UMARU MUSA YAR'ADUA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE

FACULTY OF NATURAL AND APPLIED SCIENCES

CSC 1301

INTRODUCTION TO COMPUTER SCIENCE

1ST PART OF THE LECTURE NOTE

2023

INTRODUCTION

WHY THE NEED TO STUDY THIS COURSE?

Computer Science is Foundational for Every Student *We believe that computing is so fundamental to understanding and participating in society that it is valuable for every student to learn as part of a modern education. We see computer science as a liberal art, a subject that provides students with a critical lens for interpreting the world around them. Computer science prepares all students to be active and informed contributors to our increasingly technological society whether they pursue careers in technology or not. Computer science can be life-changing, not just skill training.*

Computers are a primary means of local and global communication *for billions of people. Consumers use computers to correspond with businesses, employees with other employees and customers, students with classmates and teachers, and family members and military personnel with friends and other family members. In addition to sending simple notes, people use computers to share photos, drawings, documents, calendars, journals, music, and videos. Through computers, society has instant access to information from around the globe. Local and national news, weather reports, sports scores, airline schedules, telephone directories, maps and directions, job listings, credit reports, and countless forms of educational material always are accessible. From the computer, you can make a telephone call, meet new friends, share opinions or life stories, book flights, shop, fill prescriptions, file taxes, take a course, receive alerts, and automate your home.*

As technology continues to advance, computers have become a part of everyday life. Thus, many people believe that ***computer literacy is vital*** *to success in today's world.* ***Computer literacy****, also known as* ***digital literacy,*** *involves having a current knowledge and understanding of computers and their uses. Because the requirements that determine computer literacy change as technology changes, you must keep up with these changes to remain computer literate.*

People Using Computers

A HISTORICAL OVERVIEW OF COMPUTING

The term computer was originally applied to humans who were employed to solve sometime difficult equation. Human computer were often used to compile almanac consisting of tabulated values that could be used by navigators, for example to help them quickly find the answer to complicated trigonometric equation. The task is then called computing. Unfortunately, the results produced by human computers are not only tedious to produce but were all too prone to errors. Earliest known attempt to automate this process was an attempt by mathematician called Charles Babbage 1822.

Abacus (4000 years ago to 1975)

Used by merchants throughout the ancient world. Beads represent figures (data); by moving the beads according to rules, the user can add, subtract, multiply, or divide. The abacus remained in use until a world-wide deluge of cheap pocket calculators put the abacus out of work, after being used for thousands of years.

It is still a useful learning device for the visually impaired, as well as for anyone who wants to learn the roots of the modern calculator.

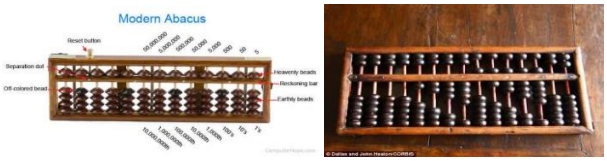


Figure 1: Typical Abacus Machines

Pascal's calculator (1642)

Over the centuries, humans built more advanced counting tools but they still required a human to manually perform the calculations. This first major step was the invention of the mechanical calculator by Blaise Pascal. A French mathematician and philosopher Blaise Pascal, the son of an accountant, invents an adding machine to relieve the tedium of adding up long columns of tax figures. The machine had a series of interlocking cogs (gear wheels with teeth around their outer edges) that could add and subtract decimal numbers. It is capable of addition and subtraction with numbers being entered by manipulating its dials.

Figure 2: Pascaline

Leibniz's calculator (1674)

German mathematician and philosopher Gottfried Wilhelm Leibniz came up with a similar but more advanced machine. Instead of using cogs, it had a "stepped drum" (a cylinder with teeth of increasing length around its edge), an innovation that survived in mechanical calculators for 300 hundred years. The Leibniz machine could do much more than Pascal's, as well as adding and subtracting, it could multiply, divide, and work out square roots. He built the first calculator to do multiplication and division.

Figure 3: Leibniz calculator

Jacquard's loom (1801)

French weaver Joseph-Marie Jacquard creates an automatic, programmable weaving machine that creates fabrics with richly detailed patterns. It is controlled by means of punched cards.

These looms took a sequence of cards with holes in them. When the loom encountered a hole, it would hook the thread underneath it. If it didn't encounter a hole, the hook wouldn't thread anything. Eventually this spun up a design pattern on fabric. These cards were known as punch cards.

Figure 4: Loom machine

Charles Babbage Difference Engine (1822):

Babbage was a gifted engineer who developed a series of machines that are now known as the greatest breakthrough on our way to modern computer. Many regard Babbage as the "father of the computer" because his machines had an input (a way of feeding in numbers), a memory (something to store these numbers while complex calculations were taking place), a processor (the number-cruncher that carried out the calculations), and an output (a printing mechanism)—the same basic components shared by all modern computers. During his lifetime, Babbage never completed a single one of the hugely ambitious machines that he tried to build.His most famous invention is the Difference Engine. It is capable of computing several sets of numbers and making hard

copies of the results. Due to a lack of funding and technological limits, he was never able to complete a full-scale functional version of this machine.

Babbage's follow up to the difference engine was a machine called the Analytical Engine. He was inspired by Jacquard's use of punch cards to automatically perform calculations instead of manually entering them by hand. Babbage used punch cards in his Analytical Engine to allow people to predefine a series of calculations they wanted to perform. As impressive as this achievement was, the Analytical engine was just a very advanced mechanical calculator. It took the powerful insights of a mathematician named Ada Lovelace to realise the true potential of the Analytical engine.

Babbage is largely remembered because of the work of Augusta Ada (Countess of Lovelace) who was probably the first computer programmer.

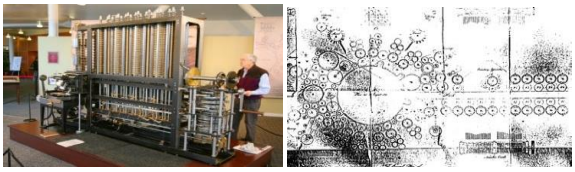


Figure 5: Difference Engine

Ada Lovelace 'Analytical Engine' (Mid 1800s)

Countess of Lovelace, the daughter of the poet, Lord Byron an enthusiastic mathematician, she helped to refine Babbage's ideas for making his machine programmable—and this is why she is still, sometimes, referred to as the world's first computer programmer. Ada programming language was named after her. She completed Babbage's Analytical engine.

She was the first person to realize that the Analytical engine could be used for more than just pure calculations. She developed the first algorithm for the engine. It was the very first example of computer programming. Because of this Lovelace's discovery that algorithms could be programmed into the analytical engine, it became the very first general purpose computing machine in history.

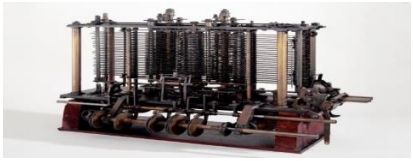


Figure 6: Analytical Engine

Hollerith's tabulating machine (1890)

Created to tally the results of the U.S. Census, this machine uses punched cards as a data input mechanism. The 1880 census took 8 years to tabulate but with the invention of Hollerith's machine the 1890 census took a year. Hollerith's Tabulating Machine Company later became part of International Business Machines (IBM).

Figure 7: Hollerith's tabulating machine

Alan Turing (1936)

He is considered to be the father of computer science, he described a theoretical device called the Turing machine or "a-machine" (a simple information processor that works through a series of instructions, reading data, writing results, and then moving on to the next instruction). Turing's ideas were hugely influential in the years that followed and many people regard him as the father of modern computing—the 20th-century's equivalent of Babbage. He also formalized the concepts of computation and algorithms. Turing later helped crack German military codes during World War II.

Zuse's Z1 (1938), and Z3 (1941)

German inventor Konrad Zuse creates the world's first programmable binary computer, in his parents' living room. In 1943, Zuse completes Z3, world's first *fully functional* programmable computer.

Howard Aiken (1944)

Howard Aiken designs Mark I, the first operational general-purpose electro-mechanical computer. Financed and built at IBM, the Mark I computer (51ft long) was based on relays (operate in milliseconds) as opposed to the use of gears. It required 3 seconds for a multiplication. Grace Hopper joined the project in July of that year and in **1947**, Aiken completed his work on the Harvard Mark II computer. He continued his work on the Mark III and the Harvard Mark IV.

Eckert and Mauchly (1946)

John Mauchly and Presper Eckert completed the Electronic Numerical Integrator and Calculator (ENIAC) at Univ of Pennsylvania.The **ENIAC** was the first electronic computer used for general purposes, such as solving numerical problems. It was used to calculate artillery firing tables for the United States Army's Ballistic Research Laboratory.

Its construction began in 1943 and was not completed until 1946. Although it was not completed until the end of World War II, the ENIAC was created to help with the war effort against German forces.

It used vacuum tubes (valves) which were completely electronic (operated in microseconds) as opposed to the relay which was electromechanical. It weighed 30 tons, used 18000 valves, and required 140 kilowatts of power and cost about $487,000. It was 1000 times faster than the Mark I multiplying in 3 milliseconds. ENIAC was a decimal machine and could not be programmed without altering its setup manually.

Von Neumann was a scientific genius and was a consultant on the ENIAC project. He formulated plans with Mauchly and Eckert for a new computer, Electronic Discrete Variable Automatic Computer (EDVAC) which was able to store programs as well as data.

This is called the stored program concept and Von Neumann is credited with it. Almost all modern computers are based on this idea and are referred to as von Neumann machines.

He also concluded that the binary system was more suitable for computers since switches have only two values. He went on to design his own computer at Princeton which was a general purpose machine.

From the 1950's, the computer age took off in full force. The years since then have been divided into periods or generations based on the technology used.

WHAT IS A COMPUTER?

A **computer** is an electronic device, operating under the control of instructions stored in its own memory, that can accept **data**, process the data according to specified rules, produce **results/information**, and store the results/information for future use.

Data and Information

Computers process data into information. **Data** is a collection of unprocessed items, which can include text, numbers, images, audio, and video. **Information** conveys meaning and is useful to people. **Data and information** are often used interchangeably; however data becomes information when it is viewed in context or in post-analysis.

While the concept of data is commonly associated with scientific research, data is collected by a huge range of organizations and institutions, including businesses (e.g., sales data, revenue, profits, stock price), governments (e.g., crime rates, unemployment rates, literacy rates) and non-governmental organizations (e.g., censuses of the number of homeless people by non-profit organizations). Data is measured, collected and reported, and analysed, to produce **information,** whereupon it can be visualized using graphs, images or other analysis tools.

CLASSIFICATION OF COMPUTERS

Computers can be classified according to

i) Data Processing (Mode of operation)

ii) Size and Speed

iii) Generation

I. Data processing (MODE OF OPERATION)

Using this classification technique, computers can be divided into Analog, Digital and Hybrid systems. They are explained as follows:

Analog Computers

Analog computers were well known in the 1940s although they are now uncommon. In such machines, numbers to be used in some calculation were represented by physical quantities - such as electrical voltages. According to the Penguin Dictionary of Computers (1970), "an analog computer must be able to accept inputs which vary with respect to time and directly apply these inputs to various devices within the computer which performs the computing operations of additions, subtraction, multiplication, division, integration and function generation…." The computing units of analog computers respond immediately to the changes which they detect in the input variables. Analog computers excel in solving differential equations and are faster than digital computers.

Digital Computers

Most computers today are digital. They represent information discretely and use a binary (two-step) system that represents each piece of information as a series of zeroes and ones. The Pocket Webster School & Office Dictionary (1990) simply defines Digital computers as "a computer using numbers in calculating." Digital computers manipulate most data more easily than analog computers. They are designed to process data in numerical form and their circuits perform directly the mathematical operations of addition, subtraction, multiplication, and division. Because digital information is discrete, it can be copied exactly but it is difficult to make exact copies of analog information.

