GE6151-COMPUTER PROGRAMMING

UNIT –I

INTRODUCTION

Generation and Classification of Computers- **Basic Organization of a** Ccmputer -Number System -Binary - Decimal - Conversion - Problems. **Need for logical analysis and thinking - Algorithm - Pseudo code - Flow Chart.**

LECTURE NOTES

GENERATIONS OF COMPUTERS

The Zeroth Generation

The term Zeroth generation is used to refer to the period of development of computing, which predated the commercial production and sale of computer equipment. The period might be dated as extending from the mid-1800s. In particular, this period witnessed the emergence of the first electronics digital computers on the ABC, since it was the first to fully implement the idea of the stored program and serial execution of instructions. The development of EDVAC set the stage for the evolution of commercial computing and operating system software. The hardware component technology of this period was electronic vacuum tubes. The actual operation of these early computers took place without be benefit of an operating system. Early programs were written in machine language and each contained code for initiating operation of the computer itself. This system was clearly inefficient and depended on the varying competencies of the individual programmer as operators.

The First Generation, 1951-1956

The first generation marked the beginning of commercial computing. The first generation was characterized by high-speed vacuum tube as the active component technology. Operation continued without the benefit of an operating system for a time. The mode was called "closed shop" and was characterized by the appearance of hired operators who would select the job to be run, initial program load the system, run the user‘s program, and then select another job, and so forth. Programs began to be written in higher level, procedure-oriented languages, and thus the operator‘s routine expanded. The operator now selected a job, ran the translation program to assemble or compile the source program, and combined the translated object program along with any existing library programs that the program might need for input to the linking program, loaded and ran the composite linked program, and then handled the next job in a similar fashion. Application programs were run one at a time, and were translated with absolute computer addresses. There was no provision for moving a program to different location in storage for any reason. Similarly, a program bound to specific devices could not be run at all if any of these devices were busy or broken.

At the same time, the development of programming languages was moving away from the basic machine languages; first to assembly language, and later to procedure oriented languages, the most significant being the development of FORTRAN

The Second Generation, 1956-1964

The second generation of computer hardware was most notably characterized by transistors replacing vacuum tubes as the hardware component technology. In addition, some very important changes in hardware and software architectures occurred during this period. For the most part, computer systems remained card and tape-oriented systems. Significant use of random access devices, that is, disks, did not appear until towards the end of the second generation. Program processing was, for the most part, provided by large centralized computers operated under mono-programmed batch processing operating systems.

The most significant innovations addressed the problem of excessive central processor delay due to waiting for input/output operations. Recall that programs were executed by processing the machine instructions in a strictly sequential order. As a result, the CPU, with its high speed electronic component, was often forced to wait for completion of I/O operations which involved mechanical devices (card readers and tape drives) that were order of magnitude slower.These hardware developments led to enhancements of the operatin g system. I/O and data channel communication and control became functions of the operating system, both to relieve the application. programmer from the difficult details of I/O programming and to pr otect the integrity of the system to provide improved service to users by segmenting jobs and running shorter jobs first (during "prime time") and relegating longer jobs to lower priority or night time runs. System libraries became more widely available and more comprehensive as new utilities and application software components were available to programmers.

The second generation was a period of intense operating system development. Also it was the period for sequential batch processing. Researchers began to experiment with multiprogramming and multiprocessing.

The Third Generation, 1964-1979

The third generation officially began in April 1964 with IBM‘s announcement of its System/360 family of computers. Hardware technology began to use integrated circuits (ICs) which yielded significant advantages in both speed and economy. Operating System development continued with the introduction and widespread adoption of multiprogramming. This marked first by the appearance of more sophisticated I/O buffering in the form of spooling operating systems. These systems worked by introducing two new systems programs, a system reader to move input jobs from cards to disk, and a system writer to move job output from disk to printer, tape, or cards.

The spooling operating system in fact had multiprogramming since more than one program was resident in main storage at the same time. Later this basic idea of multiprogramming was extended to include more than one active user program in memory at time. To accommodate this extension, both the scheduler and the dispatcher were enhanced. In addition, memory management became more sophisticated in order to assure that the program code for each job or at least that part of the code being executed was resident in main storage. Users shared not only the system‘ hardware but also its software resources and file system disk space.

The third generation was an exciting time, indeed, for the development of both computer hardware and the accompanying operating system. During this period, the topic of operating systems became, in reality, a major element of the discipline of computing.

The Fourth Generation, 1979 - Present

The fourth generation is characterized by the appearance of the personal computer and the workstation. Miniaturization of electronic circuits and components continued and Large Scale Integration (LSI), the component technology of the third generation, was replaced by Very Large Scale Integration (VLSI), which characterizes the fourth generation. However, improvements in hardware miniaturization and technology have evolved so fast that we now have inexpensive workstation-class computer capable of supporting multiprogramming and time-sharing. Hence the operating systems that supports today‘s personal computers and workstations look much like those which were available for the minicomputers of the third generation. Examples are Microsoft‘s DOS for IBM-compatible personal computers and UNIX for workstation. However, many of these desktop computers are now connected as networked or distributed systems. Computers in a networked system each have their operating system augmented with communication capabilities that enable users to remotely log into any system on the network and transfer information among machines that are connected to the network. The machines that make up distributed system operate as a virtual single processor system from the user‘s point of view; a central operating system controls and makes transparent the location in the system of the particular processor or processors and file systems that are handling any given program.

CLASSIFICATION OF COMPUTERS

There are four classifications of digital computer systems:

super-computer, mainframe computer, minicomputer, and microcomputer.

