A LECTURE NOTE
ON
TH.1B – COMPUTER APPLICATION
SEMESTER -1

Prepared by – Miss Dipa Biswas

Guest Faculty

Mechanical Engineering

GOVT. POLYTECHNIC, MALKANGIRI

UNIT - 1

Computere Organisation

Introduction to Computer

A computer is an electronic device that is designed to accept data, perform the required mathematical and logical operations at high speed and provides the output result.

* Charles Babbage is called the Father of computer.
* The first mechanical computer designed by Charles Babbage was called Analytical Engine.

Use of computer in various field

The uses of computer in various field are

> Education, Communication, Banking, Medical, Scientific Research, Entertainment, Government, Business, Designing and many more.

Evolution of Computers

Introduction

Computer evolution refers to the change in computer technology right from the time computers were first used to their current state.

Evolution of computers

→ Analytical Engine
→ Hollerith's tabulatore
→ Vacuum tubes
→ Transistors
→ Integrated circuits
→ VLSI (Microprocessor)
→ AI.

Generation of computers

The word generation means improvement on the product development process.
There are five generations of computer

* First generation (1940 - 1956) - Vacuum tube
* Second generation (1956 - 1963): Transistors
* Third generation (1964 - 1970): ICs
* Fourth generation (1971 to present) - VLSI (Microprocessor)
* Fifth generation (present and beyond): AI (Artificial Intelligence)

First Generation (1940 - 1956): Vacuum Tubes

First generation of computer started with using vacuum tubes as the basic components for memory and circuitry for CPU.

⇒ These computers were so big in size that they often required an entire room to be installed.

⇒ These computers were very expensive, hence it is mainly used for scientific purposes.

⇒ In addition to large space, they also needs a lots of electricity.

⇒ Machine level programming language were used in this generations.

Example - EDVAC, EDSAC

Disadvantages

→ Large in size
→ Difficult to program
→ Generates more heat
→ High power consuption
→ Expensive.

Second generation (1956 - 1963): Transistors

⇒ This generation used transistors.
⇒ In this generation magnetic cores were used as primary memory and magnetic tape and magnetic disks as secondary storage devices.
⇒ These computers were smaller, faster, cheaper and more energy-efficient and reliable than ~~their~~ 1st generation computers
⇒ Assembly language was used as programming language in this generation.
⇒ Examples

Disadvantages

⇒ Expensive

⇒ Limited to special purpose tasks

Third generation (1964 - 1970) - ICs

IC - It stands for integrated circuit

⇒ This computer used ICs for their internal circuit design.

⇒ Several electronic components such as transistors, resistors and capacitors were miniaturised and placed on silicon chips called integrated circuit.

⇒ ICs are smaller, faster, reliable and cheaper.

⇒ FORTRAN and PASCAL like high level language were used as programming language.

Ex: IBM-360, B6500

Disadvantage

⇒ Limited storage capacity

Fourth generation (1971 - present) - VLSI

⇒ These computers used very large scale integrated (VLSI) circuit.

⇒ GUI was used as programming language

Disadvantage

→ Difficult to manufacture

Fifth Generation (present and Beyond):

Artificial Intelligence:

Fifth generation computers are based on AI Artificial Intelligence s/w and parallel processing h/w.

→ In this generation, the VLSI technology became VLSI (Ultra large Scale Integration) technology

Ex - Robot

Disadvantage

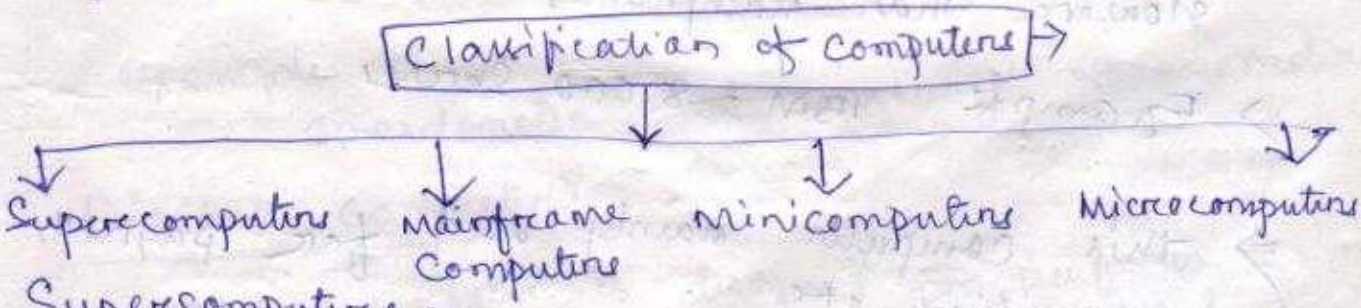
Lack of human-like intelligence

Classification of computers:

Computers can be broadly classified into four categories based on their speed, storage capacity and price.

The categories are as follows based on size and capacity.

* Supercomputers
* Mainframe computers
* Minicomputers
* Microcomputers



Supercomputers:

Supercomputers were first developed in 1980s to process large amounts of data and

⇒ A Single Supercomputer can Support thousands of users at the same time.

⇒ Supercomputers are mainly used for weather forecasting, nuclear energy research, aircraft design, automotive design, online banking, controlling industrial units.

Ex- CRAY-1, CRAY-2, CYBER 205-

Mainframe Computers

⇒ Mainframe computers are large-scale computers but smaller than Supercomputers

⇒ they are very expensive and need a very large clean room with AC.

Example

IBM S/390

⇒ This computer can support 50,000 users at the same time.

Minicomputers

⇒ Minicomputers are smaller, cheaper and slower than mainframes.

⇒ Example IBM - 8000 Series

⇒ This computer mainly used for payroll accounting etc

Microcomputers

⇒ Microcomputers commonly known as PCs, and they are very small and cheap.

⇒ This computer mainly used in school, home and office.

⇒ Example

BBC, Acorn, Apple-II

Based on operating principles

⇒ Analog computers

⇒ Digital computers

⇒ Hybrid computers

Analog Computer:

Analog computers are designed to process the analog data.

⇒ Analog computers measure the continuous changes in physical quantity and provides output as a reading on a dial or scale.

Example -
Speedometer and mercury thermometer

Digital computer

Digital computer is designed to perform Calculations and logical operations at high speed.

⇒ It accepts the digits and numbers as data

Example laptops and desktops.

