Computer

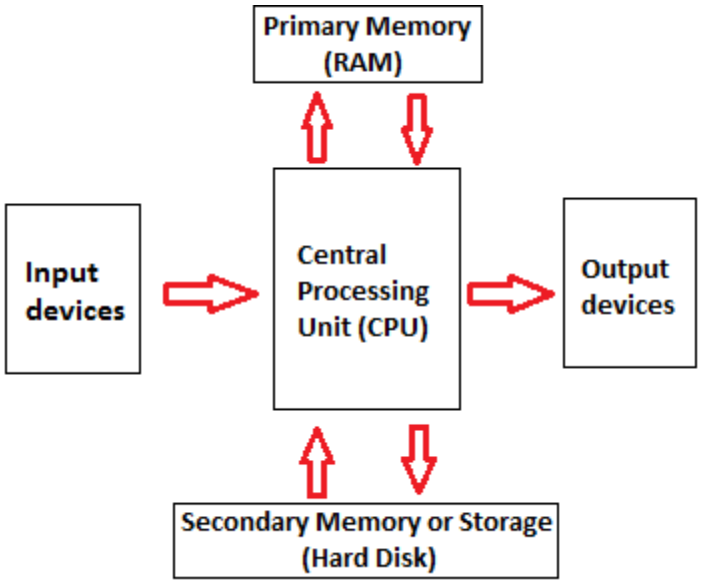
The term "computer" is derived from Latin word "computare" which means to calculate.

Computer is a programmable electronic device that accepts raw data as input and processes it with set of instructions to produce result as output. It renders output just after performing mathematical and logical operations. The device also has memory that stores the data, programs and result of processing.

Computer Components

There are 5 main computer components that are given below:

- Input Devices
- CPU
- Output Devices
- Primary Memory
- Secondary Memory



The operations of computer components are given below:

1) Inputting: It is the process of entering raw data, instructions and information into the computer. It is performed with the help of input devices such as keyboard, mouse etc..

2) Storage/Memory: The computer has primary memory and secondary storage/Memory to store data and instructions.

What is Memory?

Memory is very much like our brain as it is used to store data and instructions. Computer memory is the storage space where data is to be processed, and instructions needed for processing are stored.

What is Primary Memory?

Primary memory is the main memory of the computer system. Accessing data from primary memory is faster because it is an internal memory of the computer. The primary memory is most volatile which means data in primary memory does not exist if it is not saved when a power failure occurs.

The primary memory is a semiconductor memory. It is costlier compared with secondary memory. The capacity of primary memory is very much limited and is always smaller compares to secondary memory.

Two types of Primary Memory are:

- RAM
- ROM

What is Secondary Memory?

All secondary storage devices which are capable of storing high volume data is referred to secondary memory. It's slower than primary memory. However, it can save a substantial amount of data, in the range of gigabytes to terabytes. This memory is also called backup storage or mass storage media.

3) Processing: It is the process of converting the raw data into useful information. This process is performed by the CPU of the computer. It takes the raw data from storage, processes it and then sends back the processed data to storage.

4) Outputting: It is the process of presenting the processed data through output devices like monitor, printer and speakers.

5) Controlling: This operation is performed by the control unit that is part of CPU. The control unit ensures that all basic operations are executed in a right manner and sequence.

Input Devices

Input device enables the user to send data, information, or control signals to a computer. The Central Processing Unit (CPU) of a computer receives the input and processes it to produce the output.

Some of the popular input devices are:

1. Keyboard
2. Mouse
3. Scanner
4. Joystick
5. Light Pen
6. Digitizer
7. Microphone
8. Magnetic Ink Character Recognition (MICR)
9. Optical Character Reader (OCR)

1) Keyboard

The keyboard is a basic input device that is used to enter data into a computer or any other electronic device by pressing keys. It has different sets of keys for letters, numbers, characters, and functions. Keyboards are connected to a computer through USB or a Bluetooth device for wireless communication.

2) Mouse

The mouse is a hand-held input device which is used to move cursor or pointer across the screen. It is designed to be used on a flat surface and generally has left and right button and a scroll wheel between them. Laptop computers come with a touchpad that works as a mouse. It lets you control the movement of cursor or pointer by moving your finger over the touchpad. Some mouse comes with integrated features such as extra buttons to perform different buttons.

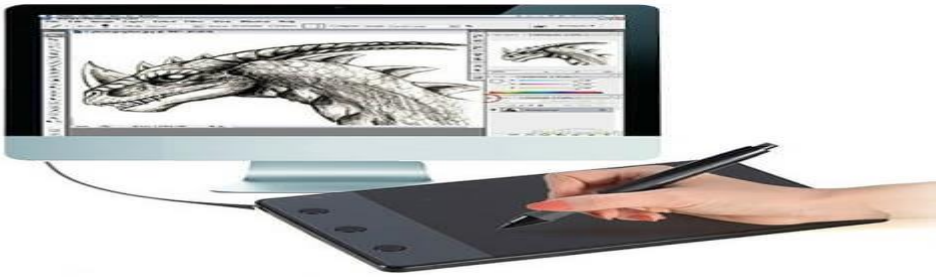
3) Scanner

The scanner uses the pictures and pages of text as input. It scans the picture or a document. The scanned picture or document then converted into a digital format or file and is displayed on the screen as an output. It uses optical character recognition techniques to convert images into digital ones.

5) Light Pen

A light pen is a computer input device that looks like a pen. The tip of the light pen contains a light-sensitive detector that enables the user to point to or select objects on the display screen. Its light sensitive tip detects the object location and sends the corresponding signals to the CPU. It is not compatible with LCD screens, so it is not in use today. It also helps you draw on the screen if needed. The first light pen was invented around 1955 as a part of the Whirlwind project at the Massachusetts Institute of Technology (MIT).

6) Digitizer



Digitizer is a computer input device that has a flat surface and usually comes with a stylus. It enables the user to draw images and graphics using the stylus as we draw on paper with a pencil. The images or graphics drawn on the digitizer appear on the computer monitor or display screen. The software converts the touch inputs into lines and can also convert handwritten text to typewritten words.

It can be used to capture handwritten signatures and data or images from taped papers. Furthermore, it is also used to receive information in the form of drawings and send output to a CAD (Computer-aided design) application and software like AutoCAD. Thus, it allows you to convert hand-drawn images into a format suitable for computer processing.