- Super-computers are very fast and powerful machines. Their internal architecture enables them to run at the speed of tens of MIPS (Million Instructions per Second). Super-computers are very expensive and for this reason are generally not used for CAD applications. Examples of super-computers are: Cray and CDC Cyber 205.

- Mainframe computers are built for general computing, directly serving the needs of business and engineering. Although these computing systems are a step below super-computers, they are still very fast and will process information at about 10 MIPS. Mainframe computing systems are located in a centralized computing center with 20-100+ workstations. This type of computer is still very expensive and is not readily found in architectural/interior design offices.

- Minicomputers were developed in the 1960's resulting from advances in microchip technology. Smaller and less expensive than mainframe computers, minicomputers run at several MIPS and can support 5-20 users. CAD usage throughout the 1960's used minicomputers due to their low cost and high performance. Examples of minicomputers are: DEC PDP, VAX 11.

- Microcomputers were invented in the 1970's and were generally used for home computing and dedicated data processing workstations. Advances in technology have improved microcomputer capabilities, resulting in the explosive growth of personal computers in industry. In the 1980's many medium and small design firms were finally introduced to CAD as a direct result of the low cost and availability of microcomputers. Examples are: IBM, Compaq, Dell, Gateway, and Apple Macintosh.

The average computer user today uses a microcomputer. These types of computers include PC's, laptops, notebooks, and hand-held computers such as Palm Pilots. Larger computers fall into a mini-or mainframe category. A mini-computer is 3-25 times faster than a micro. It is physically larger and has a greater storage capacity.

A mainframe is a larger type of computer and is typically 10-100 times faster than the micro. These computers require a controlled environment both for temperature and humidity. Both the mini and mainframe computers will support more workstations than will a micro. They also cost a great deal more than the micro running into several hundred thousand dollars for the mainframes.

processor

The term processor is a sub-system of a data processing system which processes received information after it has been encoded into data by the input sub-system. These data are then processed by the processing sub-system before being sent to the output sub-system where they are decoded back into information. However, in common parlance processor is usually referred to the microprocessor, the brains of the modern day computers.

There are two main types of processors: CISC and RISC

CISC: A Complex Instruction Set Computer (CISC) is a microprocessor Instruction Set Architecture (ISA) in which each instruction can indicate several low-level operations, such as a load from memory, an arithmetic operation, and a memory store, all in a single instruction. The term was coined in contrast to Reduced Instruction Set Computer (RISC).

Examples of CISC processors are the VAX, PDP-11, Motorola 68000 family and the Intel x86/Pentium CPUs.

RISC: Reduced Instruction Set Computing (RISC), is a microprocessor CPU design philosophy that favors a smaller and simpler set of instructions that all take about the same amount of time to execute. Most types of modern microprocessors are RISCs, for instance ARM, DEC Alpha, SPARC, MIPS, and PowerPC.

The microprocessor contains the CPU which is made up of three components--the control unit supervises all that is going on in the computer, the arithmetic/logic unit which performs the math and comparison operation, and temporary memory. Because of the progress in developing better microprocessors, computers are continually evolving into faster and better units.

Notebooks:

A laptop computer (also known as notebook computer) is a small mobile personal computer, usually weighing around from 1 to 3 kilograms (2 to 7 pounds). Notebooks smaller than an A4 sheet of paper and weighing around 1 kg are sometimes called sub-notebooks and those weighing around 5 kg a desk note (desktop/notebook). Computers larger than PDAs but smaller than notebooks are also sometimes called "palmtops". Laptops usually run on batteries.

Notebook Processor

A notebook processor is a CPU optimized for notebook computers. All computing devices require a CPU. One of the main characteristics differentiating notebook processors from other CPUs is low-power consumption. The notebook processor is becoming an increasing important market segment in the semiconductor industry. Notebook computers are an increasingly popular format of the broader category of mobile computers. The objective of a notebook computer is to provide the performance and functionality of a desktop computer in a portable size and weight. Wireless networking and low power consumption are primary consideration in the choice of a notebook processor.

Integrated Components

Unlike a desktop computer, a notebook has most of the components built-in or integrated into the computer. For desktop systems, determining which computer to buy is generally not based on what type of keyboard or mouse that is available. If you don't like the keyboard or mouse, you can always purchase something else. However, in the case of a notebook computer, the size of the keyboard or type of pointing device may be something that you need to consider unless you intend to use a regular mouse or full-sized keyboard. There are some notebooks that have a keyboard that expands when the notebook is opened which is a nice feature if you find the normal keyboard to be too small. Pointing devices vary from a touch pad to a stick within the keyboard to a roller or track-ball. Most notebooks have the video, sound, and speakers integrated into the computer and some notebooks even have a digital camera built-in which is very handy for video conferencing.

BOOTING:

In computing, booting is a bootstrapping process that starts operating systems when the user turns on a computer system. A boot sequence is the set of operations the computer performs when it is switched on which load an operating system.

Everything that happens between the times the computer switched on and it is ready to accept commands/input from the user is known as booting.

