1. Computer Organisation :-

- Introduction to Computer education of computer generation of computer.
- classification of computer
- Basic Organisation of computer (functionally block diagram
- Input and Output devices
- computer memory and classification of memory

2. Computer Software :-

- Application software overview of operating system objectives and functions of O.S. Types of operating system.
- software concept
- system software
- Batch processing
- Multiprogramming type
- Features of Dos, windows and unix
- Programming language computer interpreter computer virus.
- Detection and preventation of Virus.
- Application of computer's in different domain

3. Computer Network and Internet :-
Networking concept

- Professionaly communicating media
- Data transmission mode

Introduction of computer :—

(i) Computer is an electronic device which takes input process it and gives output to the user

(ii) Desktop, personal computer, laptop and general purpose computers.

(iii) The home appliences, electronic gadgets are also computers if it's circuit containing programe is embaided into them.

(Ex - washing machine digital camera
digital watch Refrigerator
Microwave

Ceiling fan is not a computer as it's not an electronic device but it an electrical device.
Mobile is also a computer

Hardware :—

Hardware is a comprehesive term for all the physical part of a computer as distinguished from the data it contains or operates on and the software that provides instruction for the device for the hardware to accomplish task.

Hardware is anything which is tangible in the computer system.

Software :—

Software are two types.

(i) Application software
(ii) System software

System software :—
It is for computer system, they might be software responsible to control hardware or any back ground task.

Evolution of computer :—

↳ Development of the present day computer is the result of a advancement of technologies and the need of quantify calculation

↳ Abacus was one of first counting device use for simple operation like addition, substraction, multiplication, division.

↳ In 1600 John napier invented a calculating device called Napier's bone.

↳ Blase pascale invented pascaline in 1645 and it is used for only addition and substraction Lebiniz in 1671 improved the pascal's addition machine and invented the leibniz calculator.

↳ Charless babbage invented the ~~Leibniz calculator~~ difference engine 1822. He was first person to design a fully programmable computer.

↳ Adanasoff - berry computer (ABC) was first purpose electronic computer was invented by Jb Adanasoff and c berry. It use a binary system of Arithmatic, parallel processing, regenerative memory and electronic circuit.

Generation of Computer :—

1st generation (1942 - 1956) :—

↳ It used vaccum tubes for circuitory and magnetic drum for memory.

↳ They were very much expensive and enormous/ large.

↳ Input was based on punchcards and output was displayed on print-outs.

Ex : — ENIAC

(Electronic numerical integrator and calculator

UNIVAC

Universal automatic Computer

IBM 701

2nd generation Computers (1956 - 1964) :—

↳ Here transistors replaced vaccum tubes and inplace of magnetic drum core technology was used, it is still realayed on punchcards for input and printout for outputs.

Ex :— UNIVAC — III

CDC 1604

Honeywell

3rd generation (1964 - 1975) :-

↳ Here transistors were shrinked and placed on silicon chip called semiconductors which amazingly increased the speed and efficiency of computers.

↳ Here instead of punchcards and printouts, users interacted through keyboards and monitors and used operating system.

↳ For the first time computers became accessible to a mass of audiance because they were smaller and cheap than their previous one.

Ex :- IBM360
PDP8
CDC6600

4th generation (1975 - 1989)

↳ Here ICs with VLSI technology were used. Here microprocessors are also used.

↳ Thousands of integrated circuits were onto a single silicon chip.

↳ In this generation computers also stop the development of GUIs (graphical user interface)

Ex - mouse and handheld devices.

5th generation :— (1989 - present)

↳ ICs with ULSI technology are used here.

↳ This generation computing device based on artificial intelligence are still in development process.

Ex - PARAM1000
IBM NOTEBook
PANTIUM PCs

Classification of computer :—

The modern days computers are broadly classified into

(a) Analog
(b) digital
(c) hybrid

Analog :—

Analog computers are operates on mathematical variable in the form of physical quantities that are continously varing.

These computers are mostly used in industries, in process control activity.

These computers works on Analog data such as variables in temperature, speed, voltage etc.

Advantages :—

The shows the solution in a simple and graphical in less time.

Disadvantages :—

They are not versatile or much accurate.

Digital computer :-
They are the general purpose computer that works on digital or binary field.

Advantages :-
Speed and accuracy of this computers are very high.

Disadvantages :-
The real time change required complex programming and graphical programmes.

Hybrid computer :-
These are computer that have features of both Analog and digital computer

These computer are used to control the entire process.

Based on processing capability :-
Computers are available in different shapes, sizes and weight due to this they performs different kinds of job.

Super computer :-
These are most powerful computer. Super Computers play an important role in field of computational science and are used for a wide range of computationally intensive tasks in jobs in various fields including quantum mechanics climate research weather forecasting

Mainframe Computer :—

These Computer can also process millions of instructions per seconds and capable of accessing billions of data

These computers have large primary memory and have substantial processing capabilities.

These computers are generally too expensive and are used in companies.

Mini Computer :—

These are the computers which are mostly used by the small types of business or universities.

They have fairly large primary memory that supports wide range of application area.

Workstation :—

It is a powerful desktop computer designed to meet the computing needs to engineers architectures and other proffessionals, generally unit Operating System or variation of it.

Operating system is generally designed to support multiuser environment.

UNIX

↓

Solaris

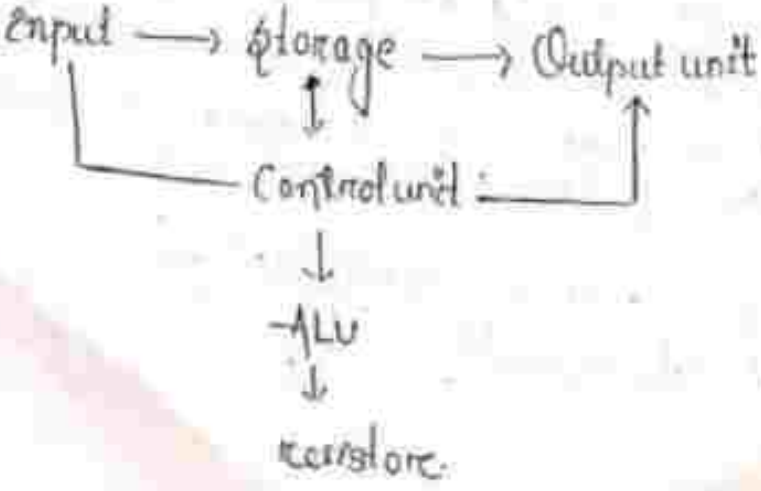
Micro Computer :—

↳ It is the smallest & cheapest available for mas
↳ These are portables and required minimum power.
↳ It processing Power is appropriate for handeling most of the task.

Storage units :—

1 nibble = 4 bits
1 byte = 8 bits
1 kilobyte = 1024 bytes
1 megabyte = 1024 kilobytes
1 Gigabyte = 1024 megabytes
1 terabyte = 1024 Gigabytes
1 pitabyte = 1024 terabytes
1 Exabyte = 1024 pitabytes
1 Zettabyte = 1024 exabytes
1 yottabyte = 1024 Zettabytes

Basic Organizations of Computer :—



Input unit :-

↳ Computer need to receive data and instruction in Order to exicuit data and instruction any problems.

↳ The input unit consists of One Or more input device.

↳ The input device accept data and instruction and Converts it to Computer understandable form and then supply the converted data to the Computer System for further processing.

Ex - mouse, Keyboard, Punchcard, Cpu

Central processing unit (Control unit + ALU)

↳ The mainunit of inside the computer is Cpu.

↳ This unit is responsible for all events inside the computer

↳ It performs all calculations, takes all decisions and controls all units of the computer.

↳ Cpu Contains Cu (control unit) which is responsible for all the activities like transfer of data, instruction etc between various subunit.

↳ Alu is the part where actual Computation takes place.

↳ Registers are small in Size and are capable of storing few bits at a time

Output unit :-

↳ After the Computer process data it needs to represent the result in a human understandable form.

Input device :—

Input device is any hardware component that allows you to enter data programme.

There are various input devices.

Keyboard :—

The most common input device used by the computer are the keyboard.

The keyboard is a way to input letters, numbers, or special characters into programme.

It mainly contains 4 types of key.

The four types of keys are given below.

- Alphabet keys
- Function keys
- Control keys
- Numerical keys

(b) Mouse :—

It is an input device that is used to control the movement of pointer on the screen and to make selection from the screen.

(c) Optical mark reader :—

It is a special type of input device or scanning device which is used to read carefully placed pen or pencil marks on a OMR sheet.

Scanner :—

It is an input device that Optically scan image or printed text and convert it to a digital image.

Magnetic ink character recognization :— (MICR)

It is a special type of input device that enables special character printed in magnetic ink to be read an input rapidly to a computer.

Output Device :—

In Order to communicate with computer and get the result we need computer devices like —

Monitor :—

The computer monitor (visual display unit) is the most common Output device. images are represented On monitor by individual dots. called pixel.

A pixel is the smallest unit On the screen that can be turned On and off Or made different shade.

Printer :—

A printer is an Output devices that produces hardcopy of a data. The resolution of a printer is expressed as DpI (dot per inch).

Metro computer uses five kinds of printer.

i) Dot matrix printer
ii) Dessyuhot printer
iii) chain printer
iv) Ink-jet printer
v) Lasser printer

Plotters :—
It is a special purpose Output devices that draws image with ink pen.

Computer memory :—
The computer memory is the part of the system use to store data and programme which may be permanently or temporarily depending on the requirement and type of memory.

Classification of computer memory :—
According to the purpose memory can be devided into 5 types.

i) Resistor memory ii) Cache, primary or ~~secondary~~ memory, Auxillary memory, Back up memory.

Resistor memory :—
It is integrated inside the C.P.U. Its memory capacity is small which is use for storing data temporarily during the exicution of a problem.

Cache :—
It is the smallest memory between cpu and main memory. It's purpose is to hold frequently needed data from the main memory dividing the execution.

Primary memory :—
It is the ~~primary~~ memory in a computer which is use to store data and instruction during executing Programme.

The

Rom – Read only Memory

ROM can be further divided (read only memory) 4 types.

P- ROM (programable ROM)

Ep-ROM (Erase programable ROM)

EEp- Electrically Erasable programmable ROM.

UVEPROM – Ultra violet Erasable Programmable ROM

↳ ROM is mainly used to store small programmes permanently.

↳ This is non-volatile in nature

Auxillary memory Or secondary memory :–

It is a memory where data Or programme are stored permanently this type of memory hold data until they are erased, hence this memories are none-volatile in nature. Generally magnetic materials Or Optical materials are used as secondary memory.

For ex – Floppy disk

Harddisk

CD-ROM

Back up memory :—

This memory is used by the user for keeping backup of important data for future use

Software Concept:—

A computer system mainly comprised of two catagory i.e.,

i) computer hardware

ii) Computer software

↳ The tengible component of computers are referred as Computer hardware.

↳ Computer Software refers to the set of programes which make the hardware Operational. The computer software can be classified into 2 types i.e.,

(a) System Software

(b) Application Software

System Software :—

Software programes that are designed to control the Operation of computer hardware is known as System Software.

