

Swami Ramanand Teertha Marathwada University, Nanded

BAHIRJI SMARAK MAHAVIDYALAYA, BASMATHNAGAR

Tq. Basmath, Dist. Hingoli, Maharashtra

STUDY NOTES

FUNDAMENTALS OF COMPUTER SCIENCE

Course Code: SCSCCT1101 (Theory)

B.Sc. First Year (Computer Science) | Semester I

Periods	30 Hours
Maximum Marks	50 (ESE: 40 + CA: 10)
Credits	2
Modules	4

Prepared for the students of B.Sc. F.Y. (Computer Science)

Department of Computer Science

Academic Year 2026-27

SYLLABUS: SCSCCT1101 FUNDAMENTALS OF COMPUTER SCIENCE

Module	Topics	Hours
1.0 Basics of Computer System	1.1 Computer Architecture: CPU (ALU, MU, CU), Input Ports, Output Ports 1.2 Input / Output Devices: Keyboard, Mouse, Monitor, Printer 1.3 Memory: Main Memory, Secondary Memory 1.4 Introduction to Algorithm 1.5 Introduction to Flowcharts and its symbols	09
2.0 Input / Output Devices	2.1 Input Devices: Keyboard, Point & Draw Devices, Data Scanning Devices 2.2 Digitizer, Electronic Card Reader, Voice Recognition Devices 2.3 Output Devices: Monitor, Printer, Plotter 2.4 Screen Image Projector, Voice Response System	09
3.0 Introduction to Number System	3.1 Introduction to Decimal, Binary, Octal, Hexadecimal 3.2 ASCII Code, Gray Code 3.3 BCD Code	05
4.0 Computer Software and OS	4.1 Introduction to Software 4.2 Types of Software 4.3 Operating System and its functions 4.4 Introduction to DOS and Windows 4.5 DOS Internal and External Commands	07
	Total	30

Reference / Text Books

1. *Fundamentals of Computers*, V. Rajaraman
2. *Computers and Common Sense*, R. Hunt and Shelly Y.
3. *Computer Fundamentals* (5th Edition), P. K. Sinha

Evaluation Scheme

Component	Marks	Details
End Semester Examination (ESE)	40	Theory paper at the end of the semester
Continuous Assessment (CA)	10	Internal tests, assignments, seminars, attendance
Total	50	2 credits (30 teaching hours)

How to Use these Notes

- Read each module in the order of the syllabus; tables give quick comparisons for revision.
- Green **Exam tip** boxes mark frequently asked points; yellow boxes give module summaries.
- Practise the numerical examples in Module 3 and the DOS commands in Module 4 on a computer.
- Use the **Question Bank** at the end for self-testing before the examination.

MODULE 1: BASICS OF COMPUTER SYSTEM

Module at a glance (9 hours)

- 1.1 Computer Architecture: CPU (ALU, MU, CU), Input Ports, Output Ports
- 1.2 Input / Output Devices: Keyboard, Mouse, Monitor, Printer
- 1.3 Memory: Main Memory, Secondary Memory
- 1.4 Introduction to Algorithm
- 1.5 Introduction to Flowcharts and its symbols

1.0 Introduction to Computer

A **computer** is an electronic device that accepts data as input, stores it, processes it according to a set of instructions (a program) and produces meaningful information as output. The word "computer" comes from the Latin word *computare*, meaning "to calculate". A modern computer, however, does much more than calculate: it handles text, images, audio, video and communication.

Data and Information

Data is a collection of raw facts and figures (for example 45, 78, "Ravi") that has no meaning by itself. **Information** is data that has been processed and organised so that it becomes meaningful and useful (for example "Ravi scored 78 marks and ranked 4th"). A computer converts data into information; this is called **data processing**.

Characteristics of a Computer

Characteristic	Meaning
Speed	A computer performs millions of operations per second; speed is measured in MIPS or GHz.
Accuracy	Results are error-free as long as the input data and program are correct (GIGO: Garbage In, Garbage Out).
Diligence	It can work for long hours without tiredness, boredom or loss of concentration.
Versatility	The same machine can do accounting, drawing, playing music, browsing, etc.
Storage	Huge amounts of data can be stored and retrieved quickly.
Automation	Once a program is loaded, the computer executes it without human intervention.
No IQ	A computer has no intelligence of its own; it only follows instructions.