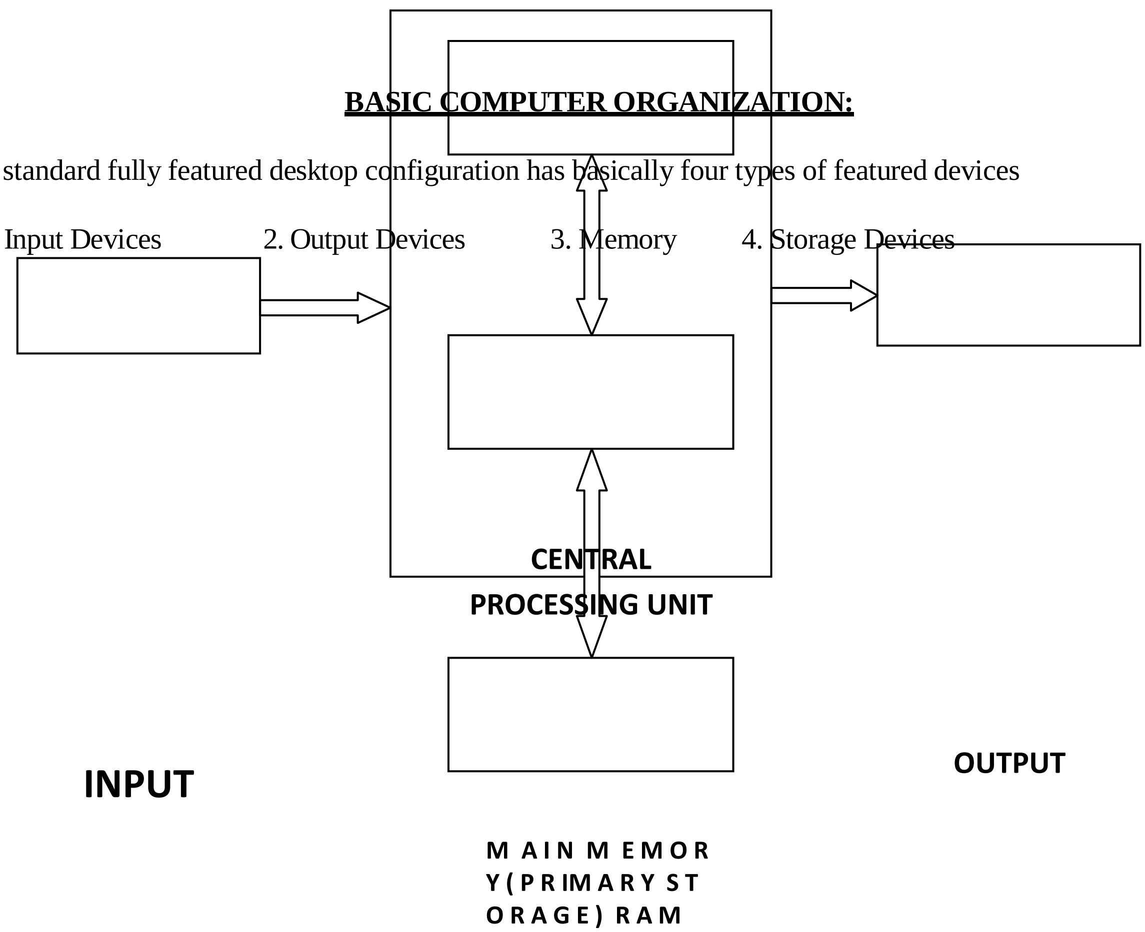
The process of reading disk blocks from the starting of the system disk (which contains the Operating System) and executing the code within the bootstrap. This will read further information off the disk to bring the whole operating system online. Device drivers are contained within the bootstrap code that support all the locally attached peripheral devices and if the computer is connected to a network, the operating system will transfer to the Network Operating system for the "client" to log onto a server

The Process of loading a computer memory with instructions needed for the computer to operate. The process and functions that a computer goes through when it first starts up, ending in the proper and complete loading of the Operating System. The sequence of computer operations from power-up until the system is ready for use

BASIC COMPUTER ORGANIZATION:

A standard fully featured desktop configuration has basically four types of featured devices

1. Input Devices 2. Output Devices 3. Memory 4. Storage Devices



SECONDARY STORAGE

Introduction to CPU

- CPU
- The Arithmetic / Logic Unit (ALU)
- The Control Unit
- Main Memory
- External Memory
- Input / Output Devices
- The System Bus

CPU OPERATION

The fundamental operation of most CPUs
- To execute a sequence of stored instructions called a program.

1. The program is represented by a series of numbers that are kept in some kind of computer memory.
2. There are four steps that nearly all CPUs use in their operation: fetch, decode, execute, and write back.
3. Fetch:

o Retrieving an instruction from program memory.
o The location in program memory is determined by a program counter (PC)
o After an instruction is fetched, the PC is incremented by the length of the instruction word in terms of memory units.

Decode :

1.The instruction is broken up into parts that have significance to other portions of the CPU.

2.The way in which the numerical instruction value is interpreted is defined by the CPU's instruction set architecture (ISA).

3.Opcode, indicates which operation to perform.

4.The remaining parts of the number usually provide information required for that instruction, such as operands for an addition operation.

5.Such operands may be given as a constant value or as a place to locate a value: a register or a memory address, as determined by some addressing mode.

Execute :

1.During this step, various portions of the CPU are connected so they can perform the desired operation.
2.If, for instance, an addition operation was requested, an arithmetic logic unit (ALU) will be connected to a set of inputs and a set of outputs.
3.The inputs provide the numbers to be added, and the outputs will contain the final sum.

4. If the addition operation produces a result too large for the CPU to handle, an arithmetic overflow flag in a flags register may also be set.

Write back :

1.Simply "writes back" the results of the execute step to some form of memory.

2.Very often the results are written to some internal CPU register for quick access by subsequent instructions.

3.In other cases results may be written to slower, but cheaper and larger, main memory.

Some types of instructions manipulate the program counter rather than directly produce result data.

INPUT DEVICES

Anything that feeds the data into the computer. This data can be in alpha-numeric form which needs to be keyed-in or in its very basic natural form i.e. hear, smell, touch, see; taste & the sixth sense …feel?

