particular order. User may first read block 14, then read block 53, and then write block 7. There are no order of reading or writing for a direct-access file.
- Direct-access method is based on a disk model, since disk allow random access to any file block.
- Direct-access files are ~~great~~ used for immediate access to large amounts of information. Databases are of this type.
- For the direct access method, the file operations, must be modified to include the block number as a parameter.



(Random Access Method)

→ The block number provided by the user to the OS is normally a relative block number, which is an index relative to the beginning of the file. Thus the first relative block of the file is 0, the next 1 and so on, even though the actual absolute address of the block may be different.

→ The use of relative block numbers allows the OS to decide where the file should be placed and helps the user from accessing portions of the file system that may not be part of his file.

Sequential Access:

Key Features:-

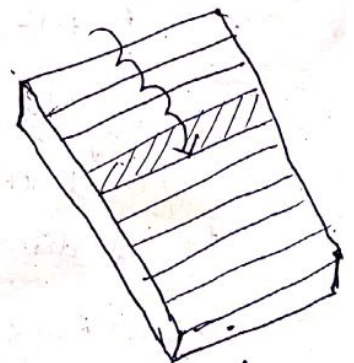
→ This is the simplest method, where information in the file is processed in order one ~~at~~ record after the other.

→ In this method, accessing of files starts from the beginning position of the files.

→ Reads and writes make up the bulk of the operations on a file.

→ In sequential access method, a read operation reads the next position of the file and automatically advances a file pointer, which tracks the I/O operation.

→ Similarly, the write operation writes next appends, to the end of the file and advances to the end of the newly written end of file.



(Sequential Access)

Indexed Sequential Access Method (ISAM)

- An indexed sequential access method is static, hierarchical, disk-based index structure that enables both range and membership queries on an ordered data file.
- ISAM initially stores records sequentially and permits both sequential and random processing.
- In this method indexes are used to locate a correct cylinder and track, and keys ^{are used} to locate a record on a track.
- Since ISAM is static, it does not change its structure if records are added or deleted from the file.
- Each index defines a different ordering of the records.
for example, In an employee database, a name index may order employees alphabetically by last name.
→ similarly, a department index may order employees by their department.
- A key is specified in each index. This key must be unique. That unique key is used to search the record in an ~~file~~ indexed file.

Data Capture :-

- Data capture is the process of identification & an extraction of data from a scanned document, often to be sent to a workflow for routing and action as part of a business process.
- There are various methods used for capturing data from unstructured documents (letters, invoices, email, fax forms).

Methods of capturing data from documents in electronic format

- 1) Single click :- It is an optical character recognition (OCR) tool that can be used to capture machine produced characters.
- 2) OCR (Optical Character Recognition) :- It is a technology that provides the ability to successfully capture machine produced characters in full page.

- OCR system can recognize ~~ma~~ different types OCR fonts, as well as typewriter and computer printed characters.
- Depending upon the capabilities of OCR product, it can be used to capture low to high volumes of data, where the information is in consistent location on the document.

ICR (Intelligent Character Recognition) :-

- ICR is the computer translation of hand printed and written characters.
- Data is entered from hand printed form through a scanner, and the image of the captured data is analyzed and translated by ICR software.
- It is more difficult than OCR process.

Bar Code Recognition :-

- Bar code recognition is a technique used to read bar codes on different types of documents (items such as proof of delivery notes, membership forms, application forms, gift aids, etc).

Intelligent Document Recognition (IDR)

- These applications are used to capture metadata from documents that are rules based.
- For example, the product will identify post codes, logos, key words, VAT registration numbers, through on going learning process, capture information from multiple document types.

Data Storage :-

→ Data storage is the process of holding data in an electro-magnetic form for access by computer processor.

→ There are two types of storage.

(i) Primary storage:- Where data is held in Random Access memory (RAM) and other memory devices that are built into computers.

(ii) secondary storage:- Where data is stored in external storage devices such as hard disks, tapes, CD, etc.

Hard disks:- A hard disk drive is used to store large amount of digital information in computer, and provides quick access. Hard disk drives are used to store operating systems, software and working data. These are suitable for any application which requires very fast access to data for both reading and writing.
→ Hard disk drives may not be suitable for applications which need portability. These are used in file servers for computer network to store large amount of data.

Floppy disks:- A floppy disk consists of a flexible disk with a magnetic coating, with a capacity of 1.44 MB. These are portable and are popular for transferring software from one PC to another.
→ Floppy disks are ~~portable~~ and very slow and lack of storage capacity.

Tape storage:- Tape is used as an external storage medium. It consists of a loop of flexible celluloid like material that can store data in the form of electromagnetic charges. A tape drive is the device that positions, writes from and reads to the tape. It is portable.

Optical disk:- Optical disk is a storage medium that can be read and written to using a low-powered laser beam. A laser reads the dots and the data is converted to an electrical signal, finally converted into original data.

CD-R (Compact Disc-Recordable) - CD-Rs are used for music recording and for file storage or transfer between PCs. These are write-once media. This means once used - they can't be erased or re-recorded upon. These can be played back in any audio CD-player. DVD are used to store very large files like movies.

Pen drive:- It is a removable and rewritable USB flash drive with capacity 64 MB to 64 GB. These are portable, operate faster, hold more data.

Memory Card :- It is a solid state electronic flash memory, device used with digital cameras, mobile, computers, telephones, music players, video game consoles etc.

Data Processing and Retrieval :-

Data Processing :-

→ Data processing is defined as a sequence of operations on data to convert it into useful information.

→ The important operations that can be performed on data are:

- Arithmetic and logical operations

- Transfer of data from one computer to another.

- Classification of data

- Arranging of data into a specific order.

→ Methods of data processing

(1) Manual Data processing

→ In manual data processing data is processed without using any machine or tool to get the results.

→ In this method all the calculations and logical operations are performed manually on data. Similarly, data is transferred manually from one place to another.

→ This method is very slow and errors may occur in the output.

Ex:- Data is processed manually in small business firms, government offices & institutions, in educational institutes mark sheets, fee receipts, financial calculations are done performed by hand.

(2) Mechanical data processing :-

→ In this method data is processed by using different devices like typewriters, mechanical printers or other mechanical devices.

→ This method of data processing is faster and more accurate than manual data processing.

Ex:- Examination boards and printing press use mechanical devices.

(3) Electronic data Processing :-

- This is the modern technique of data processing.
- In this method data is processed using computers. Data and instructions are given to the computer as input and the computer automatically processes the data according to the given set of instructions.
- This method of processing data is very fast and accurate.
- Ex:- In a computerized education environment results of students are prepared through computer, in banks accounts of customers are maintained using computers.

Data Retrieval :-

Data retrieval or data recovery is the process of restoring data that has been lost, accidentally deleted, corrupted or made inaccessible for any reason.

- Files can be lost because of the following reasons.
 - (i) File was mistakenly deleted.
 - (ii) File was corrupt and deleted by scandisk.
 - (iii) Another program delete the file.
 - (iv) File is password protected.

Methods of data recovery :-

① Physical damage to storage devices :-

- physical damage could range from breaking of tapes to scratching of data layer of CD-ROMs or hard disks suffering mechanical failures like failed motors and head crash.
- When physical damage occurs, some form of data is lost.
- Data recovery from physical damage to storage device may involve replacing parts in hard disk or recovering every readable bit from the surface through a specialized disk imaging procedure.