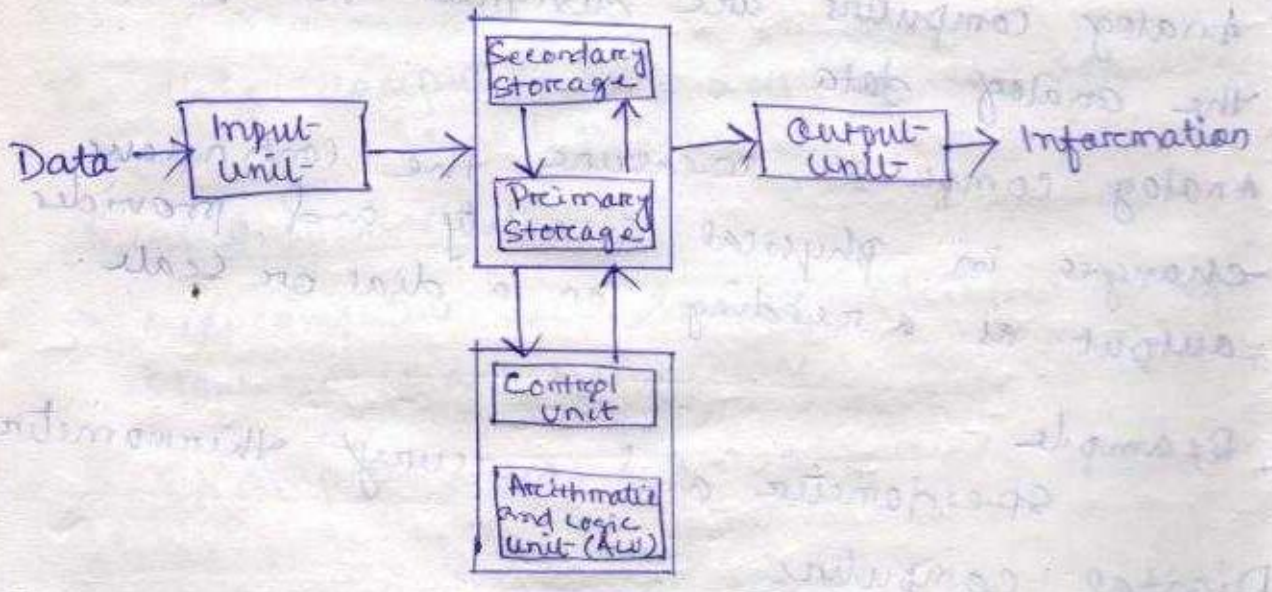
Hybrid Computers

Hybrid computer has features of both analog and ~~digital~~ digital computer.

Basic Organisation of a Computer
(Functional Block Diagram)

Computer performs five major operations

- ⇒ Accepting data or instructions as input
- ⇒ Storage of data
- ⇒ processing data
- ⇒ Displaying result (output)
- ⇒ Controlling and coordinating all operations inside a computer.



Computer interact with each unit with other units to perform the operation

Input unit-

→ Computing needs to receive data and instruction in order to solve any problem.

→ The data and instructions can be entered by using different input- devices such as keyboard, mouse, scanner, trackball etc.

→ As the computer understand only binary language, so the input- devices convert- the entered input- data into binary codes.

Central processing unit :

The control unit (CU) and Arithmatic logic unit- (ALU) of the computer are together known as the central processing unit- (CPU). The CPU is like brain which performs the following functions.

→ It performs all calculations

→ It takes all decisions

→ It controls all units of the computers.

Arithmatic Logic Unit- (ALU)

→ All calculations are performed in the Arithmatic logic Unit- of the computer.

→ The ALU performs basic operations such as addition, Subtraction, Multiplication, division and does logic operations using >, <, = etc.

Control unit

→ It controls all other units in the computer.

⇒ The control unit instructs the input unit to receive the data from the user and store it and does step by step processing of all operations.

Storage unit:

⇒ The storage unit of the computer holds data and instructions for processing.

Storage devices are of two types.

⇒ Primary Storage

⇒ Secondary storage

Primary storage

⇒ This storage is also known as the main memory.

⇒ Main memory is directly accessible by the CPU at very high speed

⇒ It is used to store the data and parts of program, intermediate result and recently generated result of the jobs that are currently being worked on by the computer.

⇒ Primary storage space is very expensive and limited storage capacity.

⇒ Main memory is volatile in nature, as soon as the computer is switched off, the information get erased.

Ex - RAM

⇒ Due to volatile nature, the primary memory is not used as permanent storage

Secondary Storage :

⇒ The secondary memory is also known as auxilliary memory.

⇒ This memory is cheaper and non-volatile. That's why used for permanent storage of data and information.

⇒ Ex - Hard disk, CD, DVD, pendrive etc

Processing

The process of performing operations on the data as per the instruction specified by the user is called processing.

⇒ Data and instructions are taken from the primary memory and transferred to the arithmatic and logic unit to perform all sorts of calculation. After completion of processing the final result is displayed using output unit.

Output unit

output is the process of giving the result of data processing. The output result are given through monitor and printer etc.

⇒ As the computer accepts data only in binary form, the result also comes in binary form which user cannot understand. So the output devices, convert the binary result into human readable language.

Input Devices:

Input devices are used to give input data/ instructions to the computer.

Input devices are:

- Keyboard
- Mouse
- Lightpen
- ~~Magnetic Tape~~ Joystick
- optical device such as Barcode reader, scanners, OCR, OMR and MICR.

Keyboard

The keyboard is the main input- device for computer.

- Basically keyboards have 80 to 110 keys and keys are categories into the following groups

Typing key: These includes the letters of alphabet

Numeric key: These include a set of 17 keys which contains number key from 0 to 9 /, *, - and +.

Fandion Keys, control key, arrow key, HOME, End, page up, page down, insert and delete, Ctrl, Alt, Esc, Print-Screen.

Advantage

- The keyboard is easy to use and cheap.

Mouse

It is an input device used in a GUI

⇒ when the mouse is moved on any surface the pointer on the screen is also moved.

⇒ The mouse has two buttons called left button and right button and a scroll.

⇒ With the help of MOUSE, user performs the following operations

- ⇒ point-
- ⇒ click
- ⇒ drag
- ⇒ scroll.

There are different types of MOUSE available

- ⇒ Mechanical mouse
- ⇒ optical "
- ⇒ cordless "

Joystick

It is a cursor control device, mainly used in computer games and computer aided design (CAD) / computer aided manufacturing (CAM) applications

⇒ It consists of a handheld lever that pivots on one end and transmits its coordinates to a computer

⇒ It has one or more push buttons called switches, whose position can also be

optical devices

Barcode Reader

It is used to capture and read information stored in a barcode.

⇒ It consists of a scanner, a decoder and a cable used to connect the reader to a computer.

⇒ The function of the barcode reader is to capture and translate the barcode into numerals and/or alphabets.

⇒ It is connected to a computer for further processing of the captured information.

Advantage

Barcode readers are cheap, portable handy and easy to use.

Image Scanner

A scanner is a device that captures images printed text and handwritting from different sources such as photographic prints, posters and magazines and converts them into digital images.

⇒ The basic types of image scanners are flatbed, hand, film and drum scanner.

⇒ The scanner used in colleges and offices are flatbed scanner.

⇒ A hand image scanner has to be manually moved across the object or image to scan.

⇒ Film scanners are usually used in photography and slides.

Optical Character Recognition:

Optical character recognition (OCR) is the process of converting printed materials into text or word processing files that can be easily edited and stored.

⇒ OCR scan the text character by character.

⇒ Analysing the scanned image to translate the character images into character codes. (e.g. ASCII)

⇒ It is commonly used to recognize text in scanned documents or images.

OMR (Optical Mark Recognition)

⇒ OMR is the process of electronically extracting data from marked fields, such as checkbox and fill-in field.

⇒ The optical mark reader is fed with an OMR sheet that pen or pencil marks in pre-defined positions to indicate each selected response.

Ex - MCQ - pattern for entrance

⇒ The OMR sheet is scanned by the reader to detect the presence of a mark by measuring the reflected light levels.

⇒ The dark or the marked areas reflect less light than the unmarked ones.

⇒ The OM reader interprets this pattern marks and spaces and stores the interpreted data in a computer for storage, analysis and reporting.

MICR: Magnetic Ink Character Reader:

It is used to verify the legitimacy of paper document, especially bank (cheques).

⇒ It consists of magnetic ink printed characters that can be recognized by high speed magnetic recognition devices.

⇒ The printed characters provide important information such as cheque number, bank routing number, checking account number for processing to the receiving party.

⇒ Ex - Bank cheque

Routing number, Account number, cheque number.

