

Bachelor of Computer Applications (BCA)

Fundamentals of Computers & PC Tools (DBCACO102T24)

Self-Learning Material (SEM I)



Jaipur National University Centre for Distance and Online Education

**Established by Government of Rajasthan
Approved by UGC under Sec 2(f) of UGC ACT 1956
&
NAAC A+ Accredited**



TABLE OF CONTENTS

Course Introduction	i
Unit 1 Introduction, Generation and Evolution of Computers	01 – 13
Unit 2 Input and Output Devices	14 – 26
Unit 3 Number System	27 – 39
Unit 4 Memory and Storage	40 – 49
Unit 5 Hardware Components of Computer	50 – 58
Unit 6 Introduction to Programming Language	59 – 69
Unit 7 Introduction to Software	70 – 77
Unit 8 Introduction to Operating System	78 – 85
Unit 9 Memory Management	86 – 96

EXPERT COMMITTEE

Prof. Sunil Gupta
(Computer and Systems Sciences, JNU Jaipur)

Dr. Deepak Shekhawat
(Computer and Systems Sciences, JNU Jaipur)

Dr. Shalini Rajawat
(Computer and Systems Sciences, JNU Jaipur)

COURSE COORDINATOR

Heena Shrimali
(Computer and Systems Sciences, JNU Jaipur)

UNIT PREPARATION

Unit Writer(s)	Assisting & Proof Reading	Unit Editor
Swarnima Gupta (Computer and Systems Sciences, JNU Jaipur) (Unit 1, 2, 3, 4)	Rashmi Choudhary (Computer and Systems Sciences, JNU Jaipur)	Shish Dubey (Computer and Systems Sciences, JNU Jaipur)
Pawan Jakhar (Computer and Systems Sciences, JNU Jaipur) (Unit 5, 6, 7)		
Heena Shrimali (Computer and Systems Sciences, JNU Jaipur) (Unit 8, 9)		

Secretarial Assistance

Mr. Mukesh Sharma

COURSE INTRODUCTION

Welcome to "Fundamentals of Computers and PC Tools," a course specifically designed to introduce you to the essential elements of computer technology and software applications. In this course, you will embark on a journey to understand the foundational aspects of computing, which are crucial for both personal and professional success in today's technology-driven world. The course has 3 credits and divided into 9 units.

This course aims to provide a thorough understanding of computer systems, starting with the basic hardware components such as processors, memory, and storage devices. You will also delve into the software side, learning about operating systems and various applications that are integral to daily computing tasks.

As you progress through the course, you will develop practical skills in navigating and utilizing different operating systems. This includes managing files and folders, customizing your desktop environment, and understanding system settings to enhance your computing experience.

A significant focus of the course is on productivity tools, where you will gain hands-on experience with essential software applications. You will learn how to effectively use word processors, spreadsheets, and presentation tools to create, format, and manage various types of documents and data.

In addition to mastering software applications, the course will equip you with basic troubleshooting and maintenance skills. You will learn how to identify and resolve common computer issues, perform routine maintenance, and ensure your system operates smoothly and efficiently.

The course also emphasizes the importance of effective internet use and digital communication. You will become proficient in browsing the internet, utilizing search engines, managing email, and using digital tools for collaboration and communication.

Finally, you will gain an understanding of basic digital security principles. This includes learning how to protect your computer and personal information through safe browsing practices, secure password management, and recognizing potential security threats.

Course Outcomes:

At the completion of the course, a student will be able to:

1. Identify the important role of computers and why computers are essential components in business and society along with their various components.
2. Demonstrate the building up of Sequential and combinational logic from basic gates.
3. Apply different categories of programs, system software and applications. Organize and work with files and folders. Utilize the Word Processor, Worksheet and PowerPoint for various applications.
4. Assess the emerging technologies in the area like Big Data, Data Mining and Cloud Computing.
5. Bridge the fundamental concepts of computers with the present level of knowledge of the students.

Acknowledgements:

The content we have utilized is solely educational in nature. The copyright proprietors of the materials reproduced in this book have been tracked down as much as possible. The editors apologize for any violation that may have happened, and they will be happy to rectify any such material in later versions of this book.

Unit : 1

Introduction, Generation and Evolution of Computer

Learning Objectives:

- Define a computer, identify its characteristics and functions;
- Understand the advancement of computers
- Learn about the Generations of Computers
- Describe the different classifications of computers

Structure:

- Introduction
- 1.1 What is Computer?
- 1.2 Computer Organization
- 1.3 Characteristics of a computer
- 1.4 Evolution of computer
- 1.5 Generations of computer
- 1.6 Classification of computer
- 1.7 Classification of computer based on size
- 1.8 Summary
- 1.9 Keywords
- 1.10 Self-Assessment Questions
- 1.11 Case Study
- 1.12 References

Introduction

The word "computer" is taken from the verb "to compute," which implies to perform calculations. Computers have a huge, obvious impact on all aspects of our daily lives. In this module, you will learn how computers operate, how they evolved, and different classifications.

1.1 What is a Computer?

A computer is an electrical device that can receive information or data and carry out a number of tasks in line with an established set of tasks. Data or information is produced as a result of this. A computer is a device that can act on data and solve issues. It accepts the input, goes through various mathematical and logical processes to process it, and then produces the desired result.

Thus, a computer can be defined as a device of electronic equipment that converts data into information. The capabilities of a computer can also be used to characterize it. High speed, precision, diligence, adaptability, and storage are key attributes of a computer.

1.2 Computer Organization

In general, a computer performs five main tasks, which are:

- (1) Accepting instructions or data via input;
- (2) Storing data;
- (3) Processing data following user requirements;
- (4) Providing results in the form of output; and
- (5) Regulating all computer-related actions.

We go over each of these operations below.

1. **Input:** Entering data and information into the computer system.
2. **Control Unit (CU):** This unit is in charge of overseeing the input, output, processing, and storage processes. It determines when data is to be received, when it is to be stopped, where data is to be stored, etc. It handles the step-by-step processing of all computer processes.
3. **Memory Unit:** Instructions and Data are stored in the computer's memory unit.

4. **Arithmetic Logic Unit (ALU):** ALU carry out addition, subtraction, multiplication, division, logic, and comparison as its main operations.
5. **Output:** This process generates information-rich outcomes from the data.

A computer system's central processing unit (CPU) refers to both the ALU and the CU.

1.3 Characteristics of a Computer

Let's examine some of a computer's qualities to understand better why they perform such a revealing role in our lives.

A computer can typically process 3–4 million instructions per second.

- **Accuracy** -Computers display a very high level of precision. Most errors are caused by human error, whether erroneous data, improper instructions, or a chip flaw.
- **Reliability** – Computers can consistently perform the same tasks without making mistakes brought on by fatigue or boredom, which are highly common in humans.
- **Versatility:** Computers are capable of doing a wide range of tasks, from simple mathematical calculations and buying tickets to sophisticated calculations and ongoing astronomical observations. The computer will function if you input the required data and follow the instructions.
- **Storage Capacity** – Computers can store enormous amounts of data at a fraction of the price of conventional file storage. Additionally, data is protected from paper's typical wear and tear.

1.4 Evolution of Computers

Man has always needed to count things, do calculations, or analyse data.

1.4.1 Early Developments

The ABACUS, an early counting tool that used beads on rods to perform calculations. It predates modern computers by thousands of years and laid the foundation for future computing devices.

A machine is added in 1642 for the development by the French scholar “Blaise Pascal (1623-1662)” represented an important step forward in developing computer systems. Gottfried Wilhelm von Leibniz improved the formula in 1671 by adding the ability to multiply, divide, and discover the square root. “Charles Xavier Thomas, often known as Thomas of Colmar, created the first mechanical calculator that was sold commercially in 1829”. Multiplication, division, addition, and subtraction done by this desktop calculator. A series of more enlightened and better replication of mechanical calculators came after that.

The "difference engine," created by “Charles Babbage in 1822”, its a mechanical calculator that operates automatically. His "analytic machine" was a general-purpose, programmable, automatic mechanical digital computer that went into production in 1833. He earned the title of "father of the modern computer" as well.

The development of punch cards, an automatic calculating employed for the first time by “Herman Hollerith and James Powers during the 1890 U.S. census”, marked a important growth in the development of computer systems.

1.4.2 Present-Day Developments

Numerous research teams put forth a lot of effort to create an autonomous digital computer by the late 1930s. ENIAC (Electronic Numerical Integrator and Calculator) was designed in 1942 by physicist John W. Mauchly and electrical engineer J. Presper of Eckert. In 1946, ENIAC began operating. It was only digital electronic computer that could do anything at that time. Relays were replaced by vacuum tubes as the logic components. It was, therefore, thousand times speeder than its electromechanical forerunners.

However, the magnitude and complexity of ENIAC were unparalleled. EDVAC (Electronic Discrete Variable Automatic Computer), a stored-program computer, followed ENIAC in 1950. “Eckert and Mauchly” founded their own business, “Eckert- Mauchly Computer Corporation, in 1947” to produce computers for the market. The business generated by the UNIVAC I in 1951 for the US Census Bureau.

1.5 Generations of Computer

Every generation has smaller and more advanced circuitry than the one before it. Today, a computer can be used to perform several like send emails, browse the Internet, play games and type documents. Additionally, it may also be used to modify or create presentations, videos, and spreadsheets. However, this complicated system has been developing ever since the first generation of computers was introduced in the 1940s.

1.5.1 The First Generation: 1946- 1958 (Vacuum Tubes)

The 1st generation of computers was characterised by size, speed, and reliability issues. They occupied entire rooms, depend on vacuum tubes for their connection and magnetic drums for their memory. They were more expensive to execute and generated a lot of heat in addition to absorbing a large amount of electricity, which was frequently the root of failures.

The first generation of computers could only works on one problem at single time and were counting on machine language, the lowest level of programming language that computers could know. Printed pages were used for output displays, and input depended on computer cards and paper tape.

Examples:, EDVAC, , IBM-701, NIAC, IBM-650, UNIVAC

Features:

- Vacuum tube technology
- untrustworthy
- Understand Machine language only
- Expensive
- Huge size

1.5.2 The Second Generation: 1959-1964 (Transistors)

The 2nd generation of computer was powered by transistors, which replaced vacuum tubes. Vacuum tubes were surpassed by transistors, which made computers minor, quicker, less price, more, more dependable and energy-efficient than their first-generation forebears. Symbolic, or assembly, languages is used in second-generation

computers in place of the opaque binary machine language, enabling programmers to explain commands in words. High-level programming languages, such as the earliest iterations of COBOL and FORTRAN, were also being created around this time.

These were also the first computers that used magnetic core technology and replace magnetic drum for storing instructions. Few Examples are: IBM 7094, Honeywell 400, CDC 3600, CDC 1604, UNIVAC 1108

Features:

transistors used

trustworthy

Smaller in size

Generate low heat

Consumed not much electricity

Quicker

1.5.3 Third Generation: 1965-1970 (Integrated Circuits IC)

The third generation of computers, which spanned from the mid-1960s to the early 1970s, is characterized by the transition from transistors to integrated circuits (ICs) for processing and memory. Here are the key features and advancements of third-generation computers:

Features

- Integrated Circuits (ICs)
- Increased Speed and Efficiency
- Reduced Size and Cost
- Improved Reliability
- Multiprogramming and Time-Sharing
- High-Level Programming Languages
- Increased Storage Capacity

Notable Computers of the Third Generation

IBM System/360: One of the most significant and influential mainframes, the IBM System/360 series, introduced in 1964, set a standard for compatibility and performance in the computing industry.

Honeywell 6000 Series: Another notable line of mainframe computers, the Honeywell 6000 series, competed with IBM's System/360 and contributed to the advancement of computing technology.

1.5.4 The Fourth generation: 1971- today (Microprocessors)

Due to the integration of several of integrated circuits into a single silicon chip, the microprocessor gave rise up to the 4th generation of computers. All computer components, including the CPU, memory, and input/output controllers, were on a single chip thanks to the Intel 4004 chip, which was developed in 1971. IBM released its first personal computer for personal use in 1981 and Apple released the Macintosh in 1984. As these mini computers gained in power, they could be connected to each other to establish networks, which eventually resulted in the rising of the Internet. GUIs, the mouse, and handheld devices all were developed in conjunction with the fourth generation of computers. This generation used all higher-level languages, such as C and C++, DBASE, etc.

Few Examples are: IBM 4341, DEC 10, STAR 1000, PUP 11

Features:

- Very large scale integration technology used
- Affordable
- Portable and reliable
- Minor in size
- Concept of internet was introduced

1.5.5 Fifth Generation

The era from 1980 onwards marks the onset of the fifth generation in computing. While certain applications such as speech recognition have been implemented, the full potential of fifth-generation computer devices, driven by artificial intelligence, is still

in progress. Advancements in superconductors and parallel processing are aiding this development. In the near future, quantum computation, molecular science, and nanotechnology are poised to revolutionize the very structure of computing systems. Fifth-generation computing endeavors to create systems capable of learning, self-organization, and natural language interaction. It heavily relies on high-level programming languages such as C, C++, Java, .Net, among others.

Few Examples are: Desktop, Laptop, NoteBook, UltraBook, Chromebook

Features:

- ULSI technology
- Advancement of true AI
- Advancement of NLP
- Development in Parallel Processing
- Advancement in Super conductor technology
- Much user friendly interactive with multimedia features

1.6 Classification of Computer

Generally, computers are categorized into three types.

1. Analog Computers
2. Digital Computers
3. Hybrid Computers

1.6.1 Analog Computers

These types of computers use physical quantities like electric current, voltage, resistance, mechanical motions, etc., to represent numerical magnitudes. These are devices made to carry out mathematical operations on numbers shown by physical quantities.

Analogue computers are extensively employed in creating facilities where heat, pressure, or liquid flow must be continuously monitored, like petrol pumps, where they translate fuel flow measurements into quantities and prices.

The inability to store vast amounts of data, the lack of analytical capabilities, and the fact that only arithmetic operations could be performed on them are some drawbacks of using them.

1.6.2 Digital Computers

Instead of the actual quantities utilised in analogue computers, a digital computer works with data that is represented as digits. This means that its input must be precise rather than continuous and may include different sets of integers, letters, and special symbols that are expressed in the proper programming language.

1.6.3 Hybrid Computers

These computers are a cross between the two categories mentioned above. Analogue or digital data may be used or produced by a hybrid computer. It can be acquired by linking both analog and digital computer together using a hybrid interface or by having the analogue unit built into a digital computer's core CPU, which finally facilitates getting input and output directly.

Usually, hybrid computers are used for scientific applications or for industrial process control. In either scenario, users can leverage the computational capabilities of the machine to handle both discrete and continuous data through specific digital subroutines when needed, while employing analogue machines for rapid integration functions.