(2) Media errors and corrupt partitions and file systems.

- Media error or damage to the file system or partition table can make the data on a hard drive to be unreadable.
- Data recovery software like Test Disk can be used to repair damaged file system or partition table.
- In this case all or some of the original data may be recovered.
- This method requires end-users ~~have~~ to be knowledgeable.

(3) Online data recovery:-

- This is a method of data recovery that is performed over the internet.
- When using this method data is regularly backed up online and therefore if any data is lost ~~to~~ user can ~~can~~ recover the ~~data~~ last backed up data.
- This method of data recovery is convenient but not common, because it required a broadband internet connection.

— o —

Important questions :

- 1) What are the differences between file and folder.
2. Define OCR
3. What do you mean by ISAM
4. What is the difference between random access method and sequential access method.
5. What do you mean by data validation?
6. How to recover missing, lost or deleted files?

Long Questions

1. What is file? Explain the various file access methods.
2. Explain briefly about different methods of data capture.
3. Define data processing and explain various methods of data processing.
4. What is data recovery? Explain briefly different methods of data recovery.

Problem Solving Methodology Chapter-5 ①

- A computer is an electronic device and it has zero IQ, it can't solve any problem on its own. So we have to write a program in a programming language and give it as input to the computer. The computer then executes that program and perform the task as mentioned in the program, and the problem get solved.
- While writing a program a programmer has to go through various steps. Those are as follows.
- (1) Get the problem and understand it.
 - (2) Frame a logic for solving the problem and represents it in the form of flowchart or algorithm.
 - (3) Choose a particular programming language to encode the algorithm and convert it into a program.
 - (4) Load the program in the computer, compile it and execute.
 - (5) After the program is executed, get the result of the problem.
- The success of solution to any problem depends on the logic of the problem solving method, which is documented in the form of algorithm or flowchart.

Algorithm :-

An algorithm is defined as a step by step method for writing the solution to a problem.

Characteristics of algorithm :

- (1) Algorithm should be definite
- (2) Algorithm should have finite number of steps.
- (3) Algorithm should mention the input required for the program clearly.
- (4) Algorithm should give an idea the output that will be obtained.

For example : Write an algorithm to print all 2-digit odd numbers.

Ans :- Algorithm :-

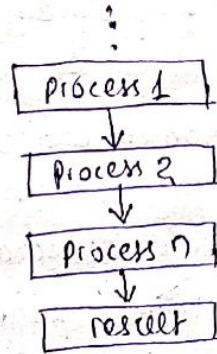
1. Initialize a variable NUM with 11
2. Print this variable NUM.
3. Add 2 to the variable NUM
4. Go on repeating step 2 and 3 UNTIL NUM becomes more than 99.

Algorithm types :-

In an algorithm, we generally write the logic of the solution to the problem. There are three types of logic.

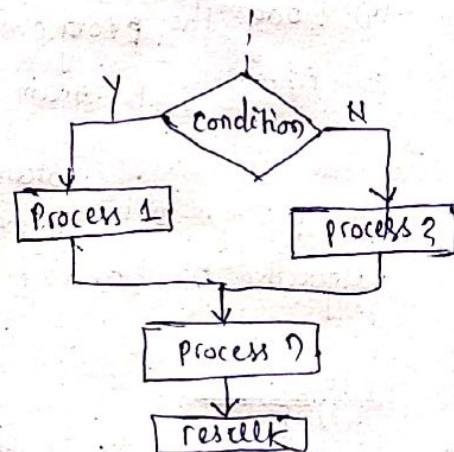
(1) sequence logic :- In a sequence logic steps are executed in a linear sequence, one after another from top to bottom.

(2) selection logic :-



(2) selection logic

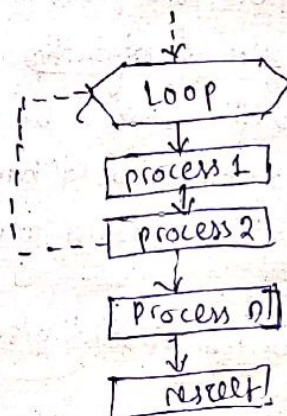
In selection logic, the steps of the solution logic are not linear. It follows different paths basing on the result of certain condition.



(3) Iteration logic :-

In iteration logic, a number of steps are repeated certain number of times depending on the iteration counter.

This is also known as Looping logic.



Pseudocode :-

Pseudocode is a method of writing the logic of a problem solution in a stepwise manner in English that follow certain programming construction.

→ It is a set of codes which may not be written by following the correct ~~syse~~ syntax of the code.

→ It is used as a program planning tool.

→ Same as algorithm, pseudocode can be written in 3 ways.

(i) Sequence logic

(ii) selection logic

(iii) Iteration logic

→ The main difference between algorithm and pseudocode is that,

In algorithm we may not use any language that resembles a programming language, but in pseudocode we use a structure which resembles a programming language.

→ Pseudocode is also known as program design language (PDL), which emphasises on the design of a program logic.

→ The previous example can be written using pseudocode as follows.

1. Begin

2. Set NUM = 1

3. Print NUM

4. Set NUM = NUM + 2


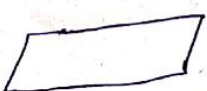


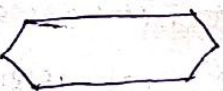

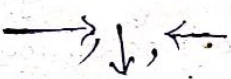
5. If NUM ≤ 99 then repeat step 3 & 4

6. END

Flowchart :-

A flowchart is defined as a program planning tool, where, the logic of the problem is represented in a pictorial manner by using a set of predefined symbols.

Basic symbols for Flowchart

<u>Symbol</u>	<u>Name</u>	<u>Use</u>
	Oval/Ellipse	Contain START or STOP
	Parallelogram	Contain input statements, and also output statements
	Rectangle	Contain any calculation or processing statement
	Rhombus	Contain condition for selection logic
	Hexagon	Contain number of iterations in the for of a starting and ending condition.
	Continuation symbol	and Used to continue the flow of flowchart
	Arrow	Used to show the direction of the flow of logic and connect the different symbols of the flowchart.

Advantage 1 -

- 1) Program logic represented in a graphical manner is easy to create.
- 2) The logic in a flowchart is easy to translate.
- 3) It can be used by non computer professionals.
- 4) It is easy to modify the logic of a problem solution.
- 5) Long and complicated problem solutions can be represented by small and simple flowcharts.

Disadvantage

- 1) The problem solution represented in a flowchart is difficult to convert into a program.
- 2) If the solution contains specific type of structure it is difficult to represent it in flowchart.

Generations of Programming Language :-

There are five generations of programming languages.

① First generation Programming language :- (1GL)

The first generation of programming language is machine language. Machine language is a set of instructions and data, that CPU can execute directly. It consists of binary codes.

② Second generation of programming language (2GL)

→ The second generation of programming language is assembly language. Assembly language is the human-readable notation and consists of symbolic instruction codes that are meaningful abbreviations or mnemonics. For this language an assembler is required to translate it into machine language.

③ Third generation of programming language (3GL)

The third generation of programming language is the procedural language, consists of a series of English-like words that a user can understand easily. These are the high level languages, that make complex programming simpler, and easier to read and write. This language needs a compiler or interpreter to be translated into machine language. Examples of this language are - PASCAL, COBOL, FORTRAN, BASIC, C, C++ etc.