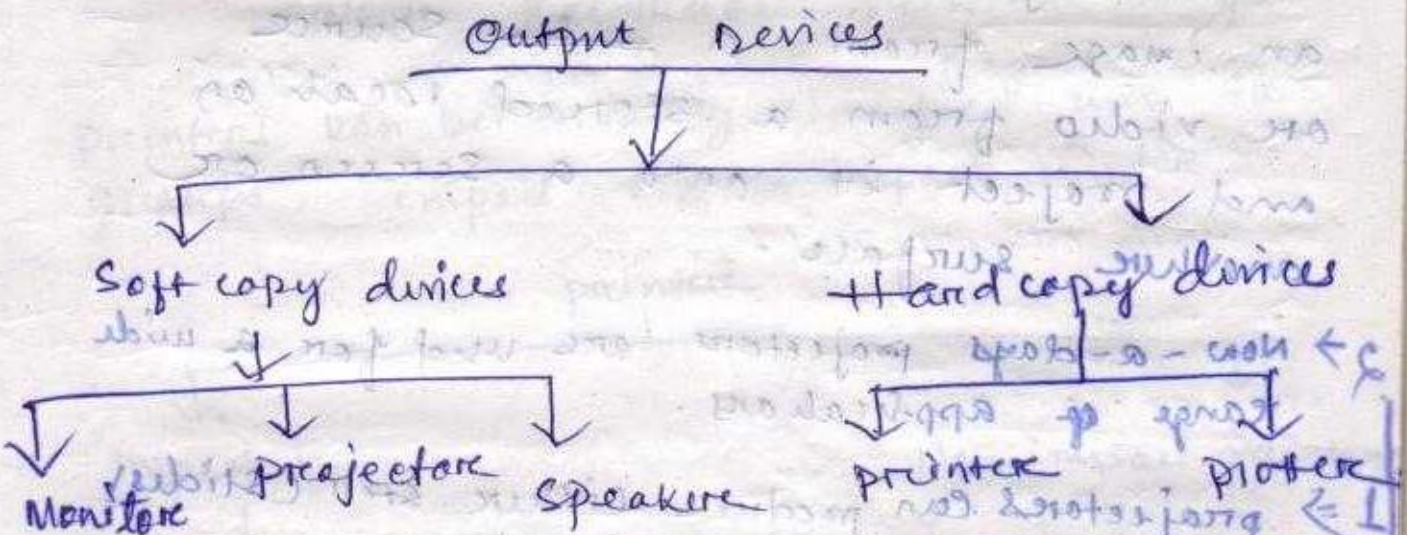
⇒ These three sets of numbers act as

the person who signs it.

⇒ MICR use can enhance security and minimize the losses caused by some types of fraud.

Output Devices :

Output devices produce output in hard-copy as well as soft copy by using hard copy devices and soft copy devices.



Soft copy Devices

The soft copy output devices produces output in electronic version.

⇒ A file that is stored on a hard disk, CD, or pen drive and is displayed on the computer screen such as monitors, projectors and audio comes from speakers.

Monitors :

The monitors is a soft copy output-device used to display video and graphics

=> The monitor is connected to either the VGA or the digital video interface (DVI) port.

=> Monitors comes in ~~three~~ four variants - CRT (cathode ray tube), LCD (liquid crystal display, ~~and~~ plasma and LED (Light emitting diode)

Projector:

A projector is a device that takes an image from a ~~video~~ source or video from a stored location and project it onto a screen or another surface.

2 => Now-a-days projectors are used for a wide range of applications.

1 => projectors can produce either still (slides) or moving images (videos)

=> It can be used to project power point presentation.

=> It can be used as computer screen to teach in classroom.

=> It can be used to play movie in a largize screen. etc

projectors can be broadly classified into two categories depending on the technology:

Hardcopy output Devices

Hardcopy output devices produce a physical form of output

Ex - printers, plotters

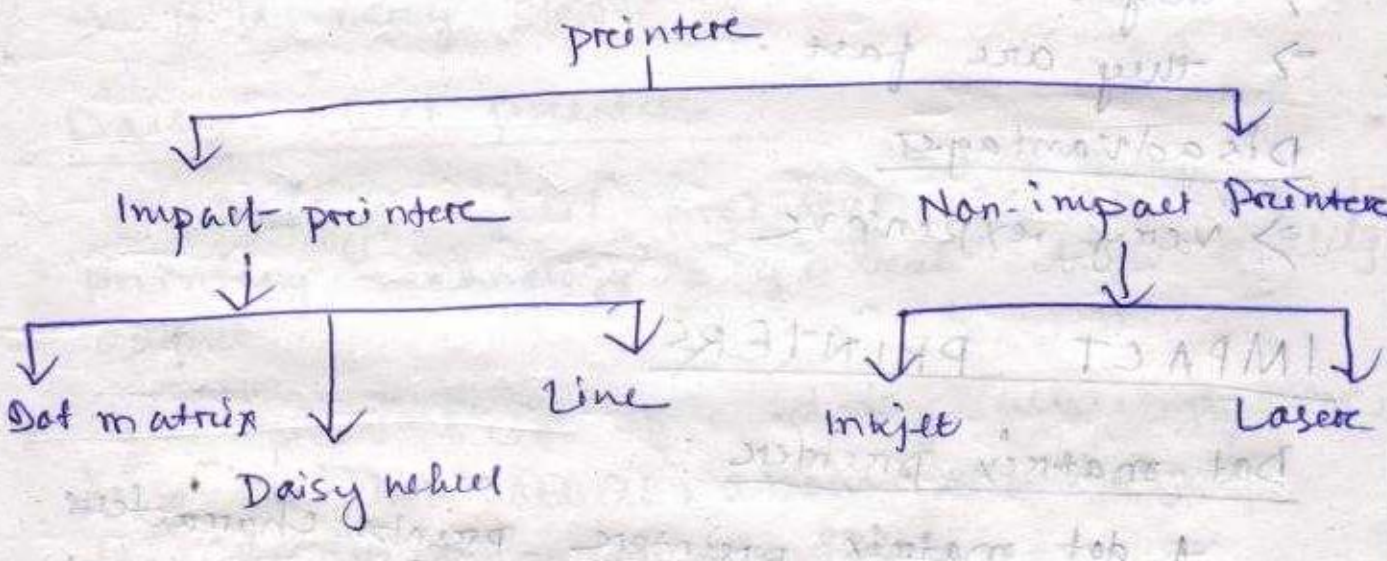
Printers

A printer is a device that takes the text and graphics information obtained from a computer and print it on to a paper.

-> Every printer have following qualities

colour, Resolution, Speed, memory

Printers can be broadly classified into two groups: impact and non-impact printers



Impact Printers

These printers print characters by striking an inked ribbon against the paper.

Ex - Dot matrix, daisy wheel and line printer

Advantage

These printers enable the users to produce

Disadvantages

⇒ Impact printers are slow

⇒ extremely noisy

Non - impact printer:

Non - impact printers are ~~much~~ produced less noise than impact printer, as their printing heads do not strike the paper. The main types of non - impact printers are Inkjet, Laser and thermal printers.

Advantage:

→ Non - impact printers produce prints of good quality.

→ they are noiseless

→ they are fast.

Disadvantages

⇒ very expensive

IMPACT PRINTERS

Dot matrix printer:

A dot matrix printer prints characters and images of all types as a pattern of dots.

→ This printer has a print head called hammer that consists of pins representing the character are image

⇒ the print head runs back and forth or in an up and down motion on the

ink - soaked cloth ribbon against the paper.

Advantages

=> the dot matrix printer can produce carbon copies.

=> The printing cost is low.