1.7 Classification of Computers Based on Size

The following classification would apply if we were to group computers according to their sizes:

1. Mini computer
2. Micro computer
3. Mainframe computer
4. Super computer

1.7.1 Supercomputers

A supercomputer has multiple processing units that work in parallel to increase speed. They are employed for more complex applications since they are, in fact, extremely large computers. There are a lot of super computers in India and mainly it is used by the meteorological service for weather forecasting.

1.7.2 Mainframe Computer

Though still substantial in size, it distinguishes itself from supercomputers by its ability to be interconnected with other computers to pool resources. Mainframe computers find application in research institutions, major financial institutions, airlines, and railway reservation systems where extensive databases are indispensable. The airline reservation system is an example of an everyday application. At their headquarters, the airlines have a mainframe computer where they keep data on all flights. Small computers installed in booking offices are connected to the main data bank so that the most recent flight information is available.

1.7.3 Super Mini Computers

These are hybrids of supercomputers and minicomputers. These are frequently employed as single-purpose computers to perform one processing task at a time.

1.7.4 Mini Computers

They are slower and have less storage than mainframe computers. They can also accommodate different terminals. Actually, they can accommodate as many as 100 terminals.

Minicomputers can accommodate several users thanks to their operating systems' multitasking and network features. The primary applications for these systems are in local government and education, as well as as front-end processors for main-frame computers. Additionally, word processing uses it. They are used in business for things like billing, inventory management, payroll, sales analysis, etc.

1.7.5 Micro Computers and Personal Computers

Digital computers, known as microcomputers, have processing units of one or more microprocessors, input/output devices, and enough memory to carry out operations. These devices often have a monitor, a keyboard, and storage for tapes and diskettes. They can be desktop or portable. They are mainly intended for stand-alone operation, although they can also be utilised as workstations in terminal emulation mode.

1.8 Summary

- A computer is a device that can act on data and solve issues.
- The central processing unit of a computer system refers to both ALU and the CU.
- The 2nd generation of computers was powered by transistors, which replaced vacuum tubes.
- Symbolic or assembly languages were described to 2nd generation computers in place of the hazy binary machine language, enabling programmers to define commands in words.
- The speed and effectiveness of computers were dramatically boosted as transistors were made smaller and mounted on silicon chips, often known as semiconductors.
- The integration of thousands of embedded circuits into a single silicon chip, the microprocessor, gave rise to 4th generation of computers
- 5th generation computing targets to create systems that can learn and self-organise while also responding to input in natural language
- Analogue computers are extensively employed in manufacturing facilities where temperatures, pressure, or liquid flow must be continuously monitored
- Hybrid computers are used for scientific applications or for industrial process control.
- A supercomputer has multiple processing units that work in parallel to increase speed.

- Minicomputers can accommodate several users thanks to their operating systems' multitasking and network features.

1.9 Keywords

- Iteration: the repetition of a process
- Forerunner: a person or thing that precedes the coming or development
- Reliability: performing consistently well
- handheld devices: devices that can be moveable and held in one's palm
- quantum: a required or allowed amount
- magnitude; a numerical quantity or value

1.10 Self-Assessment Questions

- 'Computers have a huge, obvious impact on all aspects of our daily lives'. Give reasons to justify this statement.
- Briefly explain the characteristics of a computer.
- Write a short note on the evolution of computers.
- How are computers in each generation different from one another?
- What purpose did the 3rd generation of computers serve?
- What is the history of computers, generation by generation?

1.11 Case Study

For a long time, computer science and programming were not taught as official disciplines in New Zealand schools; instead, the emphasis in computing classes was on teaching students how to utilise computers. The newly introduced computer science curriculum offers students the chance to explore a variety of computer science topics outside of programming, such as algorithms and complexity, human-computer interaction, encryption, artificial intelligence, formal languages, computer graphics, and more, in addition to a significantly improved programming focus.

ICT (Information and Communication technology) was defined within a wide and literal domain, when in 1995 the school curriculum adopted and included “technology” as a learning subject.

Questions

- Discuss how the school would have implemented the new ICT curriculum.
- Do you think it would be easy for junior and senior learners to adapt to the new curriculum?

1.12 References

- Computer Interfaces: Principles and Practice by Jonathan Wolpaw and Elizabeth Winter Wolpaw
- Computer Interfacing: An Introduction by Rajesh P. N. Rao

Unit : 2

Input and Output Devices

Learning Objectives:

- To understand the concept and function of computer interface
- Learn about Data Processing and Block Diagram
- Analyse the Computer System Architecture
- Learn about the different Input and Output Devices

Structure:

- Introduction
- 2.1 Function of Computer Interface
- 2.2 Data Processing
- 2.3 Block Diagram
- 2.4 Computer System Architecture
- 2.5 Input Devices
- 2.6 Output Devices
- 2.7 Summary
- 2.8 Key words
- 2.9 Self-Assessment Questions
- 2.10 References

Introduction

The art of linking computers and devices is known as computer interfacing. Even though the PC has many plug-and-play internal hardware options, connecting various external devices requires technical expertise. The interface is the link and interaction between the user, the software, and the hardware. It needs to be created, tested and then redesigned.

Hardware Interfaces: Hardware interface is the cables, sockets, plugs, and electrical impulses that pass through them. Examples include ATA/IDE, Ethernet, FireWire, USB, SCSI, and PCI.

Software/Programming Interfaces: Software interface is the computer languages, codes, and messages which are used to communicate with one another and hardware. Examples include the SMTP email protocol, IP network protocols, the Windows, Mac, and Linux operating systems, and the software drivers that turn on perimetric devices.

2. 1 Function of Computer Interface

Every interface implies a structure and function. Electronic signals engage hardware-level processes; data is read, written, sent, received, error-checked, etc. Instructions at the software level turn on the hardware. At higher levels, the data being transported or communicated might even make a function request (client/server, program to program, etc.).

User Interfaces: A "user interface" is the layout of the user's interaction with a computer. "Protocols" are the agreements, forms, and interactions between the various parts of a communications system or network. Keyboards, Mouse, commands, and menus are examples of user interfaces that you use to interact with a computer.

2.2 Data Processing

The terms "data processing system" and "information system" are often used interchangeably. Both systems handle the conversion of raw data into something more meaningful. We often call them information systems to highlight their practical value. After all, data is most useful when it's presented in a clear and informative way.

In simpler terms, these systems take unorganized data and turn it into something informative and easier to understand. They might achieve this by summarizing, organizing, or analyzing the data.

Numbers or Characters are known as Data which represents measurements from the real world in data processing. A single real-world measurement is referred to as a datum. Then, measured information is algorithmically derived, logically inferred, or statistically calculated from various data.

Although data conversion would be the most natural and accurate phrase, "data processing" can refer to any operation that transforms data between different formats.

The process of transforming information into data and then back into information is known as data processing. Imagine you're writing a sentence and when you press each key, the computer doesn't understand those keystrokes directly. Instead, it converts them into a special code, like a secret language. This code, called ASCII, uses numbers to represent each letter, symbol, and punctuation mark. This makes it much easier for the computer to handle the information. It's like having a common language for the computer and the keyboard. Finally, the computer translates this code back into readable characters that you see on the screen. So, all that typing gets transformed into a special code and then back again, allowing the computer to understand and display your message.

2.2.1 Data

Data is any information that can be processed by a computer, including facts, figures, and text. It comprises:

- Information about operations or transactions, such as sales, costs, inventories, payroll, and accounting.
- Non-operational information, including industry sales, forecasts, and macro-economic information.
- Metadata is information about the data, such as definitions from a data dictionary or a logical database design.

2.2.2 Information

These data's patterns, interactions, and affiliations may have information. For instance, analysing transaction data from retail point-of-sale systems might reveal when products are selling.

2.3 Block Diagram

Every time a computer is used, there are three fundamental stages it must go through in order to finish any task. Input, processing, and output are these. These steps are completed by a computer executing a program. A code is a set of detailed instructions that specify how the computer should process input to get the desired result.

2.3.1 Input

Gathering the data that the program requires into the computer is the focus of the input stage of computing. This is accomplished using input devices. The keyboard and mouse are the two most often utilised input methods.

2.3.2 Processing

The application includes instructions on how to handle the input. The computer uses the recently input data to execute these instructions during processing. The output quality that the computer generates after this phase depends on the program's instructions.

2.3.3 Output

The output stage is that in which machine will give out processed data in the form of information in a way that is valuable to the user. It is done using output devices.

2.4 Computer System Architecture

The computer's 'brain' is located here. All of the searching, sorting, maths, and decision-making happens here. By executing software instructions, the CPU get raw information from several input devices (such as a keyboard or Mouse) and transforms

it into meaningful information. After all that labour, the output data is transferred to printers and monitors.

The central processing unit (CPU) is a silicon chip microprocessor made up of micro electrical switches known as "transistors." Megahertz (MHz) or Gigahertz (GHz) are units used to express how quickly a processor performs its tasks. The processing speed of the computer increases with the MHz value. Today's typical CPU operates at 3 GHz or more.

2.5 Input Devices

Remember that before a computer can process data, it convert data into machine-readable form and upload into the computer by an input device in computer hardware. Several input devices will be introduced in this section.

2.5.1 Keyboard

The keyboard is the most popular input method for entering data and commands into computers. It is organised into four primary regions, each with a set of alphabetic, numeric, and function keys:

Function keys run across the top; letter keys are in the central portion; there is a numeric keypad to the right and keys for editing and cursor movement across the main section and the numeric keypad.

A QWERTY keyboard refers to the configuration of the keys on a keyboard, which is used in many nations. The name originates in the first six alphabetic keys on the top row of the alphabetic characters. Other keyboards include incorporated wrist support, while others offer additional Internet keys.

Hundreds of times every second, the computer's processor checks the keyboard to determine whether any keys have been depressed. A digital code is transmitted to the Central Processing Unit (CPU) when any key is pressed. "It is possible to transform this digital code into ASCII (American Standard Code of Information Interchange) coding."

For instance, 01100001 is generated to replace 'a' in binary digits, which stands for the letter 'a' in lower case. The upper case letter is represented by the binary code 01000001 when the shift key is simultaneously pressed.

Advantages:

- This accessory is commonly found attached to computers.
- It is a reliable means for text and number data input.
- Data can be entered quickly by a proficient typist.
- There are specialitykeyboards available.

Disadvantages:

- Mistakes might happen while entering data
- It may be quite time-consuming process
- Entering data, such as the specifics of diagrams and images, is particularly challenging

2.5.2 Mouse

The most common pointing device you will encounter is a mouse. You can slide it about the desk to adjust the on-screen cursor's position and movement. The mouse's buttons allow you to click and drag items about the screen and choose options from menus. 'Mouse click' is the sound made when a mouse button is pressed. The terms "double click," "click and drag," and "drag and drop" may be familiar to you.

Most mice compute the direction you are moving the mouse in using a little ball that is situated underneath them. Two rollers inside the mouse rotate in response to the movement of the ball; one roller looks movement in a north-south direction and the other roller records movements in an east-west direction. The computer moves the pointer based on information the mouse gives to it about how far and in which direction the ball is turning.

Advantages:

- They work well with desktop computers
- Most computer users are accustomed to using them.

Disadvantages:

- They require a flat surface area next to function

2.5.3 Trackball

Imagine a mouse flipped on its back, with a freely rolling ball where you'd normally place your hand. That's a trackball! The pointer on the screen is moved when you turn the ball with your hand. It operates with extremely minimum room requirements and frequently works in conjunction with computer-aided design. It features buttons like a regular mouse. Instead of a traditional mouse, you will frequently find a little trackball incorporated into laptop computers.

Advantages:

- Can be useful with laptops since they can be integrated into the keyboard or clipped on
- Ideal for use where flat space next to the computer is restricted.

Disadvantage: Not included as standard, thus they cost extra

2.5.4 Joystick

A joystick functions similarly to a tracker ball, with the exception that you move the stick rather than a rolling ball. Computer games require joysticks to be played. An ordinary joystick has eight directions in which it can be moved.

The computer uses the information provided by the joystick, which indicates the direction in which it is being moved, to, for instance, drive a racing vehicle on the screen. Several buttons on a joystick may also be used to initiate activities like firing a missile.

Advantage: When you shift the stick, a direction change is immediately felt.

Disadvantages:

- Joysticks are not particularly sturdy and can break easily
- Some people may find it challenging to use

2.5.5 Touch Screen

These screens serve the same purpose as concept keyboards. The screen has a grid of thin wires or light beams running over it. The computer senses where you have pressed when you contact the screen with your finger since the rays are blocked.

Since they are user-friendly and convenient, touch screens are frequently used as input devices in public locations including airports, museums, ATMs, etc.

Advantages:

- Easy to use
- No further peripherals are required beyond the touch screen display itself
- Neither computer system proficiency nor expertise required to use it.

Disadvantages:

- It is quite expensive
- Touch displays are not reliable

2.5.6 Digital Camera

A digital camera resembles a traditional camera. It makes use of several light sensors. When a photograph is taken, sensors positioned behind the lens convert the many colours that make up the image into digital signals (binary).

Most digital cameras allow you to view the image right away after you take it, and you can erase it if you don't like what you see. The image can then be kept in the RAM of the camera or transmitted to computer and edit photo using photo editing software.

The resolution of each image determines how much memory it uses. The amount of pixels that allows the image determines the resolution; the more pixels that make up the image, the sharper the photo.

Advantages:

- Photographs can be printed, distributed via email, or uploaded to websites.
- Unwanted photographs can be immediately erased.
- Editing or enhancing images is possible

Disadvantages

- They are typically more expensive than conventional cameras
- Photographs take up too much memory space

2.5.7 Scanner

Scanners act like copiers for your computer, capturing images and text from real-world objects. They use light to "read" the document. A light beam scans the surface, and a sensor detects the reflected colors. This information is then converted into a digital image made up of tiny squares called pixels.

There are two main types of scanners:

Flatbed Scanners: These are the most common and work similarly to a photocopier. They can handle larger objects and produce higher quality scans than handheld scanners.

Handheld Scanners: These compact scanners are ideal for on-the-go use. They are smaller and require you to move the scanner over the document you want to capture.

Benefits of Scanners:

Superior Accuracy and Resolution: Flatbed scanners can create images far more detailed and accurate than what you might get with a camera.

Preserving and Enhancing Information: Any physical document can be transformed into a digital file, making it easier to store, share, and edit on your computer.

Downsides to Consider:

Storage Hogs: Digital images can occupy a significant amount of storage space on your computer.

Quality In, Quality Out: The final scan quality depends heavily on the condition of the original document being scanned. A blurry or damaged document will result in a blurry or damaged digital copy.

2.5.8 Graphics Tablets

Graphics tablets are used by graphics designers and illustrators. For artists and designers, graphics tablets offer a natural way to create digital artwork. These tablets have a flat surface you draw on with a special pen, and your strokes appear instantly on the computer screen. This allows for highly precise drawings, accurate to within a hundredth of an inch.

The pen itself, called a stylus, might have buttons similar to a mouse. In some cases, a super-accurate puck replaces the pen for drawing.