④ Fourth generation of programming language :- (4GL)

→ The fourth generation programming language is the non-procedural language and enables users to access data in a database.
→ These are very high level programming languages and are goal oriented, limited to a very specific applications.
Ex - SQL, NOMAD and FOCUS

⑤ Fifth generation of programming language :- (5GL)

The fifth generations of programming language is visual programming language, also called as natural language.

→ This language provides a visual or graphical interface, called visual programming environment, for creating source codes. It allows people to interact with computers. People can talk to computers and the voice recognition system can convert spoken sounds into written words.
Ex - Prolog, Mercury.

Structured Programming Language

- A structured programming language, also called as modular programming, is a subset of procedural programming that enforces a logical structure of the program to make it more efficient and easier to understand and modify. Ex: - Ada, dBASE etc.
- Structured programming uses a top-down model, in which developers can divide the overall program into subsections or modules or submodules and can be coded separately. Each individual module or submodule can be loaded into memory more efficiently and can be used in other programs. After a module has been tested individually it is then integrated with other modules into the overall program structure.
- structured programming was first suggested by Corrado Bohm, and Giuseppe Jacopini.

Some Examples of Problem solving through Algorithm

Q.1. Write an algorithm to check whether a number is even or odd

Ans:- step 1: start

step 2: Read the value of x

step 3: If $(x \% 2 == 0)$ then

step 4: print number is even

step 5: else

step 6: print number is odd

step 7: stop

Q.2. Write an algorithm to find out the simple interest.

step 1: Start

step 2: Read P, R and T

step 3: Simple Interest = $(P * R * T) / 100$

step 4: print simple interest

step 5: stop

Q. Write an algorithm for the factorial of an integer.

Solⁿ

step 1: start

step 2: Read a variable n .

step 3: Set variable $fact = 1$

step 4: $fact = fact * n$

decrease n

step 5: check if n is equal to 0

if n is equal to 0 goto step 6

else goto step 4.

step 6: print $fact$.

step 7: stop

Q. Write an algorithm and draw a flowchart to find out the greatest among three numbers.

Solⁿ

step 1: start

step 2: Read three numbers
 a, b, c

step 3: if $(a > b)$

{

step 4: if $(a > c)$

step 5: print "a is greater"

step 6: else

print "c is greater"

step 7: }

step 8: if $(b > c)$

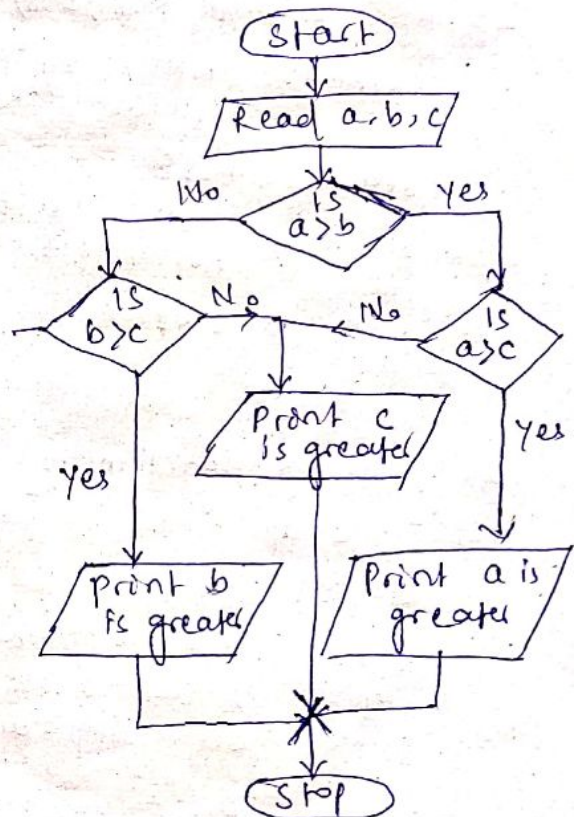
step 9: { print "b is greater"

step 10: else

print "c is greater"

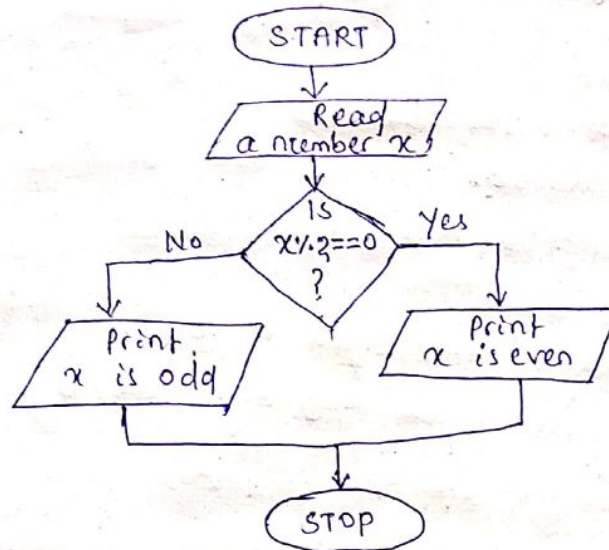
step 11: stop

Flowchart



Q. Draw a flowchart to ~~find~~ check whether a number is even or odd. (4)

Ans

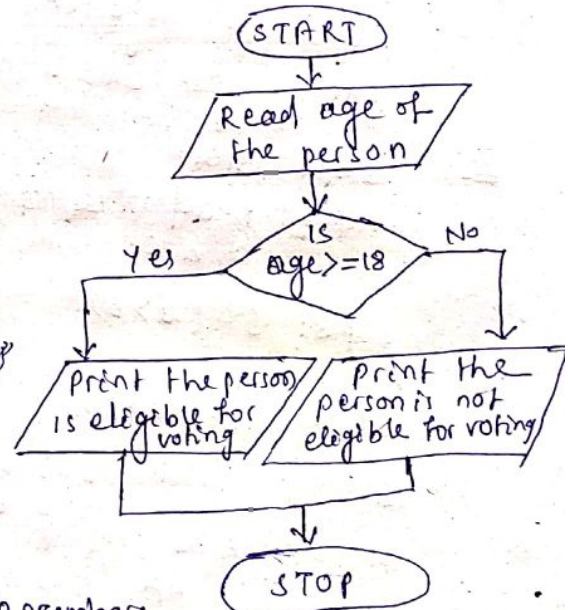


Q. Draw a flowchart. to check if a person is eligible for voting. Also write the algorithm for it.

Algorithm

- step 1: START
- step 2: READ AGE OF THE PERSON
- step 3: IF (AGE ≥ 18) THEN
- step 4: PRINT "THE PERSON IS ELIGIBLE FOR VOTING"
- step 5: ELSE
- step 6: PRINT "THE PERSON IS NOT ELIGIBLE FOR VOTING"
- step 7: STOP