=> It is used for Bulk printing.

Disadvantages

=> It creates a lots of noise when the pins strike the ribbon against the paper.

=> print- quality is poor.

=> It is very slow.

Daisy wheel printer:

Daisy wheel printers use an impact- printing technology to generate high quality output.

=> The print- head of a daisy wheel printer is a circular wheel, about 3 inches in diameter with arms are spokes.

=> The shape of the printer wheel resembles the petals of a daisy flower, thats why the name is like so.

=> The characters are embossed at the outer ends of the arms.

=> To print a character, the wheel is

to be printed is positioned just in front of the printere ribbon.

⇒ The spoke containing the required character is then hit by a hammere by striking the ribbon to leave an impression on the paper placed behind the ribbon.

⇒ The movement of all these parts is controlled by a microprocessore in the printere.

Advantage

⇒ print quality is high.

Line printere

A line printere is a high speed impact printere in which one typed line is printed at a time.

⇒ The speed of a line printere usually varies from 600 to 1200 lines per minute approximately 10-20 pages per minute.

⇒ Due to high speed line printer is used in data centers and in industrial environment.

NON- IMPACT PRINTER:

Inkjet printers: The print-head of inkjet-printers has several tiny nozzles. also called jets

⇒ As the paper moves past the print-head, the nozzles spray ink onto it, forming characters and images

⇒ To create coloured image, the dots can have different colours combined together.

⇒ inkjet printers are cheaper.

Laser printer:

⇒ this printer works at very high speeds and produces high-quality text and graphics

⇒ When a document is sent to the printer, the following steps take place.

* A laser beam draws the document on a drum (which is coated with a photo conductive material) using electrical charges.

* After the drum is charged, it is rolled in a toner

* the toner sticks to the charged image on the drum.

* the toner is transferred onto a piece of paper and fused to the paper with heat and pressure.
* After the document is printed, the electrical charge is removed from the drum and the excess toner is collected.

Plotters

A plotter is a printing device that is usually used to print vector graphics with high print quality.

→ It is widely used to ~~print vector~~ draw maps, scientific application, CAD, CAM and computer aided engineering.

→ Architech use plotter to draw blueprints of the structure they are working on.

→ There are two different types of plotter
- drum and flatbed.

Computure memory.

What is computure memory.

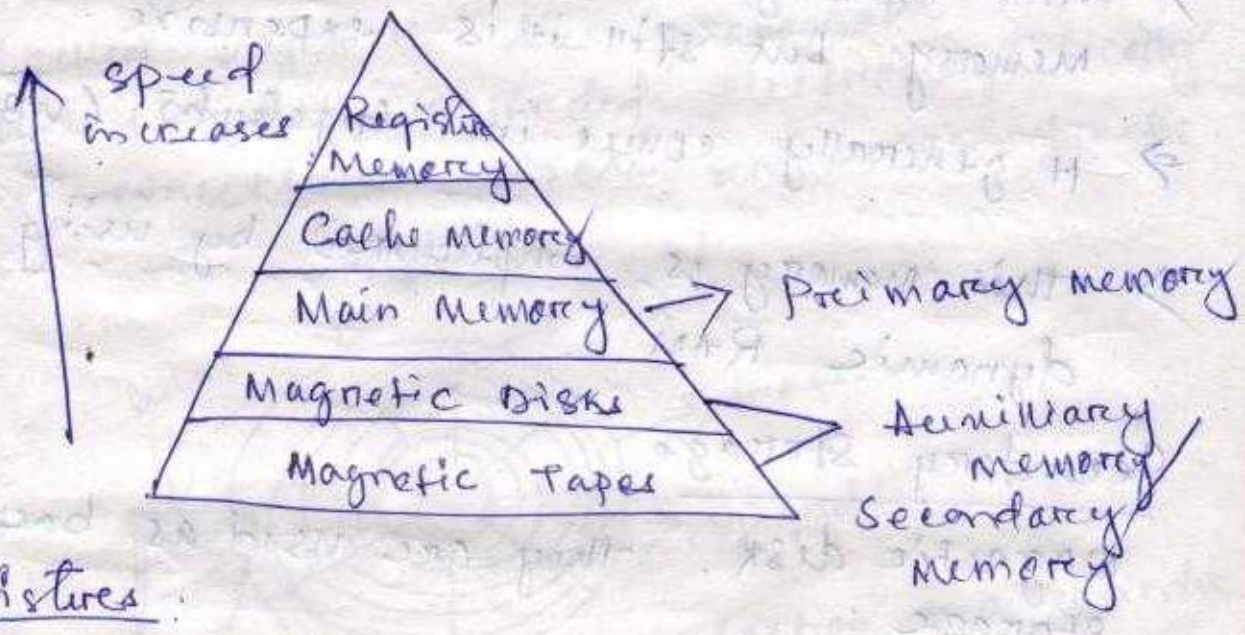
Memory is an internal storage area in the computute used to store data and progreams either temporarily or permanently.

-> computure memory classified into two greoups.

primary memory
Secondary memory

Memory Hierarchy:

Memory hierarchy helps in organising the memory, so that it can actually minimize the access time.



Registures.

The registures are present inside the CPU, As they are present inside the CPU they have least access time.

⇒ Generally it comes in kilobytes.
⇒ It is implemented using flip-flop.

Cache Memory:

Cache memory is used to store the segments of a program that are frequently accessed by the processor.

⇒ It is expensive and smaller storage capacity.
⇒ It generally comes in megabytes (MB)

Primary Memory or Main Memory:

It directly communicates with the CPU and with auxiliary memory devices through an I/O processor.

⇒ Main memory is less expensive than cache memory but still it is expensive.
⇒ It generally comes in Gigabytes (GB)
⇒ This memory is implemented by using dynamic RAM.

Secondary Storage:

Magnetic disk. They are used as backup storage.

⇒ They are cheaper than Main memory and larger in storage capacity.
⇒ Generally it comes in Terabyte (TB)

Magnetic Tape

They are used to store removable files

=> They are cheapest -

=> Generally comes in Terabyte (TB).

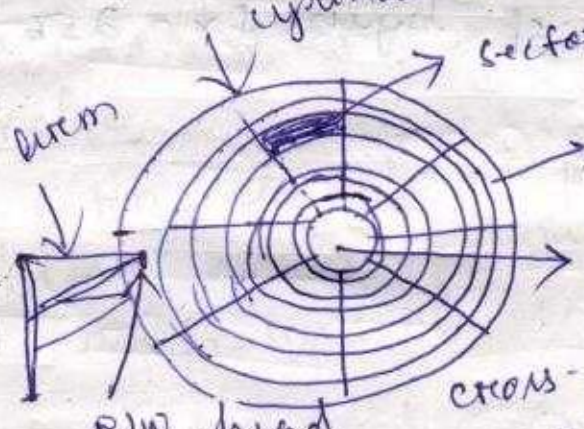
=> The Secondary storage is not directly accessible by the CPU

=> The Secondary storage devices hold data even when the computer is switched off.

Ex - Magnetic disk, magnetic tape, hard disk

Hard Disk

A hard disk is basically a set of disks, stacked together like phonograph records, that has data recorded electromagnetically in concentric circle known as tracks.



Cross-section of a cylinder

A Single hard disk includes several platters (disks) that are covered with a magnetic recording medium.

Each platter requires two read/write (R/W) heads, one for each side.

=> All the R/W heads are attached to a single access arm and so they cannot move independently.