Hybrid Computers

These are machines that can work as both analog and digital computers.It combines the best features of both types of computers, that is, it has the speed of analog computer and the memory and accuracy of digital computer. Hybrid computers are used mainly in specialized applications where both kinds of data need to be processed. Therefore, they help the user, to process both continuous and discrete data. For example a petrol pump contains a processor that converts fuel flow measurements into quantity and price values. In hospital Intensive Care Unit (ICU), an analog device is used which measures patient's blood pressure and temperature etc, which are then converted and displayed in the form of digits. Hybrid computers for example are used for scientific calculations, in defense and radar systems.

II. SIZE

Since the advent of the first computer different **types and sizes of computers** are offering different services. Computers can be as big as occupying a large building and as small as a laptop or a microcontroller in mobile & embedded systems. The four basic types of computers are.

- Supercomputer
- Mainframe Computer
- Minicomputer
- Microcomputer

Supercomputer:

The most powerful computers in terms of performance and data processing are the supercomputers. These are specialised and task specific computers used by large organisations. These computers are used for research and exploration purposes, like NASA uses supercomputers for launching space shuttles, controlling them and for space exploration purposes.

They are very expensive and very large in size. It can be accommodated in large air-conditioned rooms; some supercomputers can span an entire building.

In 1964 Seymour Cray designed the first supercomputer CDC 6600.

Uses of Supercomputer

In Pakistan and other countries Supercomputers are used by Educational Institutes like NUST (Pakistan) for research purposes. Pakistan Atomic Energy commission & Heavy Industry Taxila use supercomputers for Research purposes.

- **Space exploration:** Supercomputers are used to study the origin of the universe, the dark matters. For these studies scientists use IBM's powerful supercomputer "Roadrunner" at National Laboratory Los Alamos.
- **Earthquake studies:** Supercomputers are used to study the Earthquakes phenomenon. Besides that, supercomputers are used for natural resources exploration, like natural gas, petroleum, coal, etc.
- **Weather forecasting:** Supercomputers are used for weather forecasting, and to study the nature and extent of Hurricanes, Rainfalls, windstorms, etc.
- **Nuclear weapons testing:** Supercomputers are used to run weapon simulation that can test the Range, accuracy & impact of Nuclear weapons.

Popular Supercomputers

- IBM's Sequoia, in United States
- Fujitsu's K Computer in Japan
- IBM's Mira in United States
- IBM's SuperMUC in Germany
- NUDT Tianhe-1A in China

Mainframe computer

Although Mainframes are not as powerful as supercomputers, they are quite expensive nonetheless, and many large firms & government organizations uses Mainframes to run their business operations. The Mainframe computers can be accommodated in large air-conditioned rooms because of its size.

Super-computers are the fastest computers with large data storage capacity. Mainframes can also process and store large amount of data. Banks, educational institutions and insurance companies use mainframe computers to store data about their customers, students & insurance policy holders.

Popular Mainframe computers include:

- Fujitsu's ICL VME
- Hitachi's Z800

Minicomputer

Minicomputers are used by small businesses and firms. Minicomputers are also called as **"Midrange Computers"**. In size and power minicomputers lie between Mainframes and workstations.

These computers are not designed for a single user. In general, a minicomputer is a multiprocessing system capable of supporting from 4 to 200 users simultaneously. Individual departments of a large company or organizations use Minicomputers for specific purposes. For example, a production department can use Minicomputers for monitoring certain production process.

Popular Minicomputers include:

- K-202
- Texas Instrument TI-990
- SDS-92
- IBM Midrange computers

Microcomputer

Desktop computers, laptops, personal digital assistant (PDA), tablets and smartphones are all **types of microcomputers.** Micro-computers are widely used and are the fastest growing computers. These computers are the cheapest among the other three types of computers. They are specially designed for general usage like entertainment, education and work purposes. Well known manufacturers of Micro-computer are Dell, Apple, HP, Samsung, Sony, Toshiba, etc.

Desktop computers, Gaming Consoles, Sound & Navigation system of a car, Netbooks, Notebooks, PDA's, Tablet PC's, Smartphones, Calculators are all type of Microcomputers.

III. Computer Generations

Evolution of computer started from 16th century and resulted in the form that we see today. Thepresent-day computer, however, has also undergone rapid change during the last fifty years. This period, during which the evolution of computer took place, can be divided into five distinct phases known as *Generations of Computers*. Each phase is distinguished from others based on the type of *switching circuits* used.

First Generation Computers (1940-1956): Vacuum Tubes

The first computers used vacuum tubes for circuitry and magnetic drums for memory, and were often enormous, taking up entire rooms. They were very expensive to operate and in addition to using a great deal of electricity, the first computers generated a lot of heat, which was often the cause of malfunctions.

First generation computers relied on machine language, the lowest-level programming language understood by computers, to perform operations, and they could only solve one problem at a time, and it could take days or weeks to set-up a new problem. Input was based on punched cards and paper tape, and output was displayed on printouts.

The UNIVAC and ENIAC computers are examples of first-generation computing devices. The UNIVAC was the first commercial computer delivered to a business client, the U.S. Census Bureau in 1951.

Second Generation (1956-1963): Transistors

Transistors replaced vacuum tubes and ushered in the second generation of computers. The transistor was invented in 1947 but did not see widespread use in computers until the late 1950s. The transistor was far superior to the vacuum tube, allowing computers to become smaller, faster, cheaper, more energy-efficient and more reliable than their first-generation predecessors.

Though the transistor still generated a great deal of heat that subjected the computer to damage, it was a vast improvement over the vacuum tube. Second-generation computers still relied on punched cards for input and printouts for output.

Second-generation computers moved from cryptic binary machine language to symbolic, or assembly languages, which allowed programmers to specify instructions in words. High-level programming languages were also being developed at this time, such as early versions of COBOL and FORTRAN. These were also the first computers that stored their instructions in their memory, which moved from a magnetic drum to magnetic core technology.

The first computers of this generation were developed for the atomic energy industry.

Third Generation (1964-1971): Integrated Circuits (ICs)

The development of the integrated circuit was the hallmark of the third generation of computers. Transistors were miniaturized and placed on silicon chips, called semiconductors, which drastically increased the speed and efficiency of computers.

Instead of punched cards and printouts, users interacted with third generation computers through keyboards and monitors and interfaced with an operating system, which allowed the device to run many different applications at one time with a central program that monitored the memory. Computers for the first time became accessible to a mass audience because they were smaller and cheaper than their predecessors.

Fourth Generation (1971- Present): Microprocessors

The microprocessor brought the fourth generation of computers, as thousands of integrated circuits were built onto a single silicon chip. What in the first generation filled an entire room could now fit in the palm of the hand. The Intel 4004 chip, developed in 1971, located all the components of the computer—from the central processing unit and memory to input/output controls—on a single chip.

In 1981 IBM introduced its first computer for the home user, and in 1984 Apple introduced the Macintosh. Microprocessors also moved out of the realm of desktop computers and into many areas of life as more and more everyday products began to use microprocessors.

As these small computers became more powerful, they could be linked together to form networks, which eventually led to the development of the Internet. Fourth generation computers also saw the development of GUIs, the mouse and handheld devices.

Fifth Generation (Present and Beyond): Artificial Intelligence

Fifth generation computing devices, based on artificial intelligence, are still in development, though there are some applications, such as voice recognition, that are being used today. The use of parallel processing and superconductors is helping to make artificial intelligence a reality. Quantum computation and molecular and nanotechnology will radically change the face of computers in years to come. The goal of fifth-generation computing is to develop devices that respond to natural language input and are capable of learning and self-organization.

Classification of Computer by Purpose

The classification of computer based on purpose is divided into two 1. General purpose computer (Computer that con suite any filed such as Microcomputer) and 2. Special purpose computer (Computer that are designed for special purpose such as voting machine, ATM machine and so on).

CHARACTERISTICS AND DISADVANTAGES OF A COMPUTER

Computers of all sizes have common characteristics -- speed, reliability, storage capacity, and productivity. Computers are the foundation of business, travel, and leisure life today. Computers provide the processing speed required by all facets of society. The quick service we expect at the bank, at the grocery store, on the stock exchange, and on the Internet are dependent on the speed of computers.

Computers are extremely reliable as well. Most errors are caused by humans, not computers. Computers are capable of storing enormous amounts of data that must be located and retrieved very quickly. The capability to store and retrieve volumes of data is at the core of the Information Age.

Generally, computers have high speed, ability to store data and information. They are accurate and precise, versatile, automatic and diligent.

- **Speed**: Computer carries out every assignment and processing very fast. It works faster than calculators, four figure table, etc. it's internal speed is almost instantaneous. The speed with which computers work is generally very high compared to all other machines or system that perform similar functions.
- **Storage**: A computer would store information or data once you instruct it to do so. It stores them inside its storage devices.
- **Accuracy**: Every instruction is reliably carried out. Computers can perform operations and process data faster but with accurate results and no errors. Results can be wrong only if incorrect data is fed to the computer or a bug may be the cause of an error.
- **Versatility**: Computer is a versatile machine. They are used in various fields. They are used in Schools & Colleges, at hospitals, at government organizations and at home for entertainment & work purposes.
- **Multitasking:** Multitasking is also a computer characteristic. Computers can perform several tasks at a time. For example you can listen to songs, download movies, and prepare word documents all at the same time.
- **Communication:** Computers have the ability to communicate, but of course there needs some sort of connection (either Wired or Wireless connection). Two computers can be connected to send & receive data. Special software is used for text and video chat. Friends & family can connect over the internet and share files, photos & videos online.

Disadvantages of Using Computers

Some disadvantages of computers relate to **health risks, the violation of privacy, public safety, the impact on the labor force, and the impact on the environment.**

- **Health Risks**: Prolonged or improper computer use can lead to injuries or disorders of the hands, wrists, elbows, eyes, neck, and back. Computer users can protect themselves from these health risks through proper workplace design, good posture while at the computer, and appropriately spaced work breaks. Two behavioral health risks are computer addiction and technology overload. Computer addiction occurs when someone becomes obsessed with using a computer. Individuals suffering from technology overload feel distressed when deprived of computers and mobile devices
- **Violation of Privacy:** Nearly every life event is stored in a computer somewhere in medical records, credit reports, tax records, etc. In many instances, where personal and confidential records were not protected properly, individuals have found their privacy violated and identities stolen.
- **Public Safety:** Adults, teens, and children around the world are using computers to share publicly their photos, videos, journals, music, and other personal information. Some of these unsuspecting, innocent computer users have fallen victim to crimes committed by dangerous strangers. Protect yourself and your dependents from these criminals by being cautious in e-mail messages and on Web sites. For example, do not share information that would allow others to identify or locate you and do not disclose identification numbers, password, or other personal security details.
- **Impact on Labour Force:** Although computers have improved productivity in many ways and created an entire industry with hundreds of thousands of new jobs, the skills of millions of employees have been replaced by computers. Thus, it is crucial that workers keep their education up-to-date. A separate impact on the labor force is that some companies are outsourcing jobs to foreign countries instead of keeping their homeland labor force employed.
- **Impact on Environment:** Computer manufacturing processes and computer waste are depleting natural resources and polluting the environment. When computers are discarded in landfills, they can release toxic materials and potentially dangerous levels of lead, mercury, and flame retardants.

Green computing involves reducing the electricity consumed and environmental waste generated when using a computer. Strategies that support green computing include recycling, regulating manufacturing processes, extending the life of computers, and immediately donating or properly disposing of replaced computers. When you purchase a new computer, some retailers offer to dispose of your old computer properly.

COMPONENTS OF COMPUTER

Hardware

Hardware represents the physical and tangible components of a computer i.e. the components that can be seen and touched. Examples of Hardware are following:

- **Input devices** -- keyboard, mouse, scanner, webcam, etc.
- **Output devices** -- printer, monitor, speakers, printer, etc.
- **Secondary storage devices** -- Hard disk, CD, DVD etc.
- **Internal components** -- CPU, motherboard, RAM etc.

Software

Software is a set of programs designed to perform a well-defined function. A program is a sequence of instructions written to solve a particular problem. There are two types of software: System Software and Application Software.