Output Devices

The output device displays the result of the processing of raw data that is entered in the computer through an input device. There are a number of output devices that display output in different ways such as text, images, hard copies, and audio or video.

Some of the popular output devices are:

1. Monitor
    - CRT Monitor
    - LCD Monitor
    - LED Monitor
    - Plasma Monitor
2. Printer
    - Impact Printers
        A. Character Printers
            i. Dot Matrix printers
            ii. Daisy Wheel printers
        B. Line printers
            i. Drum printers
            ii. Chain printers
    - Non-impact printers
        A. Laser printers
        B. Inkjet printers
3. Projector

---

1) Monitor

The monitor is the display unit or screen of the computer. It is the main output device that displays the processed data or information as text, images, audio or video.

2) Printer

A printer produces hard copies of the processed data. It enables the user, to print images, text or any other information onto the paper.

Based on the printing mechanism, the printers are of two types: Impact Printers and Non-impact Printers.

- **Impact Printers: They are of two types:**
    A. Character Printers
        i. Dot Matrix printers
        ii. Daisy Wheel printers

B. Line printers
    i. Drum printers
    ii. Chain printers

- **Non-impact printers: They are of two types:**
    A. Laser printers
    B. Inkjet printers

Impact Printer

The impact printer uses a hammer or print head to print the character or images onto the paper. The hammer or print head strikes or presses an ink ribbon against the paper to print characters and images.

Impact printers are further divided into two types.

A. Character Printers
B. Line printers

A) Character Printers

Character printer prints a single character at a time or with a single stroke of the print head or hammer. It does not print one line at a time. Dot Matrix printer and Daisy Wheel printer are character printers.

B) Line Printers:

Line printer, which is also as a bar printer, prints one line at a time.

It is a high-speed impact printer as it can print 500 to 3000 lines per minute.

Drum printer and chain printer are examples of line printers.

Non-Impact Printer:

Non-impact printers don't print characters or images by striking a print head or hammer on the ink ribbon placed against the paper. They print characters and images without direct physical contact between the paper and the printing machinery. These printers can print a complete page at a time, so they are also known as page printers. The common types of non-impact printers are Laser printer and Inkjet printer:

3) Projector

A projector is an output device that enables the user to project the output onto a large surface such as a big screen or wall. It can be connected to a computer and similar devices to project their output onto a screen. It uses light and lenses to produce magnified texts, images, and videos. So, it is an ideal output device to give presentations or to teach a large number of people.

Central Processing Unit (CPU)/Processor

A processor (CPU) is the logic circuitry that responds to and processes the basic instructions that drive a computer.

The CPU is seen as the main and most crucial integrated circuitry (IC) chip in a computer, as it is responsible for interpreting most of computers commands. CPUs will perform most basic arithmetic, logic and I/O operations, as well as allocate commands for other chips and components running in a computer.

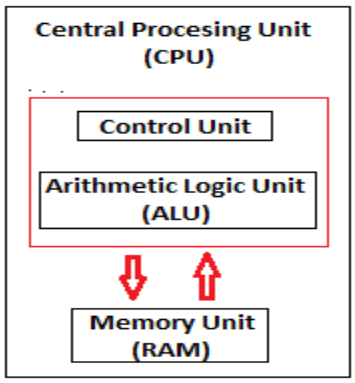
Central processing unit carries out all important functions of a computer. It receives instructions from both the hardware and active software and produces output accordingly.

It is also called processer, central processor and microprocessor.

It stores all important programs like operating system and application software. It also helps Input and output devices to communicate with each other.

Generally, a CPU has three components:

- ALU (Arithmetic Logic Unit)
- Control Unit
- Memory or Storage Unit



Memory: It is called Random access memory (RAM). It temporarily stores data, programs and intermediate and final results of processing.

Control Unit: It controls and coordinates the functioning of all parts of computer. It does not involve in processing and storing data.

ALU: It performs arithmetic and logical functions. Arithmetic functions include addition, subtraction, multiplication and division. Logical functions mainly include selecting, comparing and merging the data.

CPU Operations

The four primary functions of a processor are fetch, decode, execute and write back.

Fetch- is the operation which receives instructions from program memory from a systems RAM.

Decode- is where the instruction is converted to understand which other parts of the CPU are needed to continue the operation. This is performed by the instruction decoder

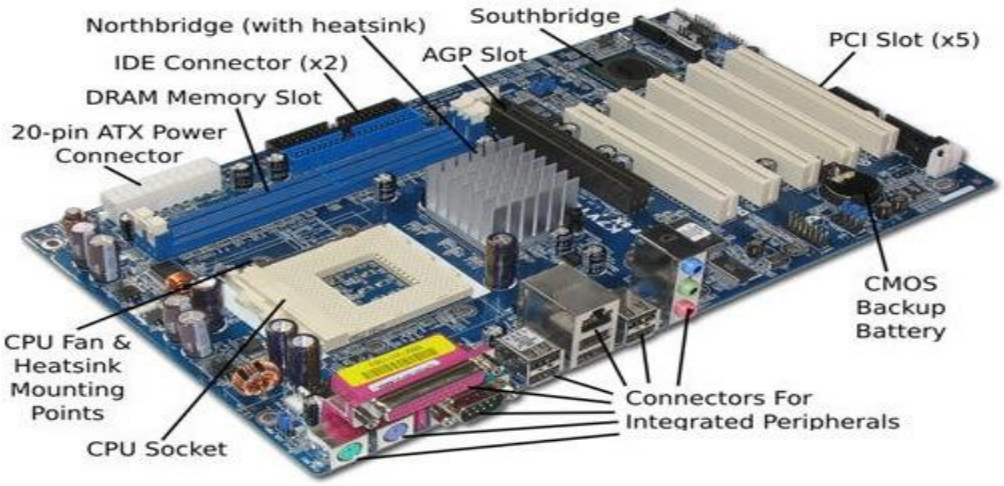
Execute- is where the operation is performed. Each part of the CPU that is needed is activated to carry out the instructions.

Hardware

All tangible physical components of computer and the devices connected to it are hardware. Some of the popular examples of computer hardware are CPU, motherboard, monitor, mouse and keyboard.