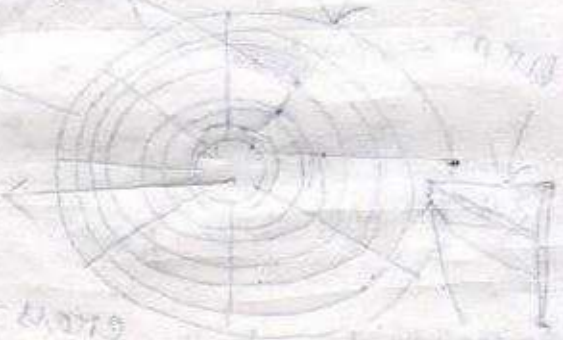
=> The R/W head can move back and forth over the platters to read or write data on them.

=> Data is actually stored on the surface of a platter in sectors and tracks.

=> Tracks are concentric

=> Sectors are pie shaped area on a track.

=> Track is divided into a number of segments called sector, which can store a fixed no. of bytes Ex - 256 or 51



Difference between compiler & Interpreter

Compiler

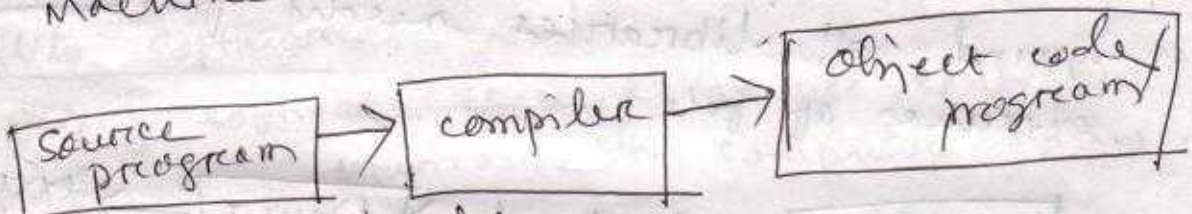
It is a translator program, that converts high level language to low level language i.e machine level language

=> A compiler is more intelligent than an assembler, it checks all kinds of limits, ranges ~~and~~ errores etc

-> compiler program run time is more

-> it occupies a large part of memory

=> it is slow because, compiler goes through the entire program and then translates the entire program into machine codes



Ex - C, c++, C# etc

Interpreter

=> An interpreter is a program that translates a programming language into comprehensible language.

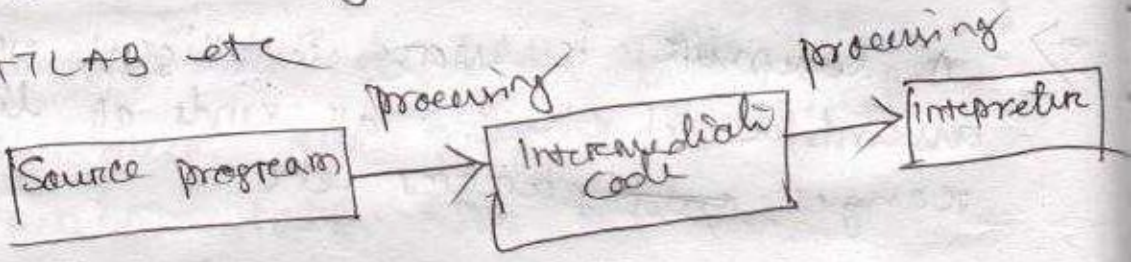
=> It translate only one statement of the program at a time

=> It read one line at a time and

⇒ It is less efficient

⇒ Interpreter takes less time for analyzing the source code

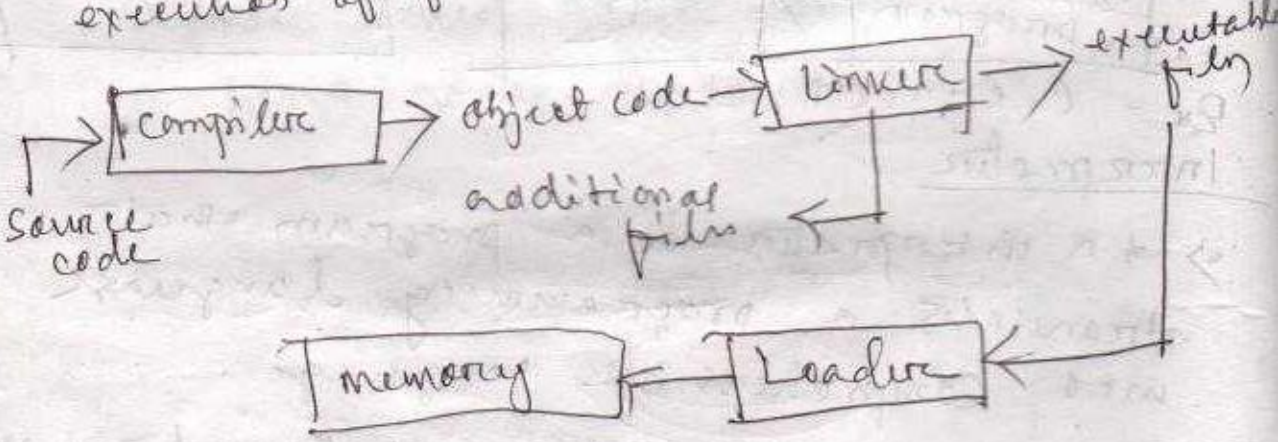
Ex - Python, Ruby, Perl, SNOBOL MATLAB etc

⇒ 

Linker

* Linker is a special program that combines the object file, generated by compiler and other piece of code to generate an executable file that has .exe extension.

⇒ In object file linker searches and append all libraries needed for execution of file.



Loader

It is a special program, which takes input of executable file from linker, loads it to main memory and prepares this code for execution by computer

⇒ Loader allocate memory space to program

There are various loaders

* Absolute loader
* Relocating "
* Direct linking
* Bootstrap

⇒ In embedded computer systems don't have loader, so the code is executed through ROM.

Utility Software:

Utility software is used analyse, configure, optimize and maintain the computer systems.

Example of utility program:

Disk defragmenters, Disk checkers, Disk cleaner, Disk space analysers, Disk partitions, File managers and backup utilities. etc etc.

UNIT-II

Types of operating System:

These are the types of operating System:

* Batch processing
* Single-user Single tasking
* Multitasking.
* Time-Sharing operating System
* Distributed OS
* Network OS
* Real time OS.

Batch processing:

In this OS data and programs that need to be processed are bundled and collected as a batch and executed together.

=> In this OS, there is very limited or no interaction between user and processor during the execution of work

=> This OS performs very well, when a large amount of data has to be processed

=> Batch processing used in payroll System.

~~Single user Single tasking~~

The process need identify each employee, calculate their monthly salary and print the corresponding pay slip

Single-user Single tasking OS

- this OS allows only one program to execute at a time.

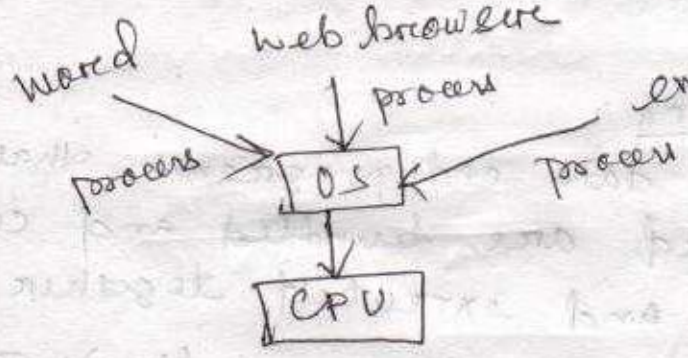
-> It is designed to allow a single user to do a single job effectively.