A major benefit of graphics tablets is how they mimic the traditional pen-and-paper workflow. This makes them ideal for design environments where sketching ideas feels more natural. However, they're not as efficient as a regular mouse for clicking menus and options on the screen.

2.6 Output Devices

2.6.1 Monitor

In simpler terms, a monitor is your computer's window to the world. Like a TV screen, it shows you what the computer is working on. Typical monitor sizes range from 17 inches to 21 inches diagonally. The sharpness of the image depends on the number of tiny colored dots, called pixels, the monitor can display. More pixels mean a clearer picture. These colored dots come in red, green, and blue (RGB), and mix together to create millions of colors.

There are two main types of monitors: CRT (Cathode-Ray Tube) and LCD (Liquid Crystal Display). LCD monitors are lighter and thinner, making them perfect for laptops and handheld devices. They work by using liquid crystals that can change the direction of light depending on an electrical charge. This allows them to control the color and brightness of each pixel.

Here's a quick rundown of the pros and cons of monitors:

Pros:

- Relatively inexpensive
- Dependable
- Display a wide range of colors for text and graphics

Cons:

- The image disappears when you turn off the computer
- Limited amount of information can be shown at once
- Screens are made of glass and can break easily

2.6.2 Printers

Printers bring your digital world to paper. These output devices take electronic information and turn it into physical copies, allowing you to see text and images on regular paper.

For vibrant colors, there are special color printers that can even create photo-quality prints with the right paper.

Choosing the right printer depends on your needs. There are three main types to consider: laser, dot matrix, and inkjet. Each has its own advantages in terms of initial cost, ongoing ink or toner expenses, print quality, and printing speed.

2.6.3 Plotter

Imagine a printer on steroids, specifically designed for engineers, architects, and scientists. This is the world of plotters, output devices that create stunningly detailed line drawings on paper far larger than any standard printer can handle. Think intricate blueprints for buildings, detailed schematics of machines, or complex layouts for printed circuit boards – plotters bring these visions to life with exceptional clarity.

But how do they differ from your everyday printer? Unlike the ink cartridge approach, plotters utilize pens that move with pinpoint accuracy across the paper, meticulously drawing continuous lines. This movement is orchestrated by a pair of computer-controlled motors, ensuring every line is exactly where it needs to be. A clever electromagnet lifts and lowers the pen on command, allowing for intricate details and clean breaks between lines.

Plotters come in different flavors, each with its own way of handling paper:

Flatbed Plotters: The paper stays put like a loyal canvas while the pens dance across its surface.

Drum Plotters: The paper takes center stage, rotating on a cylindrical drum as the pens perform their magic.

Pinch-Roller Plotters: These versatile machines borrow elements from both flatbed and drum designs, offering flexibility for various drawing needs.

The biggest advantage of plotters lies in their ability to handle massive paper sizes. This makes them ideal for projects where expansive layouts or intricate details

demand a bigger canvas. However, there's a catch. Since they meticulously draw each line, plotters tend to be slower than their printer counterparts. Additionally, the initial investment for a plotter can be higher compared to a standard printer.

2.7 Summary

- **Computers can crunch numbers and words:** In simpler terms, data, which can be facts, numbers, or text, is the raw material that computers work with.
- **Data:** The building blocks for computers: Data is essentially the foundation for everything a computer does. It's made up of facts, numbers, and text that computers can understand and manipulate.
- **The brain of the computer:** The CPU, short for Central Processing Unit, is like the brain of a computer. It's a tiny chip packed with millions of transistors, which act like microscopic switches to perform calculations and run programs.
- **Typing it in:** The keyboard's role: The keyboard is the most common way to feed information into a computer. We use it to type data and instructions for the computer to follow.
- **Joystick vs. Trackball:** A matter of preference: Similar to a trackball, a joystick is an input device used to control movement on a screen. The key difference is that you move a stick instead of a rolling ball.
- **Drawing digitally:** The graphics tablet advantage: For graphic designers and illustrators, a graphics tablet is a powerful tool. It allows them to draw and create digital art directly on a pressure-sensitive pad that mimics the feel of traditional drawing.
- **Seeing the results:** Monitors and printers take center stage: Monitors and printers are the most popular output devices. Monitors display information visually, while printers allow you to create physical copies of data on paper.
- **Camera resolution:** How many megapixels make a difference? The resolution of a digital camera refers to the number of megapixels (millions of pixels) it captures. Generally, higher megapixel counts translate to sharper and more detailed images.

2.8 Keywords

- **Input** - The input stage is the first step in any computing process. It's all about providing the program with the data it needs to function. Essentially, it's the phase where we tell the computer what information to work with.
- **Output** - The output stage acts as the final act of the computing play. Here, the computer takes the processed data and transforms it into a user-friendly format. The key is that the information is presented in a way that's clear and useful for you, the user.
- **Pixels** - The photo on a screen is made up of several of tiny coloured bullets called pixels.

2.9 Self-Assessment Questions

- What do you get by information? How is it differ from data? Explain.
- Explain the process of input – processing - output with the help of suitable examples.
- Write a short note about the architecture of a computer system.
- What is meant by the term input device? Give three examples of input devices.

Also give possible advantages and disadvantages of the same.
- What are different types of printers? How is a plotter different from a printer?

2.10 References

- Computer Fundamental, P.K. Sinha, BPB Publications – 2004
- ICT for you, Stephon Doyle, Nelson Thornes, 2003
- Information Technology, P Evans, BPB Publications, 2000 Author's Name: Sh. Varun Kumar Vetter's Name: Prof. Dharminder Kumar

Unit : 3

Number System

Learning Objectives:

- To understand the Number system.
- To understand the binary addition, binary subtraction.
- To understand Floating Point representation of numbers.

Structure:

- 3.1 Introduction to Number System.
- 3.2 Representation of Integers
- 3.3 Representation of Fractions
- 3.4 Octal & Hexadecimal representation of numbers
- 3.5 Decimal to Binary Conversion
- 3.6 Binary Addition
- 3.7 Subtraction of Numbers
- 3.8 Two's Complement Representation of Numbers
- 3.9 Addition/ Subtraction of Numbers in 2's Complement Rotation
- 3.10 Binary Multiplication
- 3.11 Binary Division
- 3.12 Floating Point Representation of Numbers
- 3.13 Summary
- 3.14 Self-Assessment Questions
- 3.15 Case Study
- 3.16 References

Introduction to Number System

We are familiar by the decimal number system which is used in our day-to-day work. 10 digits are used for four decimal numbers. To represent these decimal digits, ten separate symbols 0, 1, 2, 3, 4, 5, 6, 7, 8 and 9 are used. But a digital computer stores, understand and manipulates information composed of only zeros and ones.

3.1. “Number System” and Description

(a) “*Binary Number System*”

Base 2. Digits used : 0, 1

(b) “*Octal Number System*”

Base 8. Digits used : 0 to 7

(c) “*HexaDecimal Number System*”

(d) Base 16. Digits used : 0 to 9, Letters used : A- F

3.2 Representation of Integers

Integers are *whole numbers* or *fixed-point numbers* by the radix point *fixed* after the least-significant bit. They are contrasted to *real numbers* or *floating-point numbers*, where the position of the radix point varies. It is vital to take note that integers and floating-point numbers are treated differently in computers. They have different representations and are process differently (e.g., floating-point numbers are process in a so-called floating-point processor). Floating-point numbers will be discussed later.

Computers utilize *a fixed number of bits* to represent an integer. The commonly-used bit-lengths for integers are 8-bit, 16-bit, 32-bit or 64-bit. Besides bit-lengths, there are two representation scheme for integers:

- “Unsigned Integers”: can represent zero and positive integers.
- “Signed Integers”: can represent zero, positive and negative integers. Three representation schemes had been proposed for signed integers:
 - “Sign-Magnitude representation”
 - “1's Complement representation”
 - “2's Complement representation”

3.3 Representation of Fractions

To signify fractions may be necessary quite often inside the computer. For example, it may be needed to represent in a computer a value like +0.610 or -0.610. To represent signed fractions, it is essential to assume the binary point just after the MSB in the bit sequence. Such numbers where the binary point is supposed to be at a fixed position in the bit sequence are called fixed-point numbers.

Unsigned fractions will have the supposed binary point at the extreme left. SM,1's complement, and 2's complement fractions will have this unreal binary point just to the right of the MS bit.

If the imaginary point is at the extreme right, at that time the number is an integer. If the imaginary binary point is at the extreme left for an unsigned number, the number is an unsigned fraction. If the binary point is to the immediate right of the MS bit, the number is a signed fraction. If the binary point is in the centre of a bit sequence, the number has an integer and a fractional part.

3.4 Octal and Hexadecimal representation of numbers

3.4.1 “Octal Number System “

- Uses Eight digits, 0,1,2,3,4,5,6,7.
- Also call base 8 number system Each position in an octal number represents a 0 power of the base 8.
- Last position in an Octal number represent a x power of the base 8.
- Ex. Octal Number : 125708

3.4.2 “Hexadecimal Number System”

- Uses ten digits and 6 letters, 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F.
- Letters represent numbers opening from 10. A = 10. B = 11, C = 12, D = 13, E = 14, F = 15.
- Also call base 16 number system
- Each place in a hexadecimal number represents a 0 power of the base 16.

- Last place in a hexadecimal number represents a x power of the base 16.
Example 16^x where x represent the last position - 1.

3.5 Decimal to Binary conversion

Let's understand the conversion with the help of an example:

change $(101.101)_2 = (?)_{10}$

- 101.101

$$= 1 \times 2^2$$

$$+ 0 \times 2^1$$

$$+ 1 \times 2^0$$

$$. 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$$

$$= 1 \times 4 + 0 \times 2 + 1 \times 1 + 1 \times (1/2) + 0 \times (1/4) + 1 \times (1/8)$$

$$= 4 + 0 + 1 + (1/2) + 0 + (1/8)$$

$$= 5 + 0.5 + 0.125$$

$$= 5.625$$

Therefore $(101.101)_2 = (5.625)_{10}$

3.6 Binary Addition

The "Binary addition" table is as follows:

A+B	SUM	CARRY
0+0	0	0
1+0	1	0
0+1	1	0
1+1	0	1

Illustration 1:Add $(1010)_2$ and $(0011)_2$

$$\begin{array}{r}
 1010 \\
 + 0011 \\
 \hline
 1101 \text{ (sum)} \\
 \hline
 \end{array}$$

The adding is manipulated above as follows.**Step 1:** The smallest amount significant bits are added, i.e. $0+1 = 1$ with a carry of 0**Step 2:** The carry in the prior is added to the next higher significant bits, i.e. $0+1+1 = 0$ with a carry 1.**Step 3:** The carry in the prior is added to the next higher significant bits, i.e. $1+0+0 = 1$ with a carry 0.**Step 4:** The preceding carry is additional to the most significant bit i.e. $0+1+0 = 1$ with a carry 0. Thus the sum is 1101.**3.7 Subtraction of Numbers***The binary subtraction table is as follows:*

A-B	DIFFERENCE	BORROW
0-0	0	0
1-0	1	0
0-1	1	1
1-1	0	0

Illustration 1:

Subtract
(0101)₂ from (1011)₂

1011 (Minuend)

0101 (Subtrahend)

0110 (Difference)

The steps are described below:

Step1: the LSB in the 1st column are 1 and 1. Hence, the difference is $1 - 1 = 0$

Step2: The column, the subtraction is perform as $1 - 0 = 1$

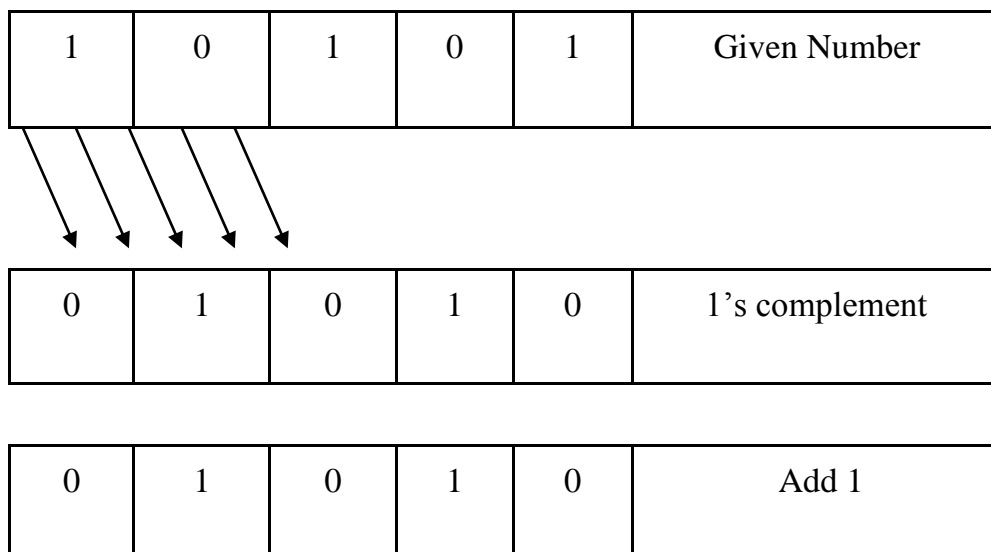
Step3: In the 3rd column, the difference is given by $0 - 1 = 1$

Step 4: In the 4th column (MSB), the difference is given by $0 - 0 = 0$ since 1 is borrowed for the third column.

3.8 Two's Complement representation of numbers

“1's complement & 2's complement”

The 1's complement of a digit is found by changing all 1's to 0's and all 0's to 1's. This is called taking a complement or 1's complement. Example of 1's Complement is as follows.



0	1	0	1	1	2's complement
---	---	---	---	---	----------------

3.9 Addition/ Subtraction of numbers in 2's complement Rotation

Addition using "2's complement"

There are three different case possible when we add two binary numbers using 2's complement, which is as follow:

Case 1: Add the positive number with a negative number when the positive number has a larger magnitude.

Initially find the 2's complement of the known negative number. Sum up with the given positive number. If we get the end-around carry 1 then the number will be a positive number and the carry bit will be leftover and remaining bits are the final result.

Example: 1101 and -1001

Find the 2's complement of the negative number 1001. So, for finding 2's complement, change all 0 to 1 and all 1 to 0 or find the 1's complement of the number 1001. The 1's complement of the number 1001 is 0110, and add 1 to the LSB of the result 0110. So the 2's complement of number 1001 is $0110+1=0111$

Addition of both the numbers, i.e., 1101 and 0111;

$$1101+0111=10100$$

By addition of both numbers, we get the end-around carry 1. We discard the end-around carry. So, the addition of both numbers is 0100.

Case 2: Adding of the positive value by a negative value when the negative number has a higher magnitude.

Initially, add a positive value with the 2's complement value of the negative number. At this time, no end-around carry is found. So, we take the 2's complement of the result to get the concluding result.