Motherboard

Motherboard is generally a thin circuit board that holds together almost all parts of computer except input and output devices. All crucial hardware like CPU, memory, hard drive and ports for input and output devices are located on the motherboard. It allocates power to all hardware located on it and enables them to communicate with each other.



Monitor

Keyboard

Mouse**.**

Instruction

An instruction is an order given to a computer processor by a computer program. At the lowest level, each instruction is a sequence of 0s and 1s that describes a physical operation the computer is to perform (such as "Add") and, depending on the particular instruction type, the specification of special storage areas called registers that may contain data to be used in carrying out the instruction, or the location in computer memory of data.

Software

It is a set of programs that enables the hardware to perform a specific task. All the programs that run the computer are software. Software is of two types; system software and application software.

- **Application software**

  Which is software that uses the computer system to perform special functions or provide entertainment functions beyond the basic operation of the computer itself. Example MS word, VLC media player etc.

System software

Which is software for managing **computer hardware** behavior, as to provide basic functionalities that are required by users, or for other software to run properly, if at all. System software is also designed for providing a platform for running application software, and it includes the following:

- **Operating systems**
  which are essential collections of software that manage resources and provides common services for other software that runs "on top" of them.

  Supervisory programs, boot loaders, shells and window systems are core parts of operating systems. In practice, an operating system comes bundled with additional software (including application software) so that a user can potentially do some work with a computer that only has one operating system.

- **Device drivers**
  which operate or control a particular type of device that is attached to a computer. Each device needs at least one corresponding device driver; because a computer typically has at minimum at least one input device and at least one output device, a computer typically needs more than one device driver.
- **Utilities**
  which are computer programs designed to assist users in the maintenance and care of their computers.\

Operating System

An operating system is a program that acts as an interface between the software and the computer hardware.

It is an integrated set of specialized programs used to manage overall resources and operations of the computer.

It is specialized software that controls and monitors the execution of all other programs that reside in the computer, including application programs and other system software.

Function of operating system

- **Processor Management:** This deals with management of the Central Processing Unit (CPU). The operating system takes care of the allotment of CPU time to different processes. When a process finishes its CPU processing after executing for the allotted time period, this is called scheduling.
- **Device Management**: The Operating System communicates with hardware and the attached devices and maintains a balance between them and the CPU. This is all the more important because the CPU processing speed is much higher than that of I/O devices.
- **Memory management:** In a computer, both the CPU and the I/O devices interact with the memory. When a program needs to be executed it is loaded onto the main memory till the execution is completed. Thereafter that memory space is freed and is available for other

programs. The common memory management techniques used by the operating system are Partitioning and Virtual Memory.

- **File Management:** The operating System manages the files, folders and directory systems on a computer. Any data on a computer is stored in the form of files and the operating system keeps information about all of them using File Allocation Table (FAT).
- **Security** − Prevents unauthorized access to programs and data by means of passwords and other similar techniques.
- **Job Accounting** − Keeps track of time and resources used by various jobs and/or users.
- **Control Over System Performance** − Records delays between the request for a service and from the system.

Compiler

A compiler is a computer program that translates computer code written in one programming language (the source language) into another language (the target language). The name compiler is primarily used for programs that translate source code from a high-level programming language to a lower level language (e.g., assembly language, object code, or machine code) to create an executable program. The complete program is compiled at a time.

Interpreter

An interpreter is a computer program that directly executes instructions written in a programming or scripting language, without requiring them previously to have been compiled into a machine language program.

The interpreter transforms the high-level program into an intermediate language that it then executes, or it could parse the high-level source code and then performs the commands directly, which is done line by line or statement by statement.

The difference between an interpreter and a compiler is given below:

Interpreter	Compiler
Translates program one statement at a time.	Scans the entire program and translates it as a whole into machine code.
It takes less amount of time to analyze the source code but the overall execution time is slower.	It takes large amount of time to analyze the source code but the overall execution time is comparatively faster.
No intermediate object code is generated, hence are memory efficient.	Generates intermediate object code which further requires linking, hence requires more memory.

○ Continues translating the program until the first error is met, in which case it stops. Hence debugging is easy.	○ It generates the error message only after scanning the whole program. Hence debugging is comparatively hard.
○ Programming language like Python, Ruby use interpreters.	○ Programming language like C, C++ use compilers.
○ It usually does’t generates additional intermediate code.	○ It generate additional intermediate code.
○ It requires less memory as it does’t generates extra object code.	○ It requires more memory as it generates extra object code.
○ Errors are displayed one by one after interpreting each line.	○ Errors are displayed at the end of the compilation process.

Interpreter execute conditional control statements at a much slower speed. Compiled programs take more memory because the entire object code has to reside in memory. Interpreter does not generate intermediate object code. As a result, interpreted programs are more memory efficient.

Assembler

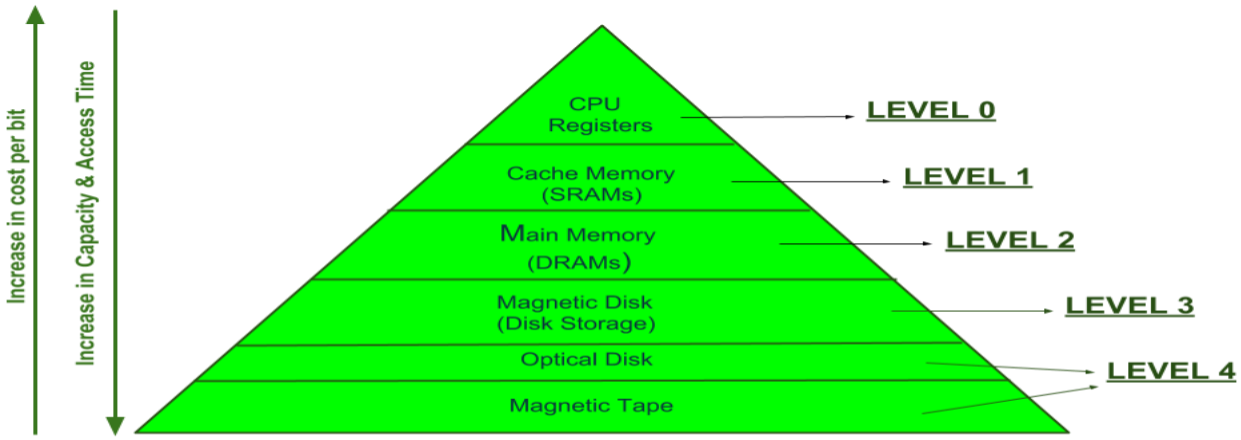
Assembler is a program that turns assembly language into machine code.