Typical input devices are:

1. Keyboard
2. Mouse
3. Joystick
4. Digitizing Tablet
5. Touch Sensitive Screen
6. Light Pen
7. Space Mouse
8. Digital Stills Camera
9. Magnetic Ink Character Recognition (MICR)
10. OpticalMarkReader (OMR)
11. Image Scanner
12. Bar Codes
13. Magnetic Reader
14. Smart Cards
15. Voice Data Entry
16. Sound Capture
17. Video Capture

The **Keyboard** is the standard data input and operator control device for a computer. It consists of the standard QWERTY layout with a numeric keypad and additional function keys for control purposes.

The **Mouse** is a popular input device. You move it across the desk and its movement is shown on the screen by a marker known as a 'cursor'. You will need to click the buttons at the top of the mouse to select an option.

Track ball looks like a mouse, as the roller is on the top with selection buttons on the side. It is also a pointing device used to move the cursor and works like a mouse. For moving the cursor in a particular direction, the user spins the ball in that direction. It is sometimes considered better than a mouse, because it requires little arm movement and less desktop space. It is generally used with Portable computers.

Magnetic Ink Character Recognition (MICR) is used to recognize the magnetically charged characters, mainly found on bank cheques. The magnetically charged characters are written by special ink called magnetic ink. MICR device reads the patterns of these characters and compares them with special patterns stored in memory. Using MICR device, a large volume of cheques can be processed in a day. MICR is widely used by the banking industry for the processing of cheques.

The joystick is a rotary lever. Similar to an aircraft's control stick, it enables you to move within the screen's environment, and is widely used in the computer games industry.

A **Digitising Tablet** is a pointing device that facilitates the accurate input of drawings and designs. A drawing can be placed directly on the tablet, and the user traces outlines or inputs coordinate positions with a hand-held stylus.

A **Touch Sensitive Screen** is a pointing device that enables the user to interact with the computer by touching the screen. There are three types of Touch Screens: pressure-sensitive, capacitive surface and light beam.

A **Light Pen** is a pointing device shaped like a pen and is connected to a VDU. The tip of the light pen contains a light-sensitive element which, when placed against the screen, detects the light from the screen enabling the computer to identify the location of the pen on the screen. Light pens have the advantage of 'drawing' directly onto the screen, but this can become uncomfortable, and they are not as accurate as digitising tablets.

The **Space mouse** is different from a normal mouse as it has an X axis, a Y axis and a Z axis. It can be used for developing and moving around 3-D environments.

Digital Stills Cameras capture an image which is stored in memory within the camera. When the memory is full it can be erased and further images captured. The digital images can then be downloaded from the camera to a computer where they can be displayed, manipulated or printed.

The Optical Mark Reader (OMR) can read information in the form of numbers or letters and put it into the computer. The marks have to be precisely located as in multiple choice test papers.

Scanners allow information such as a photo or text to be input into a computer. Scanners are usually either A4 size (flatbed), or hand-held to scan a much smaller area. If text is to be scanned, you would use an Optical Character Recognition (OCR) program to recognise the printed text and then convert it to a digital text file that can be accessed using a computer.

A **Bar Code** is a pattern printed in lines of differing thickness. The system gives fast and error-free entry of information into the computer. You might have seen bar codes on goods in supermarkets, in libraries and on magazines. Bar codes provide a quick method of recording the sale of items.

Card Reader: This input device reads a magnetic strip on a card. Handy for security reasons, it provides quick identification of the card's owner. This method is used to run bank cash points or to provide quick identification of people entering buildings.

Smart Card: This input device stores data in a microprocessor embedded in the card. This allows information, which can be updated, to be stored on the card. This method is used in store cards which accumulate points for the purchaser, and to store phone numbers for cellular phones.

OUTPUT DEVICES :

Output devices display information in a way that you can you can understand. The most common output device is a monitor. It looks a lot a like a TV and houses the computer screen. The monitor allows you to 'see' what you and the computer are doing together.

Brief of Output Device

Output devices are pieces of equipment that are used to get information or any other response out from computer. These devices display information that has been held or generated within a computer. Output devices display information in a way that you can understand. The most common output device is a monitor.

Types of Output Device

Printing:		Plotter, Printer
Sound	:	Speakers
Visual	:	Monitor

A **Printer** is another common part of a computer system. It takes what you see on the computer screen and prints it on paper. There are two types of printers; Impact Printers and Non-Impact Printers.

Speakers are output devices that allow you to hear sound from your computer. Computer speakers are just like stereo speakers. There are usually two of them and they come in various sizes.

MEMORY OR PRIMARY STORAGE :

Purpose of Storage

The fundamental components of a general-purpose computer are arithmetic and logic unit, control circuitry, storage space, and input/output devices. If storage was removed, the device we had would be a simple calculator instead of a computer. The ability to store instructions that form a computer program, and the information that the instructions manipulate is what makes stored program architecture computers versatile.

Primary Storage

Primary storage is directly connected to the central processing unit of the computer. It must be present for the CPU to function correctly, just as in a biological analogy the lungs must be present (for oxygen storage) for the heart to function (to pump and oxygenate the blood). As shown in the diagram, primary storage typically consists of three kinds of storage:

Processors Register

It is the internal to the central processing unit. Registers contain information that the arithmetic and logic unit needs to carry out the current instruction. They are technically the fastest of all forms of computer storage.

Main memory

It contains the programs that are currently being run and the data the programs are operating on. The arithmetic and logic unit can very quickly transfer information between a processor register and locations in main storage, also known as a "memory addresses". In modern computers, electronic solid-state random access memory is used for main storage, and is directly connected to the CPU via a "memory bus" and a "data bus".