System Software

The system software is collection of programs designed to operate, control, and extend the processing capabilities of the computer itself. System software are generally prepared by computer manufacturers. These software products comprise of programs written in low-level languages which interact with the hardware at a very basic level. System software serves as the interface between hardware and the end users. Some examples of system software are Operating System, Compilers, Interpreter, Assemblers, etc.

Features of system software are as follows;

- Close to system
- Fast in speed
- Difficult to design
- Difficult to understand
- Less interactive
- Smaller in size
- Difficult to manipulate and
- Generally written in low-level language.

Application Software

Application software products are designed to satisfy a particular need of a particular environment. All software applications prepared in the computer lab can come under the category of Application software. Application software may consist of a single program, such as a Microsoft's notepad for writing and editing simple text. It may also consist of a collection of programs, often called a software package, which work together to accomplish a task, such as a spreadsheet package.

Examples of Application software are following: Payroll Software, Student Record Software Inventory Management Software, Income Tax Software, Railways Reservation Software Microsoft Office Suite Software, Microsoft Word, Microsoft Excel, Microsoft Power point.

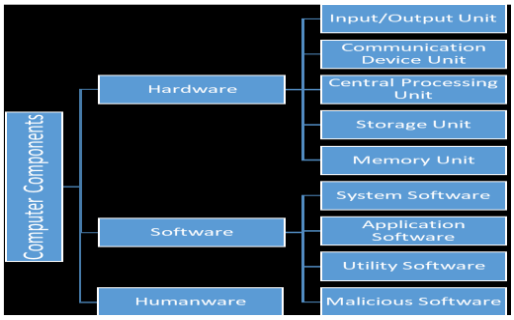
Features of application software are as follows:

- Close to user

- Easy to design and
- More interactive.

Humanware

This component refers to the person that uses the computer. More specifically, it is about the individual that makes hardware and software components productive. Typically, a great deal of testing is done on software packages and hardware parts to ensure they enhance the end-user experience to aid in creating documents, musical and video recordings, and all forms of raw and finished data.

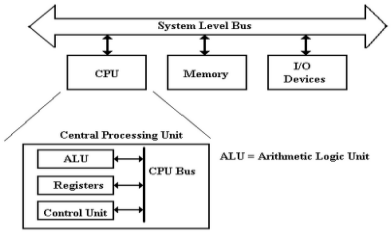


Relationship between Hardware and Software

Hardware and software are mutually dependent on each other. Both of them must work together to make a computer produce a useful output. Software cannot be utilized without supporting hardware. Hardware without set of programs to operate upon cannot be utilized and is useless. To get a particular job done on the computer, relevant software should be loaded into the hardware. Hardware is a one-time expense. Software development is very expensive and is a continuing expense. Different software applications can be loaded on hardware to run different jobs. Software acts as an interface between the user and the hardware. If hardware is the 'heart' of a computer system, then software is its 'soul'. Both are complimentary to each other.

THE BASIC COMPUTER ARCHITECTURE

It should be noted that this architecture is common to almost all computers running today, from the smallest industrial controller to the largest supercomputer. What sets the larger computers apart from the typical PC is that many larger computers are built from a large number of processor and memory modules that communicate and work cooperatively on a problem. The basic architecture is the same.



The functions of the three top-level components of a computer seem to be obvious. The **I/O devices** allow for communication of data to other devices and the users. The **memory** stores both program data and executable code in the form of binary machine language. The **CPU** comprises components that execute the machine language of the computer. Within the CPU, it is the function of the **control unit** to interpret the machine language and cause the CPU to execute the instructions as written. The **Arithmetic Logic Unit** (ALU) is that component of the CPU that does the arithmetic operations and the logical comparisons that are necessary for program execution. The ALU uses a number of local storage units, called registers, to hold results of its operations. The set of **registers** is sometimes called the register file.

DATA REPRESENTATION AND MANIPULATION

INTRODUCTION

In order for computers to process information, information must be represented in appropriate formats and stored in appropriate places. Information today comes in different forms such as text, numbers, images, audio, and video.

Computer science uses a 1 and a 0 for the two different possibilities. Hence the most basic unit of information is the binary digit (referred to as a bit of information). A bit of information can contain either a 1 or a 0. A collection of eight bits of information is generally referred to as one byte, and memory size is generally measured in the number bytes of information a computer can store.

Text

Text is represented as a bit pattern, a sequence of bits (Os or 1s). Different sets of bit patterns have been designed to represent text symbols. Each set is called a **code**, and the process of representing symbols is called coding. Today, the prevalent coding system is called **Unicode**, which uses **32 bits** to represent a symbol or character used in any language in the world. The **American Standard Code for Information Interchange (ASCII),** developed some decades ago in the United States, now constitutes the first 127 characters in Unicode and is also referred to as **Basic Latin**. It uses 7-bit strings to represent the English alphabet. Some other well-known codes are extended Binary-Coded Decimal Interchange Code (EBCDIC) uses 8-bit. The number of character that can be represented by a particular coding system depends on the bit length of its code.

Numbers

Numbers are also represented by bit patterns. However, a code such as ASCII is not used to represent numbers; the number is directly converted to a binary number to simplify mathematical operations. However, if we represent all numbers as binary numbers (rather than decimal) we can actually represent the numbers 0 to 127 with 8 bits.

The base or radix of a number system is defined as the number of digits it uses to represent the number in the system. The decimal system uses 10 as a base, and the 10 digits available are 0, 1, 2, 3 . . . 9. And binary system uses 2 as a base hence can represent only this two digits 0 and 1.

Images

Images are also represented by bit patterns. In its simplest form, an image is composed of a matrix of pixels (picture elements), where each pixel is a small dot. The size of the pixel depends on the *resolution.* For example, an image can be divided into 1000 pixels or 10,000 pixels. In the second case, there is a better representation of the image (better resolution), but more memory is needed to store the image. The collection of these encoded pixels is known as the bitmap of the image.

After an image is divided into pixels, each pixel is assigned a bit pattern. The size and the value of the pattern depend on the image. For an image made of only black and white dots (e.g., a chessboard), a 1-bit pattern is enough to represent a pixel. If an image is not made of pure white and pure black pixels, you can increase the size of the bit pattern to include grey scale. For example, to show four levels of grey scale, you can use 2-bit patterns. A black pixel can be represented by 00, a dark grey pixel by 01, a light grey pixel by 10, and a white pixel by 11.

There are several methods to represent colour images. One method is called RGB, so called because each colour is made of a combination of three primary colours: *red,* green, and blue. The intensity of each colour is measured, and a bit pattern is assigned to it. Another method is called YCM, in which a colour is made of a combination of three other primary colours: yellow, cyan, and magenta.

Files that store a bitmap image can be rather large, and various compression methods have been developed to reduce their size. Graphic Interchange Format (GIF), for example, is one such method that reduces the size of a

bitmap file by reducing the number of colours that can be assigned to a pixel to 256, and the Joint Photographic Experts Group (JPEG) developed a compression method which is commonly used to compress photographs

Audio

Audio refers to the recording or broadcasting of sound or music. Audio is by nature Different from text, numbers, or images. It is continuous, not discrete. Even when we use a microphone to change voice or music to an electric signal, we create a continuous signal.

Video

Video refers to the recording or broadcasting of a picture or movie. Video can either be produced as a continuous entity (e.g., by a TV camera), or it can be a combination of images, each a discrete entity, arranged to convey the idea of motion.

THE NUMBERING SYSTEM

Number systems are the technique to represent numbers in the computer system architecture, every value that you are saving or getting into/from computer memory has a defined number system.

Computer architecture supports following number systems.

- ***Binary number system***
- ***Octal number system***
- ***Decimal number system***
- ***Hexadecimal (hex) number system***

- ***Binary Number System:*** A Binary number system has only two digits that are 0 and 1. Every number (value) represents with 0 and 1 in this number system. The base of binary number system is 2, because it has only two digits.
- ***Octal number system:*** Octal number system has only eight (8) digits from 0 to 7. Every number (value) represents with 0,1,2,3,4,5,6 and 7 in this number system. The base of octal number system is 8, because it has only 8 digits.
- ***Decimal number system:*** Decimal number system has only ten (10) digits from 0 to 9. Every number (value) represents with 0,1,2,3,4,5,6, 7,8 and 9 in this number system. The base of decimal number system is 10, because it has only 10 digits.
- ***Hexadecimal number system:*** A Hexadecimal number system has sixteen (16) alphanumeric values from 0 to 9 and A to F. Every number (value) represents with 0,1,2,3,4,5,6, 7, 8, 9, A, B, C, D, E and F in this number system. The base of hexadecimal number system is 16, because it has 16 alphanumeric values. Here A is 10, B is 11, C is 12, D is 13, E is 14 and F is 15.

Number System Conversion

Conversion from Binary to Decimal Number System

It is in other words called conversion from base 2 to base 10. The rules for conversion from binary to decimal are given below:

1. Multiply each bit by corresponding power of 2 (base).
2. Sum each product term to get a decimal equivalent number.

Note: A power of 2 is 0 for a left bit of binary point (or for a right most bit for the number that does not contain fractional part) and increase the power by one for each bit towards left and decrease power by one towards the right of binary point.

- Example 1: convert (110011)2 to decimal.

Solution:

$(110011)_2 = 1\times2^5 + 1\times2^4 + 0\times2^3 + 0\times2^2 + 1\times2^1 + 1\times2^0$

$= 32 + 16 + 0 + 0 + 2 + 1$

$= (51)10$

- Example 2: convert $(1011.101)_2$ into decimal.

$(1011.101)_2 = 1\times2^3 + 0\times2^2 + 1\times2^1 + 1\times2^0 + 1\times2^{-1} + 0\times2^{-2} + 1\times2^{-3}$

$= 8 + 0 + 2 + 1 + 0.5 + 0 + 0.125$

$= 11 + 0.5 + 0.125$

$= (11.625)10$

Conversion of hexadecimal to decimal (base 16 to base 10)

The rules for conversion from hexadecimal to decimal are as given below:

1. Multiply each digit by corresponding power of 16(base) as in decimal to binary.
2. Sum each product term to get decimal equivalent.

- Example: convert (F4C)16 into decimal.

Solution:

$F4C_{16} = F\times16^2 + 4\times16^1 + C\times16^0$

$= 15\times256 + 4\times16 + 12\times1$

$= 3840 + 64 + 12$

$= 3916$

Therefore, $F4C_{16} = 3916_{10}$

Conversion of decimal to binary (base 10 to base 2)

The rules for conversion from decimal to binary are as given below:

1. Divide the given number by 2 and note the remainder.
2. Repeatedly divide the quotient by two and note the remainder until quotient reduced to 0.
3. Collect the remainders, last obtained first and first obtained last to binary equivalent.

- Example: convert (51)10 into binary

Solution: remainder = 1

51÷2=25

25÷ 2=12 remainder = 1

12÷ 2=6 remainder = 0

6÷ 2=3 remainder = 0

3÷ 2=1 remainder = 1

1÷ 2=0 remainder = 1

Example: convert $(A1D)_{16}$ into binary.

Solution:

Using table,

A	1	D
↓	↓	↓
1010	0001	1101

Therefore, $(A1D)_{16} = (101000011101)_2$

Conversion of decimal to Octal (base 10 to base 8)

- Decimal Number is: (12345)10

8	12345	1	LSB
8	1543	7	
8	192	0	
8	24	0	
	3	3	MSB

Therefore, Octal Number is **(30071)8**

Conversion of Octal to Decimal (base 8 to base 10)

- Octal Number is: **(30071)8**

3	0	0	7	1
4	3	2	1	0
$3*8^4$	$0*8^3$	$0*8^2$	$7*8^1$	$1*8^0$
12288	0	0	56	1

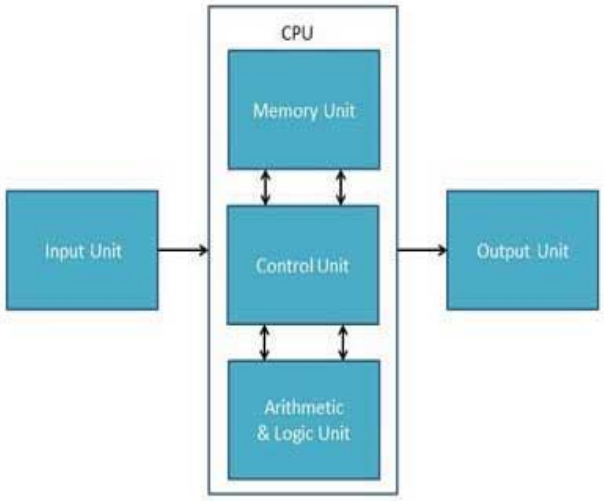
=12288+0+0+56+1

=12345

Decimal Number is: **(12345)10**

FUNCTIONAL UNITS OF A COMPUTER

All types of computers follow the same logical structure and perform the following five basic operations for converting raw input data into information.