An assembler is a program that takes basic computer instructions and converts them into a pattern of bits that the computer's processor can use to perform its basic operations.

Some people call these instructions assembler language and others use the term assembly language.

Memory Hierarchy Design and its Characteristics

In the Computer System Design, Memory Hierarchy is an enhancement to organize the memory such that it can minimize the access time. The Memory Hierarchy was developed based on a program behavior known as locality of references. The figure below clearly demonstrates the different levels of memory hierarchy:



MEMORY HIERARCHY DESIGN

This Memory Hierarchy Design is divided into 2 main types:

External Memory or Secondary Memory –
Comprising of Magnetic Disk, Optical Disk, Magnetic Tape i.e. peripheral storage devices which are accessible by the processor via I/O Module.

Internal Memory or Primary Memory –
Comprising of Main Memory, Cache Memory & CPU registers. This is directly accessible by the processor.

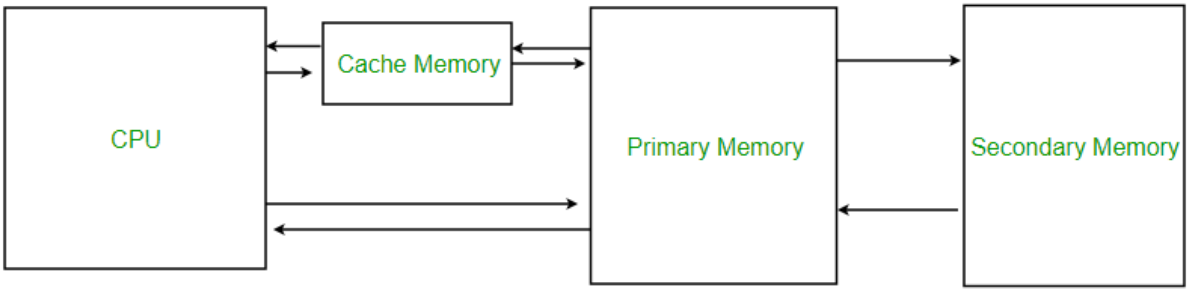
We can infer the following characteristics of Memory Hierarchy Design from above figure:

Capacity:
It is the global volume of information the memory can store. As we move from top to bottom in the Hierarchy, the capacity increases.

Access Time:
It is the time interval between the read/write request and the availability of the data. As we move from top to bottom in the Hierarchy, the access time increases.

Performance:
Earlier when the computer system was designed without Memory Hierarchy design, the speed gap increases between the CPU registers and Main Memory due to large difference in access time. This results in lower performance of the system and thus, enhancement was required. This enhancement was made in the form of Memory Hierarchy Design because of which the performance of the system increases. One of the most significant ways to increase system performance is minimizing how far down the memory hierarchy one has to go to manipulate data.

Cost per bit:
As we move from bottom to top in the Hierarchy, the cost per bit increases i.e. Internal Memory is costlier than External Memory.



Levels of memory:

Level 1 or Register –
It is a type of memory in which data is stored and accepted that are immediately stored in CPU. Most commonly used register is accumulator, Program counter, address register etc.

Level 2 or Cache memory –
It is the fastest memory which has faster access time where data is temporarily stored for faster access.

Level 3 or Main Memory –
It is memory on which computer works currently it is small in size and once power is off data no longer stays in this memory

Level 4 or Secondary Memory –
It is external memory which is not fast as main memory but data stays permanently in this memory

Register

Register is a very fast computer memory, used to store data/instruction in-execution. A Register is a group of flip-flops with each flip-flop capable of storing one bit of information. An n-bit register has a group of n flip-flops and is capable of storing binary information of n-bits.

A processor register may hold an instruction, a storage address, or any data (such as bit sequence or individual characters).The computer needs processor registers for manipulating data and a register for holding a memory address.

Cache Memory

Cache Memory is a special very high-speed memory. It is used to speed up and synchronizing with high-speed CPU. Cache memory is costlier than main memory or disk memory but economical than CPU registers. Cache memory is an extremely fast memory type that acts as a buffer between RAM and the CPU. It holds frequently requested data and instructions so that they are immediately available to the CPU when needed.

Advantages

The advantages of cache memory are as follows −

- Cache memory is faster than main memory.
- It consumes less access time as compared to main memory.
- It stores the program that can be executed within a short period of time.
- It stores data for temporary use.

Disadvantages

The disadvantages of cache memory are as follows −

- Cache memory has limited capacity.
- It is very expensive.

RAM

RAM (Random-access memory) has become a generic term for any semiconductor memory that can be written to, as well as read from, in contrast to ROM (below), which can only be read. All semiconductor memory, not just RAM, has the property of random access.

Volatile memory loses its stored data when the power to the memory chip is turned off. However it can be faster and less expensive than non-volatile memory. This type is used for the main memory in most computers, since data is stored on the hard disk while the computer is off. Major types are:

DRAM (Dynamic random-access memory) which uses memory cells consisting of one capacitor and one transistor to store each bit. This is the cheapest and highest in density, so it is used for the main memory in computers. However the electric charge that stores the data in the memory cells slowly leaks out, so the memory cells must be periodically refreshed (rewritten) which requires additional circuitry. The refresh process is handled internally by the computer and is transparent to its user.

SRAM (Static random-access memory) which relies on several transistors forming a digital flip-flop to store each bit. This is less dense and more expensive per bit than DRAM, but faster and does not require memory refresh. It is used for smaller cache memories in computers.

DRAM	SRAM
1. Constructed of tiny capacitors that leak electricity.	1.Constructed of circuits similar to D flip-flops.
2.Requires a recharge every few milliseconds to maintain its data.	2.Holds its contents as long as power is available.
3.Inexpensive.	3.Expensive.
4. Slower than SRAM.	4. Faster than DRAM.
5. Can store many bits per chip.	5. Can not store many bits per chip.
6. Uses less power.	6.Uses more power.
7.Generates less heat.	7.Generates more heat.
8. Used for main memory.	8. Used for cache.