Basic Operations (IPO Cycle)

Every computer system performs five basic functions: **Input** (accepting data), **Storage** (keeping data and instructions), **Processing** (performing arithmetic and logical operations), **Output** (presenting results) and **Control** (directing all operations). These form the Input-Process-Output (IPO) cycle.

1.1 Computer Architecture

Computer architecture describes the internal organisation of a computer: the main functional units and how they are connected. Most modern computers follow the **Von Neumann architecture** (stored-program concept, proposed by John von Neumann in 1945), in which both instructions and data are stored in the same memory and are fetched by the CPU one after another.

Block Diagram of a Computer

Stage	Unit	Role
1	INPUT UNIT (keyboard, mouse, scanner)	Accepts data and instructions from the user and converts them to binary form.
2	MEMORY UNIT (main memory)	Holds the data, the program and intermediate results during processing.
3	CENTRAL PROCESSING UNIT (ALU + CU + registers)	Fetches, decodes and executes instructions; performs calculation and decisions.
4	OUTPUT UNIT (monitor, printer)	Converts binary results back into a human-readable form.
5	SECONDARY STORAGE (hard disk, SSD)	Permanent storage of programs and data not currently in use.

Flow of data: Input Unit → Memory ⇌ CPU → Output Unit. The Control Unit sends control signals to every other unit.

The Central Processing Unit (CPU)

The CPU is called the **brain** of the computer. It is a single chip (microprocessor) that interprets and executes instructions. It has three main components: the **Arithmetic Logic Unit (ALU)**, the **Memory Unit (MU)** (registers and cache) and the **Control Unit (CU)**.

(a) Arithmetic Logic Unit (ALU)

The ALU performs all arithmetic and logical operations. It receives the operands from registers, performs the operation and stores the result back.

- **Arithmetic operations:** addition, subtraction, multiplication, division.
- **Logical operations:** comparisons such as greater than, less than, equal to, and Boolean operations AND, OR, NOT.
- It contains circuits such as adders, comparators and shift registers, and an **accumulator** that holds intermediate results.

(b) Control Unit (CU)

The Control Unit does not process data itself; it **supervises and coordinates** the activities of the whole system. Its functions are:

- Fetching the next instruction from memory.
- Decoding the instruction to find out what operation is required.
- Generating timing and control signals to the ALU, memory and I/O devices.
- Controlling the flow of data between the CPU, memory and peripherals.

(c) Memory Unit (MU) / Registers

Registers are very small, extremely fast storage locations inside the CPU. They temporarily hold data and instructions during execution.

Register	Full form	Function
PC	Program Counter	Holds the address of the next instruction to be fetched.
IR	Instruction Register	Holds the instruction currently being executed.
MAR	Memory Address Register	Holds the address of the memory location to be accessed.

Register	Full form	Function
MDR / MBR	Memory Data / Buffer Register	Holds the data being read from or written to memory.
ACC	Accumulator	Stores results of ALU operations.

Instruction Cycle (Fetch-Decode-Execute)

1. **Fetch:** the CU reads the instruction whose address is in the PC from memory into the IR.
2. **Decode:** the CU interprets the instruction and identifies the operation and operands.
3. **Execute:** the ALU or another unit carries out the operation.
4. **Store:** the result is written to a register or memory and the PC is updated.

Exam tip

The CPU is also described by its **clock speed** (measured in GHz), **word length** (32 bit / 64 bit) and **number of cores** (dual-core, quad-core, etc.).

Input Ports and Output Ports

A **port** is a socket on the computer through which peripheral devices are connected. Input ports carry data *into* the computer; output ports carry data *out of* it. Many ports are bidirectional.