Cache memory

It is a special type of internal memory used by many central processing units to increase their performance or "throughput". Some of the information in the main memory is duplicated in the cache memory, which is slightly slower but of much greater capacity than the processor registers, and faster but much smaller than main memory.

Memory

Memory is often used as a shorter synonym for Random Access Memory (RAM). This kind of memory is located on one or more microchips that are physically close to the microprocessor in your computer. Most desktop and notebook computers sold today include at least 512 megabytes of RAM (which is really the minimum to be able to install an operating system). They are upgradeable, so you can add more when your computer runs really slowly.

The more RAM you have, the less frequently the computer has to access instructions and data from the more slowly accessed hard disk form of storage. Memory should be distinguished from storage, or the physical medium that holds the much larger amounts of data that won't fit into RAM and may not be immediately needed there.

Storage devices include hard disks, floppy disks, CDROMs, and tape backup systems. The terms auxiliary storage, auxiliary memory, and secondary memory have also been used for this kind of data repository.

RAM is temporary memory and is erased when you turn off your computer, so remember to save your work to a permanent form of storage space like those mentioned above before exiting programs or turning off your computer.

TYPES OF RAM:

There are two types of RAM used in PCs - Dynamic and Static RAM.

Dynamic RAM (DRAM): The information stored in Dynamic RAM has to be refreshed after every few milliseconds otherwise it will get erased. DRAM has higher storage capacity and is cheaper than Static RAM.

Static RAM (SRAM): The information stored in Static RAM need not be refreshed, but it remains stable as long as power supply is provided. SRAM is costlier but has higher speed than DRAM.

Additional kinds of integrated and quickly accessible memory are Read Only Memory (ROM), Programmable ROM (PROM), and Erasable Programmable ROM (EPROM). These are used to keep special programs and data, such as the BIOS, that need to be in your computer all the time. ROM is "built-in" computer memory containing data that normally can only be read, not written to (hence the name read only).

ROM contains the programming that allows your computer to be "booted up" or regenerated each time you turn it on. Unlike a computer's random access memory (RAM), the data in ROM is not lost when the computer power is turned off. The ROM is sustained by a small long life battery in your computer called the CMOS battery. If you ever do the hardware setup procedure with your computer, you effectively will be writing to ROM. It is non volatile, but not suited to storage of large quantities of data because it is expensive to produce. Typically, ROM must also be completely erased before it can be rewritten,

PROM (Programmable Read Only Memory)

A variation of the ROM chip is programmable read only memory. PROM can be programmed to record information using a facility known as PROM-programmer. However once the chip has been programmed the recorded information cannot be changed, i.e. the PROM becomes a ROM and the information can only be read.

EPROM (Erasable Programmable Read Only Memory)

As the name suggests the Erasable Programmable Read Only Memory, information can be erased and the chip programmed a new to record different information using a special PROM-Programmer. When EPROM is in use information can only be read and the information remains on the chip until it is erased.

STORAGE DEVICES

The purpose of storage in a computer is to hold data or information and get that data to the CPU as quickly as possible when it is needed. Computers use disks for storage: hard disks that are located inside the computer, and floppy or compact disks that are used externally.

- Computers Method of storing data & information for long term basis i.e. even after PC is switched off.
- It is non - volatile
- Can be easily removed and moved & attached to some other device
- Memory capacity can be extended to a greater extent
- Cheaper than primary memory

Storage Involves Two Processes
a) Writing data b) Reading data

Floppy Disks

The floppy disk drive (FDD) was invented at IBM by Alan Shugart in 1967. The first floppy drives used an 8-inch disk (later called a "diskette" as it got smaller), which evolved into the 5.25-inch disk that was used on the first IBM Personal Computer in August 1981. The 5.25 -inch disk held 360 kilobytes compared to the 1.44 megabyte capacity of today's 3.5-inch diskette.

The 5.25-inch disks were dubbed "floppy" because the diskette packaging was a very flexible plastic envelope, unlike the rigid case used to hold today's 3.5-inch diskettes.

By the mid-1980s, the improved designs of the read/write heads, along with improvements in the magnetic recording media, led to the less-flexible, 3.5-inch, 1.44-megabyte (MB) capacity FDD in use today. For a few years, computers had both FDD sizes (3.5-inch and 5.25-inch). But by the mid-1990s, the 5.25-inch version had fallen out of popularity, partly because the diskette's recording surface could easily become contaminated by fingerprints through the open access area.

When you look at a floppy disk, you'll see a plastic case that measures 3 1/2 by 5 inches. Inside that case is a very thin piece of plastic that is coated with microscopic iron particles. This disk is much like the tape inside a video or audio cassette. Basically, a floppy disk drive reads and writes data to a small, circular piece of metal-coated plastic similar to audio cassette tape.

At one end of it is a small metal cover with a rectangular hole in it. That cover can be moved aside to show the flexible disk inside. But never touch the inner disk - you could damage the data that is stored on it. On one side of the floppy disk is a place for a label. On the other side is a silver circle with two holes in it. When the disk is inserted into the disk drive, the drive hooks into those holes to spin the circle. This causes the disk inside to spin at about 300 rpm! At the same time, the silver metal cover on the end is pushed aside so that the head in the disk drive can read and write to the disk.

Floppy disks are the smallest type of storage, holding only 1.44MB.

3.5-inch Diskettes (Floppy Disks) features:
- Spin rate: app. 300 revolutions per minute (rpm)
- High density (HD) disks more common today than older, double density (DD) disks
- Storage Capacity of HD disks is 1.44 MB

Floppy Disk Drive Terminology

Floppy disk - Also called diskette. The common size is 3.5 inches.