Difference between SRAM and DRAM

ROM

ROM (Read-only memory) This is designed to hold permanent data, and in normal operation is only read from, not written to. It is a nonvolatile memory preserves the data stored in it during periods when the power to the chip is turned off. It is usually used to store system software which must be immediately accessible to the computer, such as the BIOS program which starts the computer, and the software (microcode) for portable devices and embedded computers such as microcontrollers.

Mask programmed ROM In this type the data is programmed into the chip during manufacture, so it is only used for large production runs. It cannot be rewritten with new data.

PROM (Programmable read-only memory) In this type the data is written into the chip before it is installed in the circuit, but it can only be written once. The data is written by plugging the chip into a device called a PROM programmer.

EPROM (Erasable programmable read-only memory) In this type the data in it can be rewritten by removing the chip from the circuit board, exposing it to an ultraviolet light to erase the existing data, and plugging it into a PROM programmer. The IC package has a small transparent "window" in the top to admit the UV light. It is often used for prototypes and small production run devices, where the program in it may have to be changed at the factory.

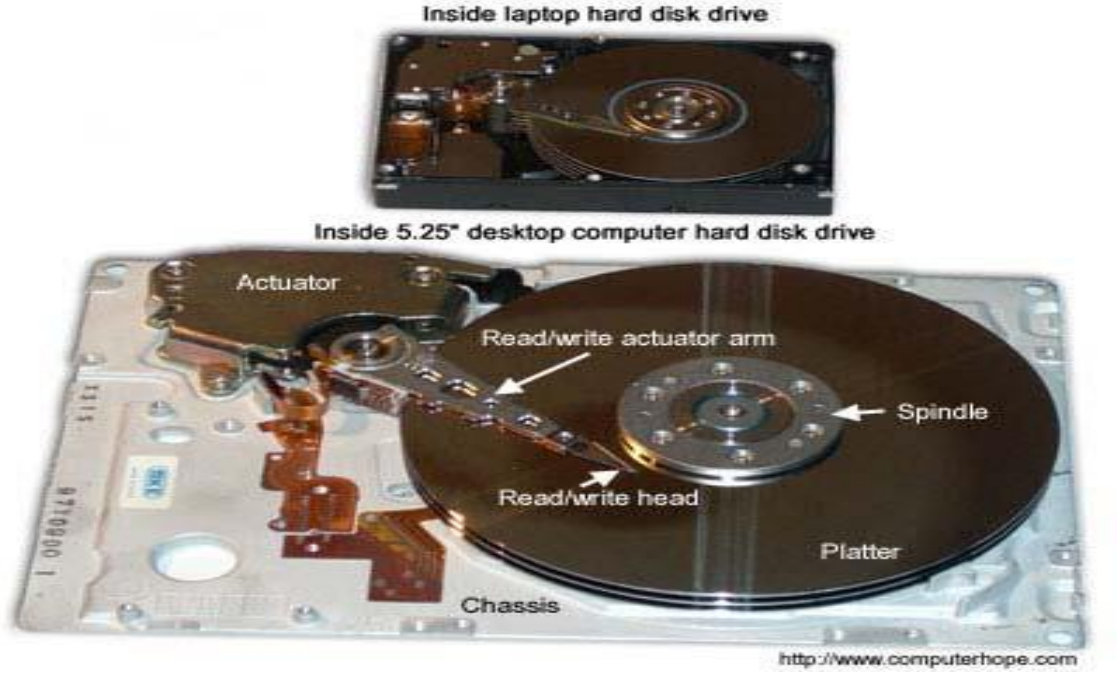
EEPROM (Electrically erasable programmable read-only memory) In this type the data can be rewritten electrically, while the chip is on the circuit board, but the writing process is slow. This type is used to hold firmware, the low level microcode which runs hardware devices, such as the BIOS program in most computers, so that it can be updated.

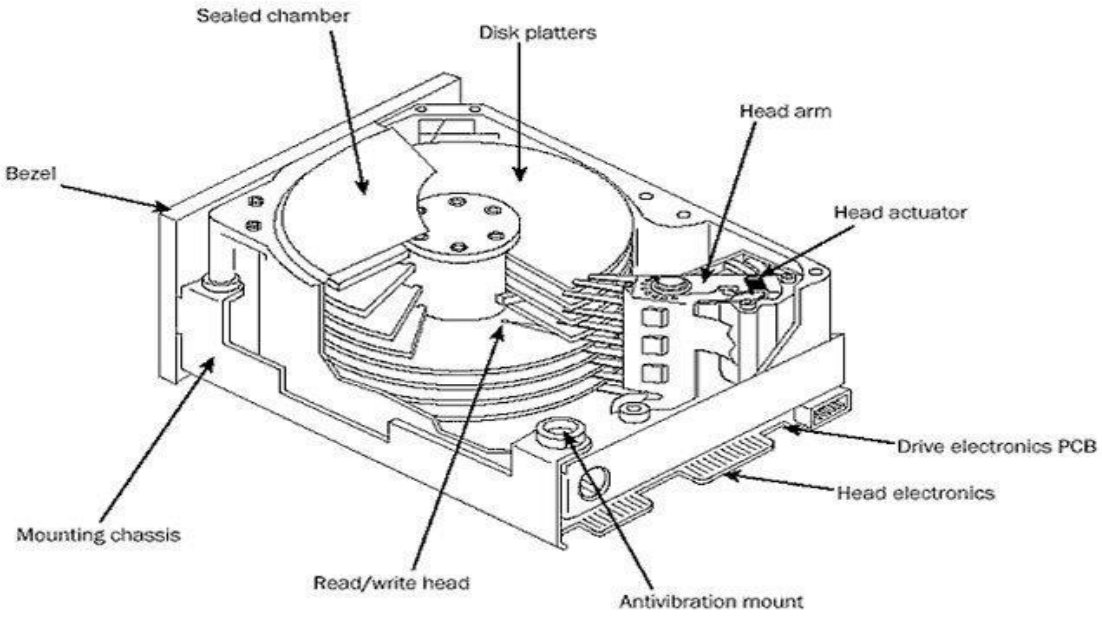
Hard disk

A hard disk drive (HDD), hard disk, hard drive, or fixed disk[b] is an electro-mechanical data storage device that uses magnetic storage to store and retrieve digital information using one or more rigid rapidly rotating disks (platters) coated with magnetic material.[2] Data is accessed in a random-access manner, meaning that individual blocks of data can be stored or retrieved in any order and not only sequentially. HDDs are a type of non-volatile storage, retaining stored data even when powered off

A hard disk is actually a set of stacked disks, like phonograph records. Each disk has data recorded electromagnetically in concentric circles, or tracks, on the disk. A head, similar to a phonograph arm but in a relatively fixed position, writes or reads the information on the tracks. Two heads, one on each side of a disk, read or write the data as the disk spins. Each read or write operation requires that data be located, an operation called a *seek*. Data

already in a disk cache, however, will be located more quickly..