~~Ex - palmtop~~

Multitasking:

⇒ This OS allows multiple applications to run simultaneously.

⇒ This OS usually used in our desktop



⇒ processes holds common processing resource such as CPU.

⇒ In this O.S multiple program run simultaneously without interfering with one another.

Ex- MS windows

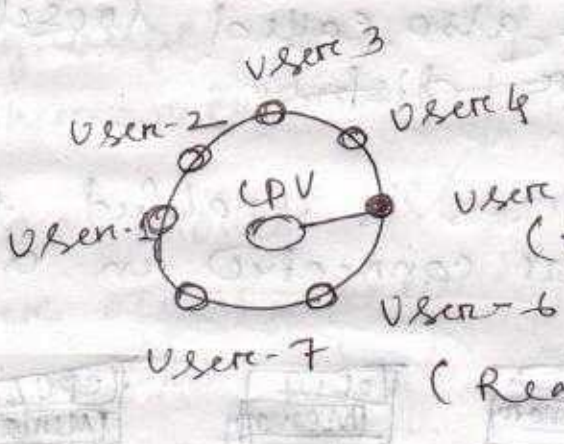
Time sharing OS

→ A time shared operating system allows multiple users to share computer simultaneously.

⇒ The CPU performs many tasks by ... so frequent that the

User can interact with each program while it is running.

⇒ A time shared OS uses CPU scheduling and multi-programming to provide each user with a small portion of a shared computer at once.



In the above fig, User-5 is in active state but User-1, User-2, User-3, User-4 User-7 is in waiting state whereas User-6 is in ready state

Active state — program is under processing

Ready state — the program is ready to execute but waiting

Distributed OS

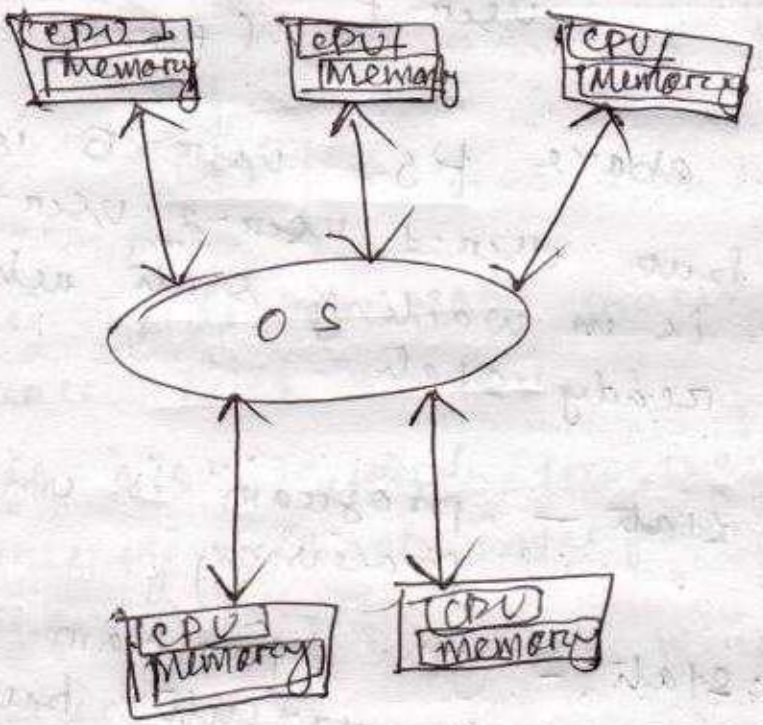
This OS is advanced OS in the computer technology.

-> By using this OS, several interconnect computers communicate with each other over a communication n/w.

=> Independent system has its own memory unit and CPU.

=> These are also called loosely coupled systems. ~~or distributed~~

=> Remote access is enabled within the devices connected in a network.



=> In the system failure of one system will not affect other system in the network, as all systems are independent from each other.

to the network.

⇒ Load reduced in the host computer

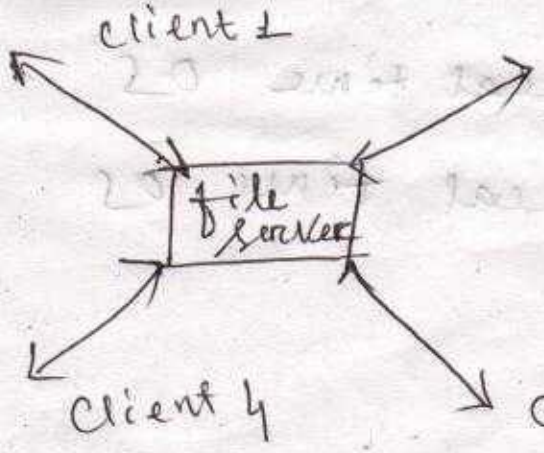
Ex - LOCUS

Network OS

⇒ This OS runs on a servere and provide the capability to manage data, users, groups, security, applications and other networking functions.

⇒ This OS allows shared file servers. printere servers, security and application overe a small private n/w.

⇒ In this system, all the users are wer aware of the system configuration within the n/w, and also there own configuration, that why this is also called tightly coupled systems.



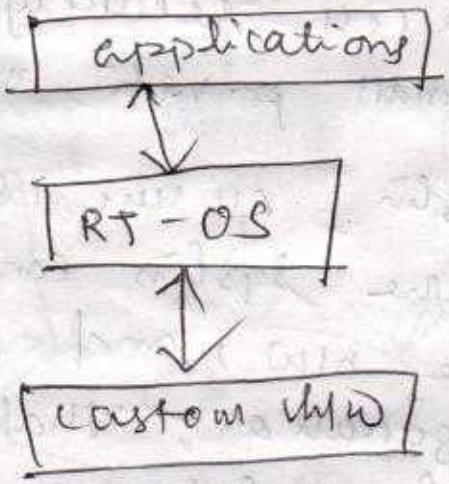
Ex - MS windows servere 2003,

Real time OS

⇒ the OS, ~~which~~ whose response time is very small is called real time OS

Response time - the time required to process and respond to the inputs

⇒ this OS used in the missile system, Air traffic control system, robot, Scientific experiments, etc



- there are two types of RTOS

⇒ Hard Real time OS

⇒ Soft real time OS

Introduction to computer Network:

A computer network is a collection of networking devices and computers that are connected with each other for the purpose of information and resource sharing among users.

→ ~~Each~~ Each device in the network is called node.

→ Computer n/w can be wired or wireless

Advantage of computer network

⇒ It facilitates sharing of files
⇒ It facilitates sharing of resources
⇒ It is cost efficient
⇒ sharing of load
⇒ It facilitates communication among devices in a network.

Protocol

Network protocol is a set of rule that is established to transmit data or to determine how the communication will be established among the devices.

Ex- TCP, IP, FTP, HTTP SMTP.

UNIT-6

Overview of C

⇒ C - programming language was developed by Dennis Ritchie at Bell Laboratories in 1972

Importance of C

⇒ programs written in C are efficient and fast

⇒ It is more faster than BASIC (Beginner's all purpose Symbolic Instruction code)

⇒ There are only 32 keywords in ANSI-C (American National Standard Institute) and its strength lies in its built-in-fun

⇒ C is highly portable - It can run on different O.S

Ex of C

```
main()
{ /* printing a/p */
printf("This is C");
}
```

⇒ The opening brace { is one beginning of one function and } brace is one end of one function

⇒ the lines beginning with /* and ending with */ are known as comment lines

⇒ Comment lines are not executable only for understanding

⇒ the printf is a predefined standard C function for printing o/p.

⇒ \n in the printf is - newline character.