Port	Typical use	Remark
PS/2 port	Keyboard and mouse (old)	Round 6-pin connector; replaced by USB.
USB port	Keyboard, mouse, pen drive, printer, phone	Universal Serial Bus; plug-and-play, hot-swappable. USB 2.0, 3.0, Type-C.
Serial port (COM)	Modem, older mouse	Sends data one bit at a time.
Parallel port (LPT)	Older printers	Sends 8 bits at a time.
VGA port	Analog monitor / projector	15-pin D-shaped connector.
HDMI port	Digital monitor, TV, projector	Carries high-definition video and audio.
Audio jacks	Microphone (input), speakers (output)	Usually colour-coded: pink, green, blue.
Ethernet (RJ-45)	Network cable	Connects the computer to a LAN.

1.2 Input / Output Devices (Introduction)

Input and output devices are collectively called **peripherals**. They allow communication between the user and the computer. This section gives a first introduction to four common devices; Module 2 covers the whole range in detail.

Device	Type	Working in brief
Keyboard	Input	A set of keys (typically 101/104); each key press generates a scan code that is converted to ASCII by the keyboard controller. Layout: QWERTY.
Mouse	Input (pointing)	Controls a pointer on the screen. Optical mice use an LED and sensor to detect movement; has left, right button and scroll wheel.
Monitor	Output	Displays text, images and video (soft copy). Types: CRT, LCD, LED. Resolution in pixels (e.g. 1920 × 1080).
Printer	Output	Produces permanent paper output (hard copy). Types: dot-matrix, inkjet, laser.

1.3 Memory

Memory is the part of a computer that stores data, instructions and results, either temporarily or permanently. The smallest unit of memory is the **bit** (binary digit: 0 or 1).

Units of Memory

Unit	Equivalent
1 Nibble	4 bits
1 Byte	8 bits
1 Kilobyte (KB)	1024 bytes
1 Megabyte (MB)	1024 KB
1 Gigabyte (GB)	1024 MB
1 Terabyte (TB)	1024 GB
1 Petabyte (PB)	1024 TB

Memory Hierarchy

Memories are arranged in a hierarchy: moving **up** the hierarchy gives higher speed and cost per byte but smaller capacity; moving **down** gives lower speed and cost but larger capacity.

Level	Memory	Speed	Capacity	Volatile?
1	Registers (inside CPU)	Fastest	Bytes	Yes
2	Cache memory (L1, L2, L3)	Very fast	KB to a few MB	Yes
3	Main memory (RAM)	Fast	GBs	Yes
4	Secondary memory (HDD, SSD)	Slow	Hundreds of GB to TB	No
5	Tertiary / backup (tape, cloud)	Slowest	Very large	No

Main Memory (Primary Memory)

Main memory is directly accessible by the CPU. Programs and data must be loaded into main memory before they can be executed. It is built from semiconductor chips and has two types: **RAM** and **ROM**.

RAM (Random Access Memory)

- Read-write memory; contents can be changed.
- **Volatile**: all contents are lost when power is switched off.
- Called "random access" because any location can be accessed in the same time.
- Used to hold the operating system, running programs and data currently in use.

Type	Full form	Features
SRAM	Static RAM	Uses flip-flops; faster, costlier, needs no refreshing. Used for cache.
DRAM	Dynamic RAM	Uses capacitors; slower, cheaper, needs periodic refreshing. Used as main memory.

Type	Full form	Features
SDRAM / DDR	Synchronous DRAM / Double Data Rate	Synchronised with the system clock; DDR3, DDR4, DDR5 are used in modern PCs.

ROM (Read Only Memory)

- **Non-volatile:** contents are retained when power is off.
- Normally data can only be read, not changed easily.
- Stores the **BIOS** / boot program that starts the computer.

Type	Full form	Features
MROM	Masked ROM	Programmed by the manufacturer during fabrication; cannot be changed.
PROM	Programmable ROM	Can be programmed once by the user using a special device.
EPROM	Erasable PROM	Erased by ultraviolet light and reprogrammed.
EEPROM	Electrically Erasable PROM	Erased electrically, byte by byte; used in flash BIOS.

Basis	RAM	ROM
Nature	Read/Write	Mostly read-only
Volatility	Volatile	Non-volatile
Use	Temporary working storage	Permanent start-up instructions
Speed / Cost	Fast, costlier per byte	Slower, cheaper
Capacity	Large (GBs)	Small (MBs)

Cache Memory

Cache is a small, very high-speed memory placed between the CPU and RAM. It stores frequently used instructions and data so that the CPU does not have to wait for the slower RAM. A *cache hit* occurs when the required data is found in the cache; a *cache miss* occurs otherwise.