A hard disk/drive unit comes with a set rotation speed varying from 4,200 revolutions per minute (RPM) to 15000 rpm

Difference between primary and secondary memory

Parameter	Primary memory	Secondary memory
Nature	The primary memory is categorized as volatile & nonvolatile memories.	The secondary memory is always a non-volatile memory.
Alias	These memories are also called internal memory.	Secondary memory is known as a Backup memory or Additional memory or Auxiliary memory.
Access	Data is directly accessed by the processing unit.	Data cannot be accessed directly by the processor. It is first copied from secondary memory to primary memory. Only then CPU can access it.
Formation	It's a volatile memory meaning data cannot be retained in case of power failure.	It's a non-volatile memory so that that data can be retained even after power failure.
Storage	It holds data or information that is currently being used by the processing unit. Capacity is usually in 16 to 32 GB	It stores a substantial amount of data and information. Capacity is generally from 200GB to terabytes.
Accesses	Primary memory can be accessed by the data bus.	Secondary memory is accessed by I/O channels.
Expense	Primary memory is costlier than secondary memory.	Secondary memory is cheaper than primary memory.

Pseudo code

- Pseudo code is an informal way of writing a program.
- It is not exactly a computer program.
- It represents the algorithm of the program in natural language and mathematical notations.
- Usually, there is no particular code syntax to write a pseudo code. Therefore, there is no strict syntax as a usual programming language. It uses simple English language.

Example: Pseudo code to add 2 numbers is as follows;

```
Sum of Two Numbers

Begin

    Set sum=0;

    Read: num1, num2;

    Set sum = num1+num2;

    Print sum;

End
```

Example: Pseudo code to find the area of a Rectangle is as follows.

```
Area of Rectangle

Begin

    Read: width, length;

    Set area = width * length;

    Print area;

End
```

Example: Pseudo code to check whether a number entered by user is prime or not.

```
Begin

    Set flag=0,i=2;
```

```
Read: n;
While i<=n/2
   Repeat
          {
                 if  n%i equal to 0 then
                        set flag=0;
                 else
                                set flag=1;
                  increment I;
          }
   if flag equal to 0
          print: "n is not a prime number";
   else
           Print: "n is a prime number";
End
```

Algorithm

An algorithm is a step by step procedure to solve a problem.

A procedure is a finite sequence of instructions, where each is carried out in a finite amount of time.

Every problem can be solved with the help of an algorithm. For example, when the user wants to login to a Facebook account, first he has to go to Facebook.com. Then he has to give the correct username and password. Then he has to click the login button. If the username and password are correct, the user can enter his account. Likewise, every problem has a sequence of steps to solve it. This is also an algorithm because it provides a correct sequence of steps to solve the problem.

Example: Write an algorithm to add two numbers entered by user.

Step 1: Start

Step 2: Declare variables num1, num2 and sum.

Step 3: Read values for num1, num2.

Step 4: Add num1 and num2 and assign the result to a variable sum.

Step 5: Display sum

Step 6: Stop

Example: Write an algorithm to check whether a number is odd or even.

Step 1: Start

Step 2: [Take Input] Read: Number

Step 3: Check: If Number%2 == 0 Then

Print : N is an Even Number.

Else

Print : N is an Odd Number.

Step 4: Exit

Write an algorithm to find the largest among three different numbers entered by user.

Step 1: Start

Step 2: Declare variables a,b and c.

Step 3: Read variables a,b and c.

Step 4: If a>b && a>c Than goto step 7

Otherwise goto step 5.

Step 5: If b>a && b>c Than goto step 8

Otherwise goto step 6.

Step 6: Print “c” is greater and goto step 9.

Step 7: Print “a” is greater and goto step 9.

Step 8: Print “b” is greater and goto step 9.

Step 9: Stop.

Write an algorithm to find the factorial of a number entered by user.

Step 1: Start

Step 2: Declare variables n,factorial and i.

Step 3: Initialize variables

factorial←1

i←1

Step 4: Read value of n

Step 5: Repeat the steps until i=n

5.1: factorial←factorial*i

5.2: i←i+1

Step 6: Display factorial

Step 7: Stop

Write an algorithm to check whether a number entered by user is prime or not.

Step 1: Start

Step 2: Declare variables n,i,flag.

Step 3: Initialize variables

flag←1

i←2

Step 4: Read n from user.

Step 5: Repeat the steps until i<(n/2)

5.1 If remainder of n÷i equals 0

flag←0

Go to step 6

5.2 i←i+1

Step 6: If flag=0

Display n is not prime

else

Display n is prime

Step 7: Stop

Write an algorithm to calculate net salary of an employee with the help of specifications given below.
HRA=20% of basic, DA=50% of basic, TA=5% of basic, Incom-tax=10% of gross.

Step 1: Start

Step 2: Declare variables basic, DA, HRA, TA, IT, gross, net.

Step 3: Read basic from user.

Step 4: HRA=.2*basic

Step 5: DA=.5*basic

Step 6: TA=.05*basic

Step 7: gross=HRA+DA+TA

Step 8: IT=.1*gross

Step 9: net=gross-IT

Step 10: Display net.

Step 11: Stop

Difference between Pseudo code and algorithm

The main difference between algorithm and pseudo code is that an algorithm is a step by step procedure to solve a given problem while a pseudo code is a method of writing an algorithm.

BASIS FOR COMPARISON	ALGORITHM	PSEUDOCODE
Comprehensibility	Quite hard to understand	Easy to interpret
Uses	Complicated programming language	Combination of programming language and natural language
Debugging	Moderate	Simpler
Ease of construction	Complex	Easier

Flowchart

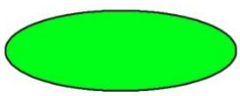
Flowchart is a graphical representation of an algorithm.