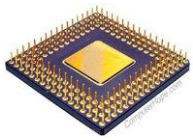
A. Input Unit

This unit contains devices with the help of which we enter data into computer. This unit makes link between user and computer. The input devices translate the information into the form understandable by computer. Examples of the input devices include; Keyboard, Mouse, Joy Stick, and Light pen, Track Ball, Scanner, Graphic Tablet, Microphone, Optical Character Reader (OCR), Bar Code Reader etc.

B. CPU (Central Processing Unit)

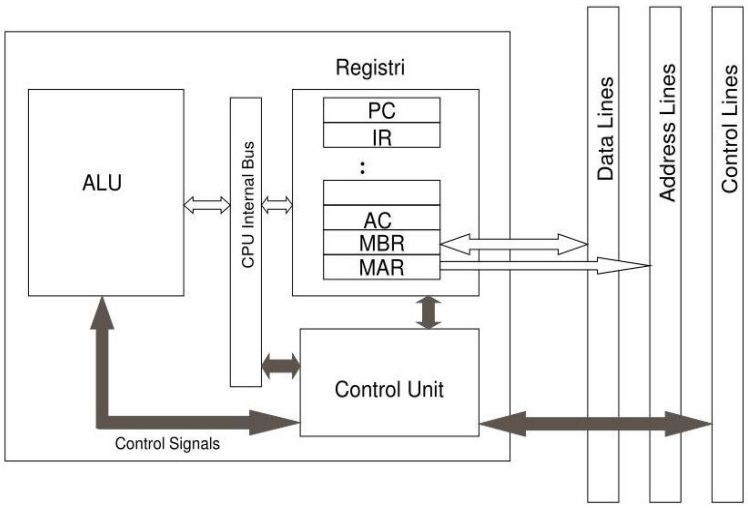
Alternately referred to as a **processor**, **central processor**, or **microprocessor**, the **CPU** (pronounced sea-pea-you) is the **central processing unit** of the computer. A computer's CPU handles all instructions it receives from hardware and software running on the computer. CPU is the component that actually executes instructions. It typically performs arithmetic and logical calculations and controls the operations of the other elements of the system. Because of this some call it the brain of computer. To be able to execute program CPU needs to access the memory, and to deal with instructions and data.

To control instructions and data flow to and from other parts of the computer, the CPU relies heavily on a chipset, which is a group of microchips located on the motherboard. Some computers utilize two or more processors. These consist of separate physical CPUs located side by side on the same board or on separate boards.



The CPU Chipset

Components of the CPU

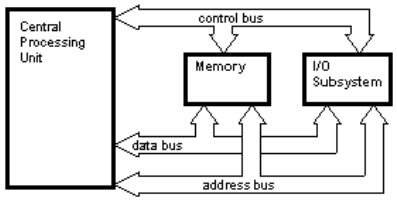


A typical CPU will consist of the following components (see Figure above):

1. ***An arithmetic/logic unit:*** The main task of the CPU is to process information. The arithmetic and logic unit (ALU) is used to perform the computer‘s data-processing functions. It is the heart of the CPU. The ALU is a sophisticated logic circuit which is made up of numerous logic gates. It performs arithmetic operations, such as addition, subtraction, multiplication and division; it also performs Boolean logical operations, such as AND, OR, NOT, and other logical operations such as comparing two numbers to see if one is greater than the other, comparing two letters to see whether they are the same, and so on.
2. ***Control unit***: Another important part of the CPU is the control unit. It is the portion of the processor that actually causes things to happen. The control unit issues control signals external to the processor, such as READ and WRITE, to cause data exchange with memory and I/O modules. It also issues control signals internal to the processor to move data between registers, to cause the ALU to perform a specific function and so on. The relationships between registers, the ALU and the control unit can be described as follows. Data are presented to the ALU in registers, and the results of operations are stored in registers. The control unit provides signals that control the operation of the ALU and the movement of data into and out of the ALU.
3. ***Registers***: A register is a small piece of electronic (or semiconductor) memory, which is used to hold certain information temporarily. Below are some typical registers including:
   - **Memory address registers (MAR):** Holds the address of the cell that the CPU is going to access.
   - **Memory buffer register (MBR):** Contains the instruction or data just read from memory, or data that is about to be written into memory.
   - **Instruction register (IR):** Holds the instruction just fetched from memory.

- **Program counter (PC):** Contains the address of the next instruction in memory, thus keeps track of current position in a machine-code program in memory.
- **Accumulator (AC):** holds temporarily the result of a calculation.
- **Control registers**: used by the control unit (to be discussed below) to control the operations of the CPU and by privileged, operation system programs to control the execution of programs.

4. ***Internal bus***: Bus is a communication line that connects elements of computer together. The CPU is connected to main memory, secondary memory and I/O devices through the system bus. The data bus is connected to the memory buffer register (MBR) and the address bus to the memory address registers (MAR). The control bus is connected to the control unit. This connection is shown in the figure below:



CPU Buses

How the CPU Runs a Program

A program residing in the memory unit of a computer consists of a sequence of instructions. These instructions are executed by the processor by going through a cycle for each instruction. This cycle is known as the **Instruction cycle**. In a basic computer, each instruction cycle consists of the following phases:

- Fetch instruction from memory.
- Decode the instruction.
- Read the effective address from memory.
- Execute the instruction.

Before we explore how a generic CPU runs a program lets look at how an instruction is formed.

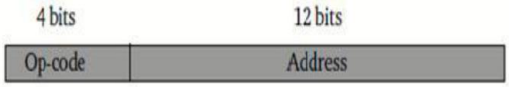
Instruction Format

Each instruction is usually consist of two parts:

An op-code: This indicates what operation is to be performed. As we just mentioned, there are four basic types of operations. These include:

- **Transfer of data between CPU and memory** (i.e. read from memory, and write to memory).
- **Transfer of data between CPU and some I/O devices** (e.g. read from a device, write to a device).
- **Data processing** (i.e. arithmetic and logical operations on data).
- **Control:** An instruction may specify that the sequence of execution be altered. For example, an instruction I (1) at address 149 may tell the CPU to execute an instruction stored at address 182. When I(1) is executed, the PC will be set to 182. Thus, on the next instruction cycle, the CPU will fetch the instruction at address 182, rather than the instruction at address 150.

Operand: This specifies the thing that is to be operated on. An operand is often an address of a cell where some real datum (number, letter, character, colour, sound pitch, etc.) is stored.

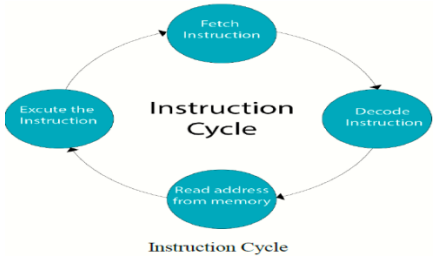


Instruction format

The Figure above shows a simple format of instructions. In this example, an instruction is 16-bit long. The first 4 bits are for the op-code, and the remaining 12 bits store one operand (an address in this case).

Instruction Cycle

The generic instruction cycle for an unspecified CPU consists of the following stages:



Instruction Cycle

- Fetch the instruction code from Memory [PC]
- Decode the instruction. This reveals that it's a multiply instruction, and that the operands are memory locations x, y, and product.
- Fetch x and y from memory.
- Multiply x and y, storing the result in a CPU register.
- Save the result from the CPU to memory location product.

1. Fetch instruction: Read instruction code from address in PC and place in IR. (IR ← Memory [PC])

2. Decode instruction: Hardware determines what the opcode/function is, and determines which registers or memory addresses contain the operands.

3. Fetch operands from memory if necessary: If any operands are memory addresses, initiate memory read cycles to read them into CPU registers. If an operand is in memory, not a register, then the memory address of the operand is known as the *effective address* or EA for short. The fetching of an operand can therefore be denoted as Register ← Memory [EA]. On today's computers, CPUs are much faster than memory, so operand fetching usually takes multiple CPU clock cycles to complete.

4. Execute: Perform the function of the instruction. If arithmetic or logic instruction utilizes the ALU circuits to carry out the operation on data in registers, this is the only stage of the instruction cycle that is useful from the perspective of the end user. Everything else is overhead required to make the execute stage happen. One of the major goals of CPU design is to eliminate overhead, and spend a higher percentage of the time in the execute stage. A detail on how this is achieved is a topic for a hardware-focused course in computer architecture.

5. Store result in memory if necessary: If destination is a memory address, initiate a memory write cycle to transfer the result from the CPU to memory. Depending on the situation, the CPU may or may not have to wait until this operation completes. If the next instruction does not need to access the memory chip where the result is stored, it can proceed with the next instruction while the memory unit is carrying out the write operation.

C. Output Unit

Output unit consists of devices with the help of which we get the information from computer. This unit is a link between computer and users. Output devices translate the computer's output into the form understandable by users. These are monitors, Graphic Plotter, Printer, Speaker etc.

D. Memory or Storage Unit

This unit can store instructions, data and intermediate results. Supply information to the other units of the computer when needed. It is known as internal storage unit or main memory or primary storage or Random access memory (RAM). Its size affects speed, power and capability. A memory unit is the amount of data that can be stored in the storage unit; that in which storage capacity is expressed in terms of Bytes. Following are the main memory storage units: Bit, Byte, Nibble and word.

Functions of the memory unit are:

- Storage of all the data and the instructions required for processing.
- Store processed result
- All inputs and outputs are transmitted through main memory.

A memory is just like a human brain. It is used to store data and instructions. Computer memory is the storage space in computer where data is to be processed and instructions required for processing are stored. The memory is divided into large number of small parts called cells. Each location or cell has a unique address which varies from zero to memory size minus one. For example if computer has 64k words, then this memory unit has 64 * 1024=65536 memory locations. The address of these locations varies from 0 to 65535.

Memory is primarily of three types

- Cache Memory
- Primary Memory/Main Memory
- Secondary Memory

Cache Memory

Cache memory is a very high speed semiconductor memory which can speed up CPU. It acts as a buffer between the CPU and main memory. It is used to hold those parts of data and program which are most frequently used by CPU. The parts of data and programs are transferred from disk to cache memory by operating system, from where CPU can access them.

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.

Cache Memory

Primary Memory (Main Memory)

Primary memory holds only those data and instructions on which computer is currently working. It has limited capacity and data is lost when power is switched off. It is generally made up of semiconductor device. These memories are not as fast as registers. The data and instruction required to be processed reside in main memory.

Characteristics of Main Memory

- These are semiconductor memories.
- It is known as main memory.
- Usually volatile memory.
- Data is lost in case power is switched off.
- It is working memory of the computer.
- Faster than secondary memories.
- A computer cannot run without primary memory.

Main Memory

Random Access Memory (RAM)

RAM (Random Access Memory) is the internal memory of the CPU for storing data, program and program result. It is read/write memory which stores data until the machine is working. As soon as the machine is switched off, data is erased. Access time in RAM is independent of the address that is, each storage location inside the memory is as easy to reach as other locations and takes the same amount of time. Data in the RAM can be accessed randomly but it is very expensive. RAM is volatile, i.e. data stored in it is lost when we switch off the computer or if there is a power failure. Hence a backup uninterruptible power system (UPS) is often used with computers. RAM is small, both in terms of its physical size and in the amount of data it can hold.