Example: 1101 and -1110

1. First, find the 2's complement of the negative number 1110. So, for finding 2's complement, add 1 to the LSB of its 1's complement value 0001.

$$0001+1=0010$$

2. Add both the numbers, i.e., 1101 and 0010;

$$1101+0010= 1111$$

3. Find the 2's complement of the result 1110 that is the final result. So, the 2's complement of the result 1110 is 0001, and add a negative sign before the number so that we can identify that it is a negative number.

Case 3: Add two negative numbers

Find the 2's complement of both the negative numbers, and then we will add both these complement numbers. In this case, we will get the end-around carry, which will be added to the LSB, and forget the final result, we will take the 2's complement of the result.

Ex . : -1101 and -1110 in five-bit register

Find the 2's complement of the negative numbers 01101 and 01110. Finding 2's complement, we add 1 to the LSB of the 1's complement of these numbers. 2's complement of the number 01110 is 10010, and 01101 is 10011.

We add the complement numbers, i.e., 10001 and 10010;

$$10010+10011= 1\ 00101$$

By adding numbers, we get the end-around carry 1. This carry is discarded and the final result is the 2's complement of the result 00101. So, the 2's complement of the result 00101 is 11011, and we add a negative sign by the number so that we can identify that it is a negative number.

“Subtraction” using 2’s complement

Following steps to subtract two binary numbers using 2's complement

- In the 1st step, find the 2's complement of the subtrahend.
- Add the complement number by the minuend.

- If we get the carry by adding both the numbers, then we throw away this carry and the result is positive, else take 2's complement of the result which will be negative.

Ex. 1: 10101 - 00111

Take 2's complement of subtrahend 00111, which is 11001. Now, Sum them. So, 10101+11001 =1 01110.

In the result, we get the carry bit 1. So we discard this carry bit and remaining is the final result and a positive number.

3.10 “Binary Multiplication”

The binary multiplication table is as follow :

A *B	PRODUCT
0 * 1	0
0 * 0	0
1 * 0	0
1 * 1	1

- ❖ Binary multiplication use add and shift process
- ❖ Binary multiplication is related to decimal multiplication.

Ex. 1:

$$\begin{array}{r}
 \text{Multiplicand * Multiplier} \\
 10110.1 \times 01001.1 = 101101 \\
 \begin{array}{r}
 101101 \\
 000000 \\
 000000 \\
 101101 \\
 000000 \\
 \hline
 011010101.11
 \end{array}
 \end{array}$$

(Final product)

The steps are described below:

stage 1: The LSB of the multiplier is taken. If the multiplier bit is 1, the multiplicand is copied as such and if the multiplier bit is 0 zero is placed in all the bit positions.

stage 2: The next higher significant bit of the multiplier is taken and the partial product is written with the shift to the left, as in step 1.

stage 3: step 2 is repeated for all other higher significant bits.

stage 4: The partial product terms are added which gives the actual product of the multiplier and the multiplicand.

3.11 “Binary Division”

The Binary division table is as follows:

A ÷ B	Result
0 ÷ 0	Not allowed
0 ÷ 1	0
1 ÷ 0	Not allowed
1 ÷ 1	1

Binary division use Subtract and Shift processes.

Binary Division is similar to Decimal division.

Division by 0 is pointless.

Illustration 1:

Dividend ÷ Divisor

$$\begin{array}{r}
 11011.1 \div 101 \\
 101.1 \quad \text{(QUOTIENT)} \\
 \text{DIVISOR } 101 \sqrt{11011.1 \text{ (DIVIDEND)}} \\
 101 \\
 \text{-----} \\
 111 \\
 101
 \end{array}$$

within the computer. The number system is divided into some categories according to the base (or radix) of the system as binary octal and hexadecimal.

3.14 Keywords

- **Integers:** Integers are *whole numbers* or *fixed-point numbers* by the radix point *fixed* after the least-significant bit.
- **Floating Point representation of numbers:** 6.6310×10^{-34} (Planck's constant). The 1st bit defines the non-zero part of the number and is called the **Mantissa**, the second part defines how many positions we want to move the decimal point, this is known as the **Exponent** and can be positive when moving the decimal point to the right and negative while moving to the left.

3.15 Self-Assessment Questions

Explain the characteristics of the Octal Number System?

Explain Binary division with the help of an example.

Explain binary addition and subtraction.

3.16 Case Study

Mr. X from school ABC organised a game of binary calculation and several students participated in the same. He provided following equation to be calculated based on the basis of binary calculation: $0 * 1 \div 1 + 1 - 0$ and asked to provide the correct answer for the same by following the BODMAS rule.

1. What is the correct answer for the same?

Answer- 1

2. What is the 1's complement of the answer?

Answer- 0

3.17 References

- Raja Raman V., "Introduction to Computers", PHI Publications
- Leon Alex & Leon Mathews, "Introduction to Computers", Vikas Publishing House
- Norton. Peter, "Introduction to Computers", TMH
- SaxenaSanjay., "A First Course in Computers", Vikas Publishing House Pvt. Ltd.
- NagpalD.P., "Computer Fundamentals", S. Chand Publications
- Bharihoke Deepak, "Fundamentals of Information Technology", Excel Books

Unit : 4

Memory and Storage

Learning Objectives:

- Defining memory cell
- Comprehending the Characteristics of Memory Cell
- Understanding types of storage in a computer
- Read-Only Memory (ROM) and types of ROM
- Random Access Memory and its types
- Serial Access Memory

Structure:

- Introduction
- 4.1 What is Computer Memory?
- 4.2 Why is computer memory so important?
- 4.3 What is a memory cell?
- 4.4 Storage
- 4.5 Quantity Information
- 4.6 Types of Computer Memory
- 4.7 Summary
- 4.8 Keywords
- 4.9 Self-Assessment Questions
- 4.10 References

Introduction

The maintenance of a computer's performance requires a variety of factors. Computers need memory to store data that the central processor unit uses to process and execute instructions in order to operate effectively. This chapter will explain to us what computer memory is, why it's crucial, and the various kinds of computer memory.

You need a way to store the data if you want to keep all your work on your computer. Many storage systems can take care of this for you. Hard disks, floppy disks, CD-ROMs, and DVDs are a few of the more popular types you have encountered.

When looking into SSDs or memory, you've encountered a memory cell. You may have also come across acronyms like SLC, MLC, or TLC. These words describe various memory cell types that are utilized by flash storage. Any time you purchase a USB flash drive, SD card, or SSD, you will discover these variations in your available alternatives.

4.1 What is Computer Memory?

Computer memory, which can be internal or external, is a mechanism used to store data and instructions on a device. It consists of several memory cells, each with a unique identification number.

The CPU, which receives and processes instructions, selects which cells to read from or write to according on the task the user has given the computer. You can use different kinds of memory, depending on your needs and the type of gadget you're using.

4.2 Why is computer memory so important?

Computer memory is necessary since gadgets cannot do functions without it. The memory of a device ensures that it turns on and functions properly. It also keeps your computer running quickly and enables you to run multiple programs at once. Additionally, you can use particular types of data to save information for later use.

4.3 What is a memory cell?

The tiniest data storage unit is a memory cell. Another name for it is a binary memory cell. One piece of data can be stored in a memory cell.

Although the computing industry uses various memory cell types, NVRAM (Non-Volatile Random Access Memory) cells are the most often employed in SSDs and other storage devices. Non-volatile cells retain the value they were given whilst in use, even after you turn them off.

A single memory cell:

- A dynamic RAM memory cell includes transistor and a capacitor.
- A static RAM memory cell contains roughly five transistors.

4.3.1 Where are memory cells found?

The human brain is analogous to a memory. It functions as a device for storing instructions and data. Computer memory is the portion of the computer that stores data that needs to be processed as well as instructions for doing so.

Memory cells of any kind, whether RAM or ROM, can be found in almost every part of the computer.

Many little segments make up the memory. A cell is the name given to each component. Every location or cell has a different & unique address, which varies from 0 to memory size -1.

For an example, this memory unit has $64 * 1024 = 65536$ memory locations if the machine has 64k words. These places have addresses ranging from 0 to 65535.

There are mainly two types of memories.

- Internal Memory: Primary/main memory and cache memory
- External memory: Magnetic disks, optical disks, etc.

4.3.2 Characteristics of Memory Cell

As we move from top to down, we notice characteristics of the memory order.

- The amount of storage capacity grows.
- Storage becomes less expensive per bit.
- The CPU's ability to access memory quickly reduces.

- The CPU's access time increases.

4.4 Storage

Bytes measure a computer's storage capacity. The smallest data unit that can be stored is a byte consisting of 8 bits (also known as binary digits). Binary numbers, such as 1 or 0, represent bits. A keyboard letter, number, or symbol equals one byte (a binary term). Considering all the files saved on your computer and the number of characters you've written, you'll require millions of bytes of storage space to ensure the safety of your work. Typically, Kilobytes (kB), Megabytes (MB), and Gigabytes (GB) are used to describe a computer's storage capacity (or, on really big systems, Terabytes!).

4.5 Quantity Information

The below table is showing the memory units:

Full Form	Units	Bytes
1 Bit	Binary Digit (0/1)	
1 Nibble	4 bits	
1 Byte	8 bits	
1 kilobyte(KB)	1024 byte	2^{10} bytes
1 Megabyte(MB)	1024 KB	2^{20} bytes
1 Gigabyte (GB)	1024 MB	2^{30} bytes
1 Terabyte(TB)	1024 GB	2^{40} bytes
1 Petabyte(PB)	1024 TB	2^{50} bytes
1 Exabyte(EB)	1024 PB	2^{60} bytes
1 Zettabyte(ZB)	1024 EB	2^{70} bytes
1 Yottabyte(YB)	1024 ZB	2^{80} bytes
1 Brontobyte	1024 YB	2^{90} bytes
1 Geopbyte	1024 Brontobyte	2^{100} bytes

4.6 Types of Computer Memory

Here is a list of the different types of computer memory:

4.6.1 Internal

Internal memory, often known as primary memory, stores brief information that the computer can access while you're using it. The chips that make up internal memory are affixed to the motherboard and require a direct connection to the computer in order to be utilized. RAM and ROM are the two primary types of internal memory, each with distinct memory subsets.

4.6.2 Random Access Memory

Random-access memory is the primary internal memory of the CPU (RAM). It acts as a method for your electronic device's temporary data storage. This is achieved by providing programs with a place to store the data you are currently using so that the apps can access the data with ease. The amount of RAM in your gadget determines its speed and performance.

RAM is also called as "volatile memory" because it loses the data it was holding if the device is shut down or turned off. For example, if you're browsing the web on your laptop and it turns off, the web pages you were viewing may not have been stored because RAM only retains information temporarily.

Have you ever failed to save your work in the moments before a computer crash? Your work is gone when you log back in. This is because it was wiped when the computer turned off and was saved in RAM. Your work would have been secure if you had saved it from RAM to the hard drive! It is stored in this area when you CUT, COPY, and PASTE data from within applications like Microsoft Word and Excel.

The size of RAM has grown as operating systems and computer programs have become more complicated. Nowadays, most PCs come with 256MB or 512MB of RAM.

RAM is of two types

- **Static RAM (SRAM)**
- **Dynamic RAM (DRAM)**

4.6.2.1 DRAM or dynamic RAM

Dynamic random access memory (DRAM) is one of the two distinct types of RAM used in contemporary gadgets like laptops, desktop computers, portable electronics, and gaming systems. It produces high-capacity memory and is the less expensive of the two forms of RAM.

It is constructed of two parts—transistors and capacitors—that must be recharged periodically in order to maintain data retention. It has volatile memory, just as RAM, and loses data when the power goes out.

Unlike SRAM, DRAM requires frequent refreshing to maintain data integrity within the memory space. This is achieved by integrating the memory with a refresh circuit, which rewrites the data numerous times per second.

4.6.2.2 SRAM, or Static RAM

In contrast to DRAM, which necessitates more frequent refreshing, static random access memory (SRAM) retains data as long as there is continuous electricity in the system. Nonetheless, due to its volatile characteristic, data is lost upon machine shutdown.

SRAM chips are devoid of capacitors and consist of a matrix of six transistors. Unlike DRAM, SRAM does not require frequent refreshing since the transistors do not rely on electricity to prevent leakage.

It costs more than DRAM since it retains power for a longer period of time, which usually prevents it from being as popular. SRAM is quicker than DRAM because it is frequently used as cache memory by users. Because the matrix contains greater space, SRAM uses more chips than DRAM for the same amount of storage, which increases the cost of production. Because cache memory needs to be small and rapid, static RAM is used.

4.6.3 Read-Only Memory (ROM)

Read-only memory (ROM) is another kind of primary internal memory. Unlike random-access memory (RAM), ROM is non-volatile and keeps data for a long period. Data storage operates regardless of whether the device is turned on.

When the power is turned off, data kept in Read Only Memory (ROM) is not permanently lost. 'Non-volatile memory' is another name for this kind of memory. A ROM chip is placed on the motherboard of a computer. The instructions needed to boot up the computer are stored on this chip. The BIOS is another name for this program.

A ROM is an ideal option whenever any data needs to be permanently stored. For instance, many automotive computer systems will have ROM chips that houses essential data to run the engine.

The various ROM kinds are as follows.

4.6.3.1 Masked ROM (MROM)

The earliest ROMs comprised hard-wired devices containing a pre-programmed set of data or instructions. These ROMs are classified as "masked ROMs" and are known for their cost-effectiveness.

4.6.3.2 Programmable Read-Only Memory (PROM)

Read-only memory (ROM) is another kind of primary internal memory; unlike RAM, ROM is non-volatile and holds data throughout time. Data storage is done without requiring the gadget to be turned on.

4.6.3.3 Erasable Programmable Read-Only Memory (EPROM)

Exposing an EPROM to UV light for over forty minutes enables its erasure, typically done using an EPROM eraser. During programming, an electrical charge is trapped in an insulated gate region, where it can persist for nearly a decade due to the absence of a leakage channel. To clear this charge, ultraviolet light is emitted through a quartz crystal window, causing the charge to dissipate. In regular operation, a sticker seals the quartz cover.

4.6.3.4 Electrically Erasable Programmable Read-Only Memory (EEPROM)

In order to program and clear the EEPROM, electrical currents are utilized. It has at least 10,000 cycles of wiping and reprogramming. It takes between 4 and 10 ms

(milliseconds) to program and wipe. An EEPROM can be selectively wiped or programmed to any location. EEPROMs can be deleted one byte at a time as an alternative to erasing the entire chip. Reprogramming is hence both flexible and time-consuming.

4.6.4 Serial Access Memory

Sequential access requires the system to look through the storage device, starting at the memory address until it locates the needed data. A Sequential Access Memory, or S, is a memory device that permits such access.

Data storage devices that read data sequentially are called serial access memory (SAM). However, information can be retrieved from random access memory (RAM) in any order. Sequential access devices are the most common form of magnetic memory.

Despite being read in order, sequential access memory still allows for uncontrolled access by "seeking" the desired position. However, this process could be more efficient.