Programmers often use it as a program-planning tool to solve a problem.

It makes use of symbols which are connected among them to indicate the flow of information and processing.
The process of drawing a flowchart for an algorithm is known as “flowcharting”.

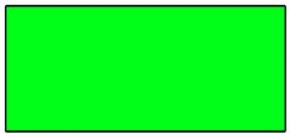
Basic Symbols used in Flowchart Designs

Terminal: The oval symbol indicates Start, Stop and Halt in a program’s logic flow. A pause/halt is generally used in a program logic under some error conditions. Terminal is the first and last symbols in the flowchart.

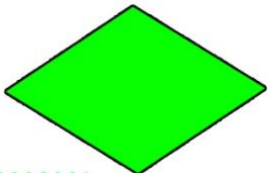


Input/Output: A parallelogram denotes any function of input/output type. Program instructions that take input from input devices and display output on output devices are gram in a flowchart.

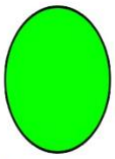
Processing: A box represents arithmetic instructions. All arithmetic processes such as adding, subtracting, multiplication and division are indicated by action or process symbol.



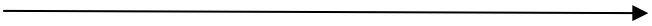
Decision: Diamond symbol represents a decision point. Decision based operations such as yes/no question or true/false are indicated by diamond in flowchart.



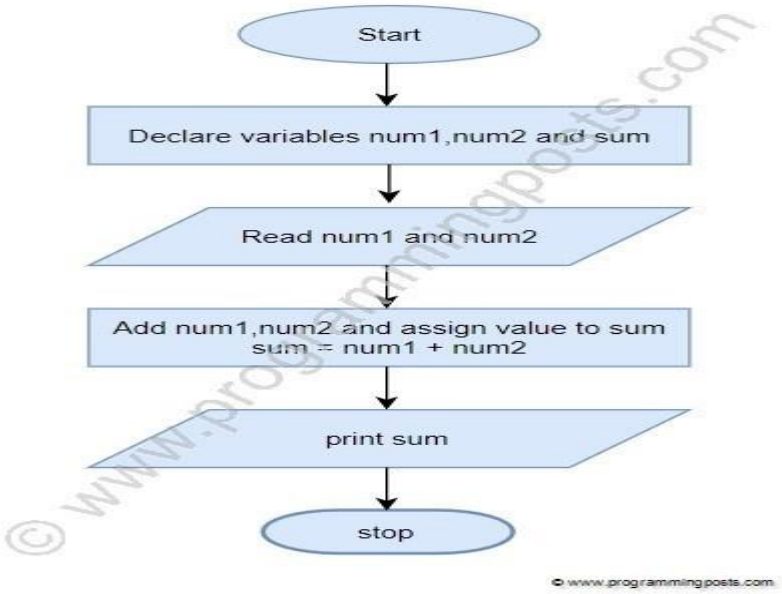
Connectors: Whenever flowchart becomes complex or it spreads over more than one page, it is useful to use connectors to avoid any confusions. It is represented by a circle.



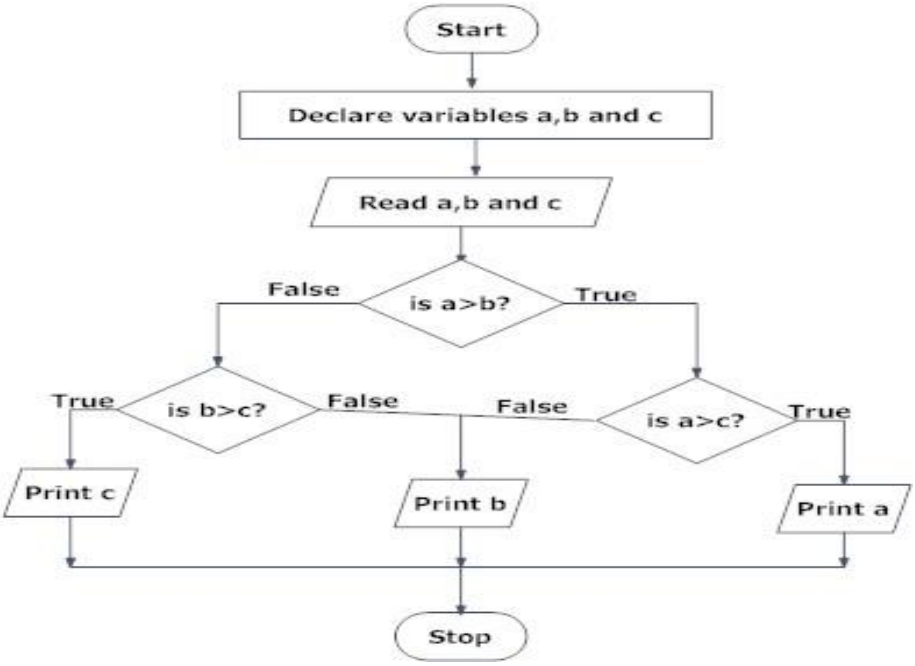
Flow lines: Flow lines indicate the exact sequence in which instructions are executed. Arrows represent the direction of flow of control and relationship among different symbols of flowchart.



Flowchart to find the sum of 2 numbers

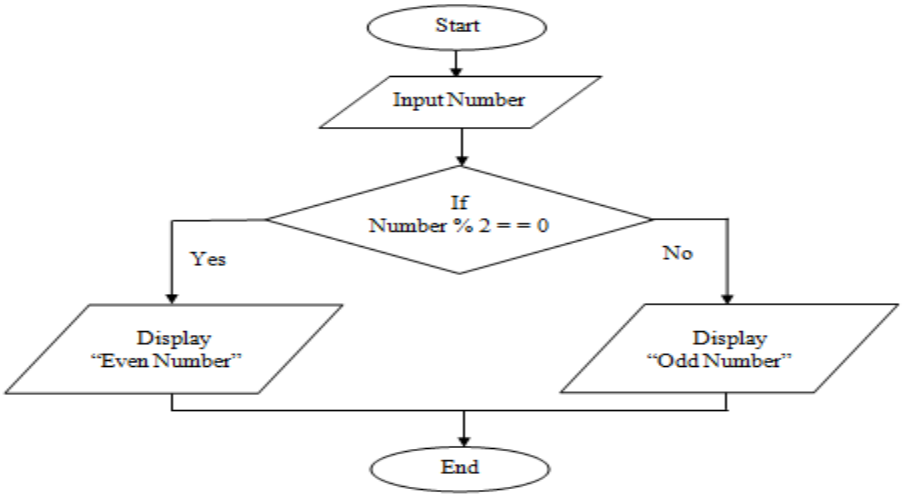


Flowchart to find the largest among three different numbers entered by user.