RAM is of two types

- ***Static RAM (SRAM)***

The word **static** indicates that the memory retains its contents as long as power is being supplied. However, data is lost when the power gets down due to volatile nature. SRAM chips use a matrix of 6-transistors and no capacitors. Transistors do not require power to prevent leakage, so SRAM need not have to be refreshed on a regular basis. Because of the extra space in the matrix, SRAM uses more chips than DRAM for the same amount of storage space, thus making the manufacturing costs higher. So SRAM is used as cache memory and has very fast access.

Characteristic of the Static RAM

- It has long life
- There is no need to refresh
- Faster
- Used as cache memory
- Large size
- Expensive

- High power consumption

➢ ***Dynamic RAM (DRAM)***

DRAM, unlike SRAM, must be continually **refreshed** in order to maintain the data. This is done by placing the memory on a refresh circuit that rewrites the data several hundred times per second. DRAM is used for most system memory because it is cheap and small. All DRAMs are made up of memory cells which are composed of one capacitor and one transistor.

Characteristics of the Dynamic RAM

- It has short data lifetime
- Need to be refreshed continuously
- Slower as compared to SRAM
- Used as RAM
- Lesser in size
- Less expensive
- Less power consumption

Secondary Memory

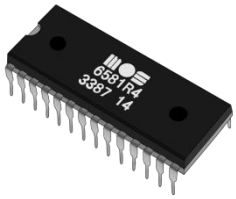
This type of memory is also known as external memory or non-volatile. It is slower than main memory. These are used for storing data/Information permanently. CPU directly does not access these memories instead they are accessed via input-output routines. Contents of secondary memories are first transferred to main memory, and then CPU can access it. For example: disk, CD-ROM, DVD etc.

Characteristic of Secondary Memory

- These are magnetic and optical memories.
- It is known as backup memory.
- It is non-volatile memory.
- Data is permanently stored even if power is switched off.
- It is used for storage of data in a computer.
- Computer may run without secondary memory.
- Slower than primary memories.

Read Only Memory

ROM stands for Read Only Memory. This is the memory from which we can only read but cannot write on it. This type of memory is non-volatile. A ROM store such instructions that are required to start a computer, this operation are referred to as bootstrap. The information is stored permanently in such memories during manufacture. ROM is divided into 3 types:



Read Only Memory (ROM)

➢ **MROM (Masked ROM)**

The very first ROMs were hardwired devices that contained a pre-programmed set of data or instructions. These kinds of ROMs are known as masked ROMs which are inexpensive.

- **PROM (Programmable Read only Memory)**

PROM is read-only memory that can be modified only once by a user. The user buys a blank PROM and enters the desired contents using a PROM program. Inside the PROM chip there are small fuses which are burnt open during programming. It can be programmed only once and is not erasable.

- **EPROM (Erasable Programmable Read Only Memory)**

EPROM will overcome the problem of PROM. EPROM chip can be programmed time and again by erasing the information stored earlier in it. EPROM chip has to be exposed to sunlight for some time so that ultra violet rays fall on the chip and that erases the data on the chip and the chip can be re-programmed using a special programming facility. There is another type memory called EEPROM that stands for Electrically Erasable Programmable Read Only Memory in which we can erase the data and re-program it with fresh content.

- **EEPROM (Electrically Erasable and Programmable Read Only Memory)**

The EEPROM is programmed and erased electrically. It can be erased and reprogrammed about ten thousand times. Both erasing and programming take about 4 to 10 ms (milli second). In EEPROM, any location can be selectively erased and programmed. EEPROMs can be erased one byte at a time, rather than erasing the entire chip.

The advantages of ROM are as follows:

- Non-volatile in nature
- These cannot be accidentally changed
- Cheaper than RAMs
- Easy to test
- More reliable than RAMs
- These are static and do not require refreshing
- Its contents are always known and can be verified

Motherboard

The motherboard serves as a single platform to connect all of the parts of a computer together. A motherboard connects CPU, memory, hard drives, optical drives, video card, sound card, and other ports and expansion cards directly or via cables. It can be considered as the backbone of a computer.

Motherboard

Features of Motherboard

A motherboard comes with following features:

- Motherboard varies greatly in supporting various types of components.
- Normally a motherboard supports a single type of CPU and few types of memories.
- Video Cards, Hard disks, Sound Cards have to be compatible with motherboard to function properly.

Motherboards, cases and power supplies must be compatible to work properly together. Popular Manufacturers include **Intel, ASUS, AOpen, ABIT, Biostar, Gigabyte, ECS and MSI.**

Description of Motherboard

The motherboard is mounted inside the case and is securely attached via small screws through pre-drilled holes. Motherboard contains ports to connect all of the internal components. It provides a single socket for CPU whereas for memory, normally one or more slots are available. Motherboards provide ports to attach floppy drive, hard drive, and optical drives via ribbon cables. Motherboard carries fans and a special port designed for power supply. There is a peripheral card slot in front of the motherboard using which video cards, sound cards and other expansion cards can be connected to motherboard.

On the left side, motherboards carry a number of ports to connect monitor, printer, mouse, keyboard, speaker, and network cables. Motherboards also provide USB ports which allow compatible devices to be connected in plug-in/plug-out fashion for example, pen drive, digital cameras etc.

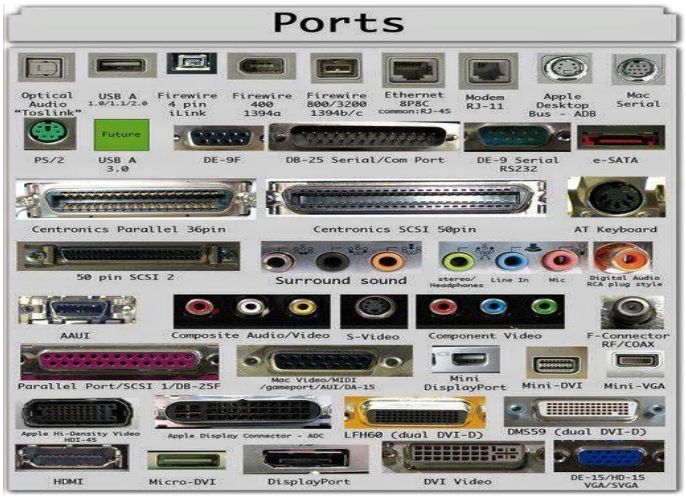
PORT

A port is a physical docking point using which an external device can be connected to the computer. It can also be programmatic docking point through which information flows from a program to the computer or over the Internet.

A port has the following characteristics −

- External devices are connected to a computer using cables and ports.
- Ports are slots on the motherboard into which a cable of external device is plugged in.
- Examples of external devices attached via ports are the mouse, keyboard, monitor, microphone, speakers, etc.

Types of ports



Serial Port

- Used for external modems and older computer mouse
- Two versions: 9 pin, 25 pin model
- Data travels at 115 kilobits per second

Parallel Port

- Used for scanners and printers

- Also called printer port
- 25 pin model
- IEEE 1284-compliant Centronics port

PS/2 Port

- Used for old computer keyboard and mouse
- Also called mouse port
- Most of the old computers provide two PS/2 port, each for the mouse and keyboard

Universal Serial Bus (or USB) Port

- It can connect all kinds of external USB devices such as external hard disk, printer, scanner, mouse, keyboard, etc.
- It was introduced in 1997.
- Most of the computers provide two USB ports as minimum.
- Data travels at 12 megabits per seconds.
- USB compliant devices can get power from a USB port.

VGA Port

- Connects monitor to a computer's video card.
- It has 15 holes.
- Similar to the serial port connector. However, serial port connector has pins, VGA port has holes.

Power Connector

- Three-pronged plug.
- Connects to the computer's power cable that plugs into a power bar or wall socket.

Firewire Port

- Transfers large amount of data at very fast speed.
- Connects camcorders and video equipment to the computer.
- Data travels at 400 to 800 megabits per seconds.
- Invented by Apple.
- It has three variants: 4-Pin FireWire 400 connector, 6-Pin FireWire 400 connector, and 9-Pin FireWire 800 connector.

Modem Port

- Connects a PC's modem to the telephone network.

Ethernet Port

- Connects to a network and high speed Internet.
- Connects the network cable to a computer.
- This port resides on an Ethernet Card.
- Data travels at 10 megabits to 1000 megabits per seconds depending upon the network bandwidth.

Game Port

- Connect a joystick to a PC
- Now replaced by USB

Digital Video Interface, DVI port

- Connects Flat panel LCD monitor to the computer's high-end video graphic cards.
- Very popular among video card manufacturers.

Sockets

- Sockets connect the microphone and speakers to the sound card of the computer.

BOOTING PROCESS

Booting is the process of starting the computer and loading the operating system. This is the process where the computer prepares itself for proper operation. When the operating system is loaded, the icons on the desktop in windows are displayed.

Booting of computer is the process of powering it on and starting the operating system. Booting loads the first piece of software that starts a computer. Because the operating system is essential for running all other programs, it is usually the first piece of software loaded during the boot process.

Types of Booting

Broadly speaking, there are two (2) main types of booting:

1. Cold Booting
2. Warm Booting

1. Cold booting is the process of starting the computer from a switched off state.

Cold boot is also a process of starting a computer from a powered-down, or off state. Cold boot is also called hard boot.

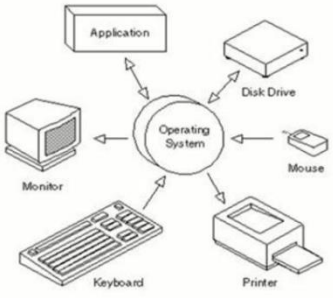
2. Warm booting this is the process of re-starting the operating system without switching on the computer.

Warm boot also refers to restarting a computer that is already turned on via the operating system. Restarting it returns the computer to its initial state. A warm boot is sometimes necessary when a program encounters an error from which it cannot recover. On PCs, you can perform a warm boot by pressing the CTRL, ALT, and Delete keys simultaneously. On Macs, you can perform a warm boot by pressing the Restart button.

OPERATING SYSTEM (OS)

What is an Operating System?

An operating system is the most important software that runs on a computer. An operating system acts as an interface between the user and the computer. It manages the computer's **memory** and **processes**, as well as all its **software** and **hardware**. It also allows you to **communicate** with the computer without knowing how to speak the computer's language. Without an operating system, a computer is useless.



Typical Devices Managed by the OS

The operating system's job

Your computer's **operating system** (**OS**) manages all the **software** and **hardware** on the computer. Most of the time, there are several different computer programs running at the same time, and they all need to access your computer's **central processing unit (CPU)**, **memory**, and **storage**. The operating system coordinates all of this to make sure each program gets what it needs.

The following are some of the services provided by the Operating System:

- Program Execution
- I/O Operations
- File System Manipulation
- Communication
- Error Detection
- Accounting
- Resource Allocation
- Protection

Components of Operating System

Operating system is basically composed of two parts:

The Kernel: This is also called the core. it provides the most basic level of control over all of the computer‘s hardware devices. It manages memory access for programs in the RAM, it determines which programs get access to which hardware resources, it sets up or resets the CPU‘s operating states for optimal operation at all times, etc. The kernel is not something, which can be used directly, although its services can be accessed through system calls.

The Shell: This is the interface between human users and the core, that why is also called User Interface. This layer just basically wraps the kernel in more acceptable clothes. This day we usually use either **Command-Line Interface (CLI)** or **Graphical User Interface (GUI**). In CLI all user instruction to a program must be typed, DOS terminal is an example. While in the GUI, visual elements like windows, icons, and menus are used for interaction.

Types of operating systems

There are many types of operating systems, the complexity of which varies depending upon what type of functions are provided, and what the system is being used for. Operating systems may be classified by both how many tasks they can perform simultaneously and by how many users can be using the system simultaneously

Operating systems usually come **pre-loaded** on any computer you buy. Most people use the operating system that comes with their computer, but it's possible to upgrade or even change operating systems.