flowchart

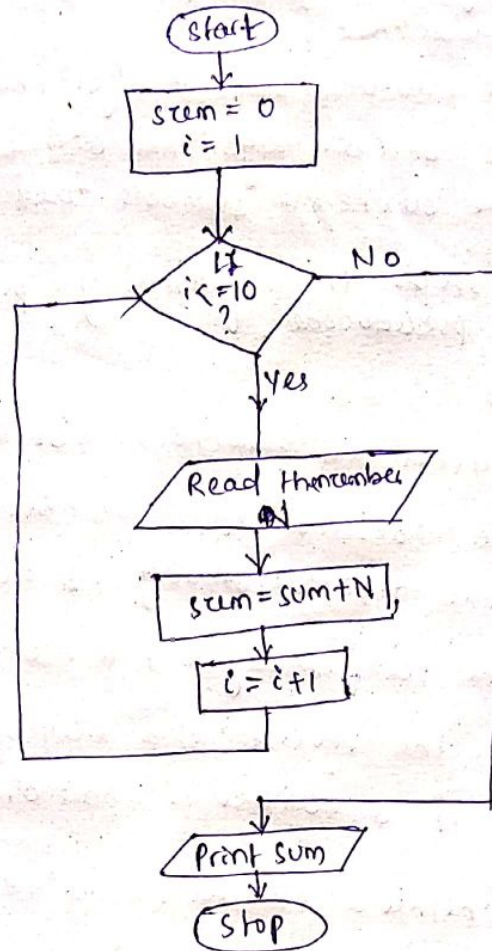


Q. Write an algorithm to swap two numbers.
Ans! - Swap mean exchange.

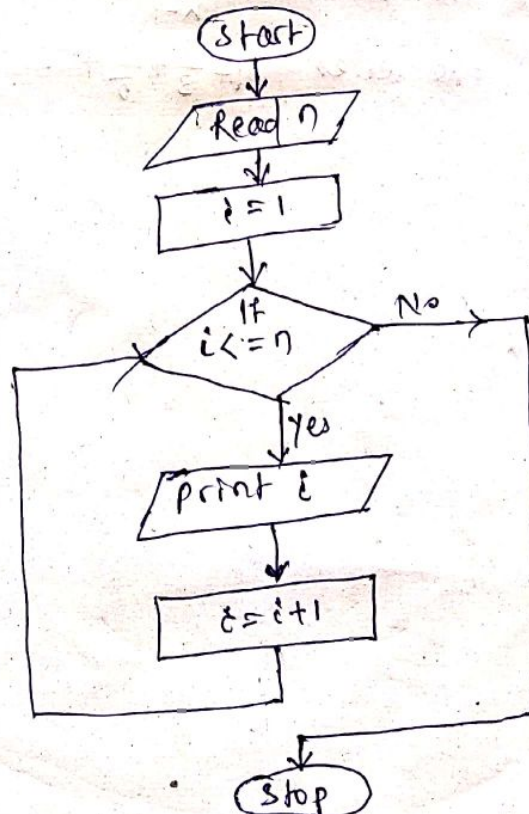
- step 1. START
- step 2. Read two numbers a and b.
- step 3. Declare a third variable temp
- step 4. set, temp = a, a = b, b = temp
- step 5. print values of a and b
- step 6. STOP

Q) Draw a flowchart to find sum of 10 random numbers. 5

sol



Q) Draw a flowchart to print first n natural numbers.



Important Questions

1. Define an algorithm.
2. What is a flowchart?
3. What are the characteristics of algorithm?
4. Differentiate between algorithm and flowchart.
5. What are the different symbols used for flowchart?
6. Write the steps in programming process?
7. Define a pseudocode.

Long questions

1. Write an algorithm and flowchart for the following:

- (i) To calculate factorial of a number.
- (ii) To find the greatest among 3 numbers.
- (iii) To find the sum of digits of a number.
- (iv) To find the reverse of a number.
- (v) To check if a number is palindrome or not.
- (vi) To find whether a number is prime or not.
- (vii) To solve a quadratic equation $ax^2 + bx + c = 0$.
- (viii) To print the sum of $\sum_{k=1}^n x^k$.
- (ix) To print the series $1 + \frac{1}{3} + \frac{1}{5} + \dots + \frac{1}{19}$.

CHAPTER-6

C is a general-purpose, high-level language that was originally developed by Dennis M. Ritchie to develop the UNIX operating system at Bell Labs. C was originally first implemented on the DEC PDP-11 computer in 1972.

In 1978, Brian Kernighan and Dennis Ritchie produced the first publicly available description of C, now known as the K&R standard. C has now become a widely used professional language for various reasons –

- Easy to learn
- Structured language
- It produces efficient programs
- It can handle low-level activities
- It can be compiled on a variety of computer platforms

Characteristics of C

- C was invented to write an operating system called UNIX.
- C is a successor of B language which was introduced around the early 1970s.
- The language was formalized in 1988 by the American National Standard Institute (ANSI).
- The UNIX OS was totally written in C.
- Today C is the most widely used and popular System Programming Language.
- Most of the state-of-the-art software have been implemented using C.
- Today's most popular Linux OS and RDBMS MySQL have been written in C.

Why use C?

C was initially used for system development work, particularly the programs that make-up the operating system. C was adopted as a system development language because it produces code that runs nearly as fast as the code written in assembly language. Some examples of the use of C might be –

- Operating Systems
- Language Compilers
- Assemblers
- Text Editors
- Print Spoolers
- Network Drivers
- Modern Programs
- Databases
- Language Interpreters
- Utilities

C Programs

A C program can vary from 3 lines to millions of lines and it should be written into one or more text files with extension ".c"; for example, *hello.c*. You can use "**vi**", "**vim**" or any other text editor to write your C program into a file. This tutorial assumes that you know how to edit a text file and how to write source code inside a program file.

A C program basically consists of the following parts –

- Preprocessor Commands
- Functions
- Variables
- Statements & Expressions
- Comments

Let us look at a simple code that would print the words "Hello World" –

```
#include <stdio.h>

void main() {
    /* my first program in C */
    printf("Hello, World! \n");

}
```

- The first line of the program `#include <stdio.h>` is a preprocessor command, which tells a C compiler to include `stdio.h` file before going to actual compilation.
- The next line `void main()` is the main function where the program execution begins.
- The next line `/*...*/` will be ignored by the compiler and it has been put to add additional comments in the program. So such lines are called comments in the program.
- The next line `printf(...)` is another function available in C which causes the message "Hello, World!" to be displayed on the screen.

Compile and Execute C Program

Following are the simple steps –to save the source code in a file, and to compile and run it.

- Open C editor and write the C- program code.
- Save the file as *hello.c*
- Open a command prompt and go to the directory where you have saved the file.
- Press `alt+f9` and press enter to compile your code.
- If there are no errors in your code, the command prompt will take you to the next line and would generate executable file.
- Now, press `ctrl+f9` to execute your program.
- You will see the output *"Hello World"* printed on the screen.

Tokens in C

A C program consists of various tokens and a token is either a keyword, an identifier, a constant, a string literal, or a symbol. For example, the following C statement consists of five tokens –

```
Printf("Hello, World! \n");
```

The individual tokens are –

```
Printf
```

```
(
```

```
"Hello, World! \n"
```

```
)
```

```
;
```

Semicolons

In a C program, the semicolon is a statement terminator. That is, each individual statement must be ended with a semicolon. It indicates the end of one logical entity.

Comments

Comments are like helping text in your C program and they are ignored by the compiler. They start with `/*` and terminate with the characters `*/` as shown below –

```
/* my first C program*/
```

Identifiers

A C identifier is a name used to identify a variable, function, or any other user-defined item. An identifier starts with a letter A to Z, a to z, or an underscore `'_'` followed by zero or more letters, underscores, and digits (0 to 9).