As supplemental storage in general-purpose computers, magnetic sequential access memory is widely used because of its higher density at a cheaper cost than RAM, wear resilience, and non-volatility. Among the SAM devices still in use are magnetic tapes, CD-ROMs, and hard drives.

4.6.5 Cache Memory

Fast-moving semiconductor memory that can speed up the CPU is called cache memory. It functions as a buffer when the CPU and main memory are in use. It keeps track of the information and software elements that the CPU uses most regularly. For CPU access, operating systems shift data and program elements from disk to cache memory.

Advantages

- Faster than main memory
- It takes less time to access

- It keeps the programs that can be run quickly in storage
- Data is stored for short-term use

4.7 Summary

- The computing industry uses a variety of memory cell types.
- Non-volatile cells retain the value they were given whilst in use even after you turn them off.
- A cell in dynamic RAM memory is consists of a transistor and a capacitor.
- A cell in static RAM memory contains roughly five transistors
- Memory cells of any kind, whether they are RAM or storage, can be found in almost every part of the computer.
- Bytes are used to measure a computer's storage capacity.
- Binary numbers, such as 1 or 0, are used to represent bits.
- Terabyte (TB) is equivalent to 1,099,000,000,000 bytes or 240.
- Floppy disks have a capacity limit of 1.44 MB.
- CD ROM disks have a capacity limit of 650 MB.
- PROM is read-only memory that a user can only modify one time only.
- Electrical currents are used to program and erase the EEPROM.
- Random Access Memory, as opposed to ROM, is volatile memory.
- RAM is of two types: Static RAM (SRAM) and Dynamic RAM (DRAM)
- DRAM, unlike SRAM, needs to be refreshed to keep the data.
- Often used as supplemental storage in general-purpose computers, Serial Access Memory has a higher density than RAM at a lower cost, is resistant to wear, and is not volatile.

4.8 Keywords

- Non-volatile memory: it is a type of computer memory that can retain stored information even after system shut down.
- Volatile memory: the memory that can keep the information only during the time it is powered up.

- Byte: a unit of information representing one item, such as a letter or a number. A byte is usually made up of a series of eight smaller units (bits)
- Mounting: Increasing
- Supplementary Storage: persistent storage for less frequently accessed non-critical data compared to primary storage

4.9 Self-Assessment Questions

- Explain three ways in which RAM is different from ROM.
- Differentiate between volatile and non-volatile memory using suitable examples.
- Define the concept of memory storage and bytes. Name the different types of bytes.
- What are the advantages of cache memory?
- Differentiate between RAM and ROM. Also, state its different types.
- Discuss how a computer's storage capacity affects its performance.

4.10 References

- Memory Management A Complete Guide (2020 Edition) by Gerardus Blokdyk
- Windows Internals System architecture, processes, threads, memory management, and more, Part 1 by Pavel Yosifovich, Mark Russinovich, Alex Ionescu, David Solomon
- Non-Volatile Memory Database Management Systems by Joy Arulraj, Andrew Pavlo, H. V. Jagadish

Unit : 5

Hardware Components of Computer

Learning Objectives:

- To understand the concept of storage capacity
- Define Internal Computer Hardware Components
- Learn about Physical Devices used to Construct Memories
- Acquire knowledge about different types of data storage and storage devices

Structure:

- 5.0 Introduction
- 5.1 Internal Computer Hardware Components
- 5.2 Physical Devices Used to Construct Memories
- 5.3 External Memory
- 5.4 Summary
- 5.5 Keywords
- 5.6 Self-Assessment questions
- 5.7 References

Introduction

In general, "computer hardware" refers to any physical part of digital or analog computer. The actual parts of a computer system are referred as "hardware," as opposed to "software," which is made of written, machine-readable instructions or programs that state how and when to operate physical parts.

Hardware and software complement each other well. A computing device requires properly synchronized hardware and software in order to function and generate useable output.

Computer hardware can separate into external and internal components. External hardware gears are attached to the computer to enhance or add functionality, whilst internal hardware components are typically those needed for the computer to function correctly.

5.1 Internal Hardware Parts of a Computer: Internal components process or store the instructions given by the program or operating system (OS) collectively. They include the following:

- **Motherboard:** This printed circuit board, which houses the central processing unit (CPU) and other essential internal hardware, acts as the center for all other hardware parts.
- **CPU:** The computer's brain, the CPU interprets and carries out digital instructions from various programs. The computer's performance and data processing efficiency are impacted by the CPU's clock speed.
- **RAM:** Also referred to as dynamic RAM, RAM is a type of temporary memory that provides programs with immediate access to data. Any data that is stored in RAM is lost when the computer shuts down since RAM is volatile memory.
- **Hard disk:** Hard disk drives are real, physical storage devices that may store a wide range of data types, including operating systems, applications, device files, images, and more, both temporary and permanent. This storage device houses all of the apps and data, and it works something like a filing cabinet. Data stored on a hard disk may be accessed substantially faster than data placed

on a floppy disk. A hard disk spins inside its metal casing millions of times per minute, which produces whirring sound it makes. Less than a hair's width above the disk, a magnetic read and write head generates 1s and 0s on the circular grooves below. The majority of hard disks are installed within computers in an inaccessible manner, despite the availability of external drives that plug into PCs.

- **“Solid-state drive” or SSD:** SSDs are solid-state storage units that can safely store data even while the computer is off since they are based on NAND flash memory technology. SSDs don't lose data.
- **Optical Drive:** Typically located in the drive bay of a device, optical drives enable computers to read and communicate with non-magnetic external media such as CDs and digital video discs. read-only memory.
- **Heat sink:** This is a passive hardware component that draws heat away from the system's components to help regulate or lower their temperature and maintain proper system operation. A heat sink is usually installed directly on top of the CPU, the internal component that produces the most heat.
- **Graphics Processor:** Usually acting as a supplement to the main CPU, this chip-based system manages graphic data processing.
- **Network Interface Card, or NIC:** A NIC is circuit board or chip that permits computer to connect to a network. It is sometimes referred to as a network adapter or local area network adapter. Usually, it makes connecting to an Ethernet network easier.

Additionally, components like as power supplies, USB ports, semiconductors, and transistors are examples of internal hardware used in computers.

5.2 Physical Tools for Memory Construction

Physical memory is the whole quantity of RAM that is installed on your machine. This RAM is used to load specific apps and files, along with the operating system, into your computer.

Physical devices: what are they? They include any input/output, storage, and computer equipment such a disk, printer, modem, or screen.

What is a computer's physical memory? The physical RAM of the computer is called physical memory, and it is usually inserted as cards (DIMMs) on motherboards. It is frequently referred to as primary memory and is the only sort of storage that the CPU can access directly. It includes the instructions for the executable program.

High physical memory utilization is due to a flaw in a process that would typically be much less resource-intensive or by running too many resource-intensive apps. Numerous factors might contribute to slow processing, some of which will be more challenging to address than others.

What are physical computer devices?

It speaks of the real components of a computer system. Hardware refers to a computer system's actual physical components. Thus, it is a collection of all the computer's touchable components, including the keyboard, mouse, display, etc. Physical memory, often referred to as random-access memory (RAM), is a quick but unstable way to store data. Physical disks are frequently timed in milliseconds (10001), but RAM modules are commonly timed in nanoseconds (10003).

What kind of memory is used in a computer?

Memory in computing refers to the actual hardware used to temporarily or permanently store data, such as program state information or programs (sequences of instructions) for use in a computer or other digital electronic device.

5.3 External Memory

Secondary memory, sometimes called external memory, is memory that is added or removed based on need and isn't directly connected to the CPU. Individuals use various kinds of external memory in their devices.

CDs, flash drives, memory cards, and external hard drives are a few examples. External memory can be used to store data from a computer. It can then be disconnected from the device and linked to another appropriate device for data transmission.

Floppy Disk

Floppy disks, which have been around since about 1980, are among the earliest types of portable storage devices still used. Using floppy disk drive to move small files between computers, you can make backup copy of your work to ensure that it is not lost. The material used to create floppy disks is known as Mylar. They have a magnetic surface that enables data to be recorded. While the first (3 1/2 inch) floppy disks were actually "floppy," the ones we use today are covered in hard plastic. Read/write head may access the disk because the disk rotates in the drive. A typical floppy disk can hold up to 1.44 Mb of data, which is roughly 300 pages of text on an A4 size page. However, as graphic images are frequently quite huge, you might discover that your work won't fit onto a floppy disk.

Prior to writing data to the disk, all disks must be formatted. The disk is divided into sectors during formatting so that data files can be put there. Pre-formatted floppy disks are frequently marketed.

Floppy Disks should be handled carefully to safeguard the data. You should avoid touching the disk's surface and keep it away from hot or cold places and strong magnetic fields like those that may be present next to speakers, as doing so could cause all of your data to be lost.

Zip Drive

Zip drive resembles a floppy drive but has a storage capacity of 100 MB, which is at least 70 times greater than a floppy. Some zip disks ,may hold up to 250 MB of data. The Zip disk requires a separate drive and is marginally thicker than a floppy disk. Zip disks are very helpful for backing up crucial data and for quickly transferring data between computers. Data is compressed to make files that are too big to fit on a floppy disk smaller.

Magnetic Tape

Magnetic tape has long been most popular media for huge data storage, back-up, archiving, and exchange. Hard disk and tape have historically had a far higher capacity-to-price ratio than hard disk, but the two have moved closer together recently. Numerous formats exist, many exclusive to specific platforms or industries, such as mainframes or a specific type of personal computer.

Though access times on tape might be slow, it write or read data is quick because the tape is a sequential access medium.

Examples of organizations that will hold more information:-

- Satellite imaging companies have a backlog of many photographs.
- Film studios that store their digital films in archives
- Companies that specialize in architecture, automobiles, and design and hold hundreds of CAD drawings.
- Scientific institutions like CERN keep the outcomes of previous experiments.
- Weather service providers.
- There are two types of magnetic tapes:
 - Tape reels, which can store a lot of data and frequently used to backup data from mainframe systems.
 - Cassettes or cartridges, which despite being relatively small can store more data to back up information stored on a PC or a small network.
- The key advantage of employing magnetic tape as backup storage is that it is reasonably inexpensive and has a high data storage capacity.

Compact Disc

James Russell invented compact discs, or CDs, which are spherical, flat optical storage media. The first CD was created on August 17, 1982, at a Philips facility in Germany. Compact discs store data that can be recovered or used at a later time. Software for your computer can be found on CDs. In addition to storing music to play in a CD player, they also save files for backup or transfer to computer.

A typical CD can store 650 MB of data or 72 minutes of music. Data on an 80-minute CD can be stored in 700 MB.

Compact Discs can be categorized on:

- End-use: Video, Audio, Photo, Graphics
- Operations: Read-Only, Recordable, Rewritable

Optical Drive

External memory known as an optical drive may store and read data using light. CDs, DVDs, and Blu-ray discs are the three most popular varieties. You insert the disc into the optical drive and the computer spins it so you can access the information on the disc.

It is scanned by an internal laser beam, which also downloads the data from the optical disk into the computer after receiving it. This sort of memory can be helpful because it often costs little, is simple to use, and holds a lot of data.

Magnetic Data Storage

Data is encoded as an electric current in magnetic storage devices, which have a covering of magnetic material. Small portions of a metal spinning disk are magnetized in this sort of memory using magnetic fields.

Each segment, which denotes a "1" or a "0," is quite large, frequently containing many terabytes of data. Users favor this sort of memory because it is inexpensive, robust, and capable of holding a large amount of data. Floppy disks, hard drives, and magnetic tape are examples of common magnetic storage media.

Solid State Drives

External memory in the form of solid-state drives makes use of silicon microchips. Solid-state drives are more contemporary than magnetic storage devices, but they share the ability to be removed from the device from which data is being stored or extracted.

Since the memory preserves the binary data's electrical storage in silicon cells—sometimes referred to as chips—they are also faster. RAM works in a similar way, but solid-state drives are able to retain memory even when a device is turned off since they make use of flash memory. Among the most common varieties are the USB flash drive and memory stick.

Virtual Storage

Virtual memory, a different kind of Secondary memory that takes the shape of a hard drive or solid-state drive and moves data from RAM to disk storage, can help a computer make up for physical memory shortages. When RAM space runs short, virtual memory moves data to a paging file, which is a portion of a hard disk that acts

as an extension of RAM. This is temporary procedure that will terminate when there is sufficient RAM available.

A user may use a lot of RAM, for instance, if they are running multiple programs on their device at once. This could cause the device to slow down and make it more difficult for the programs to work efficiently. Then, to make more RAM available for the apps to perform at their best, the data that the computer isn't using is transferred to virtual memory.

5.4 Summary

Storage capacity is measured in bytes.

- Kilobytes (KB), Megabytes (MB), and Gigabytes (GB) are common numbers used to describe a computer's storage capacity
- The key benefit of magnetic tape as backup storage is that it is reasonably inexpensive and has a high data storage capacity.
- The Zip drive resembles a floppy drive but has a storage capacity of 100 MB, which is at least 70 times greater than a floppy
- Hard disk drives are real, physical storage devices that may store a wide range of data types, including operating systems, applications, device files, images, and more, both temporary and permanent.
- Computer hardware can be separated into external and internal components. hard disk spins inside its metal cover hundreds of times per minute, which produces the whirring sound.
- Floppy disks, which have been around since roughly 1980, one of the oldest categories of moveable storage devices still in working.
- The amount of work we do on our PC at home can easily be backed up onto floppy disks or DVDs for safety.
- You may quickly back up the amount of load we do on your PC's at house onto floppy disks or DVDs for security.

5.5 Keywords

- Backup: In the context of information technology, backup is the process of making duplicate copies of data that can be restored in the case of a data loss incident.

5.6 Self-Assessment Questions

- In what sense is "storage capacity" defined? How do we calculate a computer system's storage capacity?
- Enumerate the distinctions between:
 - ROM and RAM
 - Gigabyte and Megabyte units
- Could you define "storage device" for me? Provide three storage device examples. Provide any potential benefits and drawbacks of the same as well. Using appropriate examples, describe various file types.
- Could you define "File Generations" for me? Give an explanation using appropriate examples.

5.7 References

- Information Technology, P Evans, BPB Publications, 2000 Dr. Rajinder Nathby Vetter's Name: Prof. Dharminder Kumar
- Computer Fundamental, P.K. Sinha, BPB Publications – 2004.
- ICT for you, Stephon Doyle, Nelson Thornes, 2003
- Information and Communication Technology, Denise Walmsley, Hodder Murray 2004.