↳ It provides a platform for installation and developement of application software.

↳ It facilited the execution of a programme returnin high level language and also provides error free Communication between the main computer system and attached peripheral device etc.

↳ System software consists of low level and highlevel software that interact with the computer at a very basic level.

↳ Some common system software are Operating system
device drivers
utility programs
language processors

↳ Operating system is responsible for managing the various computer software.

↳ language processor are the catagory of the system software which are responsible for translating and interpretating the programe written by using programming level.

Ex - Compiler, Interpreter.

↳ Device driver are the system software which usually comes along with a peripheral device which is used to establish communication between the devices and computer.

↳ Utility programe is the programe that is directly capable of directly interact with computer hardware for various purpose.

Application software :—

↳ An application software is any programme or groups of programme that is use for the end user.

↳ It includes such thing as data base programme, word processer and browser with spread sheet.

Overview of Operating system :—

↳ An Operating system is an interface between a computer user and computer hardware.

↳ An Operating system is a software which performs all basic function like —

i) memory management
ii) processor management
iii) Device "
iv) File "
v) Error detecting
vi) Control over
vii) System performance

(viii) Co-ordination between other software or user.

Memory management :—

↳ It refers to management of primary memory or main memory. A main memory is a large array of words or bytes. Where each word or byte has its own address. For a programme to be execute it must me in the main memory. An Operating system does some activities for memory management.

↳ Keeps tracks of primary memory i.e what part of it are in use by whom, what parts are not in use

↳ Allocates the memory when a process requests it to do so.

↳ Deallocates the memory when a process no longer needs it or has been terminated etc.

Processor ~~manage~~ management :—

In multiprogramming enviroment the Operating system decides which process gets the processer when and for how much time. This function is called as process ~~scheduling~~ scheduling.

An operating system does few activities for processer management

(i) Tracks of processor and status of process the programe responsible for task is known as traffic Controller.

(ii) Allocates the processor to approaches

(iii) Deallocates the processer when no longer required.

Device management :—

↳ An Operating system manage device communication via their respective drivers.

↳ Operating system does like few activities for device-management like -

(i) Keeptracks of all device programme responsible for the tasks is known as input - Output Controllers.

(ii) Decides which process gets the device when and for how much time.

(iii) Allocates the device in the efficient way

(iv) Deallocates the device.

File management :—

A filesystem is normally Organized into directories for easy navigation and use.

This directories may contains files and other directories.

An Operating system does the few activities like keeps track of information, location, status etc.

This collective facility is often known as file system.

Decides who gets the resources.

Allocates the resources or deallocates the resources.

Security :—

By means of password or similar other techniques it prevents unauthorized access programmes and datas.

Production of dumps error message and other debugging and error.

Control over system performance :—
Recording delays between request for a service and response from the system.

Co-ordination between other software and users.

Co-ordination and assignment of compilers, and other printers, assemblers, other software to the various uses of the computer system.

Objectives of Operating system :—
To make the computer system convenient to use in an efficient way.
It hides in an efficient way the details of the hardware resources from the user.
To manage the resources of a Computer System.

Types of Operating system :—
Real time Operating Operating
It was designed to respond to an event with in a pre-determined time.
It aims to execute real time application such as autopilot mechanism used in aircrafts.
It has very little user interphase capability and no end user

Ex - WinCE —

Single user Operating system :—
It is designed for one user to effectively use a computer in time.

Ex - Window95 2000

Multi user Operating system :—
It allows many users to take advantage of the computer resources simultaneously.
It ensures the problem with one user doesn't affect the entire community of the user.
Ex - Unics Operating system
Linux etc.

Single tasking Operating System :—
It allows only a single programme to run at a time.
Ex :- Palm Operating System.

Multy tasking Operating System :—
It allows the execution of multiple tasks On application at one time.
Ex - Apple's mac Operating system.
Microsoft Windows.

Time Sharing Operating system :—
It allows many users to share the computer resources simultaneously.
It provides advantages of quick response.
It reduces CPU idle time.

Distributed Operating System :—
It uses multiple central processors to serves multiple real-time applications.
It manages a group of independent Computers and makes them appear to a single computer.
Data processing jobs are distributed among the processors according to their efficiency.

Multi user Operating system :—
It allows many user to take advantage of the computer resources simultaneously.
It ensures the problem with one user doesnot affect the entire community of the user.
Ex - Unics Operating system
Linux, LENUX, etc.

Single tasking Operating system :—
It allows only a single program to run at a time.
Ex :- Palm Operating System.

Multy tasking Operating system :—
It allows the execution of multiple tasks or application at one time.
Ex - Apple's mac operating system.
Microsoft Windows.

Time sharing Operating system :—
It allows many user to share the computer resources simultaneously.
It provides advantages of quick response.
It reduces CPU idle time.

Distributed Operating system :—
It uses multiple central processer to multiple real-time applications.
It manages a group of independent Computers and makes them appear to a single computer.
Data processing jobs are distributed among the processers according to their efficiency.

BIOS :—

It is a built in software that determines what a computer can do without accessing programme from a disk.

Time saving Operating system :—

Time sharing Operating system is a technic which enables many people located at various terminal to use a particular computer system simultaneously.

Time sharing Or multitasking is logical extension of multi-programming Operating system.

Processor's time which is said among multiple users simultaneously is termed as time sharing Operating System.

The main difference between multi-programe batch system and time sharing system is that incase of multi programme batch system the objective is to maximize Processor use where as in time sharing system the objective is to minimize response time.

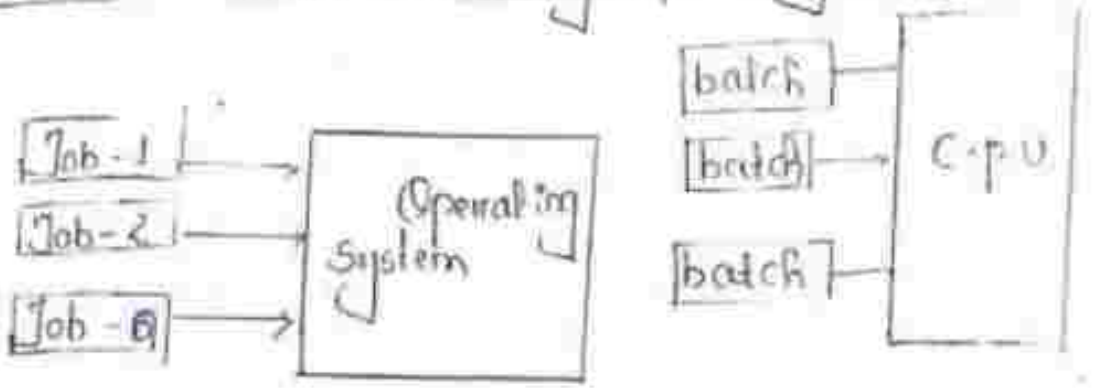
Advantages :—

- It provide advantage of quick response.
- It avoids duplication of software.
- It reduces Cpu idle time.

Disadvantages :—

- Problem of reliability.
- questions of security and integrity of user programe data.
- Problem of data communication.

Batch Processing Operating system:-



Batch processing is a technic in which an Operating System collects the ~~programme~~ data and programe together in a batch.

Before processing starts. An Operating system does few activities related to batch processing like

(i) The Operating system defines a job which has pre-defined sequence of commands Programes and data as a single unit.

(ii) The Operating System keeps a no. of jobs in the memory and execute them without any manual information.

(iii) Jobs are process as in the Order of Submission i.e 1st come 1st serve fashion.
when a job completes its execution its memory is released and the Output for the job gets copied into an Output ~~spot~~ spool. for ~~later~~ later Printing Or processing.

Advantages:-

↳ Batch processing takes much of the work of the Operator to the computer.

↳ Increased performance as a new job gets started as soon as previous job finished without any interruption.

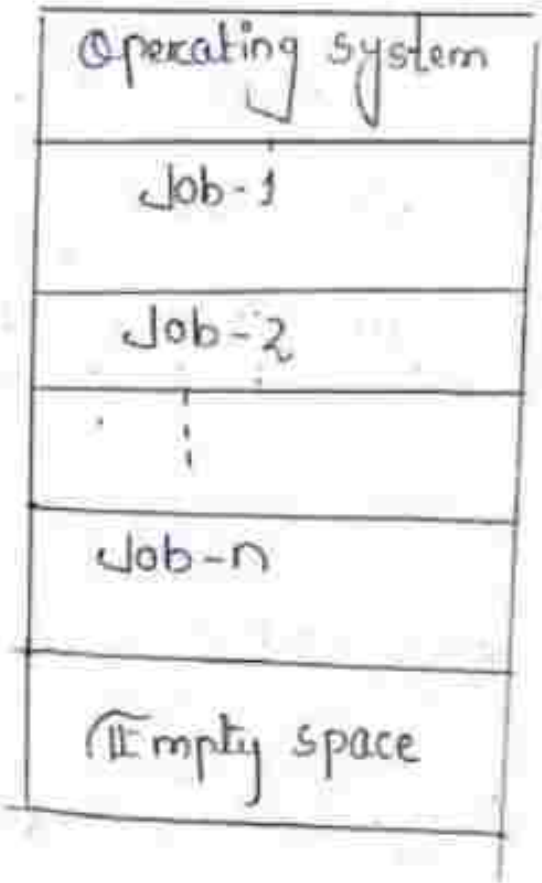
Disadvantages:-

↳ Difficult to debug programme.

↳ A job could enter an infinite loop.

↳ Due to lack of protection scheme one batch job can effect pending job.

Multiprogramming



Sharing the processor when two or more programme executes the programme over the same time then it is referred as multi programming.

Multi programming increases Cpu utilization by organizing jobs so that Cpu always have one to execute

An Operating system does few activities related to multi programming.

(i) The Operating system keeps several jobs in memory at a time.

(ii) This set of job is a subset of jobs kept in the job pool.

(iii) The Operating System picks and begins to execute one of the jobs in the memory.

(iv) Multiprogramming Operating system monitors the state of all active programmes and system resources using memory management programme to ensure that the Cpu is never idle unless there are no jobs to process.

Advantages :-

↳ High and efficient Cpu utilisation.

↳ Users feels that many programmes are allotted Cpu almost simultaneously at the same time.

Disadvantages :-

↳ Cpu scheduling is required to accomodate many jobs in memory management is required.

Features of DOS:- (Disk Operating System)

↳ It is the primary system where the user gets an environment about the input and Output device.
Ex - Monitor, Keyboard, printers etc.

↳ It is helpful in performing file management, like Creating, editing, deleting file etc.

↳ It is a single user Operating System. One user can Operate at one time.

↳ It is a characterbased interphase system.

↳ (Here we can type only letters and characters in this Operating system)

↳ It is a 16bit Operating system.

↳ It is an assembly language.

Feature of UNIX Operating system:-

↳ It is written in 'C' and assembly language

↳ It is multi tasking with protected memory.

↳ Multiple users can run multiple programmes (Each of at same time without interfacing with each other Or Crashing the system.