Floppy disk drive - The electromechanical device that reads and writes floppy disks.

Track - Concentric ring of data on a side of a disk.

Sector - A subset of a track, similar to wedge or a slice of pie.

It consists of a read/write head and a motor rotating the disk at a high speed of about 300 rotations per minute. It can be fitted inside the cabinet of the computer and from outside, the slit where the disk is to be inserted, is visible. When the disk drive is closed after inserting the floppy inside, the monitor catches the disk through the Central of Disk hub, and then it starts rotating.

There are two read/write heads depending upon the floppy being one sided or two sided. The head consists of a read/write coil wound on a ring of magnetic material. During write operation, when the current passes in one direction, through the coil, the disk surface touching the head is magnetized in one direction. For reading the data, the procedure is reverse. I.e. the magnetized spots on the disk touching the read/write head induce the electronic pulses, which are sent to CPU.

The major parts of a FDD include:

Read/Write Heads: Located on both sides of a diskette, they move together on the same assembly. The heads are not directly opposite each other in an effort to prevent interaction between write operations on each of the two media surfaces. The same head is used for reading and writing, while a second, wider head is used for erasing a track just prior to it being written. This allows the data to be written on a wider "clean slate," without interfering with the analog data on an adjacent track.

Drive Motor: A very small spindle motor engages the metal hub at the center of the diskette, spinning it at either 300 or 360 rotations per minute (RPM).

Stepper Motor: This motor makes a precise number of stepped revolutions to move the read/write head assembly to the proper track position. The read/write head assembly is fastened to the stepper motor shaft.

Mechanical Frame: A system of levers that opens the little protective window on the diskette to allow the read/write heads to touch the dual-sided diskette media. An external button allows the diskette to be ejected, at which point the spring-loaded protective window on the diskette closes.

Circuit Board: Contains all of the electronics to handle the data read from or written to the diskette. It also controls the stepper-motor control circuits used to move the read/write heads to each track, as well as the movement of the read/write heads toward the diskette surface.

Electronic optics check for the presence of an opening in the lower corner of a 3.5-inch diskette (or a notch in the side of a 5.25-inch diskette) to see if the user wants to prevent data from being written on it.

Hard Disks

Your computer uses two types of memory: primary memory which is stored on chips located on the motherboard, and secondary memory that is stored in the hard drive. Primary memory holds all of the essential memory that tells your computer how to be a computer. Secondary memory holds the information that you store in the computer.

Inside the hard disk drive case you will find circular disks that are made from polished steel. On the disks, there are many tracks or cylinders. Within the hard drive, an electronic reading/writing device called the head passes back and forth over the cylinders, reading information from the disk or writing information to it. Hard drives spin at 3600 or more rpm (Revolutions Per Minute) - that means that in one minute, the hard drive spins around over 7200 times!

Optical Storage

- Compact Disk Read-Only Memory (CD-ROM)
- CD-Recordable (CD-R)/CD-Rewritable (CD-RW)
- Digital Video Disk Read-Only Memory (DVD-ROM)
- DVD Recordable (DVD-R/DVD Rewritable (DVD-RW)
- Photo CD

Optical Storage Devices Data is stored on a reflective surface so it can be read by a beam of laser light. Two Kinds of Optical Storage Devices

- CD-ROM (compact disk read-only memory)
- DVD-ROM (digital video disk read-only memory)

Compact Disks

Instead of electromagnetism, CDs use pits (microscopic indentations) and lands (flat surfaces) to store information much the same way floppies and hard disks use magnetic and non-magnetic storage. Inside the CD-Rom is a laser that reflects light off of the surface of the disk to an electric eye. The pattern of reflected light (pit) and no reflected light (land) creates a code that represents data.

CDs usually store about 650MB. This is quite a bit more than the 1.44MB that a floppy disk stores. A DVD or Digital Video Disk holds even more information than a CD, because the DVD can store information on two levels, in smaller pits or sometimes on both sides.

Recordable Optical Technologies

- CD-Recordable (CD-R)
- CD-Rewritable (CD-RW)

- PhotoCD
- DVD-Recordable (DVD-R)
- DVD-RAM

CD ROM - Compact Disc Read Only Memory.

Unlike magnetic storage device which store data on multiple concentric tracks, all CD formats store data on one physical track, which spirals continuously from the center to the outer edge of the recording area. Data resides on the thin aluminum substrate immediately beneath the label. The data on the CD is recorded as a series of microscopic pits and lands physically embossed on an aluminum substrate. Optical drives use a low power laser to read data from those discs without physical contact between the head and the disc which contributes to the high reliability and permanence of storage device.

To write the data on a CD a higher power laser are used to record the data on a CD. It creates the pits and land on aluminum substrate. The data is stored permanently on the disc. These types of discs are called as WORM (Write Once Read Many). Data written to CD cannot subsequently be deleted or overwritten which can be classified as advantage or disadvantage depending upon the requirement of the user. However if the CD is partially filled then the more data can be added to it later on till it is full. CDs are usually cheap and cost effective in terms of storage capacity and transferring the data.

The CD‘s were further developed where the data could be deleted and re written. These types of CDs are called as CD Rewritable. These types of discs can be used by deleting the data and making the space for new data. These CD‘s can be written and rewritten at least 1000 times.

CD ROM Drive

CD ROM drives are so well standardized and have become so ubiquitous that many treat them as commodity items. Although CD ROM drives differ in reliability, which standards they support and numerous other respects, there are two important performance measures.