C does not allow punctuation characters such as `@`, `$`, and `%` within identifiers. C is a **case-sensitive** programming language. Thus, *Manpower* and *manpower* are two different identifiers in C.

Keywords

The following list shows the reserved words in C. These reserved words may not be used as constants or variables or any other identifier names.

auto	else	long	switch
break	enum	register	typedef
case	extern	return	union
char	float	short	unsigned
const	for	signed	void
continue	goto	sizeof	volatile
default	if	static	while
do	int	struct	_Packed
double			

Whitespace in C

A line containing only whitespace, possibly with a comment, is known as a blank line, and a C compiler totally ignores it.

Whitespace is the term used in C to describe blanks, tabs, newline characters and comments. Whitespace separates one part of a statement from another and enables the compiler to identify where one element in a statement, such as `int`, ends and the next element begins.

Ex: `int mark;`

In the above line there is a white space between **int** and **mark**.

Data types in C

Data types in C refer to an extensive system used for declaring variables or functions of different types. The type of a variable determines how much space it occupies in storage and how the bit pattern stored is interpreted.

The types in C can be classified as follows –

Sr.No.	Types & Description
1	Basic Types They are arithmetic types and are further classified into: (a) integer types and (b) floating-point types.
2	Enumerated types They are again arithmetic types and they are used to define variables that can only assign certain discrete integer values throughout the program.
3	The type void The type specifier <i>void</i> indicates that no value is available.
4	Derived types They include (a) Pointer types, (b) Array types, (c) Structure types, (d) Union types and (e) Function types.

The array types and structure types are referred collectively as the aggregate types. The type of a function specifies the type of the function's return value. We will see the basic types in the following section, where as other types will be covered in the upcoming chapters.

Integer Types

The following table provides the details of standard integer types with their storage sizes and value ranges –

Type	Storage size	Value range
char	1 byte	-128 to 127 or 0 to 255
unsigned char	1 byte	0 to 255
signed char	1 byte	-128 to 127
int	2 or 4 bytes	-32,768 to 32,767 or -2,147,483,648 to 2,147,483,647
unsigned int	2 or 4 bytes	0 to 65,535 or 0 to 4,294,967,295
short	2 bytes	-32,768 to 32,767
unsigned short	2 bytes	0 to 65,535
long	8 bytes	-9223372036854775808 to 9223372036854775807
unsigned long	8 bytes	0 to 18446744073709551615

Floating-Point Types

The following table provide the details of standard floating-point types with storage sizes and value ranges and their precision –

Type	Storage size	Value range	Precision
float	4 byte	1.2E-38 to 3.4E+38	6 decimal places
double	8 byte	2.3E-308 to 1.7E+308	15 decimal places
long double	10 byte	3.4E-4932 to 1.1E+4932	19 decimal places

The void Type

The void type specifies that no value is available. It is used in three kinds of situations –

Function returns as void

There are various functions in C which do not return any value or you can say they return void. A function with no return value has the return type as void. For example, **void exit (int status);**

Function arguments as void

There are various functions in C which do not accept any parameter. A function with no parameter can accept a void. For example, **int rand(void);**

Pointers to void

A pointer of type void * represents the address of an object, but not its type. For example, a memory allocation function **void *malloc(size_t size);** returns a pointer to void which can be casted to any data type.

Variable

A variable is nothing but a name given to a storage area that our programs can manipulate. Each variable in C has a specific type, which determines the size and layout of the variable's memory; the range of values that can be stored within that memory; and the set of operations that can be applied to the variable.

The name of a variable can be composed of letters, digits, and the underscore character. It must begin with either a letter or an underscore. Upper and lowercase letters are distinct because C is case-sensitive. Based on the basic types explained in the previous chapter, there will be the following basic variable types –

-
1. **Char**: Typically a single octet(one byte). It is an integer type.
 2. **Int**: The most natural size of integer for the machine.
 3. **Float**: A single-precision floating point value.
 4. **Double**: A double-precision floating point value.
 5. **Void**: Void represents the absence of type

Defining a variable in C

A variable definition tells the compiler where and how much storage to create for the variable. A variable definition specifies a data type and contains a list of one or more variables of that type as follows –

```
type variable_list;
```

Here, **type** must be a valid C data type including char, w_char, int, float, double, bool, or any user-defined object; and **variable_list** may consist of one or more identifier names separated by commas. Some valid declarations are shown here –

```
int    i, j, k;
char   c, ch;
float  f, salary;
double d;
```

The line **int i, j, k;** declares and defines the variables i, j, and k; which instruct the compiler to create variables named i, j and k of type int.

Variables can be initialized (assigned an initial value) in their declaration. The initializer consists of an equal sign followed by a constant expression as follows –

```
type variable_name = value;
```

Some examples are –

```
extern int d = 3, f = 5;    // declaration of d and f.
int d = 3, f = 5;          // definition and initializing d and f.
byte z = 22;               // definition and initializes z.
char x = 'x';              // the variable x has the value 'x'.
```

For definition without an initializer: variables with static storage duration are implicitly initialized with NULL (all bytes have the value 0); the initial value of all other variables are undefined.

Example

Try the following example, where variables have been declared at the top, but they have been defined and initialized inside the main function –

```
#include <stdio.h>

// Variable declaration:
extern int a, b;
extern int c;
extern float f;

int main () {

    /* variable definition: */
    int a, b;
    int c;
    float f;

    /* actual initialization */
    a = 10;
    b = 20;

    c = a + b;
    printf("value of c : %d \n", c);

    f = 70.0/3.0;
    printf("value of f : %f \n", f);
```



```
    return 0;
}
```

When the above code is compiled and executed, it produces the following result –

```
value of c : 30
value of f : 23.333334
```

Constants

Operators in C

An operator is a symbol that tells the compiler to perform specific mathematical or logical functions. C language is rich in built-in operators and provides the following types of operators –

- Arithmetic Operators
- Relational Operators
- Logical Operators
- Bitwise Operators
- Assignment Operators
- Misc Operators

We will, in this chapter, look into the way each operator works.

Arithmetic Operators

The following table shows all the arithmetic operators supported by the C language. Assume variable **A** holds 10 and variable **B** holds 20 then –

Show Examples

Operator	Description	Example
+	Adds two operands.	A + B = 30
-	Subtracts second operand from the first.	A - B = -10
*	Multiplies both operands.	A * B = 200
/	Divides numerator by de-numerator.	B / A = 2
%	Modulus Operator and remainder of after an integer division.	B % A = 0
++	Increment operator increases the integer value by one.	A++ = 11
--	Decrement operator decreases the integer value by one.	A-- = 9

Relational Operators

The following table shows all the relational operators supported by C. Assume variable **A** holds 10 and variable **B** holds 20 then –

Show Examples

Operator	Description	Example
==	Checks if the values of two operands are equal or not. If yes, then the condition becomes true.	(A == B) is not true.
!=	Checks if the values of two operands are equal or not. If the values are not equal, then the condition becomes true.	(A != B) is true.
>	Checks if the value of left operand is greater than the value of right operand. If yes, then the condition becomes true.	(A > B) is not true.
<	Checks if the value of left operand is less than the value of right operand. If yes, then the condition becomes true.	(A < B) is true.
>=	Checks if the value of left operand is greater than or equal to the value of right operand. If yes, then the condition becomes true.	(A >= B) is not true.
<=	Checks if the value of left operand is less than or equal to the value of right operand. If yes, then the condition becomes true.	(A <= B) is true.