↳ Very efficient Virtual memory can run so many Progra with a modest amount of physical memory.

↳ Portability (It uses high level language that is easy comprehend, modified and transfer to other machines. We can change language Code according to the requirement of new hardware On your Computers.

Disadvantages:-

↳ It is not very user friendly Operating system.
↳ It basic interface is command language. Even an expreinced user can make this mistake using the interface.
↳ It is machine independent means it hides the machine architecture from the user making it easier to write application that can run on micro, mini and mainframe computers.
↳ The hubs of unix operating system. The kernel manages the application and peripheal on a system together the kernel and shell carryout are requests and commands.
↳ We communicate without Our system through the Unix shell which translate to the kernel.

Kernel:-
It is the centerpart of an Operating system. It manges the Operation of the Computers - (most portably memory and Cpu type)

Shell:-
↳ It is an environment in which we can run Our command programmes and our shellscripts.
↳ It provides us with an interface to the unix system.
↳ It gathers input from us an execute programmes based On the input.

Features of Windows:-
↳ It is speed Or fast
↳ Compartable
↳ hardware requirement.
↳ Search and Organisation.

↳ Safe and security
↳ Interface and desktop.
↳ task or taskmenu

Advantages —
↳ It provides readymade solution that can be implemented just about anyone whoever used a computer

↳ Using window in the working place is over 95% of all virus and mallicious software are written for the window Operating system.

Feauture of linux :-
↳ Multitasking and Multiuser.
↳ portable
↳ ~~Support~~ Virtual memory

Compiler :—
↳ A programe that converts instructions (highlevel) into a machine Code or lower level form so that they can be read and execute by a computer.

Ex - MS. ~~viol~~ Visual studio.

Interpretator :—
Interprinter is a Computer programme which execute a statement directly.

Ex. python, lisp

Ocam - Objective Catagorical abstract machine language.

2) Differenciate between compiler and Interpreter :—

Compiler	Interpreter
↳ It is a special type of programe that process statements written in a particular programming language and turns them into machine language.	↳ It is a special type of compiler programe that directly exicutes instructions written in a programming language.
↳ It usually generates intermediate code in the form of object file.	↳ It doesn't creat an intermediate object file / code.
↳ Faster exicution of control statement as compared to the interpriter.	↳ Slower exicution of control statement as compared to compiler
↳ Detects errors in the programe get displayed after the entire programe is read by compiler.	↳ Detected errors in the programe get displayed after each instruction read by the interpriter.
↳ Ex - Hs-Visual studio, C++.	↳ Ex - python, lisp.

Programming language :—

↳ It is a vocabulary or a set of grammatical rules for instructing a computer device to perform specific task.

↳ The term programming language usually refers to high level language such as basic, C, C++, Java, COBOL etc.

↳ In programming language has the unique set of keyboard and a specific syntax for organising computer programme instruct.

Computer Virus :—

→ A computer virus is a mellicious software programme loaded into a users a knowledge and performs mellicious actions.

→ Computer virus replicate itself and spread from one computer to another by attaching itself to an existing programme.

→ There are various types of virus are —

(1) Resident Virus

Resident Virus :—

→ It fixes themselves into the system memory and get activated whenever the operating system run and infects all the files that are then open.

Ex — Vienna Virus.

* Direct action Virus
It comes into action when the file containing the virus executed once it infects the file into the folder that are specify.

Ex:- polyboot B.
→ Anti EXE

* Boot sector Virus:—
This type of virus effect the boot sectors of the harddisk this also known as master boot sector. Or master boot record virus.

Ex - Elkern

* Macro Virus :—
It infects file that are created using certain application and programme that contains extensions like .doc, .ppt etc.

* File system Virus :—
It infect the directory of your computers by changing the path that indicate the location of the file. It is also known as cluster virus. Of directory.

Ex - Dir - 2.

Application of computer in different domain / field :-

In different domain computers have become an essential part of modern human life. There are many computers uses in different fields like -

↳ Computers acts at educational institution.

↳ Computers have its dominion use in educational field which can make learning better which can make distance learning make easy.

↳ Researchers have massive uses of these computers in that work from the starting to till the end of their scholarly of their.

↳ Computers acts in health Center. Most of the medical information can now be digitized quick.

↳ From the prescription, different types of therapy to the patient are also possible, because of the computer.

↳ ECGs, MRI, Ctiscan, Xerox is not possible without Computers.

↳ Computers acts financial institution.

↳ The four most important thing is to store information about different account holders in a database to be available at any time in banks.

↳ For keeping the records of the cash flows and keeping information regarding your account.

↳ Computers act in different sectors.

↳ Computers are the main help in development.

↳ Design & the maintainance are possible only through computer.

↳ Computer builds the links between the soldier and commanders through the satelite.

↳ Computers for entertainment - Computers are now maximum entertainer and primary time pass machine. We can use computer for playing games, watching movie, listening movie. We can use computer for drawing picture.

↳ Computer acts in e-commerce :—
Electronic shopping, Online shopping added favour to purchasers and merchants. Electronic banking is now in our hand where every bank has online supports for transaction of monitory issue.

Malware :—

↳ It is a programe designed to gain access to computer system normally for the benifit of some third party without the users permission.

↳ Malware includes Virus, worm, trojen horse, ransomware, spyware and other malicious. on programe.

Virus :—

↳ A virus is a malicious executable code attached to another executable file.

↳ Viruses can be harmless or they can be modified or delete data.

Worm :—

↳ Worms replicates themselves. On the system attaching themselves to different files and looking for pathways between computers.

↳ Worm usually slowdown networks of computer system system.

↳ After a worm affects a host it is able to spread very quickly over the network.

Spyware :—

↳ It's purpose is to steal private information from a computer system for a third party.

↳ Spyware collects information and sends it to the Hacker.

Trogen horse :—

↳ It's a malware that carried out malicious operations under the appearance of a desired operation such as playing an online game.

↳ A trogan horse vary from a virus because the trogan binds itself to non-executable files. such as image, audio, file.

Ransomware :—

↳ Ransom ware grasp a computer system or data it contains until the victim makes a payment.

Ransomware encrypts data in the computer with a key which unknown to the user. The user has to pay a ransom to the criminals to retrieve the data.

Once the amount is paid the victim can resume using his/her system.

COMPUTER NETWORKING AND INTERNET :—

↳ A network is a set of devices connected by communication link.

↳ A network must be able to meet a certain number of criteria

↳ The most important of this are performance, reliability and security.

LAN (local area networking) :—

↳ It is usually privately owned and links devices in a single office, building, campus.

↳ It's size is limited to a few kilometer.

MAN

It is designed to extend over an entire city. It may be a single network such as a cable television network.

WAN (Wide area network) :—

It provides long distance transmission of data voice, image and video information over large geographic areas that may comprises if country, continent or even the whole world.

Data transmission mode :—

i) Simplex
ii) Half duplex
iii) Full duplex

Defination :—

Data transmission mode defines the signal flow between to connected device. It is devided into three types.

(1) Simplex :—

(i) Here communication is unidirectional as on a one way stream.

(ii) Only one of two devices on a link can transmit & the other can only receive.

(iii) Here performing mode of transmission is worst.

Ex - keyboard, Monitor.

Half duplex :—

(i) Here each station can both transmit and recieve but not at same time.

(ii) When one device is sending the other can only receive.

Full duplex :—

Here both station can transmit and received simultaneously. The full duplex mode is like two way street with traffic flowing in both directions at the same time.

Ex- Telephone

Protocol :—

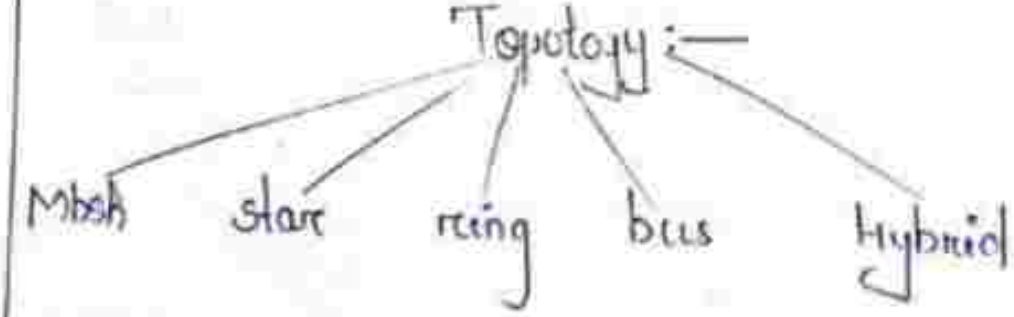
It is a digital language through which we communicate with other on the internet.

Protocol means that it is a set of mutually accepted and implemented rules at both ends of the communication channel.

There are different types of protocol's use in internet like Tcp, ARP

Tcp – transmission control protocol
Ip – Internet protocol
ARP – average resolution protocol
Dhcp – dynamic host configuration protocol
DNS – domain name system.
FTP – file transfer protocol

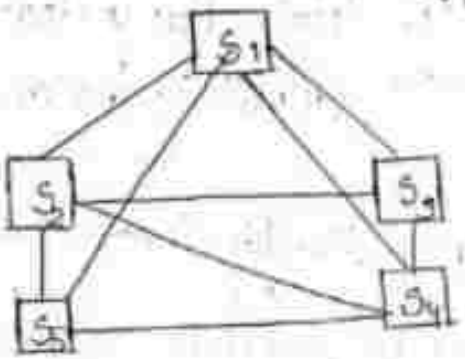
Topology :—



Two or more devices to a link forms a topology. The topology of a network is a geometric representation of relationship of all the links and linking devices to another.

There are basically five types of topology.

Mesh topology –



→ In the mesh topology every device has a dedicated point to point link to every other device.

→ A mesh topology offers several advantages over other network topology. Like the uses of dedicated links, guarantees that each connection can carry its own data load. by eleminating the traffic problem.

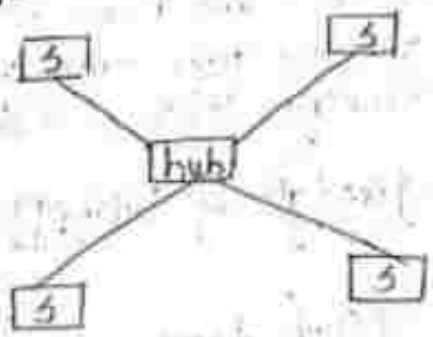
→ Fault detection can be detected easily. It is robust in nature and reliable.

Disadvantages :—

→ Instalation and reconnection are difficult.

→ The bulk of wiring can be greater than the available space.

Star topology :—



↳ In star topology each device has a dedicated point to point link only to a central controller usually called as a hub.

↳ Here the devices are not directly linked to one another

↳ If one device want to send data to another it sends the data to the controller (hub), which then relay the data to the other connected device. and has less cabling needed

↳ It is easier to install and fault detection is also easy

Disadvantages :—

↳ If hub goes down then everything will go down.

↳ The device can work without hub

Bus topology :—

Bus topology on the other hand is multipoint one long cable act as backbone to link all the devices in a network.