⇒ All the C programs need a main().

Ex -

main() →

int main()

main(void) → the function does no argument

void main(void) → the function return nothing

int main(void)

Sample program

Adding two nos

```
#include <stdio.h>
main()
{
    int num1, num2;
    int sum;
    num1 = 10;
    num2 = 15;
    sum = num1 + num2;
    printf("the sum is : %d\n", sum);
}
```

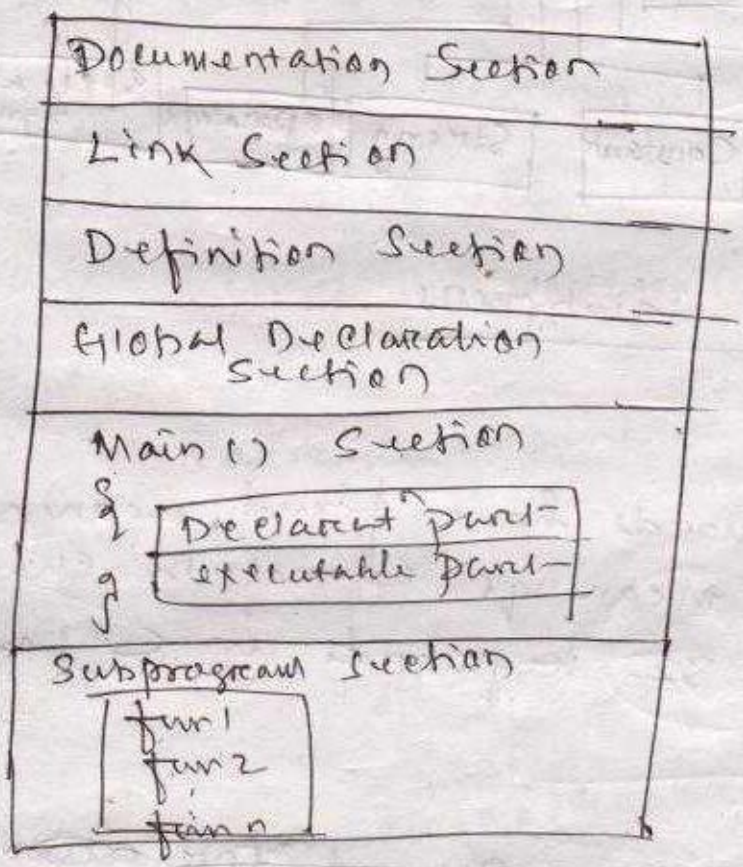
o/p - the sum is : 25

⇒ #include is preprocessor directives

Basic structure of C programs

=> C programs consists of group of building blocks called function.

=> A function is a subroutine that may include one or more statement to perform a specific task



=> The documentation section consists of set of comment lines.

=> The Link section provides instructions to the compiler to link function from the System library

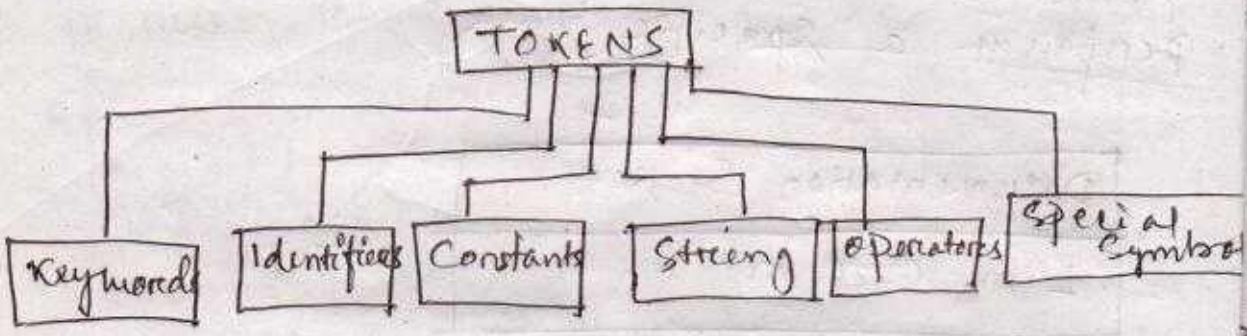
=> The definition section defines all symbolic constants.

=> The variable which is used in more than one function, such variables are global variable and declared in declaration section.

Overview of C

TOKENS

The smallest individual units in the C - program is known as C - token.



Keywords and Identifiers

Keywords

All the keywords have fixed meanings and these meanings cannot be change

⇒ There are 32 keywords in C - Langu

Identifiers

~~Identifiers are used for names of~~

Identifiers are the names of the variables, functions and arrays.

⇒ Identifiers are user - defined nam which consists of sequence of letters and digits.

Rules of Identifiers

⇒ First character must be an alphabet (or underscore).

⇒ Must consists of only letters, digits ⇒

Managing input and output operations
program takes data as i/p and process it
and display the result is known as
information

> C programs use scanf() to read
data from keyboard and printf() to
display result.

Reading and writing Single char:

> Reading a single character can be
done using getchar().

Syntax

variable-name = getchar();

Ex

char ch;
ch = getchar();

writing a character can be done
using putchar

Syntax:

putchar(variable-name);

Ex: answer = 'y';
putchar(answer);

Formatted input

Formatted input refers to input data, that
has been arranged in particular format

Points for Scanf

* Each variable to be read must have a field specification.
* For each field specification, there must be a variable address of proper type.
* Any non-whitespace character used in format string must have a matching character in the user input.
* Never end the format string with whitespace. It is a fatal error.
* the scanf reads until
  - i - it found a whitespace character as a numeric specification
  - ii - the maximum no of characters have been read.
  - iii - An error is detected
  - iv - The end of file is reached.

Formatted output

C-Language uses printf function for formatted output.

Syntax

printf ("control string", arg1, arg2...)

* control string contains the following
  - Characters to be printed on the screen as it is.
  - Format specifications for each item to be displayed.
  - Escape sequence chars such as \n, \t

C - Programs

Q-1

WAP using C - to calculate Sum of array elements using pointer

```
#include<stdio.h>
#include<conio.h>
void Main()
{
int numArray[10];
int i, Sum=0;
int *ptr;
printf("\n Enter 10 numbers :");
for(i=0; i<10 ; i++)
scanf("%d", &numArray[i]);
ptr = numArray; /* a= &a[0] */
for(i=0 ; i<10 ; i++)
{
  Sum = Sum + *ptr;
  ptr++;
}
printf("The Sum of array of
    numbers is :%d", Sum);
}
```

output

Enter 10 numbers :

10, 11, 12 13 14 15 16 17 18 19 = 114

Q. WAP in C to find greatest in 3

```
#include<stdio.h>
int main()
{
 int a,b,c;
 printf("\n Enter the value of a,b
 Scanf("%d %d %d", &a, &b, &c)
 if((a>b) && (a>c))
 printf("\n a is greatest");
 if((b>c) && (b>a))
 printf("\n b is greatest");
 if ((c>b) && (c>a))
 printf("\n c is greatest");
 return (0);
}
```

output

Enter the value of a, b & c

15, 16, 17

c is greatest

3 - WAP in C to reverse a given number.

```
#include <stdio.h>
int main()
{
  int num, rem, rev=0;
  printf("\n Enter any no to be reversed: ");
  scanf("%d", &num);
  while (num>=1);
  {
      rem = num%10;
      rev = rev*10 + rem;
      num = num/10;
  }
  printf("\n Reversed Number : %d", rev);
  return(0);
}
```

Output-

Enter any no to be reversed : 457

Reversed Number : 754

3-4 WAP in C to find Sum of two numbers

```
#include <stdio.h>
int main()
{
  int a, b, Sum;
  printf("\n Enter two nos: ");
  scanf("%d %d", &a, &b);
  Sum = a+b;
```

An array is a kind of data structure that can store a fixed size sequential collection of elements of same data type

⇒ An array is used to store a collection of data.