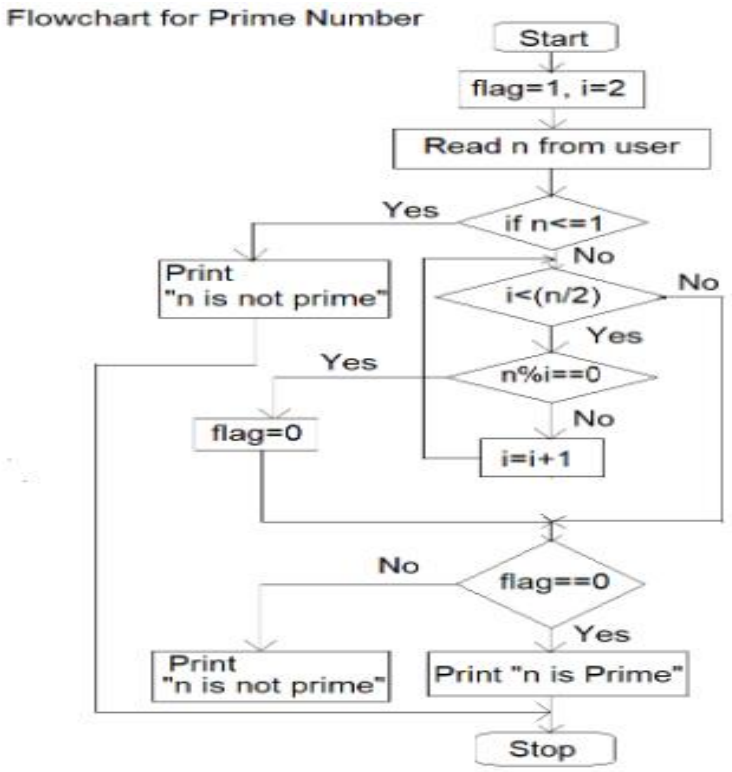


Correct the above flowchart changing Print b to Print c and Print c to Print b.

Flowchart to check a number is odd or even.



Flowchart to check a number is prime or not.



Correct above flowchart by replacing rectangle with parallelogram i.e. read and print.

Source Code

Source code is the set of instructions and statements written by a programmer using a computer programming language. This code is later translated into machine language by a compiler. The translated code is referred to as object code.

Source code is the source of a computer program. It contains declarations, instructions, functions, loops and other statements, which act as instructions for the program on how to function.

```
/* Source code for Hello World program */

#include<stdio.h>

#include<stdio.h>


Void main()

{

printf("Hello World");

getch();

}
```

Executable code

Software in a form that can be run in the computer. It typically refers to machine language, which is the set of native instructions the computer carries out in hardware. Executable files in the DOS/Windows world use .EXE and .COM file extensions, while executable files in Unix and Mac do not require specific extensions. They are identified by their file structure.

C is a high-level language and it needs a compiler to convert it into an executable code so that the program can be run on our machine.

What goes inside the compilation process?

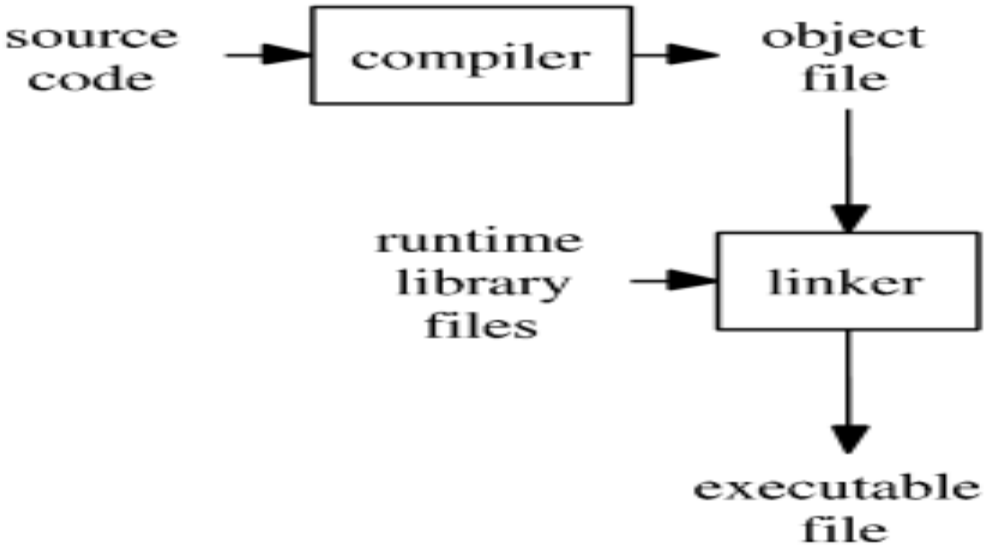
Compiler converts a C program into an executable. There are four phases for a C program to become an executable:

Pre-processing

Compilation

Assembly

Linking



Pre-processing

This is the first phase through which source code is passed. This phase include:

- Removal of Comments
- Expansion of Macros
- Expansion of the included files.
- Conditional compilation

Compiling

The next step is to compile and produce an; intermediate compiled output file **filename.s**. This file is in assembly level instructions

The snapshot shows that it is in assembly language, which assembler can understand.

Assembly

In this phase the filename.s is taken as input and turned into **filename.o** by assembler. This file contain machine level instructions. At this phase, only existing code is converted into machine language, the function calls like printf() are not resolved

Linking

This is the final phase in which all the linking of function calls with their definitions are done. Linker knows where all these functions are implemented. Linker does some extra work also, it adds some extra code to our program which is required when the program starts and ends. For example, there is a code which is required for setting up the environment like passing command line arguments.