In term of the number of users accessing computer:

- **Single-user OS** allows only one user access to the computer at a time, earlier OS that uses batch system are typical example. Operating systems such as Windows 95, Windows NT Workstation and Windows 2000 professional are essentially single user operating systems.
- **A multi-user OS** allows multiple users to access a computer system at the same time. Windows Server and UNIX all falls under this category.

In terms of how many tasks can be performed simultaneously:

- **A single-tasking OS** has only one running program
- **A multi-tasking OS** allows more than one program to be running at a time, from the point of view of human time scales.

Understanding Applications

What is an application?

You may have heard people talking about using a **program**, an **application**, or an **app**. But what exactly does that mean? Simply put, an **app** is a type of software that allows you to **perform specific tasks**. Applications for desktop or laptop computers are sometimes called **desktop applications**, while those for mobile devices are called **mobile apps**.

When you open an application, it runs inside the **operating system** until you close it. Most of the time, you will have more than one application open at the same time, which is known as **multitasking**.

- ***App** is a common term for an **application**, especially for **simple applications** that can be downloaded **inexpensively** or even **for free**. Many apps are also available for **mobile devices** and even some **TVs**.*

Desktop applications

There are countless desktop applications, and they fall into several categories. Some are more **full featured** (like **Microsoft Word**), while others may only do **one or two things** (like a **clock** or **calendar** app). Below are just a few types of applications you might use.

- **Word processors:** A word processor allows you to write, edit and store letters, design a flyer, and create many other types of documents. The most well-known word processor is **Microsoft Word**.
- **Web browsers:** A **web browser** is the tool you use to access the **Internet**. Most computers come with a web browser **pre-installed**, but you can also download a different one if you prefer. Examples of browsers include **Internet Explorer**, **Mozilla Firefox**, **Google Chrome**, and **Safari**.
- **Media players:** If you want to listen to **MP3s** or watch **movies** you've downloaded, you'll need to use a **media player**. **Windows Media Player** and **iTunes** are popular media players.
- **Games:** There are many types of games you can play on your computer. They range from card games like **Solitaire** to action games like **Halo**. Many action games require a lot of **computing power**, so they may not work unless you have a powerful computer.

Mobile apps

Desktop and laptop computers are not the only devices that can run applications. You can also download apps for mobile devices like **smart phones** and **tablets**. Here are a few examples of mobile apps.

- **Gmail**: You can use the Gmail app to easily view and send emails from your mobile device. It is available for **Android** and **iOS** devices.
- **Instagram**: You can use Instagram to quickly share photos with your friends and family. It is available for **Android** and **iOS**.
- **Duolingo**: With a combination of quizzes, games, and other activities, this app can help you learn new languages. It is available for **Android** and **iOS**.

Installing new applications

Every computer and mobile device will come with some applications already built in, such as a web browser and media player. However, you can also purchase and install new apps to add more functionality.

Computer Viruses, Worms and Vaccines

Computer Viruses

Computer virus is a small software program that is designed to spread from one computer to another and to interfere with computer operation. A true virus is capable of self-replication on a machine. Virus may spread between files or disks.

- There are estimated 30,000 computer viruses in existence.

- Over 300 new ones are created each month.
- First virus was created to show loopholes in software.
- Today almost 87% of all viruses are spread through the internet.

Symptoms of Virus Attack

- Computer runs slower than usual
- Computer no longer boots up
- Screen sometimes flicker
- Speaker beeps periodically
- System crashes for no reason
- Files/directories sometimes disappear
- Denial of Service (DoS)

Types Computer Viruses

- Trojan Horse
    - Often disguised as legitimate software
    - Requires Windows to work
    - Once infected, runs in the background
- Worm
    - Spread over network connection
    - Worms replicate
    - Consumes bandwidth and overload webservers
- Macro
    - Specific to certain applications such as MS Word and Excel
    - Comprise a high percentage of the viruses
    - Automatically begins sequence of actions when program is opened
- E-mail viruses
    - An e-mail virus travels as an attachment to e-mail messages, and usually replicates itself by automatically mailing itself to dozens of people in the victim's e-mail address book.
    - Some e-mail viruses do not even required double-clicking -- they launch when you view the infected message in the preview pane of your e-mail software.

Worms

A worm is a small piece of software that uses computer networks and security holes to replicate itself. A copy of the worm scans the network for another machine that has a specific security hole. It copies itself to the new machine using the security hole, and then starts replicating from there, as well. Worms use computer time and network bandwidth when they replicate. A worm called Code Red made huge headlines in 2001. Experts predicted that this worm could clog the Internet so effectively that things would completely grind to a halt.

A worm usually exploits some sort of security hole in a piece of software or the operating system. For example, the Slammer worm (which caused mayhem in January 2003) exploited a hole in Microsoft's SQL server.

Worms normally move around and infect other machines through computer networks. Using a network, a worm can expand from a single copy incredibly quickly. The Code Red worm replicated itself more than 250,000 times in approximately nine hours on July 19, 2001.

To protect computer against viruses you need to do the following:

- Gain knowledge about viruses

- Proper configurations
- Run only necessary programs
- Install Anti-virus software and update

AntiVirus/Vaccines

Vaccines or Antivirus software is a computer program that detects, prevents, and takes action to disarm or remove malicious software programs, such as viruses and worms. New viruses, worms, and other threats are created by cyber terrorists and discovered every day. So updating antivirus software is periodically mandatory.

List of some top AntiVirus Programs which are very useful to users:

Norton AntiVirus – Norton AntiVirus is the most popular and secure virus scanner for checking boot sector records at startup. The live update feature automatically installs new updates for regular protection against viruses.

AVG Free Edition – AVG Resident Shield provides real-time protection executions of files and programs. It features a smart e-mail scanner, virus updates and virus vault for secure handling of the files which are infected by viruses. The base version for windows is Free for private and non-commercial use.

McAfee VirusScanfor Windows: This antivirus package detects all virus types, including Word and Excel macros; boot-sector infections; and file, multipartite, stealth, polymorphic, and encrypted viruses.

Kaspersky Anti-Virus Personal Pro – It is a commonly used virus protection solution offering full protection against macro viruses and unknown viruses.

ESET NOD32 Antivirus – ESET NOD32 Anti-virus is available as an anti-virus for small businesses, individuals and for large networks.

avast! H//866/+

.

.2317o.0…….me //

Edition – A free antivirus solution for scanning disk, CDs, and E-mail.

Panda Antivirus Platin0um – It is a complete virus protection package for home and business users. It comes with an easy installation and automatic protection from latest viruses.

Avira - Windows and Linux antivirus, firewall, anti-spam, recovery solutions against malware infection.

PART 2

PROBL/EM SOLVING METHOD

This is the sequential process of analyzing information related to a given situation and generating appropriate response options.

Regardless of the area of study, computer science is all about solving problems with computers. The problems that are needed to be solved can come from any real-world problem or perhaps even from the abstract world. We need to have a standard systematic approach to solving problems. Furthermore, a computer must be able to understand the instructions in order to carry them out.

The purpose of writing a program is to solve a problem. Problem solving, in general, consists of multiple steps:

1. Problem Definition.
2. Designing a solution.
3. Implementing the solution.
4. Testing the solution and fixing any problems that exist.

Although this approach applies to any kind of problem solving, it works particularly well when developing software.

1. Problem Definition: This is always the first step in solving any type of problem and it includes some sub-steps to be followed in defining a problem. Before solving any problem, someone needs to
    - Understand the Problem: Sounds obvious? Lack of attention to this step has been the cause of many misguided efforts. If we attempt to solve a problem we don't completely understand, we often end up solving the wrong problem or at least going off on improper tangents. We must understand the needs of the people who will use the solution. These needs often include understated hints that will affect our approach to the solution. Some important questions should be answered to achieve this step;
        - What input data/information is available, what does it represent, and in what format is it in?
        - Is anything missing?
        - Do I have everything that I need? o What am I going to have to compute? o What will be the output of the program; is it a picture, graph or a text?
        - How will the program interact with the computer user?
    - Break the problem into manageable pieces: After knowing what exactly the problem is, then we break the problem into manageable pieces for more understanding and then later design a solution.
2. Designing a Solution: A solution to any problem can be designed by developing an algorithm (a sequence of instruction for solving a problem). To develop an algorithm, we need to represent the instructions in some way that is understandable to a person who is trying to figure out the steps involved. Two commonly used representations for an algorithm is by using (1) pseudo code, or (2) flow charts. When developing software, we don't write one big program. We design separate pieces that are responsible for certain parts of the solution, subsequently integrating them with other parts. Our first inclination toward a solution may not be best one unless if we consider different solutions to choose the best one. There is one important sub step that must be considered in this
    - Consider alternatives to the solution and refining the solution. The earlier we consider alternatives, the easier it is to modify our approach.
3. Implementing the solution: This is the act of taking the design and putting it in a usable form. When developing a software solution to a problem, the implementation stage is the process of actually writing the program. Too often programming is thought of a writing code. But in most cases, the final implementation of the solution is one of the last and easiest steps. The act of designing the program should be more interesting and creative than the process of implementing the design in a particular programming language.
4. Testing: We test our solution to find any errors that exist so that we can fix them and improve the quality of the software. Testing efforts attempt to verify that the program correctly represents the design, which in turn provides a solution to the problem.

PROGRAM DEVELOPMENT CYCLE

This simply means planning your program using a sequence of steps. It will enable you to use your time efficiently and help you design error-free programs that produce the desired output. Generally, the more time spent on carefully planning a program results in less time spent debugging and redesigning a program.

As we write programs, we must keep in mind that the computer will only do what we instruct it to do. Because of this, we must be very careful with our instructions.

The steps involved in the PDLC are; problem specification, analysis, design, coding, testing and debugging, documentation.

1. Problem Specification

   The program development cycle always starts with the problem specification and ends with a concrete and correct program. Specifying the problem requirements forces you to state the problem clearly and gain a clear understanding of what is required for its solution. Your objective is to eliminate unimportant aspects and to focus on the root problem, and this may not be as easy as it sound.
2. Analysis

   Be sure you understand what the program should do, that is, what the output should be. Have a clear idea of what data (or input) are given and the relationship between the input and the desired output.
3. Design: Plan the solution to the problem.

   Find a logical sequence of precise steps that solve the problem. Such a sequence of steps is called an algorithm. Every detail, including obvious steps, should appear in the algorithm. In the next section, we discussed some of the popular methods used to develop the logic plan: flowcharts and pseudocode. These tools help the programmer break a problem into a sequence of small tasks which the computer can perform to solve the problem. Planning also involves using representative data to test the logic of the algorithm by hand to ensure that it is correct.
4. Coding: Translate the algorithm into a programming language.

   Coding is the technical word for writing the program. This is the process of translating the algorithm into the syntax of a given programming language of choice and entered into the computer. The programmer uses the algorithm devised in Step 2 along with knowledge of the programming language he intends to use (java, visual basic, C, C++, python, etc.).
5. Testing and debugging: Locate and remove any errors in the program.

   Testing is the process of finding errors in a program, and debugging is the process of correcting errors that are found. (An error in a program is called a bug.) As the program is typed, most programming language points out certain types of program errors. Other types of errors will be detected when the program is executed; however, many errors occurring due to typing mistakes, flaws in the algorithm, or incorrect usages of the programming language rules can be uncovered and corrected by careful detective work. An example of such an error would be using addition when multiplication was the proper operation.
6. Documentation: Organize all the material that describes the program.

   Documentation is intended to allow another person or the programmer at a later date, to understand the program. Internal documentation consists of statements in the program that are not executed, but point out the purposes of various parts of the program. Documentation might also consist of a detailed description of what the program does and how to use the program (for instance, what type of input is expected). For commercial programs, documentation includes an instruction manual. Other types of documentation are the flowchart, pseudocode, and top-down chart that were used to construct the program. Although documentation is listed as the last step in the program development cycle, it should take place as the program is being coded.