Logical Operators

Following table shows all the logical operators supported by C language. Assume variable **A** holds 1 and variable **B** holds 0, then –

Show Examples

Operator	Description	Example
&&	Called Logical AND operator. If both the operands are non-zero, then the condition becomes true.	(A && B) is false.
	Called Logical OR Operator. If any of the two operands is non-zero, then the condition becomes true.	(A B) is true.
!	Called Logical NOT Operator. It is used to reverse the logical state of its operand. If a condition is true, then Logical NOT operator will make it false.	!(A && B) is true.

Bitwise Operators

Bitwise operator works on bits and perform bit-by-bit operation. The truth tables for &, |, and ^ is as follows –

p	q	p & q	p q	p ^ q
0	0	0	0	0

0	1	0	1	1
1	1	1	1	0
1	0	0	1	1

Assume A = 60 and B = 13 in binary format, they will be as follows –

A = 0011 1100

B = 0000 1101

A&B = 0000 1100

A|B = 0011 1101

A^B = 0011 0001

~A = 1100 0011

The following table lists the bitwise operators supported by C. Assume variable 'A' holds 60 and variable 'B' holds 13, then –

Show Examples

Operator	Description	Example
&	Binary AND Operator copies a bit to the result if it exists in both operands.	(A & B) = 12, i.e., 0000 1100
	Binary OR Operator copies a bit if it exists in either operand.	(A B) = 61, i.e., 0011 1101
^	Binary XOR Operator copies the bit if it is set in one operand but not both.	(A ^ B) = 49, i.e., 0011 0001
~	Binary One's Complement Operator is unary and has the effect of 'flipping' bits.	(~A) = ~(60), i.e., -0111101
<<	Binary Left Shift Operator. The left operands value is moved left by the number of bits specified by the right operand.	A << 2 = 240 i.e., 1111 0000
>>	Binary Right Shift Operator. The left operands value is moved right by the number of bits specified by the right operand.	A >> 2 = 15 i.e., 0000 1111

Assignment Operators

The following table lists the assignment operators supported by the C language –

Show Examples

Operator	Description	Example
=	Simple assignment operator. Assigns values from right side operands to left side operand	C = A + B will assign the value of A + B to C

+=	Add AND assignment operator. It adds the right operand to the left operand and assigns the result to the left operand.	C += A is equivalent to C = C + A
-=	Subtract AND assignment operator. It subtracts the right operand from the left operand and assigns the result to the left operand.	C -= A is equivalent to C = C - A
*=	Multiply AND assignment operator. It multiplies the right operand with the left operand and assigns the result to the left operand.	C *= A is equivalent to C = C * A
/=	Divide AND assignment operator. It divides the left operand with the right operand and assigns the result to the left operand.	C /= A is equivalent to C = C / A
%=	Modulus AND assignment operator. It takes modulus using two operands and assigns the result to the left operand.	C %= A is equivalent to C = C % A
<<=	Left shift AND assignment operator.	C <<= 2 is same as C = C << 2
>>=	Right shift AND assignment operator.	C >>= 2 is same as C = C >> 2
&=	Bitwise AND assignment operator.	C &= 2 is same as C = C & 2
^=	Bitwise exclusive OR and assignment operator.	C ^= 2 is same as C = C ^ 2
=	Bitwise inclusive OR and assignment operator.	C = 2 is same as C = C 2

Misc Operators → sizeof & ternary

Besides the operators discussed above, there are a few other important operators including **sizeof** and **? :** supported by the C Language.

Show Examples

Operator	Description	Example
sizeof()	Returns the size of a variable.	sizeof(a), where a is integer, will return 4.
&	Returns the address of a variable.	&a; returns the actual address of the variable.
*	Pointer to a variable.	*a;
? :	Conditional Expression.	If Condition is true ? then value X : otherwise value Y

Operators Precedence in C

Operator precedence determines the grouping of terms in an expression and decides how an expression is evaluated. Certain operators have higher precedence than others; for example, the multiplication operator has a higher precedence than the addition operator.

For example, $x = 7 + 3 * 2$; here, x is assigned 13, not 20 because operator * has a higher

precedence than +, so it first gets multiplied with 3*2 and then adds into 7.

Here, operators with the highest precedence appear at the top of the table, those with the lowest appear at the bottom. Within an expression, higher precedence operators will be evaluated first.

Show Examples

Category	Operator	Associativity
Postfix	() [] -> . ++ --	Left to right
Unary	+ - ! ~ ++ -- (type)* & sizeof	Right to left
Multiplicative	* / %	Left to right
Additive	+ -	Left to right
Shift	<< >>	Left to right
Relational	< <= > >=	Left to right
Equality	== !=	Left to right
Bitwise AND	&	Left to right
Bitwise XOR	^	Left to right
Bitwise OR		Left to right
Logical AND	&&	Left to right
Logical OR		Left to right
Conditional	?:	Right to left
Assignment	= += -= *= /= %= >>= <<= &= ^= =	Right to left
Comma	,	Left to right

The ? : Operator(Conditional operator)

Syntax: Exp1 ? Exp2 : Exp3;

Where Exp1, Exp2, and Exp3 are expressions. Notice the use and placement of the colon.

The value of a ? expression is determined like this –

- Exp1 is evaluated. If it is true, then Exp2 is evaluated and becomes the value of the entire ? expression.
- If Exp1 is false, then Exp3 is evaluated and its value becomes the value of the expression.

Branching:

Branching is so called because the program chooses to follow one branch or another.

if statement

This is the most simple form of the branching statements.

It takes an expression in parenthesis and an statement or block of statements. if the expression is true then the statement or block of statements gets executed otherwise these statements are skipped.

NOTE: Expression will be assumed to be true if its evaluated values is non-zero.

if statements take the following form:

Show Example

```
if (expression)
    statement;

or

if (expression)
{
    Block of statements;
}

or

if (expression)
{
    Block of statements;
}
else
{
    Block of statements;
}

or

if (expression)
{
    Block of statements;
}
else if(expression)
{
    Block of statements;
}
else
{
    Block of statements;
}
```

switch statement:

The switch statement is much like a nested if .. else statement. Its mostly a matter of preference which you use, switch statement can be slightly more efficient and easier to read.

Show Example

```
switch( expression )
{
```



```

        case constant-expression1:  statements1;
        [case constant-expression2: statements2;]
        [case constant-expression3: statements3;]
        [default : statements4;]
    }

```

Looping

Loops provide a way to repeat commands and control how many times they are repeated. C provides a number of looping way.

while loop

The most basic loop in C is the while loop. A while statement is like a repeating if statement. Like an If statement, if the test condition is true: the statements get executed. The difference is that after the statements have been executed, the test condition is checked again. If it is still true the statements get executed again. This cycle repeats until the test condition evaluates to false.