Secondary Memory (Auxiliary Memory)

Secondary memory provides **permanent, large-capacity** storage. It is not directly accessed by the CPU; data is first brought into RAM. It is non-volatile and cheaper per byte than RAM, but slower.

Type	Examples	Technology and notes
Magnetic	Hard disk (HDD), floppy disk, magnetic tape	Data stored as magnetised spots on platters or tape; HDD capacity in GB/TB; tape is sequential access.
Optical	CD (700 MB), DVD (4.7 GB), Blu-ray (25 GB)	Data read by a laser beam from pits and lands on the disc surface.
Solid-state / Flash	SSD, pen drive, memory card	No moving parts; uses NAND flash chips; fast, durable and silent.

Remember

Hard disk structure: **platters → tracks → sectors → cylinders**. Access time = seek time + rotational latency + transfer time.

1.4 Introduction to Algorithm

An **algorithm** is a finite, step-by-step set of well-defined instructions for solving a problem or accomplishing a task. The word comes from the name of the Persian mathematician *Al-Khwarizmi*. Before writing a program, a programmer first designs an algorithm.

Characteristics of a Good Algorithm

- **Input:** zero or more well-defined inputs.
- **Output:** at least one output is produced.
- **Definiteness:** every step is clear and unambiguous.
- **Finiteness:** it terminates after a finite number of steps.
- **Effectiveness:** every step is simple enough to be carried out exactly.
- **Independence:** it is independent of any programming language.

Example 1: Find the Sum of Two Numbers

```
Step 1: Start
Step 2: Read the numbers A and B
Step 3: Calculate SUM = A + B
Step 4: Display SUM
Step 5: Stop
```

Example 2: Find the Largest of Three Numbers

```
Step 1: Start
Step 2: Read A, B, C
Step 3: If A > B and A > C then LARGEST = A
Step 4: Else if B > C then LARGEST = B
Step 5: Else LARGEST = C
Step 6: Display LARGEST
Step 7: Stop
```

Example 3: Check Whether a Number is Even or Odd

```
Step 1: Start
Step 2: Read N
Step 3: If N mod 2 = 0 then print "Even"
Step 4: Else print "Odd"
Step 5: Stop
```

Example 4: Factorial of a Number

```
Step 1: Start
Step 2: Read N
Step 3: Set F = 1, I = 1
Step 4: Repeat Steps 5-6 while I <= N
Step 5:     F = F * I
Step 6:     I = I + 1
Step 7: Display F
Step 8: Stop
```

Algorithm	Program
Plan of the solution	Implementation of the plan
Written in plain English / pseudocode	Written in a programming language (C, Java, ...)
Language independent	Language dependent
Written at the design stage	Written at the coding stage

1.5 Introduction to Flowcharts and its Symbols

A **flowchart** is a pictorial (graphical) representation of an algorithm using standard symbols connected by arrows. It shows the sequence of operations and decisions, making the logic easy to understand, communicate and debug.

Standard Flowchart Symbols

Symbol (shape)	Name	Purpose
Oval / Rounded rectangle	Terminator	Marks the START or STOP (END) of the flowchart.
Parallelogram	Input / Output	Used for reading input (READ) and displaying output (PRINT).
Rectangle	Process	Used for calculations and assignments, e.g. $SUM = A + B$.
Diamond	Decision	Tests a condition; has two exits, YES/TRUE and NO/FALSE.
Circle	Connector	Joins parts of a flowchart on the same page (on-page connector).
Pentagon (home plate)	Off-page connector	Joins parts of a flowchart across pages.
Arrow	Flow line	Shows the direction of flow of the logic.
Rectangle with double side bars	Predefined process	Calls a subroutine or function defined elsewhere.
Hexagon	Preparation	Used for loop initialisation / setup.