- Data transfer rate
- Average access

Data transfer rate: Data transfer rate means how fast the drive delivers sequential data to the interface. This rate is determined by drive rotation speed, and is rated by a number followed by _X‘. All the other things equal, a 32X drive delivers data twice the speed of a 16X drive. F ast data transfer rate is most important when the drive is used to transfer the large file or many sequential smaller files. For example: Gaming video.

CD ROM drive transfers the data at some integer multiple of this basic 150 KB/s 1X rate. Rather than designating drives by actual KB/s output drive manufacturers use a multiple of the standard 1X rate. For example: a 12X drive transfer data at (12*150KB/s) 1800 KB/s and so on.

The data on a CD is saved on tracks, which spirals from the center of the CD to outer edge. The portions of the tracks towards center are shorter than those towards the edge. Moving the data under the head at a constant rate requires spinning the disc faster as the head moves from the center where there is less data per revolution to the edge where there is more data. Hence the rotation rate of the disc changes as it progresses from inner to outer portions of the disc.

CD Writers

CD recordable and CD rewritable drives are collectively called as CD writers or CD burners.

They are essentially CD ROM drives with one difference. They have a more powerful laser that, in addition to reading discs, can record data to special CD media.

Pen Drives / Flash Drives

- Pen Drives / Flash Drives are flash memory storage devices.
- They are faster, portable and have a capability of storing large data.
- It consists of a small printed circuit board with a LED encased in a robust plastic
- The male type connector is used to connect to the host PC
- They are also used a MP3 players

NUMBER SYSTEMS

Binary	Decimal	Octal	Hexadecimal
0000	00	0	0
0001	01	1	1
0010	02	2	2
0011	03	3	3
0100	04	4	4
0101	05	5	5
0110	06	6	6
0111	07	7	7
1000	08	10	8
1001	09	11	9
1010	10	12	A
1011	11	13	B
1100	12	14	C
1101	13	15	D
1110	14	16	E
1111	15	17	F

DECIMAL NUMBERS

In the decimal number systems each of the ten digits, 0 through 9, represents a certain quantity. The position of each digit in a decimal number indicates the magnitude of the quantity represented and can be assigned a weight. The weights for whole numbers are positive powers of ten that increases from right to left, beginning with $10^o = 1$ that is 10^3 10^2 10^1 10^o

For fractional numbers, the weights are negative powers of ten that decrease from left to right beginningwith 10^{-1} that is 10^2 10^1 10^o. 10^{-1} 10^{-2} 10^{-3}
The value of a decimal number is the sum of digits after each digit has been multiplied by its weights asin following examples

Express the decimal number 87 as a sum of the values of each digit.
The digit 8 has a weight of10 which is 10 as indicated by its position. The digit 7 has a weight of 1 which is 10^o as indicated by its position.

$$87 = (8 \times 10^{1)} + (7 \times 10^0)$$

BINARY NUMBERS

The binary system is less complicated than the decimal system because it has only two digits, it is a basetwo system. The two binary digits (bits) are 1 and 0. The position of a 1 or 0 in a binary number indicates its weight, or value within the number, just as the position of a decimal digit determines the value of that digit. The weights in a binary number are based on power of two as:

$$\ldots\ldots 2^4\ 2^3\ 2^2\ 2^1\ 2^0.\ 2^{-1}\ 2^{-2}\ \ldots.$$

With 4 digits position we can count from zero to 15.In general, with n bits we can count up to a number equal to Ķ - 1. Largest decimal number = Ķ - 1.A binary number is a weighted number. The right-most bit is the least significant bit (LSB) in a binary whole number and has a weight of
2^o =1. The weights increase from right to left by a power of two for each bit. The left-most bit is the most significant bit (MSB); its weight depends on the size of the binary number.

BINARY – TO – DECIMAL CONVERSION

The decimal value of any binary number can be found by adding the weights of all bits that are 1 and discarding the weights of all bits that are 0

Example
Let‘s convert the binary whole number 101101 to decimal

Weight:2^5 2^4 2^3 2^2 2^1 2^0
x

Binary no:	**1**	**0**	**1**	**1**	**0**	**1**
Value	**32**	**0**	**8**	**4**	**0**	**1**

Sum = 45

HEXADECIMAL NUMBERS

The hexadecimal number system has sixteen digits and is used primarily as a compact way of displaying or writing binary numbers because it is very easy to convert between binary and hexadecimal.Long binary numbers are difficult to read and write because it is easy to drop or transpose a bit. Hexadecimal is widely used in computer and microprocessor applications. The hexadecimal system has a base of sixteen; it is composed of 16 digits and alphabetic characters. The maximum 3-digits hexadecimal number is FFF or decimal 4095 and maximum 4-digit hexadecimal number is FFFF or decimal 65.535.

BINARY –TO – HEXADECIMAL CONVERSION

Simply break the binary number into 4-bit groups, starting at the right-most bit and replace each 4-
bit group with the equivalent hexadecimal symbol as in the following example

Convert the binary number to hexadecimal: 1100101001010111

Solution:

1100	**1010**	**0101**	**0111**
C	**A**	**5**	**7**

HEXADECIMAL – TO – DECIMAL CONVERSION

One way to find the decimal equivalent of a hexadecimal number is to first convert the hexadecimal number to binary and then convert from binary to decimal.

Convert the hexadecimal number 1C to decimal

1 C

0001 1100= $\mathbf{2^4 + 2^3 + 2^2 = 16 + 8 + 4 = 28}$

OCTAL NUMBER

Like the hexadecimal system, the octal system provides a convenient way to express binary numbers and codes. However, it is used less frequently than hexadecimal in conjunction with computers and microprocessors to express binary quantities for input and output purposes.