Advantages –

↳ It is easy instalation.

↳ less cabling star and mesh topology.

Ring topology:-

In ring topology each devices has a delicated point to point connection only with the two devices on either side of it.

-A signal is passed along the link. One direction from device to device until it reaches it destination.

Advantanges:-

↳ Easy to instlan.

↳ Easy managing is easier as to add or remove a device from the topology only two links are required to be chang

Disladvantages:-

-A link failure can fail the entire network as the single will not travel forward due to failure.

Hybrid topology:-

↳ Data traffic issue since all the data is circulating in a ring.

↳ -A combination two or more topology is known as hybrid -topology.

↳ Ex - a combination of star topology and ring topology

Advantages-

We can choose the topology based on requirement.

Disadvantages-

Installation is difficult

Design is complex so maintainance is high.

Repeater :—

↳ It operates at physical layer. Its job is to regenerate the signal over the same network before the signal become too weak or corrupt. So as to extend the length to which the signal can be transmitted over the same network.

Bridge :—

It is used to divide a large network into smaller segment. It connects different type of media as a network becomes more complex bridges make sure your network speed does not drop drastically.

Switch :—

↳ It contains many ports to connect different network segment. They are similar to hub but offers greater performance.

↳ When a network contains a large no. of devices switches are needed instead of hub to make sure communication between devices doesnot slow down.

INTERNET

↳ Internet is a global wide area networks that connects computer system across the world.

↳ It carries many network services most prominently the world wide wave including social media from electronic mail social media.

↳ Isp is an acronim that stands for internet service provider.

↳ An Isp is a company that provides a internet acess Orgnazination and home users.

↳ Internet service provider in India are like jio, airtel, Vodafone, Idea, Bsnl.

Different types of Internet connectivity :—

↳ There are various Options available to connect the internet to set up and internet connection we must have an internet service provider.

↳ Dial Up connection requires users to link there phone line to a Computer in Order to access the internet.

↳ It doesn't permit user to make Or receive phone calls through their home phone service while using the internet.

Dsl :—

↳ Digital Subscriber line.

↳ It uses to Copper wire telephone line connected to One's home so service is delivered at the same time as

↳ Customer can still place calls. while suffering internet.

Satelite :—

↳ In certain area where broadband connection is not yet offered a satelite internet option may be available.

Cable :—

↳ Through use of a cabel modem user can excess the internet over cabel T.V. line.

↳ Cabel modem can provide extremely fast excess to the internet.

Isdn :— (Integrated service digital network)

↳ It allows user to send data voice and video contained over digital telephone line.

↳ The instalation of an Isdn adaptor is required at both ends of the transmission means on the part of the user as well as the internet axis provider.

Broadband :—

↳ This highspeed internet connection provided through either cable or telephone company.

↳ This one of the fastest option available, broadband internet uses multiple data channels to send large quantity of information.

E-mail :—

↳ E-mail is a paperless method of sending messages, notes or letters from one person to another even many people simultaneously via the internet.

↳ E-mail is very fast compared to the normal post.

The features of E-mail includes -

↳ One to One communication
↳ One to many communication.
↳ Instant communication.

Physical presence of recipient is not required.

↳ Most inexpensive mail service, 24 hours a day and 7 days in a week

WWW :- (World Wide Wave)

↳ It is a way of exchanging information between the computers on the internet, tying them together into a vast collection of interactive multimedia resources.

↳ It is a system based on hypertext and HTTP (Hypertext transfer protocol) for providing, organising and accessing a wide i.e variety of resources.
available via the internet.

File transfer Protocol (FTP) :-

↳ FTP is a protocol used for transferring files from one computer to another typically from your computer to a web server.

↳ FTP is the preferred method of exchanging files because it's faster than a other protocol like HTTP.

The basic steps to use FTP are -

(a) Connect the FTP server.
(b) Navigate the file structure to find the file you want
(c) Transfer the file.

Chatting :-
Instant messaging just is usually between two people or may be more

Ex - Whatsapp
Facebook
Instagram
Telegram
Hike
Wechat

Internet relay chat :- (IRS)
It requires software on client many chat rooms

Internet conferencing :- (Wave conferencing)
It refers to a service that allows conferencing events to be held with remote location.

These are sometime referred as online workshop.

Application of waves conferencing include meeting transfers events features or share presentation of any computer

E-paper :- (Electronic newspaper)
An electronic newspaper is the online version of a newspaper.

Either as a stand alone publication or as online version of printed one.

↳ going Online creats more Opportunities for newspaper such as competing with broadcast journalism in presenting breaking news in a more timely manner.

↳ E-shopping :-

↳ It is a form of E-commerce which allows consumers to directly buy goods or services from a seller over the internet using a way browser.

↳ It is also known as E-wave shopping, E-shop, wave shop, virtual store etc.

↳ An Online shop evokes the physical analogy of buying Produc or services at a bricks and mortar retailer or shopping Centre.

↳ The process is called business to consumer. (b to c Online Shop)

↳ In the case where the business buys from another business the process is called business-to business Online shopping.

Ex of some Online retailing Corporation are -

(i) Amazon
(ii) Myntra
(iii) Shein

Types of connection :—

There are two types of connection.

(i) point to point connection.

(ii) Multipoint connection.

i) It provide a dedicated link between the two device. The entire capacity of link is transmission those two devices.

ii) A multipoint connection is one in which more than specific device shares a single link.

In a multipoint environment the capacity of the channel is share.

Networking device :—

A networking devices holds the key to generate a network to which communication occur. These device is responsible for smooth data transmission among different nodes of some network and nodes of different netwo

Networking device :—

A networking device holds the key to generate a network to which communication

Hub :—

It is a centralized device that connect multiple devices. when hub receiveves the data signal from a connected devices of any of it accept that export it from those signal to all others connected devices from the remaining ports.

4) File Management & data processing

→ Files

- File is nothing but an electronic document.
- It's contents can be ordinary text or it can be an executable program.
- Each file is given a file name to identify it. The file name is in the form:

Filename . Extension

- Filename can be consist of Alphabets or combinations of alphabets, numerals & special characters. Extension indicates the type of f
- Example:

xyz.Doc , abc.wav , pqrppt.mp3

Here filename is xyz.
Extension name is .Doc which indicates document file.

→ Folder:

→ Folder contains a group of files. Folder is other called as directory. Folder may have a set of files under it. It may have other folders under it also. The files & folders can be arranged in hierarchical manner.

→ File organization:

- It refers to the way data is stored in a file. File organization is very important because it determines the methods of access, efficiency, flexibility & storage device to use. There are methods of organizing files on a storage medi

i) sequential file organization.
ii) random or direct file organization.
iii) serial file organization.
iv) indexed sequential file organization.

i) Sequential file organization -
- Records are stored & accessed in a particular order stored using a key field.
- Retrieval requires searching sequentially through the entire file record by record to end.
- Here the record in a file are sorted in a particular order. Better file searching method like the binary search technique can be used to reduce the time used for searching a file.
- Since the records are sorted it is possible to know in which half of the file a particular record being searched is located. Hence this method repeatedly divides the set of records in the file into two halves & searches only the half on which the record is found.
- Ex:

Let a file has records with keyfields 10, 20, 30, 40, 50, 60 & the computer is searching for a record with keyfield 50, it starts at 40 upwards in its search ignoring the first half of the set.

Advantages -
- Sorting makes it easy to access records.
- The binary chop technique can be used to reduce record search times as little time is taken.

Disadvantages -
- The sorting does not remove the need to access other records as the search looks for particular records.
- The requirement that all records be of the same size is sometimes difficult to enforce.
- Sequential records cannot support modern technologies.

ii) Random or direct file organisation -
- Here records are stored randomly but accessed directly.
- To access a file stored randomly, a record key is used to determine where a record is stored on the storage media.
- Magnetic & optical disks allow data to be stored & accessed randomly.

Advantages -
- Quick retrieval of records.
- The records can be of different sizes.

iii) Serial file organisation -
- Records in a file are stored & accessed one after another.
- The records are not stored in any way on the storage medium this type of organisation is mainly used on magnetic tapes.

- It is simple.
- It is cheap.

Disadvantages -
- It cannot support modern high speed requirements for quick record access.
- It is difficult to access because in this we have to access all preceding records before retrieving the one being searched.

iv) Indexed sequential file organisation -
- Almost similar to sequential method only that an index is used to enable computer to locate individual records on the storage media.
- Ex: On a magnetic drum, records are stored sequential on the tracks. However each record is assigned an index that can be used to access it directly.

* Data file - it contains records in sequential scheme.
* Index file - it contains the primary key/s its address in the data file.

Deciding on a file organisation :-
- The major factors to be considered while file organisation should be chosen are the following:
* Type of activity: records processed in a day, if large number of records are accessed at a time, direct access file organisation

of very few records are accessed, sequential file organization will be more suitable & cheaper. files that are frequently updated on the direct storage device such as disk on a direct

Data processing :-

- Data processing means, processing the input data to produce some meaningful & purposeful information. Computer has the capability of processing high volume of data in less time with higher accuracy. Data processing involves 5 distinct steps.

a> data capturing
b> data validation
c> processing / execution
d> data storage.
e> data retrieval.

→ Data capturing encompasses the activities of inputing data to the computer. Before giving input to the system the required data are to be first identified & put in the defined format called source data layout. The aim of this layout is to have faster data entry. To reduce the volume of data & also have better organisation & easy access to data those can suitably be coded. After data are ready they can be entered to the computer through keyboard. This is something sometimes called data capture through intelligent terminals. The other form of data capture is through scanners or optical devices. In this type of data capturing data are not entered rather data are captured from the the source document or paper.

→ After data capturing the data is validated. Data validation involves checking of input data to fit the requirements or specifications. For example price of a book can be numeric. Only if by mistake alphabetic data are entered then it is checked & error is shown to revalidate the data. This prevents invalid & unspecified data to enter into the system causing errors.

→ The valid input data are stored in file or database & processed as per the instructions. The instructions are put in programs or software. This software or programs when executed does the processing of input data & produces the output. Again outputs are stored in files in memory.

After processing of data, the outputs are produced. The output may be formed in different ways depending on the requirements specifications. The same set of data can be printed in tabular form or in form of graphs. There are variety of ways for presenting data. The printed output is sometimes called hardcopy. There can be provision of answering to queries of user where the answer is displayed on the monitor screen itself. So it depends on the requirements of users.

→ An important step of data processing is maintaining database. Database is nothing but collection of data which is controlled centrally with many provisions of data security. This is where usually data are stored for reference.

—x—

⑤ Problem solving Introduction

Algorithm -

→ Algorithm is defined as the step by step solution of problem in user language.

→ It is considered as an effective procedure for solving a problem in finite number of steps.

The characteristics of algorithm are -

i) precise i.e. clear & accurate/particular, exact

ii) Unambiguous i.e not open to more than one interpretation / definite / specific

iii) Finite termination i.e fixed or limited end

iv) Unique solution

Once algorithm is written, it can be coded into program using any programming language.