Flowchart for "Largest of Two Numbers" (described in text)

Step	Symbol	Content
1	Oval	START
2	Parallelogram	READ A, B
3	Diamond	Is $A > B$?
4	Parallelogram (YES path)	PRINT "A is largest"
5	Parallelogram (NO path)	PRINT "B is largest"
6	Oval	STOP

Rules for Drawing Flowcharts

- A flowchart has only one START and one STOP.
- Flow is normally from top to bottom and left to right.
- Use only standard symbols and keep them neat and uniform in size.
- A decision symbol has one entry and two exits; process symbols have one entry and one exit.

- Flow lines must not cross each other; use connectors if the chart is large.

Advantages and Limitations

Advantages	Limitations
Better communication of logic	Complex programs make large, untidy charts
Effective analysis and documentation	Difficult to modify; the chart may have to be redrawn
Helps in debugging and testing	No standard for the amount of detail to show
Acts as a guide while coding	Time-consuming to draw by hand

Types of Program Logic (Structures)

- **Sequence:** statements executed one after another.
- **Selection (Decision):** a branch is chosen on a condition (if-else).
- **Iteration (Loop):** a set of steps is repeated while a condition holds.

Example: Flowchart Logic for the Sum of the First N Natural Numbers

```
START
READ N
SUM = 0, I = 1
WHILE I <= N (decision diamond)
    SUM = SUM + I
    I = I + 1
PRINT SUM
STOP
```

MODULE 2: INPUT / OUTPUT DEVICES

Module at a glance (9 hours)

2.1 Input Devices: Keyboard, Point & Draw Devices, Data Scanning Devices

2.2 Digitizer, Electronic Card Reader, Voice Recognition Devices

2.3 Output Devices: Monitor, Printer, Plotter

2.4 Screen Image Projector, Voice Response System

2.0 Introduction

An **input device** is a hardware component used to enter data and instructions into a computer. An **output device** receives the processed result from the computer and presents it to the user in a human-readable form. Input devices convert human-readable data into machine-readable (binary) form; output devices do the reverse.

Basis	Input device	Output device
Direction	User → Computer	Computer → User
Function	Feeds data and commands	Presents results
Conversion	Human form → binary	Binary → human form
Examples	Keyboard, mouse, scanner	Monitor, printer, plotter

2.1 Input Devices

2.1.1 Keyboard

The keyboard is the most common input device. It is similar to a typewriter keyboard but has additional keys. When a key is pressed, the keyboard encoder generates a unique **scan code**, which is sent to the computer and converted into the corresponding character code (ASCII).

Key group	Keys	Purpose
Alphanumeric keys	A-Z, 0-9, punctuation	Enter text and numbers. Standard layout is QWERTY.
Function keys	F1 to F12	Perform special tasks depending on the software (e.g. F1 = Help, F5 = Refresh).
Numeric keypad	0-9, +, -, *, /, Enter	Fast numeric entry; activated with Num Lock.
Cursor / navigation keys	Arrows, Home, End, PgUp, PgDn	Move the cursor on the screen.
Modifier keys	Shift, Ctrl, Alt, Caps Lock	Change the meaning of other keys.
Special keys	Esc, Tab, Enter, Backspace, Delete, Insert, Print Screen	Editing and control operations.

Types of keyboard: QWERTY (standard), Dvorak, ergonomic, wireless (Bluetooth/RF), virtual (on-screen/touch) and multimedia keyboards. **Advantages:** familiar, cheap, accurate for text. **Limitations:** slow for graphics work; needs typing skill.

2.1.2 Point and Draw Devices

These devices let the user control the position of a pointer or draw on the screen without typing. They are essential for Graphical User Interfaces (GUI).

(a) Mouse

A mouse is a hand-held pointing device that moves a cursor over the screen. Main actions: **click**, **double-click**, **right-click**, **drag and drop** and **scroll**.

- **Mechanical mouse:** a rubber ball rolls against internal rollers; obsolete today.
- **Optical mouse:** an LED/laser and a small camera sensor track movement; no moving parts.
- **Wireless mouse:** communicates through Bluetooth or an RF dongle.

(b) Trackball

A trackball is like an upside-down mouse: a ball is rotated with the fingers or palm while the device stays fixed. It needs little desk space and is used in laptops (older), kiosks and CAD workstations.