Unit : 6

Introduction to programming language

Learning Objectives:

- Computer languages or programming languages
- Programming languages, often known as computer languages, can be divided into three main categories: machine, assembly, and high-level languages.
- Commonly utilized programming language tools, including compiler, interpreter, assembler, and linker
- Decision table and flowchart concepts
- Qualities of a good programming language, and things to think about

Structure:

- Introduction
- 6.1 Features of Programming Languages
- 6.2 Hierarchy of Computer language
- 6.3 Characteristics of Programming Languages
- 6.4 Benefits of Programming language
- 6.5 The Need for Assembly Language
- 6.6 Advantages of Using Assembly Language
- 6.7 Languages of the High Level and Low Level
- 6.8 Language Processors: Assembler, Compiler and Interpreter
- 6.9 Algorithm
- 6.10 Flow Chart
- 6.11 Decision Table
- 6.12 Summary
- 6.13 Keywords
- 6.14 Self-Assessment Questions
- 6.15 References

Introduction

A computer needs instructions in order to perform any task using the data that the user provides. A code is a set of instructions given to a machine so it can perform specific tasks. The machine should perform the tasks that are specified by the software. A computer's interpretation and use of data are managed by a program.

A computer program is a collection of procedures and instruction sets that can be used to operate a computer system and perform certain tasks. A machine is programmed by software to do a certain processing task or series of operations. A collection of coding guidelines that direct a computer through a series of calculations or operations is referred to as a program. The keyboard, a magnetic disk, or any other device can be used to enter these commands into the computer.

6.1 Features of Programming Languages

A programming language is a grammar and set of instructions used to write software applications. Among the essential traits of programming languages are:

Syntax: A computer language's syntax is a collection of conventions and structures used to write code.

Data Types: In a program, various types of values can be stored, such as strings, numbers, and Booleans.

Variables: named memory regions with value storage capabilities

- **Operators:** Symbols that carry out operations on values, like comparison, addition, and subtraction
- **Control Structures:** Expressions like if-else statements, loops, and function calls that regulate a program's flow
- **Libraries and Frameworks:** Assemblies of previously written code that help expedite development and accomplish common tasks
- **Programming paradigms:** the language's object-oriented, procedural, or functional programming styles

Among the most widely used programming languages are Java, C++, Ruby, JavaScript, and Python. Every language is better suited for specific tasks and has pros and cons of its own.

Programming languages are formal languages that provide a set of rules that a computer must follow in order to carry out particular functions. In addition to controlling and modifying computer systems, it is used to develop software and applications. Programming languages come in a wide variety, each with its own syntax, command set, and organization. The most popular programming languages are C++, JavaScript, Python, Java, and C#. The programming language to be used depends on the specific requirements of a project, including the platform to be utilized, the intended audience, and the intended outcome. In order to adapt to the changing needs of society, more established programming languages undergo modifications and new ones are developed throughout time(Figure 6.1).

6.2 Hierarchy of Computer language

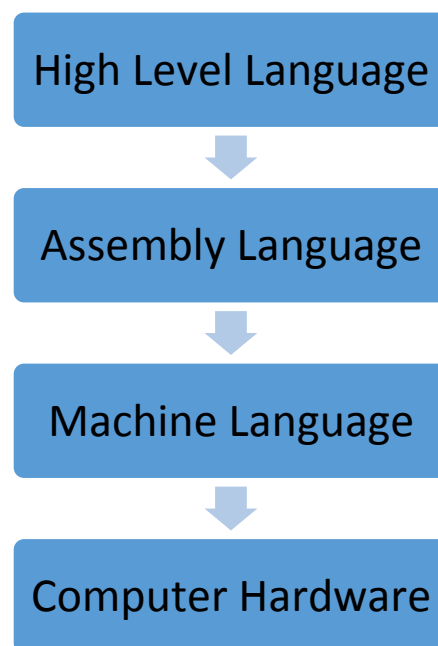


Figure 6.1: Hierarchy of Computer Languages

There have been many programming languages some of them are listed below: Most Popular Programming Languages –

C / C++/Python / SCALA/ Java/R /Swift / JavaScript/ Ruby / Go

6.3 Characteristics of Programming Languages

A programming language needs to be:

- Simple and easy to learn and use

- Easy to comprehend, and human readable
- Efficient and high in order for it to be easily converted into machine code
- Well organized and documented
- Should be semantically and syntactically consistent
- Should have the quality of ‘abstraction’ which increases usability

6.4 Benefits of Programming language

- **Enhanced Productivity:** Developers may create code more quickly and efficiently because to the abstractions provided by programming languages.
- **Portability:** High-level programming languages are portable, meaning they may be used to create programs that run across a wide range of platforms and operating systems.
- **Readability:** Effective programming languages can help both the original author and other developers understand and read code more easily.
- **Large Communities:** There are sizable user and developer communities for many programming languages, which can offer assistance, libraries, and tools.

Programming language drawbacks

- **Complexity:** Particularly for beginners, learning some programming languages can be challenging and complex.
- **Performance:** High-level programming languages have a tendency to run programs more slowly than lower-level languages.
- **Limited Functionality:** Some programming languages could lack native support for specific job types or might need additional libraries to carry out specific tasks.
- Diverse programming languages can cause fragmentation, which can make it difficult to collaborate with other engineers and share code.

6.5 The Need for Assembly Language

The computer directly decodes the 0-1 combinations that make up machine language code. However, the following issues exist with the machine language:

- Writing in 0-1 Assembly Language forms is challenging for most people and heavily dependent on the machine.
- DEBUGGING in programming is challenging as it's quite difficult to decipher the machine code. Thus, program logic will be challenging to comprehend.

To get around these issues, computer manufacturers created terms that sound like English and stand in for a machine's binary instructions. An instruction's symbolic code is known as a mnemonic. The letters that make up the mnemonic for a specific command suggest the action that instruction is supposed to take.

The mnemonic ADD is used, for example, to add two numbers. These mnemonics facilitate the symbolic representation of machine language instructions, with each machine instruction being represented by a single equivalent symbolic instruction. This is referred to as an assembly language.

6.6 Advantages of Using Assembly Language

Due to its ability to analyze the instruction set, addressing modes, interrupts, etc., assembly language gives you more flexibility over how to handle certain software and hardware components.

Assembler programming creates executable modules that are smaller and more compact. Because of this closer proximity to the machine, assembly programming may enable you to create highly optimized applications. As a result, programs run more quickly.

Programs written in assembly language have at least 30% more data in them than equivalent programs written in high-level language. The reason behind this is that, in contrast to assembly language, which generates a single line of code for each instruction, compilers currently produce a lengthy list of code for each instruction. This will be especially true for programs that deal with strings in C.

Assembly language, on the other hand, is machine-dependent. There are unique sets of instructions for every CPU. Assembly programs are not portable as a result.

Assembly language has extremely few limitations or regulations; almost everything is up to the programmer's discretion. This allows programmers a great deal of flexibility when building their systems.

6.7 Languages of the High Level and Low Level

There are two categories of programming languages: High level languages and low level languages. High level languages are easier for programmers to understand, interpret, and compile than low level languages. This is the main difference between the two types of languages. Low-level language is significantly easier for robots to understand than for humans. Examples of high level languages are C, C++, Java, and Python. Here are some differences between high-level and low-level languages (Table 6. 1).

Table 6. 1: Table Illustrating the Difference Between High-level and Low-level Languages

High-level Languages	Low-level Languages
It is a programmer-friendly language.	It is a machine friendly language.
High-level language is less memory efficient.	Low-level language is high memory efficient.
It is easy to understand.	It is tough to understand.
Debugging is easy.	Debugging is complex comparatively.
It is simple to maintain.	It is complex to maintain comparatively.
It is portable.	It is non-portable.
It can run on any platform.	It is machine-dependent.
It needs a compiler or interpreter for translation.	It needs an assembler for translation.
It is widely used for programming.	It is not used nowadays.

6.8 Language Processors: Assembler, Compiler and Interpreter

Compilers and interpreters translate programs written in high-level languages into machine code that can be understood by computers. Assemblers also translate programs written in assembly language or at a low level into machine code. The compilation process consists of several steps. Programmers can use tools to help them write error-free code.

High-Level language is machine-readable even when the mnemonics used to express instructions in assembly language are not. Computers can understand instructions

expressed as 0s and 1s, or machine code. A computer program requires a lot of work to write entirely in machine code. Programs created in high-level languages like Java, C++, Python, etc. are referred to as having source code. Before these source codes can be run directly by the computer, they must be translated into machine language.

Programs created in high-level languages are therefore translated into machine code by system software called a language processor, after which the program is referred to as object code or an object program.

Three sorts of language processors are available for use:

6.8.1 Compiler

A compiler is a type of language processor that can read an entire high-level program written in source code and translate it into an equivalent machine code program in one sitting. For instance, C, C++, Java, and C#.

A compiler can convert source code into object code correctly if there are no errors in it. At the conclusion of the compilation process, the compiler uses line numbers to identify any errors in the source code. The errors need to be corrected before the compiler can correctly recompile the source code. The object program can then be run multiple times without requiring another translation after that.

6.8.2 Assembler

The Assembler is used to translate an Assembly language program into machine code. An assembler receives a set of assembly language instructions from the source code. Machine code, often known as computer-understandable object code, is the assembler's output. In a sense, Assembler is the first interface that enables communication between machines and people. We need an Assembler to fill the communication gap between humans and machines so they can interact. Mnemonics, or instructions, such as ADD, MUL, MUX, SUB, DIV, MOV, and so on makeup assembly language code. Essentially, the assembler can convert these mnemonics into binary code.

6.8.3 Interpreter

A language processor known as an interpreter takes a single statement from a source program, translates it into machine code, runs it, and then advances to the next line. The translator stops translating and generates an error message if there is a problem in the statement. The interpreter moves on to the next line for execution after the error has been corrected. Programming or scripting language instructions are directly performed by an interpreter without requiring translation into machine or object code beforehand. An interpreter reads it aloud, line by line, before acting on it.

For instance, Python, Perl, and Matlab.

6.9 Algorithm

The logic of a program is described by its algorithms. It offers a detailed explanation on how to address a particular issue. It is defined as a series of guidelines that, when followed in the correct order, result in the desired effects. For an instruction set to be classified as an algorithm, it needs to possess the following characteristics:

1. All instructions must be brief and unambiguous.
2. There should be a time limit for completing each instruction.
3. Repeating a command or set of commands repeatedly is not advised.
4. When the directions are followed, the desired results are achieved.

Natural language statements are not appropriate for use in complicated or specialized algorithms due to their lengthy and confusing character.

Pseudo code and flowcharts are organized methods of expressing algorithms that avoid many of the ambiguities seen in ordinary language claims, all the while staying self-determining of a particular implementation language.

Programming languages are typically used to express algorithms in a form that can be executed by a computer. As such, they are often used for algorithm definition or documentation.

6.10 Flow Chart

A flowchart is a visual representation of an algorithm. Programmers often utilize computers as a tool for program planning, utilizing them to graphically organize the steps needed to solve an issue. It employs different shaped boxes to indicate different kinds of instructions. The actual instructions are provided in these boxes in plain, uncomplicated language.

6.10.1 Symbol for flowcharts

Just a few symbols are needed in a flowchart to indicate the necessary phases. The American National Standards Institute (ANSI) has standardized the basic symbols used in flowcharts.

Terminal Symbol: denotes the start, stop, and halt points in the logic flow of a program. It is a flowchart's first and last symbol.

Input/Output Symbol: they represent all input/output functions in a program.

Processing Symbols: represent instructions for computation and data transport. This symbol also designates the logical operations that transfer data from one area of the main memory to another (assignment statement).

Decision Symbol: denotes a decision point, or a place where a line can split off and go in one of two or more different directions.

Arrowheads: The exact order in which the instructions are carried out is shown by flow lines with arrowheads, which also show the flow of operation. A flowchart typically moves from left to right and from top to bottom. Only when the standard flow is not followed do arrowheads become necessary.

Connectors: When a flowchart is too complex and the number and direction of the flow lines become confusing, or when the flowchart takes up more than one page, it can be beneficial to substitute connection symbols for the flow lines.

6.11 Decision Table

A decision table is a planned rule logic entry that is displayed as a table with conditions (represented by the row and column titles) and actions (represented by the

crossing points of the conditional cases in the table), both of which are referred to as conditional cases. Decision tables work well for business rules with multiple criteria. Just like the if/then rule set, the decision table is controlled by the interplay of events and actions. The primary differentiation is in the fact that several conditions influence an action in a decision table, and multiple actions can be associated with every set of criteria. If all requirements are satisfied, the corresponding act or acts are performed.

6.12 Summary

- A computer program is a set of instructions and processes that can be used to carry out certain activities on a computer system.
- High-level programming languages have a tendency to run programs more slowly than lower-level languages.
- Some programming languages could lack native support for specific job types or might need additional libraries to carry out specific tasks.
- Programs written in assembly language have at least 30% more data in them than equivalent programs written in a high-level language
- Programs created in high-level languages are converted into machine code that a computer can understand via compilers and interpreters
- Algorithms describe a program's logic.
- Pseudocode and flowcharts are structured approaches to express algorithms.

6.13 Keywords

- **Complexity:** the state of being difficult to understand
- **Diversity:** a wide variety of something
- **Computer-understandable:** something easily understood by the computer
- **Mnemonics:** an abbreviation for an operation
- **Perplexing:** a state of confusion

6.14 Self-Assessment Questions

- Briefly explain a computer program and features of programming languages.
- Write a short note describing the characteristics of programming languages.
- What are the benefits of programming languages?
- Why do we need to use Assembly Language? What are the advantages of Using Assembly Language?
- Differentiate between Languages of the High-Level and Low Level
- Explain Language Processors: Assembler, Compiler and Interpreter
- Define (1) Algorithm (2) Flow Chart (3) Decision Table

6.15 References

- Introduction To Computers (Special Indian Edition) January 2005 by Peter Norton
- Basic Data Types: The Complete Reference by Herbert Schildt
- Fundamentals of Computers - by Reema Thareja
- Computer System Architecture - by Mono M Morris

Unit : 7

Introduction to Software

Learning Objectives:

- To understand the concept of a computer program
- Grasp the meaning of the terms 'Program' and 'Software'
- Learn about the different types of Software, their Functions and Features
- Know more about Programming Software and Components of a Programming Software
- Learn about Application Software: It's Functions & Features
- Differentiate between System Software and Application Software

Structure:

- Introduction
- 7.1 Types of Software
- 7.2 System Software
- 7.3 Programming Software
- 7.4 Application Software
- 7.5 Difference between System Software and Application Software
- 7.6 Summary
- 7.7 Keywords
- 7.8 Self- Assessment Questions
- 7.9 References

Introduction

A computer system's two main parts are software and the other is hardware. Software is a collective term for a group of computer programs and related data instructing a computer on what to do and how to do it. It is a set of instructions that tells the hardware how to carry out each task.

Let's begin by understanding the difference between a program and software.

Program Vs Software

Aspect	Program	Software
Definition	A specific set of instructions designed to perform a particular task.	A collection of programs and associated resources providing a complete solution.
Scope	Typically a single, standalone entity.	Can be a single program or a suite of programs working together.
Composition	Consists of source code and executable code.	Includes programs, documentation, libraries, and user interfaces.
Examples	A text editor, a web browser, a game.	Operating systems, application suites (e.g., Microsoft Office), ERP systems.
Structure	Focused and limited in functionality.	Comprehensive, includes additional components for usability.