ALGORITHMS

This is a step by step procedure designed to solve a problem or perform an operation, and which (like a map or flowchart) will lead to the sought result if followed correctly. Algorithms have a definite beginning and a definite end, and a finite number of steps. An algorithm produces the same output information given the

same input information, and several short algorithms can be combined to perform complex tasks such as writing a computer program. Cookbook recipes, a diagnosis, a problem solving routine, are some common examples of simple algorithms. Algorithms are suitable for solving structured problems, however, unsuitable for problems where value judgments are required. See also heuristics and lateral thinking.
Pseudo code: Uses English-like phrases with some Visual Basic terms to outline the task.

Hierarchy charts: Show how the different parts of a program relate to each other.
Example of a simple addition algorithm:

- Step 1: Start
- Step 2: Declare variables num1, num2 and sum.
- Step 3: Read values num1 and num2.
- Step 4: Add num1 and num2 and assign the result to sum.

sum←num1+num2

- Step 5: Display sum
- Step 6: Stop

Qualities of a good algorithm

a) Inputs and outputs should be defined precisely.

b) Each step in algorithm should be clear and unambiguous.

c) Algorithm should be most effective among many different ways to solve a problem.

d) An algorithm shouldn't have computer code. Instead, the algorithm should be written in such a way that, it can be used in similar programming languages

PROGRAMMING TOOLS

What does Programming Tool mean?

Computer tools help in designing, organizing and planning thoughts. We have program design tools aimed to help a programmer to organize his/her thoughts. Normally, these tools are applied immediately after developing the algorithm. Some of the tools are hierarchy chart, pseudocode and flow charts.
A programming tool may be any software program or utility that aids software developers or programmers in creating, editing, debugging, maintaining and/or performing any programming or development-specific task. It is also known as a software development tool.
Programming tools were initially designed to support or complement programming languages by providing the functionality and features these languages did not have.
Typically, they are standalone utilities that provide or support a particular task within any phase of the development/programming cycle. For example, a debugger is a programming tool that helps programmers identify and resolve bugs within a program’s source code. Compilers, linkers, assemblers, disassemblers, load testers, performance analysts, GUI development tools and code editors are also all forms of programming tools.

Hierarchy Chart

The hierarchy chart is a modular representation of the program logic. It is like the organizational chart, where each rank and subordinate ranks are represented using the rectangle in a tree like structure. The use of symbol in

the hierarchy chart is uniform unlike the flowchart. The logic flow from the main operation to sub-operation is represented in form of a tree from beginning to the end.

Flowcharts

This is one of the main program design tools also known as process flowchart or process flow diagram. A flow chart is a step by step diagrammatic representation of the logic paths to solve a given problem. Or A flowchart is visual or graphical representation of an algorithm.

A flow chart is a type of diagram representing a process using different symbols containing information about steps or a sequence of events. Each of these symbols is linked with arrows to illustrate the flow direction of the process.

Flowcharts are a methodology used to analyse, improve, document and manage a process or program.

Flowcharts are helpful for:

a) Aiding understanding of relationships among different process steps
b) Collecting data about a particular process
c) Helping with decision making
d) Measuring the performance of a process
e) Depicting the structure of a process
f) Tracking the process flow
g) Highlighting important steps and eliminating the unnecessary steps

A flowchart consists of special geometric symbols connected by arrows. Within each symbol is a phrase presenting the activity at that step. The shape of the symbol indicates the type of operation that is to occur. For instance, the parallelogram denotes input or output. The arrows connecting the symbols are called flow lines and they show the progression in which the steps take place.

Flowcharts should “flow” from the top of the page to the bottom. Although the symbols used in flowcharts are standardized, no standards exist for the amount of detail required within each symbol.

The main advantage of using a flowchart to plan a task is that it provides a pictorial representation of the task, which makes the logic easier to follow. We can clearly see every step and how each step is connected to the next.

The major disadvantage with flowcharts is that when a program is very large, the flowcharts may continue for many pages, making them difficult to follow and modify.

Symbols used in flowcharts

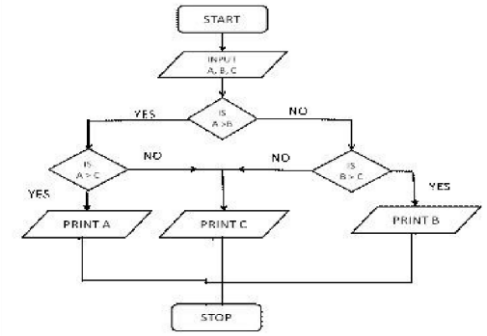
The symbols that we make use while drawing flowcharts as given below are as per conventions followed by International Standard Organization (ISO).

Name	Symbol	Description
Oval		Rectangle with rounded sides used to indicate either START/STOP of the program.
Input/Output Indicators		Parallelograms are used to represent input and output operations. Statements like INPUT, READ and PRINT
Process Indicators		Rectangle is used to indicate any set of processing operation such as for storing arithmetic operations.
Decision Maker		Diamond is used for indicating the step of decision making and is known as decision box. Are used to test the conditions or ask questions and depending upon the answers, the appropriate actions are taken by the computer
Flow Lines		This indicates the direction being followed in the flowchart. In a Flowchart, every line must have an arrow on it to indicate the direction. The arrows may be in any direction
On-page Connectors		Circles are used to join the different parts of a flowchart and these circles are called on-page connectors. In complicated problems, a flowchart may run in to several pages. The parts of the flowchart on different pages are to be joined with each other using the circles
Off-page Connectors		This connector represents a break in the path of flowchart which is too large to fit on a single page. It is similar to on-page connector. The connector symbol marks where the algorithm ends on the first page and where it continues on the second.

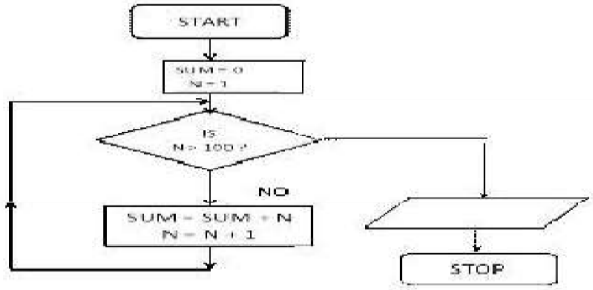
Table 1: The Flowchart Symbols

Simple Problems

Draw a flowchart to find out the biggest of the three unequal positive numbers.



Draw a flowchart for adding the integers from 1 to 100 and to print the sum.



When to use a Flowchart

a) To develop understanding of how a process is done

b) To study a process for improvement

c) To communicate to others how a process is done

d) When planning a project

Pseudocode

Pseudocode is an abbreviated version of actual computer code (hence, pseudocode). It consists of English like statements which perform the specific operations. It does not use any graphical representation. In pseudo code, the program is represented in terms of words and phrases, but the syntax of program is not strictly followed
Pseudocode allows the programmer to focus on the steps required to solve a problem rather than on how to use the computer language. The programmer can describe the algorithm java-like form without being restricted by the rules of java. When the pseudocode is completed, it can be easily translated into the java language.
Pseudocode has several advantages. It is compact and probably will not extend for many pages as flowcharts commonly do. Also, the plan looks like the code to be written and so is preferred by many programmers. It is very easy to read, understand and modify.
Example 1: Write a pseudo code to perform the basic arithmetic operations.

Read n1, n2

Sum = n1 + n2

Diff = n1 – n2

Mult = n1 * n2

Quot = n1/n2

Print sum, diff, mult, quot
End.

Write a pseudocode for finding the area of a room

.Begin process.
.input room length
.input room width

.AREA = length*width

.print AREA

.End process

COMPUTER PROGRAMMING

Computer is only an electronic machine that cannot do anything on its own unless it is directed using a program. The computer intelligence is artificial i.e. man made. The computer is built in such a way that it obeys any instruction given to it without adding or subtracting anything.
A computer is just an ingenious arrangement of metal, plastic and wire that cannot do anything except what it is directed to do by a human programmer.
A program is a series of logically and sequentially arranged instructions for solving problems. Programs are written using any of the programming languages. While a programmer is a trained specialist on computer programming languages and logic that can use any of the programming languages to direct computer on how to solve a given problem.
In many ways, programming is like a puzzle. You have task to perform (input, calculations, comparisons, rearranging of items, and output). As a programmer your job is to decompose a task into individual, ordered steps of input, calculating, comparing, rearranging and outputting. For example: suppose the task is to find the

sum of two numbers. First, your program needs to read (input the numbers into the computer). Next, your program needs to add two numbers together (calculate). Finally, your program needs to write (output) the sum.
Notice that, this program consists of steps, called instructions which are performed in order (First, Next, Finally).

Performing operations in order, one after another is called sequential processing.

The order in which the individual statements, instructions or function calls of an imperative or a declarative program are executed or evaluated by the computer is called flow of control. This is critical in programming.
You can't calculate the sum of two numbers before you have read the two numbers, and you can't output a sum before you have calculated it. Programming therefore, requires the programmer to specify the ordering of instructions (flow of control) of the program.
There are four different ways that the flow of control can progress through a program.

- Sequential execution
- Method call
- Selection
- Looping

Because getting the flow of control correct is essential to getting a program to produce correct output. Programmers use a programming tool like pseudocode or algorithms to help them design the flow of control before writing the code.

PROGRAMMING LANGUAGES

Computers programs (software) are developed using a programming language. Just like human beings, the computer has languages which are made up of a set of symbols (and often group of symbols) and rule for combining the symbols so that they represent instruction for a computer to carry out a specific operation.

Types of Programming Language

Basically they are two types of Programming languages, namely:

Low-Level Languages

CPUs are designed to understand a fixed number of instructions. These instructions (that are fetched, decoded and executed) need to be represented as bit patterns in order for them to be stored and for the CPU to understand them (in the same way as data needs to be represented in binary form). The collection of instructions a CPU can understand is known as the **CPUs instruction set**. There are two representation of the low level language and they include:

a. **The machine language or the machine code**: Machine language consists of strings of binary numbers (i.e. 0s and 1s) and it is the only one language, the processor directly understands. Machine language has the advantage of very fast execution speed and efficient use of primary memory.

 Advantages

o It is directly understood by the processor so has faster execution time since the programs written in this language need not to be translated.

o It doesn‘t need larger memory

 Disadvantages

o It is very difficult to program since all the instructions are to be represented by 0s and 1s.

o Use of this language makes programming time consuming.

o It is difficult to find error and to debug.

o It can be used by experts only.

b. Assembly Language: Assembly language is also known as low-level language because to design a program programmer requires detailed knowledge of hardware specification. This language uses mnemonics code (symbolic operation code like ADD for addition, MUL for multiplication) in place of 0s and 1s. The program is converted into machine code by assembler. The resulting program is referred to as an object code.

 Advantages

o It is makes programming easier and faster than Machine language since it uses mnemonics code for programming. Eg: ADD for addition, SUB for subtraction, DIV for division, etc.

o Error can be identified and debugged much easily compared to Machine Language.

 Disadvantages

o Programs written in this language is not directly understood by computer so translators are used.

o It is a hardware dependent language so programmers are forced to think in terms of computer‘s architecture rather than to the problem being solved.

o Programmers must know its mnemonics codes to perform any task

High-level Languages

High-level languages are programming languages that allows for a program to written in form readable to human beings. High level languages were developed to overcome the limitations of Machine and assembly languages. In high level language a program is written in a form that resembles the statement of the given problem in English, then later converted into machine language by **translator programs** (interpreter or compiler). In high level languages, there are certain syntax and punctuation which must be learned but in most cases these languages are designed to be problem oriented rather than machine oriented. High level language can be further categorized as:

a. **Procedural-Oriented language**

Procedural Programming is a methodology for modeling the problem being solved, by determining the steps and the order of those steps that must be followed in order to reach a desired outcome or specific program state. These languages are designed to express the logic and the procedure of a problem to be solved. It includes languages such as Pascal, COBOL, C, FORTAN, etc.

 Advantages:

o Because of their flexibility, procedural languages are able to solve a variety of problems.

o Programmer does not need to think in term of computer architecture which makes them focused on the problem.