(c) Joystick

A joystick is a stick that pivots on a base and reports its direction and angle. It is used mainly for video games, flight simulators and controlling robots or cranes.

(d) Light Pen

A light pen is a pen-shaped, light-sensitive device. When its tip touches a CRT screen, it detects the moment the electron beam passes and so identifies the screen position. It is used for selecting and drawing directly on the screen.

(e) Touch Screen

A touch screen allows the user to select items by touching the display with a finger or stylus. Types: **resistive**, **capacitive** (used in smartphones) and **infrared**. Used in ATMs, mobile phones, tablets, kiosks and point-of-sale terminals.

(f) Touchpad

A flat, touch-sensitive surface below the keyboard of a laptop; finger movement controls the pointer.

(g) Graphics Tablet / Stylus

A flat board on which the user draws with a stylus. It is popular with designers and artists because it gives pen-like control and pressure sensitivity.

2.1.3 Data Scanning Devices

Data scanning devices capture data directly from its source in a machine-readable form, avoiding manual typing. This increases speed and reduces errors.

(a) Image Scanner

An image scanner converts printed text, pictures or photographs into a digital image. Light reflected from the document is sensed by a **CCD (Charge-Coupled Device)** array, and the analog signal is converted to digital form. Types: **flatbed**, **sheet-fed**, **hand-held** and **drum** scanners. Resolution is measured in **dpi** (dots per inch).

(b) Optical Character Recognition (OCR)

OCR software converts the scanned image of printed or typed text into editable text by recognising the shapes of characters. Used in digitising books, number-plate reading and passport processing.

(c) Optical Mark Recognition (OMR)

OMR detects the presence or absence of a mark (pencil shading) in a pre-defined position on a form. It is used for evaluating objective-type answer sheets (e.g. MCQ examinations), surveys and lottery tickets. It is fast and accurate, but suitable only for marking, not for reading text.

(d) Magnetic Ink Character Recognition (MICR)

MICR reads characters printed with special **magnetic ink** (iron-oxide based) and a special font. It is used mainly by banks for reading the cheque number, bank code and account number at the bottom of a cheque. It is highly secure and difficult to forge.

(e) Bar Code Reader

A bar code is a pattern of vertical lines of different thickness and spacing that represents data. A bar code reader uses a laser or LED to scan the pattern and decode it. Used in supermarkets, libraries, hospitals and logistics. **QR codes** are two-dimensional codes that store more data and are read by mobile cameras.

Device	Reads	Typical application
Image scanner	Pictures and printed pages	Digitising documents, photos
OCR	Printed/typed characters	Converting books to editable text
OMR	Pencil marks / bubbles	MCQ answer sheets, surveys
MICR	Magnetic-ink characters	Cheque processing in banks
Bar code reader	Bar code / QR code	Billing, inventory control

2.2 Other Input Devices

2.2.1 Digitizer

A digitizer (digitising tablet) converts hand-drawn drawings, maps and sketches into digital data. The user moves a **stylus or puck** over a flat tablet containing a grid of fine wires; the device records the (x, y) co-ordinates of each point and sends them to the computer. It is widely used in **CAD/CAM**, architecture, cartography (map making) and engineering design.

2.2.2 Electronic Card Reader

An electronic card reader reads data stored on a plastic card. Common cards and readers:

Card	How data is stored	Examples
Magnetic stripe card	Data is recorded on a magnetic stripe at the back; read by swiping.	Older ATM / debit / credit cards, hotel keys
Smart card (chip card)	A microprocessor chip embedded in the card stores data and can process it.	EMV bank cards, SIM cards, Aadhaar-linked cards
RFID / NFC card	A tiny antenna and chip transmit data wirelessly over short range.	Metro cards, access badges, contactless payments

Card readers are used in ATMs, point-of-sale (POS) machines, attendance and access-control systems. Security is improved by a PIN or biometric verification.

2.2.3 Voice Recognition Devices

A voice recognition (speech recognition) system takes spoken words through a **microphone**, converts the sound wave to digital form and compares it with stored patterns to identify the words. It is used for dictation, voice commands in smartphones and virtual assistants (Google Assistant, Alexa, Siri), and voice-based authentication.