⇒ Instead of declaring individual variables such as number 0, number1, ... number n you can declare one array variable such as number and use

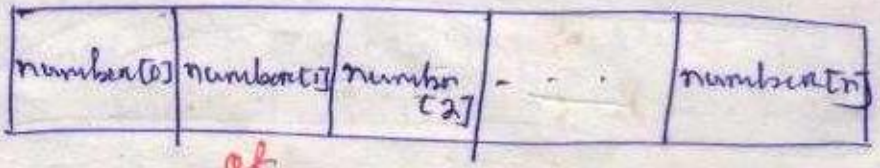
number[0], number[1] .. number[n].

to represent individual variables.

⇒ A particular element in an array is accessed by an index.

⇒ An array consists of ~~cost~~ contiguous memory locations.

⇒ The lowest address corresponds to the 1st element and highest address to the last elements.



Declaration of Arrays

⇒ To declare an array in C, a programmer specifies the type of elements and the no of elements required by an array are as follows.

datatype arrayName [arraysize];

Syntax

```
int acc_bal[1000];
```

Initializing an array :

You can initialize an array in C either one by one or using statement.

Ex

```
int acc_bal[5]={500, 1000, 100, 5, 50};
```

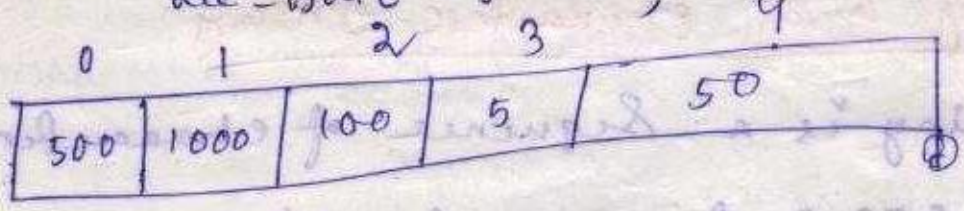
→ the no of values between { } cannot be larger than the no of elements.

Accessing Array Elements

An element is accessed by indexing the array name.

Ex -

```
acc_bal[4] = 5;
```




```
acc_bal[0] = 500;
acc_bal[1] = 1000;
acc_bal[2] = 100;
acc_bal[3] = 5;
acc_bal[4] = 50;
```

One dimensional Arrays

A list of items can be given one variable name using only one subscript. Such a variable is called one-dimensional array.

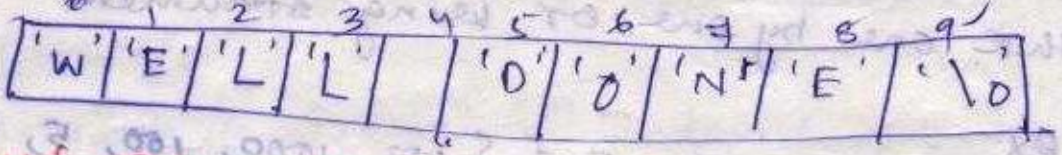
Syntax :

```
int num[3];
```

num[1], num[2], num[3], ... num[n],

number[2] = 100;
number[3] = 5;
number[4] = 50;

char name[10] = "WELL DONE";



Multi-Dimensional Arrays

The general form of a multi-dimension array is

Syntax:

type array-name [s1] [s2] [s3] - [sn];

Ex

int matrix[2][4][2];

String and Character Array:

A string is a sequence of characters.

Any group of characters between double quotation mark is a string constant.

Ex

"Good Morning";

printf("\n Good Morning");

These are the following operations performed on character strings:

* Reading and writting strings
* Combining strings
* Copying one string to another

Declaring and initializing string variables

⇒ C does not support strings as a data type. That's why it allows to represents a string as character Arrays.

Syntax

```
Char Str-name [size];
```

Ex

```
Char Std-name[20];
```

⇒ When the compiler assigns a character string to a character array. It automatically adds a null character ('\0') at the end of the string.

⇒ So the size should be equal to the maximum no. of characters in the string plus one.

```
Char Std-name[10] = "M. Harshita";
```

= {'M', ' ', 'H', 'a', 'r', 's', 'h', 'i', 't', 'a', '\0'}

⇒ Std-name is 10 character long and one

Function in C:

A Function is a group of statments that together perform a task.

⇒ Every C program has at least one fun which is main().

Definition of Function:

The function definition ~~and~~ include the following elements.

1. Function Name
2. Function Type
3. List of parameters
4. Local variable declaration
5. Function statments
6. A return statments.

All the above six elements are grouped together into two parts.

i - function header (1st three)
ii - function body. (2nd three)

The general format of a function d i.e.

```
function-type function-name (parameter-
{
    local variable declaration;
    executable stmt1;
    executable stmt2;
    -
    -
    return statment;
```

(→ Function header; local variable declaration … return statment → Function body)

Function Header:

Function header consists of three parts function type or return type, function name and the formal parameters list.

Function type / Return type:

A function may return a value. The return-type is the data type of the value the function returns. Some function perform the operation without returning a value. ~~then~~ In that case the return type is void.

Function Name

This is the actual name of the function.

The function name and parameter list together constitute the function signature.

Parameters:

When a function is invoked, you pass a value to the parameters. This value is known as actual parameter or argument.

> The parameter lists refers to the type, order and no of parameters of a function.

> A function may contain no parameters i.e optional.

Ex

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

Function Declaration:

A function declaration tells the compiler about a function name and how to call a function.

Syntax:

```
return_type function_name(parameter li
int max (int nam1, int num2).
```

Calling a function:

To use a function after the definition of function, you have to call that function to perform the defined task.

When a program calls a

and after the completion of task by function, it returns a value when its function ending closing brace is reached and it returns the program control back to the main program.

To call a function, you need to pass the required parameters along with function name.

Ex:

```
#include <stdio.h>
/* function declaration */
int max (int n1, int n2);
int main()
{ /* local variable */
   int a = 100;
   int b = 200;
   int result;
  /* calling a function to get MAX value */
   result = max (a, b);
   printf ("max value is :%d \n", result);
   return 0;
}
/* function body */
int max (int n1, int n2)
{
   int maximum
   int result;
```

Function Arguments

⇒ If a function is to use arguments, it must declare variables that accepts the values of the arguments.

⇒ these variables are called formal parameters of the function

⇒ Formal parameters behaves like other local variables inside the function.

⇒ while calling a function, there are two ways in which arguments can be passed to a function.

a- call by value – it copies the actual value of an argument into the formal parameters of the function

b- call by reference – it copies the address of an arguments into the formal parameters of the function.

Local Variables

Variables that are declared inside a function or block are called local variables.

add to ppt

Ex :-

```
#include <stdio.h>
int main()
{
    /* local variables */
    int a, b, c;
```

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

return 0;

}

Global variables

the variables which are defined outside of a function, usually top of the programs.

→ global variable holds their values throughout the lifetime of the program and can be accessed by any function inside the program.

Ex:

```
#include<stdio.h>
int g;  /* global variable declaration */

int main()
{
    /* local variable declaration */
    int a, b;
    /* variable initialization */
    a = 10;
    b = 20;
    g = a+b;
    printf("the value of a = %d,
        b = %d, g = %d \n", a, b, g);
    return 0;
```