 Disadvantages:

- It is easier but needs higher processor and larger memory.
- It needs to be translated therefore it has more execution time.

b. **Problem-Oriented language**

It allows the users to specify what the output should be, without describing all the details of how the data should be manipulated to produce the result. This is one step ahead from procedural-oriented languages. These languages are usually result oriented and include database query language. E.g: Visual Basic, C#, Java etc. The objectives of Problem-Oriented languages are to:

- Increase the speed of developing programs.

- Minimize user‘s effort to obtain information from computer.
- Reduce errors while writing programs.

 Advantages:

- Programmer need not to think about the procedure of the program. So, programming is much easier.

 Disadvantages:

- It is easier but needs higher processor and larger memory.
- It needs to be translated also and as such, has higher execution time.

c. **Natural language**

Natural language is still in developing stage where we could write statements that would look like normal sentences and programs would be developed from those statements. This implies independence from syntax rules and other protocols needed in utilizing the programming language.

 Advantages:

- Easy to program, since, the program uses normal sentences, they are easy to understand.
- The programs designed using Natural language will have artificial intelligence (AI) and would be much more interactive and interesting.

Disadvantage:

- Demands can be high in terms of the computing resources (memory, storage and CPU)
- It is an expensive approach as it may require high end computers devices that are not easy to purchase.

Translator

A computer stores and uses information in binary format, therefore the computer cannot understand programs written in either high-level or assembly language. Program code written in either high level or assembly language must be translated to a binary machine code that the computer recognizes.

A translator is a program that translates a program written in high-level or assembly language into Machine language. The translator is resident in the main memory of the computer and uses a **high-level or low-level program as input data.** The **output from the translator** is a program in **machine-readable code**. In addition to translation a translator will report on any grammatical errors made by the programmer in the language statements of the program. Translation from **high-level** language is by a **compiler or interpreter** and from **low-level** language by **assembler**.

Types of Translators

 Assemblers

An assembler is a program that translates a computer program written in assembly language into machine code.

 Compilers and Interpreters

A compiler is a program that translates a source code written in high level language into machine code. The **compiler reads the entire piece of source code**, collecting and reorganizing the instructions. Compilation: is the process of translating a computer program from its original or source program in machine code

The **interpreter**, unlike the compiler, reads and interprets the source code **line by line** to produce the machine code.

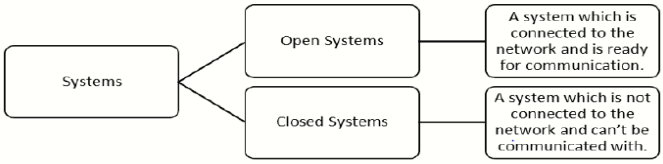
 Byte-code/Pseudo-Code

There are techniques that try to obtain some of the best of both compiled and interpreted languages. These typically compile the program to some other language that is not as high-level as the original source code, but

which is not as low-level as machine code. This intermediate level is designed to be executed very efficiently by an interpreter. This intermediate language is called byte-code or pseudo-code. The idea is used most recently in Java.

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.

NETWORKING BASICS



Computer Network

A computer network is a group of computers linked to each other that enables the computer to communicate with another computer and share their resources, data, and applications. In other words, it is the interconnection of multiple devices, generally termed as **Hosts** connected using multiple paths for the purpose of sending/receiving data or media.

Uses of Computer Network

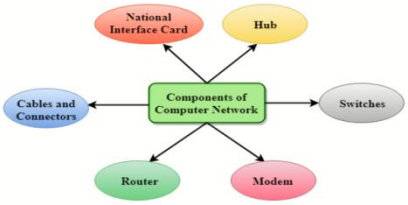
 Resource sharing: Resource sharing is the sharing of resources such as programs, printers, and data among the users on the network without the requirement of the physical location of the resource and user.

 Server-Client model: Computer networking is used in the **server-client model**. A server is a central computer used to store the information and maintained by the system administrator. Clients are the machines used to access the information stored in the server remotely.

 Communication medium: Computer network behaves as a communication medium among the users. For example, a company contains more than one computer has an email system which the employees use for daily communication. Other network based applications include WhatsApp where one can make phone call, video call and send instant messages all over the world

 E-commerce: Computer network is also important in businesses. We can do the business over the internet. For example, amazon.com is conducting business transactions over the internet, i.e., they are doing their business over the internet.

Components of Computer Network



Major components of a computer network are:

a. NIC (National Interface Card)

NIC is a device that helps the computer to communicate with another device. The network interface card contains the hardware addresses so that it transfers the data to the correct destination.

There are two types of NIC: wireless NIC and wired NIC.

 Wireless NIC: All the modern laptops use the wireless NIC. In Wireless NIC, a connection is made using the antenna that employs the **radio wave technology**.

 Wired NIC: Cables use the **wired NIC** to transfer the data over the medium.

b. Hub

Hub is a central device that splits the network connection into multiple devices. When computer requests for information from a computer, it sends the request to the Hub. Hub distributes this request to all the interconnected computers.

c. Switches

Switch is a networking device that groups all the devices over the network to transfer the data to another device. A switch is better than Hub as it does not broadcast the message over the network, i.e., it sends the message to the device for which it belongs to. Therefore, we can say that switch sends the message directly from source to the destination.

d. Cables and Connectors

Cable is a transmission media that transmits the communication signals. **There are three types of cables:**

- **Twisted pair cable:** It is a high-speed cable that transmits the data over **1Gbps** or more.
- **Coaxial cable:** Coaxial cable resembles like a TV installation cable. Coaxial cable is more expensive than twisted pair cable, but it provides the high data transmission speed.
- **Fibre optic cable:** Fibre optic cable is a high-speed cable that transmits the data using light beams. It provides high data transmission speed as compared to other cables. It is more expensive as compared to other cables.

e. Router

Router is a device that connects the Local Area Network to the internet. The router is mainly used to connect the distinct networks or connect the internet to multiple computers.

f. Modem

Modem connects the computer to the internet over the existing telephone line. A modem is not integrated with the computer motherboard. A modem is a separate part on the PC slot found on the motherboard. Based on the differences in speed and transmission rate, a modem can be classified in the following categories:

 Standard PC modem or Dial-up modem

 Cellular Modem

 Cable modem

Types of Computer Network

A computer network as earlier mentioned is a group of computers linked to each other that enables the computer to communicate with another computer. However, the network formed by the computer can be categorized by their sizes. These include:

 Local Area Network (LAN)

 Personal Area Network (PAN)

 Metropolitan Area Network (MAN)

 Wide Area Network (WAN)

We will now take a brief look at each of these network types

a. Local Area Network (LAN)

Local Area Network is a group of computers connected to each other in a small area such as building, office. It is used for connecting two or more personal computers through a communication medium such as twisted pair, coaxial cable, etc. It is less costly as it is built with inexpensive hardware such as hubs, network adapters, and Ethernet cables. The data is transferred at an extremely faster rate and provides higher security.

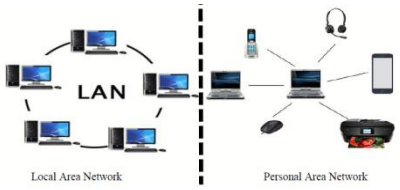
b. Personal Area Network (PAN)

Thomas Zimmerman was the first research scientist to bring the idea of the Personal Area Network. PAN is used for connecting the computer devices of personal use thus ―Personal Area Network‖. It is usually arranged within an individual person, typically within a range of 10 meters but can cover an area of **30 feet**. Personal computer devices used to develop the personal area network are the laptop, mobile phones, media player and play station.

There are two types of Personal Area Network:

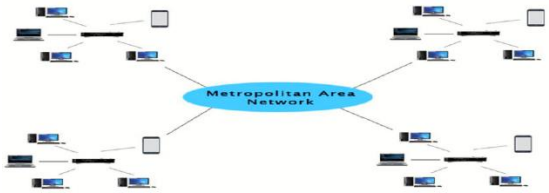
i. Wired Personal Area Network: Wired Personal Area Network is created by using the USB

ii. Wireless Personal Area Network: Wireless Personal Area Network is developed by simply using wireless technologies such as Wi-Fi, Bluetooth. It is a low range network.



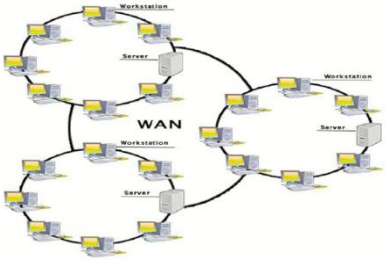
c. Metropolitan Area Network (MAN)

A metropolitan area network is a network that covers a larger geographic area by interconnecting a different LAN to form a larger network. Government agencies use MAN to connect to the citizens and private industries. In MAN, various LANs are connected to each other through a telephone exchange line. It has a higher range than Local Area Network (LAN) and as such MAN is used in communication between the banks in a city, Airports, colleges within a city, and even communication in the military zones or cantonments.



d. Wide Area Network (WAN)

Wide Area Network is a network that extends over a large geographical area such as states or countries. WAN is quite bigger network than the LAN and MAN and it is not limited to a single location, but it spans over a large geographical area through a telephone line, fibre optic cable or satellite links. The **internet** is one of the biggest WAN in the world. A Wide Area Network is widely used in the field of Business, government, and education.



NETWORK TOPOLOGY

Network Topology refers to layout of a network. In other words, it defines the structure of the network of how all the components are interconnected to each other.

Types of Network Topology

The topologies include: The **Bus, Ring, Star, Tree, Mesh, and Hybrid topologies.**

a. Bus Topology

The bus topology is designed in such a way that all the stations are connected through a single cable known as a backbone cable. Each node is either connected to the backbone cable by drop cable or directly connected to the backbone cable. When a node wants to send a message over the network, it puts a message over the network. All the stations available in the network will receive the message whether it has been addressed or not. The configuration of a bus topology is quite simpler as compared to other topologies. The backbone cable is considered as a **"single lane"** through which the message is broadcast to all the stations.

b. Ring Topology

Ring topology is like a bus topology, but with connected ends. The node that receives the message from the previous computer will retransmit to the next node. The data flows in one direction, i.e., it is unidirectional. The data flows in a single loop continuously known as an endless loop. It has no terminated ends, i.e., each node is connected to other node and having no termination point. The data in a ring topology flow in a clockwise direction. The most common access method of the ring topology is **token passing**. In a ring topology, a token is used as a carrier. **Token passing** is a network access method data carried in token is passed from one device to another device until the destination address matches. Once the token received by the destination device, then it sends the acknowledgment to the sender.

c. Star Topology

Star topology is the most popular topology in network implementation. Star topology is an arrangement of the network in which every node is connected to the central hub, switch or a central computer. Hubs or Switches are mainly used as connection devices in a **physical star topology**. The central computer is known as a **server**, and the peripheral devices attached to the server are known as **clients**.

d. Tree Topology

Tree topology combines the characteristics of bus topology and star topology. A tree topology is a type of structure in which all the computers are connected with each other in hierarchical fashion. The top-most node in tree topology is known as a **root node**, and all other nodes are the descendants of the root node. There is only one path exists between two nodes for the data transmission. Thus, it forms a **parent-child hierarchy**.

e. Mesh Topology

Mesh topology is an arrangement of the network in which computers are interconnected with each other through various disused connections. There are multiple paths from one computer to another computer. It does not contain the switch, hub or any central computer which acts as a central point of communication. The Internet is an example of the mesh topology. Mesh topology is mainly used for WAN implementations where communication failures are a critical concern. Mesh topology is mainly used for wireless networks.

f. Hybrid Topology

The combination of various different topologies is known as Hybrid topology. A Hybrid topology is a connection between different links and nodes to transfer the data. When two or more different topologies are combined together is termed as Hybrid topology and if similar topologies are connected with each other will not result in Hybrid topology. For example, if there exist a ring topology in one branch of Zenith bank and bus topology in another branch of Zenith bank, connecting these two topologies will result in Hybrid topology.