The octal system is composed of eight digits, which are:0, 1, 2, 3, 4, 5, 6, 7

To count above 7, begin another column and start over: 10, 11, 12, 13, 14, 15, 16, 17, 20, 21 and so on. Counting in octal is similar to counting in decimal, except that the digits 8 and 9 are not used.

ALGORITHM

Algorithm

- Set of step-by-step instructions that perform a specific task or operation
- –Natural‖ language NOT programming language

Pseudocode

- Set of instructions that mimic programming language instructions

Flowchart

- Visual program design tool
- –Semantic‖ symbols describe operations to be performed

FLOWCHARTS

Definitions:

A flowchart is a schematic representation of an algorithm or a stepwise process, showing the steps as boxes of various kinds, and their order by connecting these with arrows. Flowcharts are used in designing or documenting a process or program.

A flow chart, or flow diagram, is a graphical representation of a process or system that details the sequencing of steps required to create output.

A flowchart is a picture of the separate steps of a process in sequential order.

TYPES:

High – Level Flowchart

A high-level (also called first-level or top-down) flowchart shows the major steps in a process. It illustrates a "birds-eye view" of a process, such as the example in the figure entitled High-Level Flowchart of Prenatal Care. It can also include the intermediate outputs of each step (the product or service produced), and the sub-steps involved. Such a flowchart offers a basic picture of the process and identifies the changes taking place within the process. It is significantly useful for identifying appropriate team members (those who are involved in the process) and for developing indicators for monitoring the process because of its focus on intermediate outputs.

Most processes can be adequately portrayed in four or five boxes that represent the major steps or activities of the process. In fact, it is a good idea to use only a few boxes, because doing so forces one to consider the most important steps. Other steps are usually sub-steps of the more important ones.

Detailed Flowchart

The detailed flowchart provides a detailed picture of a process by mapping all of the steps and activities that occur in the process. This type of flowchart indicates the steps or activities of a process and includes such things as decision points, waiting periods, tasks that frequently must be redone (rework), and feedback loops. This type of flowchart is useful for examining areas of the process in detail and for looking for problems or areas of inefficiency. For example, the Detailed Flowchart of Patient Registration reveals the delays that result when the record clerk and clinical officer are not available to assist clients.

Deployment or Matrix Flowchart

A deployment flowchart maps out the process in terms of who is doing the steps. It is in the form of a matrix, showing the various participants and the flow of steps among these participants. It is chiefly useful in identifying who is providing inputs or services to whom, as well as areas where different people may be needlessly doing the same task. See the Deployment of Matrix Flowchart.

ADVANTAGES OF USING FLOWCHARTS

The benefits of flowcharts are as follows:

1. **Communication**: Flowcharts are better way of communicating the logic of a system to all concerned.
2. **Effective analysis**: With the help of flowchart, problem can be analysed in more effective way.
3. **Proper documentation**: Program flowcharts serve as a good program documentation, which is needed for various purposes.

4. **Efficient Coding**: The flowcharts act as a guide or blueprint during the systems analysis and program development phase.
5. **Proper Debugging**: The flowchart helps in debugging process.
6. **Efficient Program Maintenance**: The maintenance of operating program becomes easy with the help of flowchart. It helps the programmer to put efforts more efficiently on that part

Advantages:

- Logic Flowcharts are easy to understand.They provide a graphical representation of actions to be taken.
- Logic Flowcharts are well suited for representing logic where there is intermingling

among many actions.

Disadvantages:

- Logic Flowcharts may encourage the use of GoTo statements leadingsoftware design that is unstructured with logic that is difficult to decipher.
- Without an automated tool, it is time-consuming to maintain Logic Flowcharts.
- Logic Flowcharts may be used during detailed logic design to specify a module.
- However, the presence of decision boxes may encourage the use of GoTo statements, resulting in software that is not structured. For this reason, Logic Flowcharts may be better used during Structural Design

LIMITATIONS OF USING FLOWCHARTS

1. **Complex logic**: Sometimes, the program logic is quite complicated. In that case, flowchart becomes complex and clumsy.
2. **Alterations and Modifications**: If alterations are required the flowchart may require re- drawing completely.
3. **Reproduction**: As the flowchart symbols cannot be typed, reproduction of flowchart becomes a problem.
4. The essentials of what is done can easily be lost in the technical details of how it is done.

GUIDELINES FOR DRAWING A FLOWCHART

Flowcharts are usually drawn using some standard symbols; however, some special symbols can also be developed when required. Some standard symbols, which are frequently required for flowcharting many computer programs.

The following are some guidelines in flowcharting:

(a) In drawing a proper flowchart, all necessary requirements should be listed out in logical order.
(b) The flowchart should be clear, neat and easy to follow. There should not be any room for ambiguity in understanding the flowchart.
(c) The usual direction of the flow of a procedure or system is from left to right or top to bottom.
(d) Only one flow line should come out from a process symbol.

(e) Only one flow line should enter a decision symbol, but two or three flow lines, one for each possible answer, should leave the decision symbol.
(f) Only one flow line is used in conjunction with terminal symbol
(g) Write within standard symbols briefly. As necessary, you can use the annotation symbol to describe data or computational steps more clearly.
(h) If the flowchart becomes complex, it is better to use connector symbols to reduce the number of flow lines. Avoid the intersection of flow lines if you want to make it more effective and better way of communication.
(i) Ensure that the flowchart has a logical *start* and f*inish.*
(j) It is useful to test the validity of the flowchart by passing through it with a simple test data.

SAMPLE PSEUDOCODE

```
ALGORITHM
  Sample
    GET Data
    WHILE There Is Data
        DO Math Operation
        GET Data
    END WHILE
END ALGORITHM
```