In summary, software is a collection of programs, whereas a program is a set of instructions that a computer executes. For instance, a calculator is software, but its programs for addition, subtraction, and other operations are what make it what it is.

A collection of programs, guidelines, and processes that direct computers on what to perform is called software. Most of it is composed of application and system software. The type of software that acts as an interface between application and system software is known as system software. Low-level programming languages are used in the development of system software. Application software has a running route thanks to system software, which also maintains system resources up to date. It is essential to remember that the system requires system software to operate. It is a program with several uses.

7.1 Types of Software

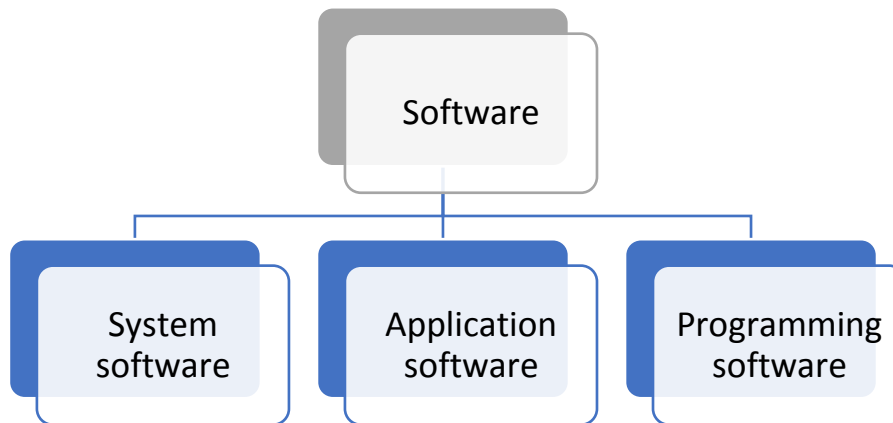


Figure 7.1: Types of Software

7.2 System Software

The type of software that acts as an interface between application and system software is known as system software. Low-level programming languages are used in the development of system software. Application software runs on a path provided by system software, which also maintains system resources up to date.

It is software designed to operate computer hardware, giving it basic functionality and serving as a platform for other applications. The operating system and all other utility apps that oversee computer resources on a basic level are considered part of the system software(Figure 7.1).

Example: Once the computer is turned on, the Basic Input/Output System (BIOS) controls the data flow between the operating system and other connected devices, like the hard drive, video adapter, and so on. Tools like the system restore and disk defragmenter is also included in the software.

7.2.1 Functions of System Software

- Resource Management
- Process Management
- Memory Management
- File System Management

- Device Management
- User Interface
- Security and Access Control

7.2.2 Features of System Software

- Resource Management
- Process Management
- Device Management
- File System Management
- User Interface
- Security and Protection
- System Utilities

7.2.3 Types of System Software

Operating System: At the heart of any computer setup lies the operating system. It serves as the overseer, managing all resources such as the hard drive, printer, and CPU. Moreover, it provides support to a plethora of other software programs. Operating systems encompass a variety of options including Linux, macOS by Apple, Microsoft Windows, and more.

Language Processor: System software uses a language processor to translate human-readable language into machine language. It turns computer programs into machine-readable instructions.

Device Driver: It is a program or software that works with a device to complete specific tasks. Before you can execute the software, a driver must be installed.

7.3 Programming Software

It consists of the programs and apps that software engineers use as tools to build, test, support, and maintain other programs and applications. The components of programming software are a compiler, a debugger, an interpreter, a linker, and a text editor (Table 7.1).

Table 7.1 : Components of Programming Software

Compiler	They convert a high-level language program into a low-level language program.
Assembler	They convert an assembly language program into low-level language programs.
Interpreter	It processes high-level language line by line and simultaneously produces low-level programs.
Linker	Most low-level languages enable programmers to create big programs with many modules. All of the modules produced by the language translator have their object code organized by the linker into a single program.
Debugger	It is software used to find programming faults and errors. It pinpoints the location of computer code faults.
Text editor	It is a program that enables computer users to work with texts. It is used for documentation and allows us to edit data that is already present in a document or file.

7.4 Application Software

Programs classified as application software run in response to user queries. It uses the platform that the system software offers to function. High-level programming languages are used in the development of the application software. It's software with a particular purpose. The High-level maintenance system can always operate without application software, but the system itself cannot operate without it. This is the key difference between system software and application software.

A few examples are word processors, presentation programs, data management tools, desktop publishing applications, and web browsers.

7.4.1 Functions of Application Software

- Management of Information and Data
- Document Interchange Systems for Document Management
- Creation of Images and Videos
- Collaboration, Audio and Video Conferencing, Emails, and Text Messaging
- Financial, Accounting, and Payroll Management
- Resource Management (ERP and CRM systems)

7.4.2 Features of Application Software

- High-level languages are used to write application software.
- More storage space is needed for application software than for system software.
- Each application software program completes a specific task.
- In comparison to system software, application software is simpler to develop.

7.4.3 Types of Applications Software

- **General-Purpose Software:** This application software is used to carry out a variety of jobs; it is not merely confined to one particular duty. MS Word, MS Excel, etc. are a few examples.
- **Customized Software** is used to carry out duties created for certain businesses. As an illustration, consider the reservation systems for railroads and airlines.
- **Utility Software:** It is employed to support the computer's architecture. It is made to take care of the system's needs while also optimizing and maintaining it (Figure 7.2).

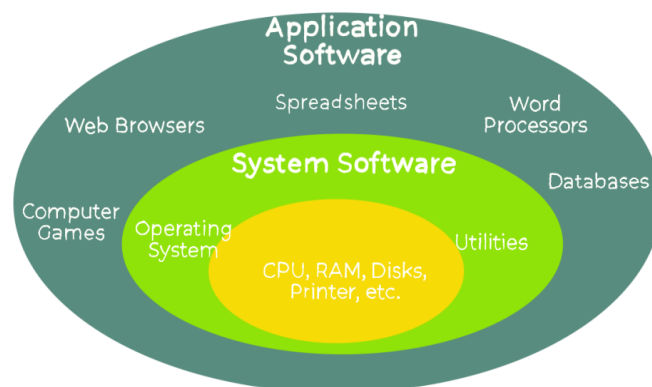


Figure 7.2: System Software and Application Software

Table 7.2 Difference between System Software and Application Software

System Software	Application Software
Low-level languages are used to write the system software.	High-level languages are used to write application software.
It is general-purpose software.	It is a specific purpose software.
A system cannot function without system software.	Application software does not interrupt the working of the system.
The system software operates in the background till computer shuts down	Application software runs in the front end as per the user's request.

Figure 4: System Software and Application Software

7.5 Summary

- Software is the collection of all programs that are performed sequentially or simultaneously to carry out a certain task. It is the result of several programs combined.
- A program is a collection of lines of computer code that processes input and executes instructions to produce output.
- System Software is the interface between application software and the system.
- System software uses a language processor to translate human-readable language into machine language.
- The components of programming software are a compiler, a debugger, an interpreter, a linker, and a text editor.
- Application software is the kind of program that executes when the user requests it.

7.6 Keywords

- **Disk defragmenter:** the act of reorganizing data on a hard disk drive (HDD) or other storage media to make room for it and to facilitate efficient access and storage
- **Memory Management:** It controls and coordinates a computer's main memory.
- **Processor Management:** aligns processes with an organization's strategic goals, designs and implements process architectures, and establishes process measurement systems
- **File Management:** It is the action of putting information into a structured framework so that it may be easily retrieved.
- **Error Detection:** refers to the techniques used to detect noise or other impairments introduced into data

7.7 Self-Assessment Questions

- What are the differences between System software and application software?
- Describe the features and functions of Application Software.
- Explain the components of programming software with suitable examples.
- What are the salient features and different types of System Software?

7.8 References

- "An Introduction to Systems Programming" by Leland L Beck
- "System Software" by Ikvinderpal Singh
- "Computer Organization and System Software" by Madhulika Jain and Prof Satish Jain

Unit : 8

Introduction to Operating System

Learning Objectives:

- Understanding the concept of an Operating System
- Learning about the Functions of the Operating System
- Knowledge of the different views of Operating System

Structure:

- Introduction to Operating System
- 8.1 Functions of the Operating System
- 8.2 View of Operating System
- 8.3 Keywords
- 8.4 Self-Assessment Questions
- 8.5 Case Study
- 8.6 References

Introduction

An operating system (OS) serves as a mediator between the hardware of computers and their users. A computer's operating system is a piece of software that manages all of its fundamental functions, including managing peripherals like printers and disc drives, processing input and output, and managing memory and processes.

An operating system is a type of software that manages how different applications are run and serves as a conduit between a computer's user and hardware.

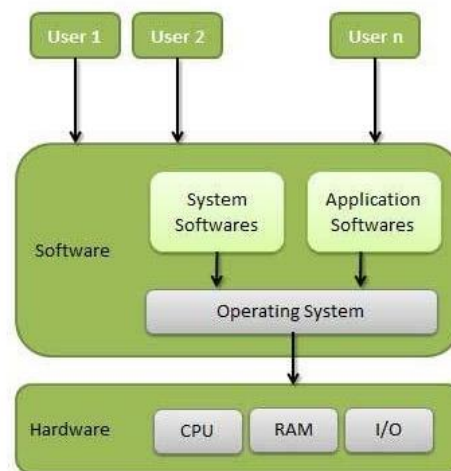


Figure 1: Explaining the Concept of an Operating System

The most popular operating systems include z/OS, Linux, Windows, VMS, OS/400, and AIX. Two goals of the operating system include:

- (1) Controlling the hardware of the computer, and
- (2) Giving the user a user-interactive interface and interpreting user commands so that the operating system may interact with the hardware.

An essential component of practically every computer system is the operating system.

8.1 Functions of the Operating System

8.1.1 Managing Hardware

- Managing and controlling a computer system's different hardware resources is the operating system's primary goal.
- The processor, RAM, disc space, and other hardware resources are included.

- The monitor displayed the output result. The operating system offers an error management process, displays an error notification, and connects with the hardware.
- The operating system cannot communicate with a device that is not running properly.

8.1.2 Providing an Interface

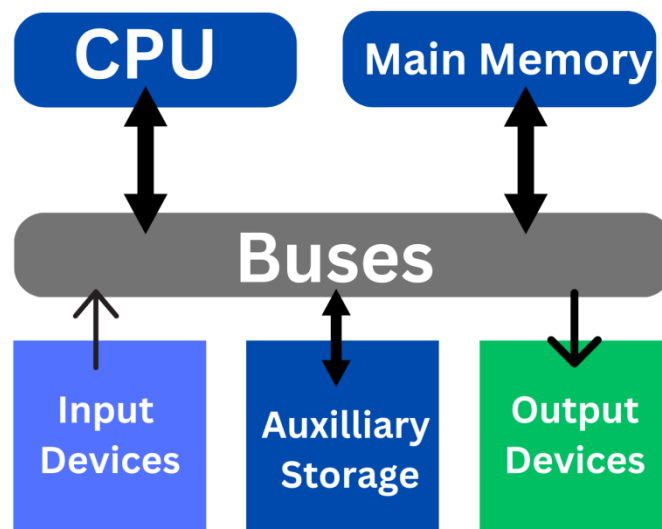


Figure 2: Operating System Serving as an Interface

- The operating system arranges programs to make it simple for users to access, utilise, and save them.
- It gives apps a consistent and reliable mechanism to interact with the hardware without requiring the user to know specific device information.
- The application shows the relevant error message when a program isn't working correctly.

The components of a computer system are separated into 5 parts

- Computer Hardware
- Operating System
- Utilities
- Application Programs
- End User

The computer's operating system is the initial program that is encoded and is permanently stored in memory. It controls and programs various users. It is an application that communicates with the hardware directly.

8.1.3 System Goals

The primary goal of an operating system is to run programs and to make the computer system user-friendly. One of its secondary goals is to use the computer hardware best.

8.2 View of the Operating System

An operating system is a configuration that allows user applications to interface with the hardware of a computer. The operating system does not have any built-in features, but it does provide a favorable environment in which different applications and programs can perform important duties.

The operating system can be seen from both the system's and the user's point of view.

They go by the names of system view and user view, respectively. Here are a few more specifics on these:

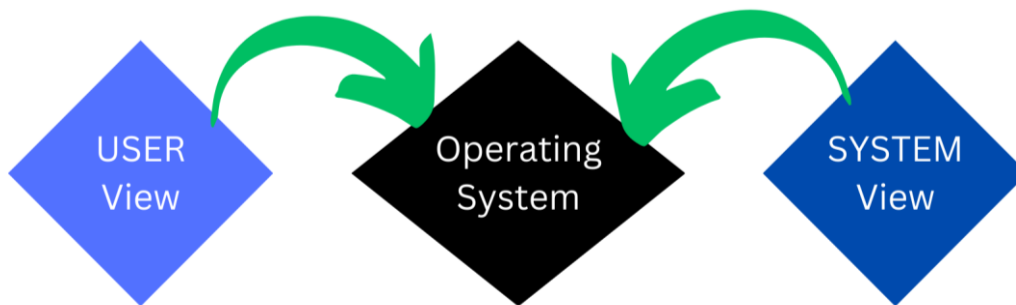


Figure 3: Views of Operating System

8.2.1 User View

Depending on the interface employed, different users see computers differently. Consider Windows XP, Vista, Windows 7, and more versions as examples.

The system interface that users utilise determines the user view. The various user view experiences can be explained in the following ways:

- If the user interacts with a personal computer, the operating system was designed with ease of use in mind. The operating system doesn't have to worry about resource usage, but the system's performance is taken into consideration. This is because all of the resources are used by the personal computer and there is no resource sharing.
- If the user is using a system connected to a mainframe or a minicomputer, the operating system is mostly focused on resource utilisation. This is due to the possibility of several terminals linked to the mainframe and the operating system's careful distribution of all resources including CPU, memory, and I/O devices among them.
- The operating system needs to focus on both network sharing and individual resource use if the user is using a workstation that is networked to other workstations. This happens because the workstation can't only use its own resources; it also has to share files and other resources with other workstations on the network.

Given that there is no user contact, some gadgets have very little or no user view. Examples include **embedded computers** in household appliances, vehicles, etc.

8.2.2 System view

From the standpoint of the computer system, the operating system acts as a conduit for data between hardware and software. While necessary, it has the highest degree of intimacy while controlling the hardware.

The different operating system view types are explained as follows:

- The operating system is viewed by the system as a resource allocator. To function, processes require a range of resources, such as memory, CPU time, I/O devices, and file storage space. To maximize computer system performance, it is the operating system's job to allocate these resources among the processes in an intelligent manner.
- The operating system may also be configured with a control software. It regulates each and every I/O device and process to guarantee error-free computer system functioning. It guarantees trouble-free and proper operation of

the I/O devices. Operating systems might be considered a way to make the use of hardware simpler. Computers were required to promptly address user problems. Working directly with computer hardware is not an easy task, though. Operating systems were therefore designed to be easily able to interface with hardware.