Algorithm uses 3 different constructs.

i) Sequence

ii) Branching or decision making

iii) Repetition

Here sequence says that instructions are to be executed in what order or sequence. Branching involves testing of condition & based on the outcome of the condition testing different instructions are executed. Repetition means one or more instructions shall be repeated for a number of times. This is otherwise called as loop. There are different types of loops such as ~~while~~, do-while, for etc.

Examples:

① Algorithm to find out sum of two numbers to be taken as input.

step 1 - Read the 1st number x

step 2 - Read the 2nd number y.

step 3 - Sum = x + y

step 4 - print sum

This is an example where only sequence is exhibited.

(2) Algorithm to find out larger between numbers to be taken as input -

step 1 - Read the 1st number x.
step 2 - Read the 2nd number y.
step 3 - If x > y
Then print x
Else if x < y
Then print y
Else print "both are equal"

This is an example where branching is exhibited.

(3) Algorithm to find out sum of first 10 natural numbers
step 1 - i=1, sum=0
step 2 - Repeat steps 3 & 4 while i <> 10
step 3 - sum = sum + i
step 4 - i = i+1
step 5 - print sum

This is an example where repetition is exhibited

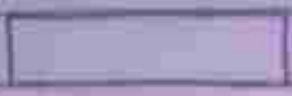
Flowchart →

Flowchart is a graphical or symbolic representation of the process of solution to a problem or algorithm. It helps to visualise the complex logic of the solution of the problem in a simplified manner through diagrammatic representation.

Each step of the algorithm is represented using a symbol & a short step description. The different symbols used for the flowchart are.


start or stop


input or output


process

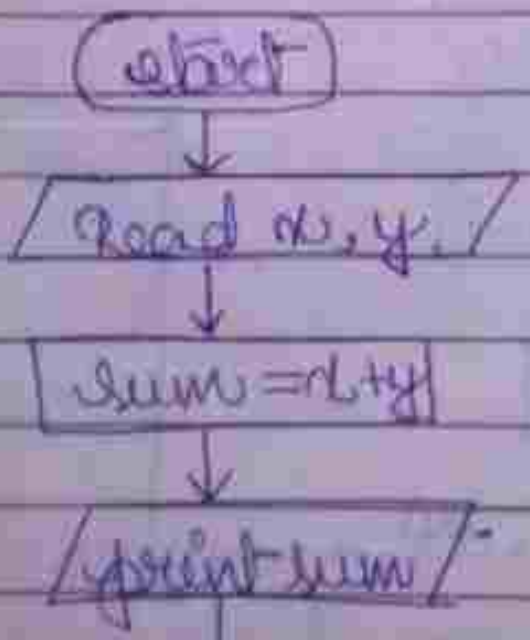

Decision symbol.

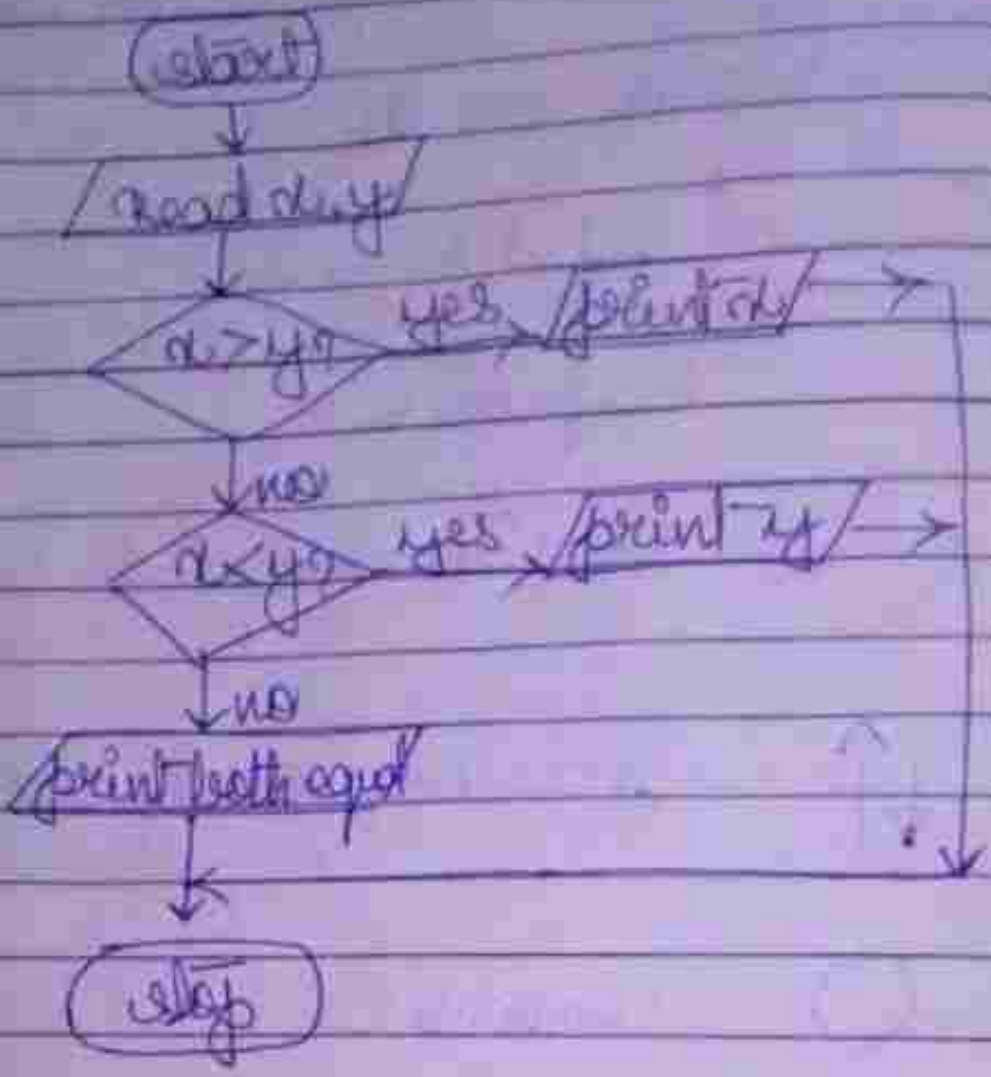

Arrows for sequence.


Connectors

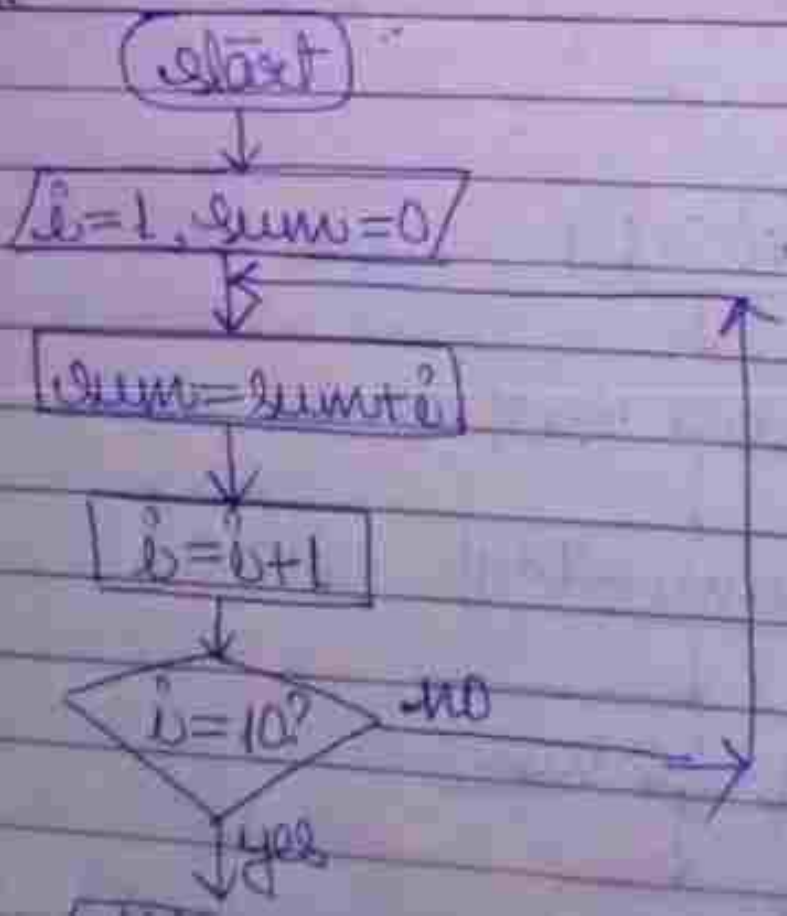
Example:

① Flowchart to find out sum of two numbers to be taken as inputs.





③ Flowchart to find out sum of first 10 natural numbers.



pseudocode

- It is a concise descriptive algorithm in English language that uses programming language constructs & contains features of the programming that can be easily converted to program. It focuses on the logic of the algorithm without giving more on the syntax of programming language. This is meant for understanding the logic of the program easily. Flowchart can be considered as an alternative of pseudocode.

programming language :-

- programming language is a tool to express the logic or instructions for understanding of the computer.
- Any programming language has two components
  - Syntax
  - Semantics

→ Syntax refers to the rules to be followed for writing valid program statements. compilers can detect errors in the syntax while compiling the program.

→ Semantics is associated with the logic of the program. Compilers can not detect the semantic errors. The user of programmer can diagnose semantic errors.

pseudocode -

pseudocode has no syntax like any of the programming language & thus can't be compiled or interpreted by the computer.

- Pseudocode is detailed yet readable description of what a computer program or algorithm must do, expressed in a formally - styled natural language. Rather than in a programming language. It is sometime used as a detailed step in the process of developing a program.

Structured programming language →

Structured programming is also known as modular programming. In this type of programming technique, the program shall be broken into several modules. This helps in managing memory efficiently as the required module of the program will be loaded into the memory only & not the entire program. This will also enhance code re-writting, understanding, debugging & modifying the individual module of the programs.