Basic syntax of while loop is as follows:

Show Example

```

while ( expression )
{
    Single statement
    or
    Block of statements;
}

```

for loop

for loop is similar to while, it's just written differently. for statements are often used to process lists such a range of numbers:

Basic syntax of for loop is as follows:

Show Example

```

for( expression1; expression2; expression3)
{
    Single statement
    or
    Block of statements;
}

```

In the above syntax:

- expression1 - Initialises variables.
- expression2 - Conditional expression, as long as this condition is true, loop will keep executing.
- expression3 - expression3 is the modifier which may be simple increment of a variable.

do...while loop

do ... while is just like a while loop except that the test condition is checked at the end of the loop rather than the start. This has the effect that the content of the loop are always executed at least once.

Basic syntax of do...while loop is as follows:

Show Example

```
do
{
    Single statement
    or
    Block of statements;
}while(expression);
```

break and continue statements

C provides two commands to control how we loop:

- break -- exit from loop or switch.
- continue -- skip 1 iteration of loop.

You already have seen example of using break statement. Here is an example showing usage of **continue** statement.

```
#include

main()
{
    int i;
    int j = 10;

    for( i = 0; i <= j; i ++ )
    {
        if( i == 5 )
        {
            continue;
        }
        printf("Hello %d\n", i );
    }
}
```

This will produce following output:

```
Hello 0
Hello 1
Hello 2
Hello 3
Hello 4
Hello 6
Hello 7
Hello 8
Hello 9
Hello 10
```

Chapter-7

Function

A function is a group of statements that together perform a task. Every C program has at least one function, which is **main()**, and all the most trivial programs can define additional functions.

You can divide up your code into separate functions. How you divide up your code among different functions is up to you, but logically the division is such that each function performs a specific task.

A function **declaration** tells the compiler about a function's name, return type, and parameters. A function **definition** provides the actual body of the function.

The C standard library provides numerous built-in functions that your program can call. For example, **strcat()** to concatenate two strings, **memcpy()** to copy one memory location to another

location, and many more functions.

A function can also be referred as a method or a sub-routine or a procedure, etc.

Defining a Function

The general form of a function definition in C programming language is as follows –

```
return_type function_name( parameter list ) {  
    body of the function  
}
```

A function definition in C programming consists of a *function header* and a *function body*. Here are all the parts of a function –

- **Return Type** – A function may return a value. The **return_type** is the data type of the value the function returns. Some functions perform the desired operations without returning a value. In this case, the return_type is the keyword **void**.
- **Function Name** – This is the actual name of the function. The function name and the parameter list together constitute the function signature.
- **Parameters** – A parameter is like a placeholder. When a function is invoked, you pass a value to the parameter. This value is referred to as actual parameter or argument. The parameter list refers to the type, order, and number of the parameters of a function. Parameters are optional; that is, a function may contain no parameters.
- **Function Body** – The function body contains a collection of statements that define what the function does.

Example

Given below is the source code for a function called **max()**. This function takes two parameters num1 and num2 and returns the maximum value between the two –

```
/* function returning the max between two numbers */  
int max(int num1, int num2) {  
  
    /* local variable declaration */  
    int result;  
  
    if (num1 > num2)  
        result = num1;  
    else  
        result = num2;  
  
    return result;  
}
```

Function Declarations

A function **declaration** tells the compiler about a function name and how to call the function. The actual body of the function can be defined separately.

A function declaration has the following parts –

```
return_type function_name( parameter list );
```

For the above defined function max(), the function declaration is as follows –

```
int max(int num1, int num2);
```

Parameter names are not important in function declaration only their type is required, so the following is also a valid declaration –

```
int max(int, int);
```

Function declaration is required when you define a function in one source file and you call that function in another file. In such case, you should declare the function at the top of the file calling the function.

Calling a Function

While creating a C function, you give a definition of what the function has to do. To use a function, you will have to call that function to perform the defined task.

When a program calls a function, the program control is transferred to the called function. A called function performs a defined task and when its return statement is executed or when its function-ending closing brace is reached, it returns the program control back to the main program.

To call a function, you simply need to pass the required parameters along with the function name, and if the function returns a value, then you can store the returned value. For example –

```
#include <stdio.h>

/* function declaration */
int max(int num1, int num2);

int main () {

    /* local variable definition */
    int a = 100;
    int b = 200;
    int ret;

    /* calling a function to get max value */
    ret = max(a, b);

    printf( "Max value is : %d\n", ret );

    return 0;
}

/* function returning the max between two numbers */
int max(int num1, int num2) {

    /* local variable declaration */
    int result;

    if (num1 > num2)
        result = num1;
    else
        result = num2;

    return result;
}
```

o/p: Max value is : 200

Function Arguments

If a function is to use arguments, it must declare variables that accept the values of the arguments. These variables are called the formal parameters of the function.

Formal parameters behave like other local variables inside the function and are created upon entry into the function and destroyed upon exit.

While calling a function, there are two ways in which arguments can be passed to a function –

Sr.No.	Call Type & Description
--------	-------------------------

1.Call by value

This method copies the actual value of an argument into the formal parameter of the function. In this case, changes made to the parameter inside the function have no effect on the argument.

2.Call by reference

This method copies the address of an argument into the formal parameter. Inside the function, the address is used to access the actual argument used in the call. This means that changes made to the parameter affect the argument.

By default, C uses call by value to pass arguments. In general, it means the code within a function cannot alter the arguments used to call the function.

Scope Rules

A scope in any programming is a region of the program where a defined variable can have its existence and beyond that variable it cannot be accessed. There are three places where variables can be declared in C programming language –

- Inside a function or a block which is called **local** variables.
- Outside of all functions which is called **global** variables.
- In the definition of function parameters which are called **formal** parameters.

.Local Variables

Variables that are declared inside a function or block are called local variables. They can be used only by statements that are inside that function or block of code. Local variables are not known to functions outside their own. The following example shows how local variables are used. Here all the variables a, b, and c are local to main() function.

```
#include <stdio.h>

int main () {

    /* local variable declaration */
    int a, b;
    int c;

    /* actual initialization */
    a = 10;
    b = 20;
    c = a + b;

    printf ("value of a = %d, b = %d and c = %d\n", a, b, c);

    return 0;
}
```

Global Variables

Global variables are defined outside a function, usually on top of the program. Global variables hold their values throughout the lifetime of your program and they can be accessed inside any of the functions defined for the program.