- The kernel, which controls all application programs on a computer system and runs continually in the background, is another way to think of an operating system. The commonly accepted definition of an operating system is as follows.

The operating system is regarded by computers as the software that interacts with hardware the most. Hardware and software resources that an operating system may require to solve a problem include CPU time, memory, file storage capacity, and I/O devices. Consequently, these resources are managed by the operating system. A few of the fundamental tasks an operating system carries out:

- **Security:** It uses passwords and similar methods to prevent unauthorised access to programmes and data.
- **Tracking:** It keeps record of the time and things of various users.
- **Error-detecting:** It employs debugging tools and monitors the creation of dumps, traces, and error messages.

8.3 Keywords

- **Conducive:** Making a certain situation or outcome likely or possible
- **Embedded computer:** A combination of computer hardware and software designed for a specific function.
- **Resource allocator:** Optimise business activities.
- **Kernel:** The central or most important part of something.

8.4 Self-Assessment Questions

- What inconveniences can a user face while interacting with a computer system, which is without an operating system?
- Explain the meaning and purpose of ‘Views’ in Operating Systems.

- Write a brief summary about the Functions of Operating Systems.

8.5 Case Study

Title: Choosing the Right OS

Introduction

The coming of various new versions of OS has brought about increased competition within companies due to the use of latest and improvised operating procedures to facilitate work progress. The famous operating systems include Microsoft Windows, Linux, and Mac OS.

Case Study:

It is necessary to evaluate the numerous systems that can run on different desktop environments before installing a specific operating system. The hardware components and the specification of their software needs are among the factors that need to be taken into account.

Windows by Microsoft

This is the most widely used and prudent operating system in the world. This is due to the fact that it is more user-friendly and provides a number of visually appealing graphical interfaces.

OS X

One important software system component that the well-known technology company Apple Inc. offers is the Mac operating system. Within its operating system, it offers a variety of services and goods, such as shared application frameworks and general graphical interface design, which are mostly compatible with Mac or Apple-branded machines.

OS Linux

This multi-version open-source operating system is available for free on the global software provisional platform.

8.6 References

- “Introduction to Information Technology”, ITL Education Solutions Ltd., Pearson Education
- Nagpal D.P., “Computer Fundamentals”, S. Chand Publications
- Leon Alex & Leon Mathews, “Introduction to Computers”, Vikas Publishing House
- Norton. Peter, “Introduction to Computers”, TMH

Unit : 9

Memory Management

Learning Objectives:

- Types of Operating Systems
- Operating System Management Tasks
- Memory Management

Structure:

- 9.0 Introduction
- 9.1 Types of Operating Systems
- 9.2 Operating System Management Tasks
- 9.3 Memory Management
- 9.4 Summary
- 9.5 Keywords
- 9.6 Self-Assessment Questions
- 9.7 Case Study
- 9.8 References

Introduction

Operating system, also known as the resource manager, is in charge of managing all resources since it oversees all essential operations, including managing files, processing, and memory. Thus, operating system acts as a conduit between the user and machine. Numerous operating systems exist, several of which are enumerated here. Now let's look at them.

9.1 Types of Operating Systems

Operating systems have been around since the first generation of computers and have continued to develop. This chapter will discuss some important and most widely used operating system types.

9.1.1 Batch Operating System

A “Batch operating System” doesn't allow for direct communication between users and computers. Users prepare their work on an offline device, like a punch card, and hand it in to computer operator. To speed up processing, jobs with comparable criteria are grouped together and executed collectively. When the programmers turn in their code, the operators group the programs based on comparable characteristics.

The exertion with batch systems are listed in the following list:

- Not enough interaction exists between the user and the task.
- The CPU is usually idle because mechanical I/O devices function more slowly than the CPU.
- It's difficult to give the desired priority.
- Operating systems for time sharing

9.1.2 Multiprogramming or Time-sharing Operating System

This method allows several users at various terminal to access a specific computer system concurrently. Multiprogramming logically lead to time-sharing or multitasking. Time-Sharing is allocating a processor's time simultaneously to several users. The CPU does several tasks by switching back and forth between them, yet this happens so frequently. As a Result, the user can get a quick reply. Time-sharing

systems have been added to computers originally designed as batch systems.

<u>Advantages</u>	<u>Disadvantages</u>
<ul style="list-style-type: none">• quick response time• avoids software duplication• reduces CPU idle time	<ul style="list-style-type: none">• Reliability issue• Integrity and security of data compromised• Data communication issue

9.1.3 Distributed Systems Operating System

For distributed systems to accommodate a big number of users and real-time applications, multiple central processors are required. Data processing tasks are distributed across the processors in a coordinated manner.

The processors can speak with each other using a variety of communication channels. We call these systems loosely coupled or scattered. In a distributed system, the processors may differ in size and capability. These processors are referred to as sites, nodes, computers, and other designations.

Some advantages of distributed systems are as follows:

- The remaining locations could be able to continue operating even in the event that one fails.
- Enhanced client support
- Lessening the host computer's workload.
- Quicker data processing

9.1.4 Real-Time Operating System

When Data processing system processes and responds to inputs so quickly that it exerts control over external events, it is said to be operating in real-time. A system's reaction time is the amount of time it takes to evaluate user input and display the most recent information that is required. Real-time systems can be used as a control device in a particular application and are useful when there are stringent time limits on a processor's operation or the data flow. Clear, defined time limitations are necessary for a real-time operating system to operate. Scientific research, medical imaging

equipment, and real-time operating systems fall into two categories.

Inflexible Real-Time Systems

An inflexible real-time system is a type of real-time system that has strict constraints and requirements, making it rigid and difficult to modify once it is operational. These systems are designed to perform specific tasks within a predetermined timeframe, and any deviation from these constraints can lead to system failure or unacceptable performance.

Real-time soft systems

Real-time soft systems, also known as soft real-time systems, are a type of real-time system where meeting deadlines is important but not absolutely critical. In these systems, deadlines can be missed occasionally without catastrophic consequences, though it may result in degraded performance or reduced quality of service.

9.2 Operating System Management Tasks

9.2.1 Processor Management

When a CPU cannot process a program's instructions, it cannot function. A process is a single-user word processor or other time-shared user program.

A system task, such sending output to a printer, can also be referred to as a process. To finish a task, a process requires a variety of resources, such as memory files, CPU time, and I/O devices. The process receives these resources either at the outset or during operation. The OS is responsible for these process management duties.

- Adding and eliminating user and system processes
- Offering a way to synchronize processes;
- Allowing for the stopping and beginning of processes;
- Providing a process mechanism

9.2.2 Memory Management

The operation of a modern computer system depends heavily on the main memory. Large collections of words or bytes, range in size from tens of thousands to billions,

make up the Main Memory. Main memory stores data that is shared instantly between the CPU and an I/O device. The central processor reads and writes data to main memory during the data fetch cycle in addition to receiving instructions from main memory during the instruction fetch cycle.

Usually, main memory is the only substantial storage device that the CPU can directly address and access. As in the case of the CPU processing data from a disc. Without being in memory, instructions cannot be carried out by the CPU. The OS is in charge of the following tasks related to memory management.

- Monitoring the use of memory by apps;
- Choosing which ones to open in memory when available;
- Allocating and releasing memory space as needed

9.2.3 File Management

File management is one of the most important components of the operating system since a computer may store data on a variety of physical media types. The magnetic tape, magnetic disc, and optical disc are the most widely used media. These characteristics include capacity, data transmission rate, access speed, and access mode (random or sequential). For easy computer use, the operating system (OS) provides a common logical representation of information storage.

The following file management duties fall under the purview of the OS:

- Adding and removing directories and files
- Mapping files for backup storage
- Using non-volatile storage for data backups

9.2.4 Storage Management

Operating programs from main memory is a computer's fundamental function. The primary memory cannot hold all of the programs and data, and the information it holds is lost in the event of a power outage. The computer system needs to provide supplementary storage in order to safeguard primary memory. The operating system is responsible for the following disk management functions.

- Unfettered use of space

- Management of storage
- Organizing Discs

9.2.5 Batch Processing

Batch processing is a method which involves gathering the instructions and data in a batch then process begins. The following operations related to batch processing are performed by an OS:

- A job is categorized by the OS as a single entity, consisting of predefined commands, programs, and data.
- Numerous tasks are stored in memory by the OS, which manages their execution without requiring input from the user.

*Jobs are given in the order, they are applied or in a first come, first served manner.

<u>Advantages</u>	<u>Disadvantages</u>
Batch Processing increases performance as a new operation starts as soon as the preceding job is finished, without any manual involvement.	The programme is difficult to debug, there is no protection against a batch job affecting pending jobs, and a job could go into an infinite loop.

9.2.6 Multitasking

When the CPU switches between several tasks at once, this is known as multitasking. The frequency of switches allows users to interact with each running program.

The OS performs and manages multitasking in a way that allows it to handle numerous tasks or run multiple programmes simultaneously. These operating systems enable interactive computer use at a reasonable cost. Each user receives a small fraction of a time-shared CPU by applying the multiprogramming and CPU scheduling concepts in a time-shared operating system.

<u>Advantages</u>	<u>Disadvantages</u>
<ul style="list-style-type: none"> • It Provides the pros of a quick response 	<ul style="list-style-type: none"> • Difficulty in reliability • The issue of data communication

- | | |
|---|--|
| <ul style="list-style-type: none">• Reduces CPU idle time | |
|---|--|

9.2.7 Multiprogramming

Multiprogramming is the term for CPU sharing when two or more programmes run simultaneously in memory. Multiprogramming relies on a solitary shared processor. The CPU is used more frequently through many programs because there is always a job for it to complete.

An OS performs the following multiprogramming-related tasks.

- Several tasks are kept running in the OS memory at once.
- Jobs kept in the job pool are a subset of this set of occupations.
- The OS chooses and launches the execution of each task stored in memory.

9.2.8 Interactivity

The ability of people to interact with a computer system is referred to as interactivity.

An operating system performs the following interactive tasks:

- Gives the user a way to communicate with the system
- Controls input devices to receive user input
- Controls output devices so that users can see outputs

9.3 Memory Management

Term "main memory" describes the internal, physical memory of the computer. It is distinguished from external mass storage devices like disk drives by the word main. RAM refers to main memory as well. Only data stored in the Main memory can be altered by the computer. As a result, each application we run and each file we access needs to be copied into the main memory from a storage device. The following are a few tasks that fall under memory management:

9.3.1 Swapping

For a process to be executed, it must be in memory. However, in some cases, main memory in a time-sharing system is required to hold all of the processes that are now running. Thus, overflowing processes are stored on disk and then injected to execute

dynamically. By swapping, each approach is brought to the main memory, where it is operated for a while before being put back into the disk.

9.3.2 Contiguous Memory Allocation

Within a contiguous memory allocation, a contiguous memory block houses each process. Memory is divided into multiple fixed-size segments. The available blank blocks are called memory "holes." The collection of holes is searched in order to select the optimal hole for allocation.

9.3.3 Memory Protection

Memory access privileges on a computer can be controlled thanks to a phenomena called memory protection. Preventing processes from accessing RAM that hasn't been allocated to them is its main objective. Consequently, it prevents a process bug from damaging other processes or the operating system. It stops the offending process by sending it a segmentation fault or storage violation exception.

9.3.4 Memory Allocation

Memory allocation is the process by which computer programs are assigned memory or space. Three types exist:

- First Fit: The program is assigned the first suitably large hole.
- Best Fit: The application is designated to cover the smallest sufficiently sized gap.
- Worst Fit: The program is given the biggest opening that needs to be filled.

9.3.5 Fragmentation

When the majority of the free blocks that are accessible are too small to handle requests, fragmentation occurs. The typical phrase used to characterize it is incapacity to utilize the memory that is available. In these situations, processes are loaded and unloaded from memory. Consequently, demands are fulfilled by free holes via external fragmentation. Block sizes are the unit of measurement used to allocate physical memory, which is separated into fixed-size blocks. RAM may receive

memory that is marginally higher than what was requested due to internal fragmentation.

9.3.6 Paging

One method for addressing the fragmentation problem is paging. Paging is a memory management technique that allows the physical address region of a process to be non-contiguous. In this case, physical memory is divided into pages, or units of equal size. Pages of a particular process are loaded into the memory frames that are accessible.

9.3.7 Segmentation

Segmentation is a memory management technique that enhances the user's perception of memory. A single process's virtual address space can be divided into segments that can be placed in physically distinct memory locations by employing segmentation.

9.4 Summary

- An operating system (OS) serves as an interface between computer hardware and users.
- Multiprogramming logically leads to time-sharing or multitasking.
- When a data processing system processes and responds to inputs so quickly that it affects the environment, it is said to be operating in real-time.
- A process is a time-shared user program.
- The system's reaction time refers to how long it takes to process user input and present the necessary, updated information.
- Multitasking operating systems are often known as time-sharing systems.
- Multiprogramming is the term for sharing the CPU when two or more programs are run simultaneously in memory.

9.5 Keywords

- Conducive: making a certain situation or outcome likely or possible
- Employed: make use of

- Embedded computer: a combination of computer hardware and software designed for a specific function
- Resource allocator: optimise business activities
- Kernel: the central or most important part of something

9.6 Self-Assessment Questions

- What are the advantages of multiprogramming?
- Differentiate between
 - (a) Batch processing system and Real-Time Processing System
 - (b) Real-Time System and Timesharing System
- Explain Real Time and Distributed operating systems and their characteristics.
- Differentiate between:
 - (a) multiprocessing and multiprogramming
 - (b) paging and segmentation

9.7 Case Study

Title: Monopolistic Approach

Introduction

In a market with a monopoly, one company controls the market because it is the only seller, giving it the advantage of setting prices, whereas in a market with perfect competition, many companies are active and selling their products there without having any control over the market. In light of this, Microsoft's Vista operating system represents the company's most effective setup. As there is no other firm that can compete with Microsoft in delivering Vista, the companies that are incorporating Vista as the operating system in the computers that they create are under Microsoft's control.

Since no other company can provide Vista at a price lower than Microsoft's, Microsoft controls the businesses that incorporate it as the operating system in the PCs they make. In fact, Microsoft pursues monopolistic entries, and by creating Vista, the company ensured that the money was well spent because all Vista users must pay Microsoft and no one else in order to use this operating system. If Microsoft had not

been granted a patent for Vista, any business would likely not have invested the time, money, and resources necessary to create and market copies of the complicated service, with the understanding that others would also profit from it.

Question: Discuss the pros and cons of the given case study.

9.8 References

“Introduction to Information Technology”, IITL Education Solutions Ltd., Pearson Education

- Nagpal D.P., “Computer Fundamentals”, S. Chand Publications
- Leon Alex & Leon Mathews, “Introduction to Computers”, Vikas Publishing House
- Norton. Peter, “Introduction to Computers”, TMH