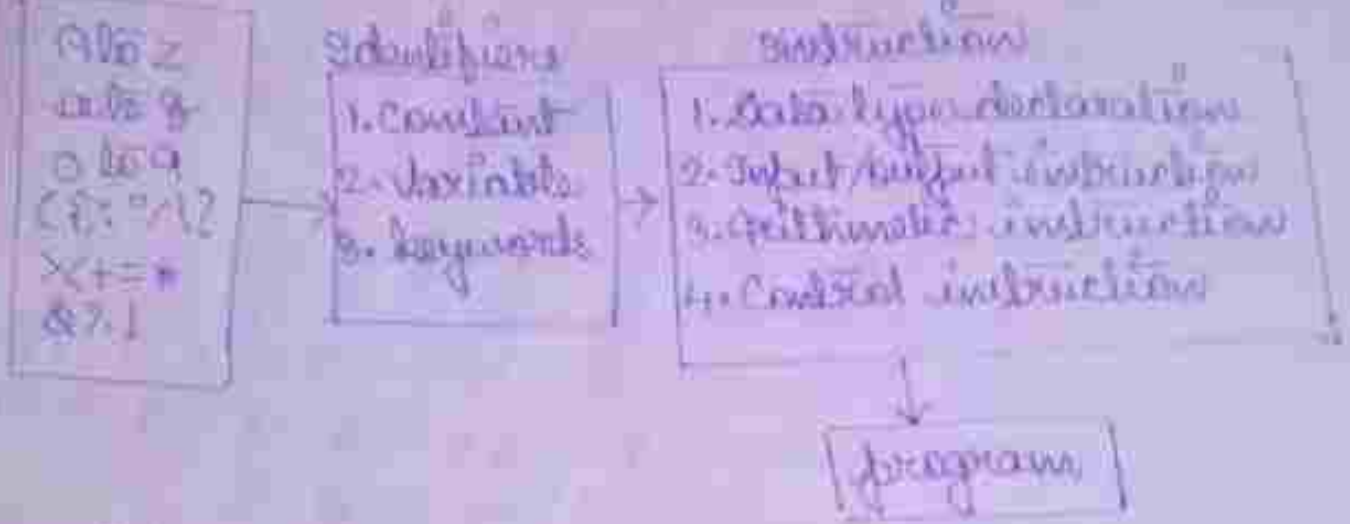
Ergonomics - The study & development of systems for improving & assisting the memory

programming language

- History of C language - C is a mid-level programming language developed by Dennis Ritchie in the year of 1972 in Bell Laboratories by USA.

- Why C is important - Basic syntax are similar in C
  - Core libraries of Android are written in C.
  - Linux OS is developed in C.
  - Almost all embedded devices are written in C.

→



Identifiers - Smallest identifying unit in the program is known as identifiers. It can be categorised as constant, variable, keywords, function etc.

1) Constant - Any information or data is constant. Every program use inputs to do process. Data For example, program to add two numbers we have to calculate simple interest, we need some data to process. In addition of two numbers prog we need two numbers. Similarly in SI prog we need p, t & r and then only we can process it so

Data = information = constant

Constants

- Primary constant
  - (i) Integer [-55, 25, 0]
  - (ii) Real [2.4, 3.56, -1.3]
  - (iii) character ['a', 'b', '+', ' ']
- Secondary constant
  - (i) Array
  - (ii) String
  - (iii) Pointer
  - (iv) Structure
  - (v) Enumeration

ii) Variables - Variables are the names of memory locations where we store data. Variable name is any combination of alphabet, digit & underscore. No other symbol is permitted. Valid variable name cannot start with digit.

iii) Keywords - keywords are predefined words, whose meaning is already known to compiler. There are 32 keywords according to the ANSI standards, but sometime compilers may support additional keywords too.

List of keywords

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

terminated with a semicolon.

Datatypes - Datatypes are keywords which are used for various purposes to specify the kind of data. One use of datatypes is the declare variables.

int
char
float
double
void

Declaration statements -

int a, b = 5;

float b1;

char ch, w;

double d1;

- Here variables a & b are of type int, b1 is of float, ch & w are of type char and d1 is of type double.

Header files -

- # is called preprocessor directive, this used after is called preprocessor command, one of the most popular preprocessor command is "include".
- "include" tells the compiler to insert the contents of "stdio" & "conio" etc a particular place.
- standard input output

printf() is a standard output function.
scanf() is a standard input function.
these functions are already defined inside the language
library. These functions are not defined by the programmer
hence are known as predefined function.
- If we want to use printf() in a program then we need
to include "stdio" header file & similarly if we want to use
getch() function then we need to include "conio" in a
headerfile. If we want to use squareroot function which is
sqrt() then we have to include "math" in headerfile.

Escape sequence -
\n → new line character
\b → same as backspace key
\t → same as tab key
\r → carriage return, it moves cursor at the
starting place of the current line
\\ → To print backslash single line

Printing values of the variables & expressions -

- To print values of the variable we have to use format
specifiers
%d for int
%c for char
%f for float
%lf for double
%d, %c, %f, %lf are called format specifiers. They
are special symbols used to tell printf about the
kind of data should be.

Arithmetic instruction :

3 + 4 → operator, → operand, → Expression

i) unary operator (one operand)
ii) Binary operator (two operands)
iii) Ternary operator (three operands)

Note: There is no BODMAS rule in C language.

→ According to the behaviour & priority table:

i) unary
ii) arithmetic
iii) Bitwise
iv) Relational
v) Logical
vi) conditional
vii) Assignment

i) unary : (+, -, ++, --, sizeof)

Ex: -3, +8

→ ++ (increment operator)

```
void main()
{
int x=3;
x++;
printf("%d",x);
getch();
}
```

O/P 4

x++ is post increment
++x is pre increment

```
→ void main()
{
int x=3;
++x;
printf("%d",x);
getch();
}
```

O/P 4

```
main()
{
int x=3, y;
y=x++;
printf("%d %d", x, y);
getch();
}
```

o/p 4, 3

```
main()
{
int x=3, y;
y=++x;
printf("%d %d", x, y);
getch();
}
```

o/p 4,4

→ x-- is post decrement
--x is pre decrement

```
void main()
{
int x=3, y;
y=x--;
printf("%d %d", x, y);
getch();
}
```

o/p 3,3

```
void main()
{
int x=3, y;
printf("%d %d", x, y);
getch();
}
```

o/p 2,2

→

```
void main()
{
int a;
float b;
char ch;
double d;
printf("\n%d", sizeof(int));
printf("\n%f", sizeof(float));
printf("\n%c", sizeof(char));
printf("\n%lf", sizeof(double));
getch();
}
```

o/p

4
4
1
8

```
void main()
{
int a, b, c, d;
a = 3+4;
b = 3-4;
c = 3*4;
d = 3/4;
printf("%d %d %d %d", a, b, c, d);
getch();
}
```

O/P 7 -1 12 0

iii) Bitwise operators (&, |, ^, ~, >>, <<);

→ Bitwise AND (&)	→ Bitwise OR (|)	→ Bitwise XOR (^)
0 & 0 → 0	0 | 0 → 0	0 ^ 0 → 0
0 & 1 → 0	1 | 0 → 1	0 ^ 1 → 1
1 & 0 → 0	1 | 1 → 1	1 ^ 0 → 1
1 & 1 → 1	0 | 1 → 1	1 ^ 1 → 0

→ Bitwise NOT (~)

~0 → 1

~1 → 0

```
- void main()
{
int x;
x = 5&12;
printf("%d", x);
getch();
}
```

5 = 0000000 0000000 00000000 00000101

12 = 0000000 0000000 00000000 00001100

& ________________________________

4 = 0000000 0000000 0000000 00000100

(O/P - 4)

```
- void main()
{
int x;
x = 12>>2;
printf("%d", x);
getch();
}
```

12 = 0000000 0000000 0000000 00001100

>> ________________________________

3 = 0000000 0000000 0000000 0000011

>> → right shift operation

<< → left shift operation

```
{
int x;
x = 'A' << 2 ;
printf("%d", x);
getch( );
}
```

<< 65 = 0000000001000001 (65)
48 = 0000000100000100 (260)

O/P = 48

iv) relational operators (<, >, <=, >=, ==, !=):

Result of these operators are either 1 (for true) or 0 (false)

5 > 4 is 1
4 < 4 is 0
3 != 4 is 1
5 > 4 > 3 is 0

v) Logical operators (!, &&, ||):

! → Logical NOT
&& → Logical AND
|| → Logical OR

```
- void main( )
{
int x;
x = !(5>4);
printf("x = %d", x);
}
```

O/P x = 0

! True = False
! False = True

```
- void main( )
{
int x;
x = !4;
printf("x = %d", x);
getch( );
}
```

O/P x = 0

```
int main( )
{
int x;
x = 5>3&&4<0
printf("%d", x);
```

O/P 0

Exp 1	Exp 2	result
True	True	True
True	False	False
False	True	False
False	False	False

```
int x, y=5;
x=y>10||y<7;
printf("%d", x);
getche();
}
```

true	true	true
true	false	true
false	true	true
false	false	false

o/p 1

v) conditional operator:

Expression1 ? expression2 : expression3;

vi) assignment operator (=, +=, -=, *=, /=, %=, &=, \=, ^=);

x=4; → valid expression, here value 4 is assigned to the variable x.

4=x; } These are invalid expressions.
3=4;

- Compound assignment operator (+=, -=, *=, /=, %=, &=, \=, ^=)

```
void main()
{
int x=5;
x+=4;
printf("x=%d", x);
getche();
}
```

o/p x=9

x+=4; → x=x+4;

x-=3; → x=x-3;

x*=5; → x=x*5;

x/=7; → x=x/7;

x%=3; → x=x%3;

Decision control instructions:

Control instructions:

A program is a set of instructions. Each instruction of the program is executed by the processor. It executes instructions one by one. Processor's control moves from one line to another. This movement of control is known as flow of the program.

Sometimes, it is required that the program's flow should not be sequential. For example, we want to execute first line of our program, then second line but after executing second line, we may want to skip few lines & jump to the 5th line. In such situations, we use control instructions.

- Control instruction determines the programmer to decide the flow of the program.

There are four types of control instructions.

i) Decision control instruction.
ii) Iterative control instruction.
iii) Switch case control instruction.
iv) Goto jump instruction.

i) Decision control instruction-

Decision control instruction is also known as selection control instruction.

a) if
b) if-else
c) conditional operator
d) if-else-if / multiconditional if statement

a) if statement :-

Syntax:-

```
if( some condition )
{
```

statement 1;
statement 2;
.
.
.
}

Here 'if' is a keyword which tells the compiler that what follows is a decision control statement.
- Immediately after 'if' keyword there is condition.
- condition can be any valid expression in C language.
- condition is always evaluated as true or false.
- If the result of expression is non-zero it is considered as "true" otherwise "false".
- Immediately after true condition there is a block of code. This block is known as if block. Statements in if block will be executed only when condition is "true".
- When condition is "false" control skips if block & executes statements written after if block.

Eg:- program to check whether user pass or fail

```
void main()
{
int marks;
clrscr();
printf("Enter marks");
scanf("%d", &marks);
if (marks >= 33)
{
printf("you are pass");
}
```

```
if(marks<90)
{
printf("you are fail");
getch();
}
```

→ if else statement

Syntax - if (some condition)
{
statement 1;
statement 2;
:
:
}
else
{
st 1;
st 2;
:
}

- This is similar to if but the else block is not by default.
- If the condition of if is true if block is executed and if the condition of if is false else block is executed. We can use if statement without else block but else must be in conjunction with if block.
- else block should appear immediately after

if block otherwise can execute during compilation.

Ex- program to check whether pass or fail using if-else.

```
void main()
{
int marks;
clrscr();
printf("Enter marks");
scanf("%d", &marks);
if(marks >= 33)
{
printf("you are pass");
}
else
{
printf("you are fail");
}
getch();
}
```

c) Nested if statement -

- Syntax:

```
if(condition-1)
{
if(condition-2)
{
st-1;
}
else
```

{	}
st 1;	else
} }	{
else	st 3;
{	}
if(condition 2)	}
{	st 4;
st 2;	

When an if statement occurs within another if statement, then such type of statement is called nested if statement. The statement is little complex than the simple if or if else statement but it is very useful to solve real problems.