A global variable can be accessed by any function. That is, a global variable is available for use throughout your entire program after its declaration. The following program show how global variables are used in a program.

```

#include <stdio.h>

/* global variable declaration */
int g;

int main () {

    /* local variable declaration */
    int a, b;

    /* actual initialization */
    a = 10;
    b = 20;
    g = a + b;

    printf ("value of a = %d, b = %d and g = %d\n", a, b, g);

    return 0;
}

```

A program can have same name for local and global variables but the value of local variable inside a function will take preference. Here is an example –

```

#include <stdio.h>

/* global variable declaration */
int g = 20;

int main () {

    /* local variable declaration */
    int g = 10;

    printf ("value of g = %d\n", g);

    return 0;
}

```

When the above code is compiled and executed, it produces the following result –

```
value of g = 10
```

Formal Parameters

Formal parameters, are treated as local variables with-in a function and they take precedence over global variables. Following is an example –

```

#include <stdio.h>

/* global variable declaration */
int a = 20;

int main () {

    /* local variable declaration in main function */
    int a = 10;
    int b = 20;
    int c = 0;

    printf ("value of a in main() = %d\n", a);
    c = sum( a, b);
    printf ("value of c in main() = %d\n", c);
}

```



```

    return 0;
}

/* function to add two integers */
int sum(int a, int b) {

    printf ("value of a in sum() = %d\n",  a);
    printf ("value of b in sum() = %d\n",  b);

    return a + b;
}

```

When the above code is compiled and executed, it produces the following result –

```

value of a in main() = 10
value of a in sum() = 10
value of b in sum() = 20
value of c in main() = 30

```

Initializing Local and Global Variables

When a local variable is defined, it is not initialized by the system, you must initialize it yourself. Global variables are initialized automatically by the system when you define them as follows –

Data Type	Initial Default Value
int	0
char	'\0'
float	0
double	0
pointer	NULL

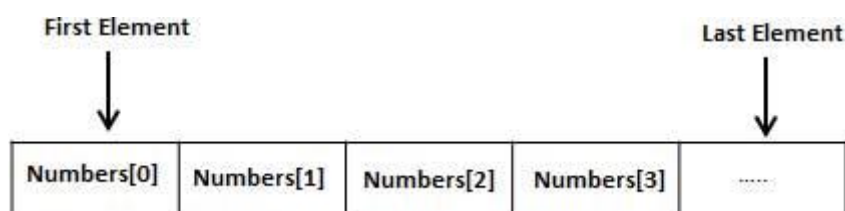
It is a good programming practice to initialize variables properly, otherwise your program may produce unexpected results, because uninitialized variables will take some garbage value already available at their memory location.

Array

Arrays are a kind of data structure that can store a fixed-size sequential collection of elements of the same type. An array is used to store a collection of data, but it is often more useful to think of an array as a collection of variables of the same type.

Instead of declaring individual variables, such as number0, number1, ..., and number99, you declare one array variable such as numbers and use numbers[0], numbers[1], and ..., numbers[99] to represent individual variables. A specific element in an array is accessed by an index.

All arrays consist of contiguous memory locations. The lowest address corresponds to the first element and the highest address to the last element.



Declaring Arrays

To declare an array in C, a programmer specifies the type of the elements and the number of elements required by an array as follows –

```
type arrayName [ arraySize ];
```

This is called a *single-dimensional* array. The **arraySize** must be an integer constant greater than zero and **type** can be any valid C data type. For example, to declare a 10-element array called **balance** of type double, use this statement –

```
double balance[10];
```

Here *balance* is a variable array which is sufficient to hold up to 10 double numbers.

Initializing Arrays

You can initialize an array in C either one by one or using a single statement as follows –

```
double balance[5] = {1000.0, 2.0, 3.4, 7.0, 50.0};
```

The number of values between braces { } cannot be larger than the number of elements that we declare for the array between square brackets [].

If you omit the size of the array, an array just big enough to hold the initialization is created. Therefore, if you write –

```
double balance[] = {1000.0, 2.0, 3.4, 7.0, 50.0};
```

You will create exactly the same array as you did in the previous example. Following is an example to assign a single element of the array –

```
balance[4] = 50.0;
```

The above statement assigns the 5th element in the array with a value of 50.0. All arrays have 0 as the index of their first element which is also called the base index and the last index of an array will be total size of the array minus 1. Shown below is the pictorial representation of the array we discussed above –

	0	1	2	3	4
balance	1000.0	2.0	3.4	7.0	50.0

Accessing Array Elements

An element is accessed by indexing the array name. This is done by placing the index of the element within square brackets after the name of the array. For example –

```
double salary = balance[9];
```

The above statement will take the 10th element from the array and assign the value to salary variable. The following example Shows how to use all the three above mentioned concepts viz. declaration, assignment, and accessing arrays –

```
#include <stdio.h>

int main () {

    int n[ 10 ]; /* n is an array of 10 integers */
    int i,j;

    /* initialize elements of array n to 0 */
    for ( i = 0; i < 10; i++ ) {
```



```

        n[ i ] = i + 100; /* set element at location i to i + 100 */
    }

    /* output each array element's value */
    for (j = 0; j < 10; j++ ) {
        printf("Element[%d] = %d\n", j, n[j] );
    }

    return 0;
}

```

When the above code is compiled and executed, it produces the following result –

```

Element[0] = 100
Element[1] = 101
Element[2] = 102
Element[3] = 103
Element[4] = 104
Element[5] = 105
Element[6] = 106
Element[7] = 107
Element[8] = 108
Element[9] = 109

```

Pointers

A **pointer** is a variable whose value is the address of another variable, i.e., direct address of the memory location. Like any variable or constant, you must declare a pointer before using it to store any variable address. The general form of a pointer variable declaration is –

```
type *var-name;
```

Here, **type** is the pointer's base type; it must be a valid C data type and **var-name** is the name of the pointer variable. The asterisk * used to declare a pointer is the same asterisk used for multiplication. However, in this statement the asterisk is being used to designate a variable as a pointer. Take a look at some of the valid pointer declarations –

```

int    *ip;    /* pointer to an integer */
double *dp;    /* pointer to a double */
float  *fp;    /* pointer to a float */
char   *ch     /* pointer to a character */

```

The actual data type of the value of all pointers, whether integer, float, character, or otherwise, is the same, a long hexadecimal number that represents a memory address. The only difference between pointers of different data types is the data type of the variable or constant that the pointer points to.

How to Use Pointers?

There are a few important operations, which we will do with the help of pointers very frequently. **(a)** We define a pointer variable, **(b)** assign the address of a variable to a pointer and **(c)** finally access the value at the address available in the pointer variable. This is done by using unary operator * that returns the value of the variable located at the address specified by its operand. The following example makes use of these operations –

```

#include <stdio.h>

int main () {

    int  var = 20;    /* actual variable declaration */
    int  *ip;         /* pointer variable declaration */

    ip = &var; /* store address of var in pointer variable*/

```

```

printf("Address of var variable: %x\n", &var );

/* address stored in pointer variable */
printf("Address stored in ip variable: %x\n", ip );

/* access the value using the pointer */
printf("Value of *ip variable: %d\n", *ip );

return 0;
}

```

When the above code is compiled and executed, it produces the following result –

```

Address of var variable: bffd8b3c
Address stored in ip variable: bffd8b3c
Value of *ip variable: 20

```

NULL Pointers

It is always a good practice to assign a NULL value to a pointer variable in case you do not have an exact address to be assigned. This is done at the time of variable declaration. A pointer that is assigned NULL is called a **null** pointer.

The NULL pointer is a constant with a value of zero defined in several standard libraries. Consider the following program –

```

#include <stdio.h>

int main () {

    int  *ptr = NULL;

    printf("The value of ptr is : %x\n", ptr );

    return 0;
}

```

When the above code is compiled and executed, it produces the following result –

```

The value of ptr is 0

```

In most of the operating systems, programs are not permitted to access memory at address 0 because that memory is reserved by the operating system. However, the memory address 0 has special significance; it signals that the pointer is not intended to point to an accessible memory location. But by convention, if a pointer contains the null (zero) value, it is assumed to point to nothing.

To check for a null pointer, you can use an 'if' statement as follows –

```

if(ptr)      /* succeeds if p is not null */
if(!ptr)     /* succeeds if p is null */

```