Ex: WAP to find out biggest no. among 3 nos

```
void main()
{
int a, b, c;
clrscr();
printf("Enter three nos");
scanf("%d %d %d", &a,&b,&c);
if(a>b && a>c)
printf("%d is biggest", a);
else if (b>a && b>c)
printf("%d is biggest", b);
else
printf("%d is biggest", c);
getch();
}
```

→ 4) If else if / multicondition if statement –

When in a complex problem condition arise in a sequence then we can use multicondition if statement to solve the problem in a simple manner.

syntax –

```
if (con-1)
{
st-1;
}
else if (con-2)
{
st-2;
}
else if (con-3)
{
st-3;
}
:
else if (cond.-n)
{
st-n;
}
else
{
default st;
}
statement-x;
```

Ex: Code for display the color according to the code.

```
void main()
{
int code;
clrscr();
printf("Enter the color code");
scanf("%d", &code);
if(code == 1)
{
printf("\n color is red");
}
else if(code == 2)
{
printf("\n color is green");
}
else if(code == 3)
{
printf("\n color is white");
}
else
{
printf("\n no color code defined");
}
printf("\n End of the program");
getch();
}
```

⇒ Conditional operator

Conditional operator is the only operator in C language which requires three operands hence known as ternary operator.

Syntax-

Expression1 ? expression2 : expression3;

- Expression1 is condition evaluated as true or false. When expression1 is true expression2 is selected otherwise expression3 is selected.

Ex:

WAP to find greater between two numbers:

```
void main()
{
int x, y;
clrscr();
printf("Enter two numbers");
scanf("%d%d", &x, &y);
x>y? printf("%d is greater", x) : printf("%d is great
getch();
}
```

- Occasionally we have to write more than one statement in expression2 or expression3, in that case use comma to separate statements. For example conditional operator provides an ease of writing selective assignments.

Another conditional operator can be use in a conditi operator as an expression2 or expression3, this is called nesting.

Q. WAP to find greater among three numbers using nested if else.

```
void main()
{
int a,b,c;
clrscr();
printf("Enter three numbers");
scanf("%d%d%d",&a,&b,&c);
if(a>b)
{
if(a>c)
printf("%d is greater",a);
else
printf("%d is greater",c);
}
else
{
if(b>c)
printf("%d is greater",b);
else
printf("%d is greater",c);
}
getch();
}
```

- Since, if-else statement is considered as a single statement, thus we can remove curly braces used within the body of if block and else block.

ii) Iterative control

a) for loop
b) while loop
c) do while loop

a) for loop -

Syntax:
```
for(initialization; condition; flow)
{
st-1;
st-2;
}
```

Ex: WAP to print 'learning C' six times on the screen using for loop

```
void main()
{
int i=1;
for(i=1;i<=6;i++)
{
printf("learning C\n");
}
```

b) while loop -

Syntax:
```
while(some condition)
{
block of statements;
}
statement x;
```

Ex: WAP to print 'learning C' five time on the screen.

```
void main()
{
int i=1;
```

printf("learning c \n");
i++;
}

c) do-while loop:-

Syntax:-

```
do
{
block of statements;
}
while(some condition);
statement 2;
```

Ex:- C program for printing "learning c" five times.

```
void main()
{
int i=1;
do
{
printf("learning c\n");
i++;
}
while(i<=5);
}
```

it is difficult to solve such type of complex problem with the help of ladder if statement, then there is need of such type of statement which should have different alternatives or different cases to solve this problem in simple & easy way. For this purpose switch statement is used.

Syntax:

```
switch (expression)
{
case value1;
 block1;
 break;
case value2;
block2;
break;
 .
 .
 .
case valuen;
 blockn;
 break;
default;
default block;
}
```

Ex: Write a program to print the color according to the code

```
void main()
{
int code;
printf("\n main menu");
printf("\n1. for color red");
printf("\n2. for color green");
printf("\n Enter the color code");
scanf("%d", &code);
```

```
case 1:
printf("\n color is Red");
break;
case 2:
printf("\n color is Green");
break;
default:
printf("\n color does not found");
}
getch();
}
```

break statement :

- This statement is always used with a decision making statement like if & switch statement. The statement statement will quit from the loop when the condition is true.
- The purpose of break is to terminate loop's execution immediately as it encounters.

Syntax:

```
while(condition1)
{
if (condition 2)
{
break;
}
statement 1;
}
```

Ex:

```
void main()
{
int i=1,n;
```

```
printf("Enter a number");
scanf("%d",&x);
if(x>0)
break;
i++;
}
i==6? printf("Ends normally"):printf("applied break");
getch();
}
```

Case:

<u>continue:</u>

- The keyword continue is used only in the body of loop & it is used to move control at termination condition in the case of while & do while loop & also in for loop

syntax:

```
while(condition1)
{
st 1;
st 2;
if(condition2)
{
continue;
}
st 3;
st 4;
}
st 5;
```

```
main()
{
int x;
while(1)
{
printf("enter the correct number");
scanf("%d",&x);
if(x==1)
exit(0);
else
{
printf("this is the correct value");
break;
}
}
getch();
}
```

goto:

The goto moves the control to the specified address called label name.

syntax:

```
goto label_name;
.....
.....
label_name:
```

Eg:

```
main()
{
int sum=0;
for(int i=0;i<=0;i++)
{
sum=sum+i;
```

```
void addition()
{
}
addition();
printf("%d", sum);
return 0;
}
```

Type conversion:

- To convert one data type to another data type is known as conversion.

Ex:

```
main()
{
int a=5;
float b;
b=a;
printf("b=%f", b);
return 0;
}
```

o/p
b=5.000000

Expressions:

- It is a combination of variables, constants & operators written according to the syntax.

algebraic expression	C expression
$a+b-c$	a+b-c
$a \times b + c - d$	a * b + c - d
ab/c	a * b / c
$3x^2+2x$	3*x*x+2*x

Date: 15.01.20

(?) Universal features of C

Functions

- Function is more about the arrangement of the code as has some name for identification.
- The format of function is

```
return_type function_name(argument)
{
statements;
}
```

- The correct way to write a program is

```
void add()
{
 int a, b, c;
 printf("Enter two numbers");
 scanf("%d %d", &a, &b);
 c = a+b;
 printf("Sum is %d", c);
}
```

- Function names must be unique. We can have any numbers of function in a C program. We can define function in any order, main() is also a function. We cannot define a function inside a function.
- Function definition is different from function declaration & function call.
- There are two types of functions
  - i) predefined functions
  - ii) user-defined functions.

- add( ), main( ) are user-defined function, whereas printf, scanf are prime examples of predefined functions
- function call is a representative of function code.

```
void add( )
{
int a, b, c;
printf("Enter two numbers");
scanf("%d%d", &a, &b);
c = a + b;
printf("sum is %d", c);
}
```

function definition — int a, b, c; … printf("sum is %d", c);

function call

- Function declaration is also known as function prototype. It is a single line statement written for the compiler to process. The general format of function prototype is:

return_type function_name(argument);

- void main( ); ← function declaration (global declaration)

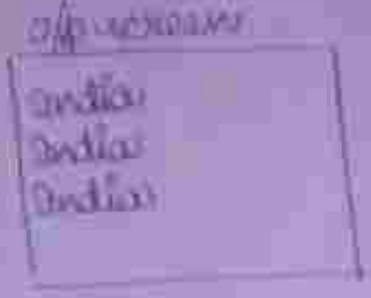
```
main( )
{
add( ); ← function call
}
void add( )
{
int a, b, c;
printf("Enter two numbers");
scanf("%d%d", &a, &b);
c = a + b;
printf("sum is %d", c);
}
```

function definition — int a, b, c; … printf("sum is %d", c);

Roll No: 19.0420

```
#include<stdio.h>
#include<conio.h>
main()
{
clrscr();
add();
issues();
add();
add();
}
add()
{
int a,b,c;
printf("Enter two numbers");
scanf("%d %d",&a,&b);
c=a+b;
printf("Sum is %d",c);
}
issues()
{
int a;
printf("Enter a number");
scanf("%d",&a);
if(a%2==0)
printf("Even");
else
printf("Odd");
}
```

```
main()
{
a();
b();
a();
}
a()
{
printf("\nIndia");
}
b()
{
a();
}
```



ways to define

ways to define a function

```
/* Takes nothing, Returns nothing */
#include<stdio.h>
#include<conio.h>
main()
{
void add(void);
clrscr();
add();
getch();
}
void add()
{
int a, b, c;
printf("Enter two numbers");
scanf("%d %d", &a, &b);
c = a + b;
printf("sum is %d", c);
}
```

ii) /* Takes something, returns nothing */

```
#include<stdio.h>
#include<conio.h>
void main()
{
int x,y;
void add(int, int);
clrscr();
printf("Enter two numbers");
scanf("%d%d",&x,&y);
add(x,y);
getch();
}
void add(int a, int b)
{
int c;
c = a+b;
printf("Sum is %d",c);
}
```

iii) /* Takes nothing, Returns something */

```
#include<stdio.h>
#include<conio.h>
int add(void);
void main()
{
int s;
clrscr();
s = add();
printf("Sum is %d",s);
getch();
}
```

```
int add()
{
int a, b, c;
printf("Enter two nos");
scanf("%d%d", &a, &b);
c = a+b;
return(c);
}
```

2) /* Takes something, returns something */

```
#include<stdio.h>
#include<conio.h>
int add(int, int);
void main()
{
int s, x, y;
clrscr();
printf("Enter two nos");
scanf("%d%d", &x, &y);
s = add(x, y);
printf("sum is %d", s);
getch();
}
int add(int a, int b)
{
int c;
c = a + b;
return(c);
}
```

Benefits of functions:

i) Easy to read, debug & modify
ii) better memory utilization
iii) achieves modularization.

Recursion

- When a function call itself, then it is called recursion.
-

```
main()
{
int k;
k = sum(3);
printf("%d", k);
}

int sum(int a)
{
int s;
if(a==1)
return(a);
s = a+sum(a-1);
return(s);
}
```

```
main()
k [ ]   k=sum(3);
        printf("%d",k);

sum(int a)
a [ ]   if(a==1)
        return(a);
s [ ]   s=a+sum(a-1);
        return(s);

sum(int a)
a [ ]   if(a==1)
        return(a);
s [ ]   s=a+sum(a-1);
        return(s);

sum(int a)
a [ ]   if(a==1)
        return(a);
s [ ]   s=a+sum(a-1);
        return(s);
```

- Not every problem can be solved using recursion.
- If a problem can be solved using recursion, then it must have an iterative solution too.
- We can think of a function which is calling itself, like:

```
int fun()
{
  ....
  fun();
  ....
}
```

→ There are several types of recursion

i) Linear recursion - A linear recursive function is a function that only makes a single call to itself each time the function runs.

ii) Tail recursion - A function call is said to be tail recursive if there is nothing to do after the function returns except return its value.

iii) Binary recursion - Some recursive functions don't have just one call to themselves, they have two or more. Functions with two recursive calls are referred as binary recursive functions.

iv) Indirect recursion - A recursive function doesn't necessarily need to call itself. Some recursive functions work in pairs. Function A calls B to call A.

Arrays:

- Array is a linear collection of similar elements.
- Array is also known as subscript variable.
- Creating array is creating a group of variables & all its elements are accessed via single name & different subscript values.

Array declaration-

- To declare an array in C, a programmer specifies the type of the elements & the number of elements required by an array.

<datatype> arrayname[arraysize];

- This is called single dimension array. The arraysize must be an integer constant greater than zero & type can be any valid C datatype.

For example:

```
int a[5];
float k[4];
```

Initializing array-

```
int a[5] = {34, 58, 72, 93, 12};
int a[5] = {34, 56};
int a[ ] = {34, 58, 72, 93, 12};
int a[5];
```

- Initializing an array with lesser values than its size is allowed & the remaining array blocks will contain zero. Array can also be initialized during declaration when its size is not mentioned. In declaration 5 is the size of an array or we can say an array is a collection of 5 variables here.

Program to calculate average of 10 nos.

```
#include<stdio.h>
#include<conio.h>
int main()
{
 int i, marks[10], sum=0;
 float avg;
printf("Enter 10 nos");
for(i=0; i<=9; i++)
scanf("%d", &marks[i]);
for(i=0; i<=9; i++)
sum = sum + marks[i];
avg = sum/10.0;
printf("Average is %f", avg);
getch();
return(0);
}
```

Declaration of arrays -

int a[ ]; error

int a[5];

int a[5] = {9,6,8,0,2};

int a[ ] = {9,6,8,0,2}

int a[5] = {9,6,8,0,2,4,7}; error

int a[2] = {9,6};

Two dimensional arrays:

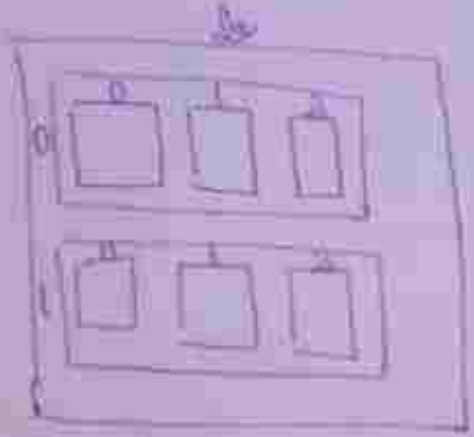
→ C programming language allows multidimensional arrays. The general form of a multidimensional array declaration is:

<data type> arrayname [size1][size2]...[sizeN];

→ The simplest form of the multidimensional array is the two dimensional array.

int a[6];

int b[2][3];



Initialization of two dimensional arrays:

int b[2][3]={12,65,78,45,33,21};

int b[ ][ ]={12,65,78,45,33,21};

int b[2][ ]={12,65,78,45,33,21};

int b[ ][ ]={12,65,78,45,33,21};

→ Last two styles are invalid & lead to compile time error.

```
#include<stdio.h>
#include<conio.h>
main()
{
int a[3][3],b[3][3],c[3][3],i,j;
clrscr();
printf("Enter 9 numbers for 1st matrix");
for(i=0;i<=2;i++)
 for(j=0;j<=2;j++)
 scanf("%d",&a[i][j]);
printf("Enter 9 numbers for 2nd matrix");
for(i=0;i<=2;i++)
 for(j=0;j<=2;j++)
 scanf("%d",&b[i][j]);
for(i=0;i<=2;i++)
 for(j=0;j<=2;j++)
 {
 c[i][j]=a[i][j]+b[i][j];
 printf("%d",c[i][j]);
 }
 printf("\n");
}
getch();
}
```

Strings in C :-

> The string in C programming language is managed in a one-dimensional array of characters. String is a sequence of characters terminated by a null character '\0'. Null character is a non-printing symbol with ASCII code 0.

Eg:

```
char city[8] = {'R','O','U','R','K','E','L','A','\0'};
```

or

```
char city[8] = "Rourkela";
```

```
#include<stdio.h>
#include<conio.h>
main()
{
char s[8] = {'S','N','D','I','A','\0'};
int i;
clrscr();
for(i=0; s[i]!='\0'; i++)
printf("%c",s[i]);

getch();
}
```

can be written as "SNDIA";

can be written as

```
printf("%s",s);
```

or

```
puts(s);
```

```
#include<stdio.h>
#include<conio.h>
main()
{
char a[20];
int i;
clrscr();
printf("Enter your name");
scanf("%s", a);  //&a[0]
for(i=0; a[i]!='\0'; i++)
printf("%c", a[i]);
getch();
}
```

OR

```
printf("%s", &a[0]);
OR
puts(a);
OR
puts(&a[0]);
```

→ We have seen when space is a delimiter in string and we want the entire space then we have to write gets(a);

```
gets(a);
```

```
#include<conio.h>
int main()
{
char text[20];
clrscr();
printf("Enter your name");
scanf("%s",&text[0]);
printf("%s",&text[0]);
return(0);
}
```

O/P:
Enter your name: Amit
Amit

→
```
#include<stdio.h>
#include<conio.h>
int main()
{
char text[20];
clrscr();
printf("Enter your name");
gets(&text[0]);
printf("%s",&text[0]);
return(0);
}
```

O/P:
Enter your name: Amit Gupta
Amit Gupta

→ There are several predefined functions declared in string.h header file.

- strlen() ⇒ prototype: int strlen(char);
  It calculates the length of the given string.
- strrev() ⇒ prototype: char* strrev(char);
  It is used to reverse the given string.
- strlwr() ⇒ prototype: char strlwr(char);
  It converts the given string into its corresponding lower case.

— strupr() → prototype: char strupr(char);
It converts the given string into its corresponding upper case.

— strcpy() → prototype: char strcpy(char, char);
It copies the second string into first.

— strcmp() → prototype: int strcmp(char, char);
It used to compare two strings & it returns 0 when string is correct otherwise not

Ex. - strcmp("AMAR", "AMIT");

— strcat() → prototype: char strcat(char, char);
It simply concatenates two strings.
Ex: strcat(s, "student");

Structure:

— It is a collection of heterogeneous data. Just like an array it is also a group of variables but can be of dissimilar types. Using structure we can create data types known as user defined data types. Such data types are also known as user-defined data types.

Defining a structure:

```
struct <tag>
{
type variable name;
type variable name;
.....
};
```

- Struct is a keyword. tag is any choice it becomes the name of user defined datatype. variables declaring inside structure body are called member variable. (Remember to put semicolon at the end of structure body)

unions:

- union is a keyword in c language. The syntax of union is very similar to structure. It is also used to create custom data type. A union is a special datatype available in C that allows storing different data types in the same memory location.

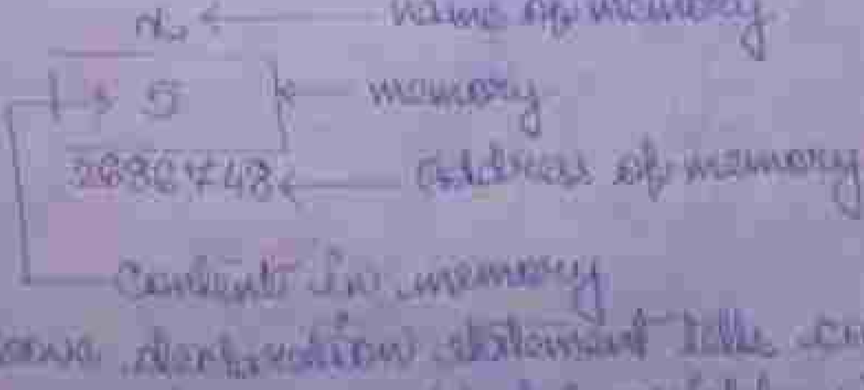
Syntax:

```
union<tag>
{
Type variablename;
Type variablename;
-----
};
```

pointers:

- pointer is a variable which is used to hold address of another variable. for example

int n=5;



- Here, the above declaration statement tells compiler about the name of memory block (variable n), amount of memory occupied by that block & type of content.

- Every byte has a logical reference number known as address of variable. In above example address of x is 1038442 though this address number could be anything & unpredictable to us, it is always in the range 0 to 4294967297.
- Addresses are also known as reference. Address number is always an integer, even if the variable is not of type int.

Address operators:-

& is called address of operator, also known as referencing operator. It is a unary operator, thus requires only one operand, & returns the reference number of variable specified as the operand.

Declaration of pointer variable

However we cannot assign anything to &x, we can assign address of x (&x) into some other variable. For example, if j is variable then we can write j = &x; // valid statement

- This is the correct way as variable is in the left side of assignment operator.

int * j;

- Asterisk symbol before j in the declaration statement tells the compiler that this is a special variable meant only to store address of another variable. This special variable is called pointer.

Size of pointer:

Since pointer contains only address of another variable, they consume 4 bytes in memory.

```
#include<stdio.h>
#include<conio.h>
int main()
{
int i=5, *j; clrscr();
j=&i;
clrscr();
printf("\n%u",&i);
printf("\n%u",&j);
printf("\n%u",&j);
printf("\n%d",*&j);
printf("\n%d",i);
printf("\n%d",*(&i));
printf("\n%d",*j);
getch(0);
}
```

o/p

```
2686748
2686748
2686744
2686748

5
5
5
Process terminated with exit code 0
Press any key to continue.
```

Pointer & arithmetic:

1) Rule 1 - Take care for the compatibility

Example -

```
int main()
{
int i=3; int k, i=3, *j;
char *j; j = &i;
j=&i; k = *j + *j;
    printf("%d", k);
    return(0);
    }
```

O/p 6

Rule 2 — we cannot add two addresses

Example -

```
int main()
{
int *k, i=3, *j;
j = &i;
k = j + j;
printf("%d", k);
return(0);
}
```

we cannot add, multiply, divide two address
divide

Rule 3 - we cannot multiply scalar value to an address

```
int main()
{
int *k, i=3, *j;
j = &i;
k = j*5;
printf("%d", k);
return(0);
}
```

Pointers :- Pointing to the pointer gives address of the immediately next block of the same type.

```
int main()
{
int *ptr, i = 3, *j;
j = &i;
ptr = j + 1;
printf("%d", ptr); return(0);
}
```

Call by reference:

When a function is called by passing addresses of variables, it is known as call by reference. Function can only access its own memory & cannot access variables of other functions, but if we pass address of variables during function call, we actually give power to the called function to access variables of calling function via address of variables.

Scope of variables in C:

Scope generally refers to the visibility of variables. Which part of a program can be seen or use it. There are mainly two types of scope of variables:

a) Local variables : It can be accessed from that part of a program where it is declared.

b) Global variables : It can be accessed from any part of the program.

Storage classes in C

There are four storage classes in C.

a) Auto - It is used for a local variable defined within a block.

b) register - It is used to store the variable in the registers.

c) Static - It is a local variable which is capable of returning a value even when control is transferred to the function call.

d) extern - It is a global variable.

- Storage class represents the visibility & a location of a variable.