- **Speaker-dependent:** must be trained for each user; high accuracy.
- **Speaker-independent:** works for any speaker; lower accuracy.

Limitations: affected by background noise, accents and speaking speed; may need a pause between words in older systems.

2.3 Output Devices

Output may be **soft copy** (displayed on screen or played as sound, not permanent) or **hard copy** (printed on paper, permanent).

2.3.1 Monitor (Visual Display Unit, VDU)

The monitor is the primary output device. It displays text, graphics and video as a matrix of tiny dots called **pixels** (picture elements).

Important Terms

Term	Meaning
Pixel	Smallest addressable dot on the screen.
Resolution	Number of pixels horizontally × vertically, e.g. 1366 × 768 (HD), 1920 × 1080 (Full HD), 3840 × 2160 (4K).
Screen size	Measured diagonally in inches (e.g. 15.6", 21").
Refresh rate	Number of times the screen is redrawn per second (Hz), e.g. 60 Hz, 144 Hz.
Dot pitch	Distance between two pixels of the same colour; smaller means a sharper image.
Aspect ratio	Ratio of width to height, e.g. 4:3, 16:9.
Colour depth	Number of bits per pixel; 24-bit gives about 16.7 million colours.

Types of Monitors

Type	Technology	Advantages	Disadvantages
CRT	Electron gun fires a beam on a phosphor-coated screen.	Low cost, good colour	Bulky, heavy, high power, radiation
LCD	Liquid crystals controlled by a fluorescent backlight.	Thin, light, low power	Limited viewing angle
LED	LCD with LED backlighting.	Brighter, slimmer, energy saving	Costlier than LCD
OLED	Each pixel emits its own light.	Deep blacks, flexible, very thin	Expensive, ageing of pixels
Plasma	Gas cells glow when electrically charged.	Large screens, wide angle	High power, heavy

Monitors can also be classified as **monochrome**, **grayscale** and **colour (RGB)** displays, and by interface as **VGA** (analog) and **HDMI/DVI/DisplayPort** (digital).

2.3.2 Printer

A printer produces permanent hard copy of text and graphics on paper. Printers are classified into **impact** printers (a mechanism strikes the paper through an inked ribbon) and **non-impact** printers (no striking; use ink spray, heat or laser). Printing speed is measured in **CPS** (characters per second), **LPM** (lines per minute) or **PPM** (pages per minute); quality in **dpi**.

(A) Impact Printers

- **Dot-matrix printer:** a print head with 9 or 24 pins strikes an inked ribbon to form characters from dots. Can make carbon copies; noisy; low quality; used for invoices and bank pass books.

- **Daisy-wheel printer:** a wheel with fully formed characters on petals; letter-quality output but slow and cannot print graphics.
- **Line printer (drum / chain):** prints an entire line at a time; very fast (up to 3000 LPM); used in large data centres.

(B) Non-Impact Printers

- **Inkjet printer:** sprays tiny droplets of liquid ink through nozzles; good colour output; cheap to buy but ink is costly. Speed 5 to 20 PPM.
- **Laser printer:** a laser beam forms an image on a photosensitive drum; **toner** powder sticks to the charged area and is fused on paper by heat. Fast, high quality, quiet; resolution 600 to 1200 dpi or more.
- **Thermal printer:** heat-sensitive paper darkens when heated; used in ATM receipts, billing machines and bar-code labels.
- **3D printer:** builds a solid object layer by layer from a digital model.

Basis	Dot-matrix	Inkjet	Laser
Type	Impact	Non-impact	Non-impact
Quality	Low	Good	Excellent
Speed	Slow	Medium	Fast
Noise	Noisy	Quiet	Very quiet
Colour	No (mostly)	Yes	Yes (colour lasers)
Running cost	Low	High (ink)	Moderate (toner)
Carbon copies	Possible	Not possible	Not possible

2.3.3 Plotter

A plotter is an output device that draws **high-quality, large-size vector graphics** such as engineering drawings, architectural plans, maps and posters. Unlike a printer, which prints dots, a plotter draws continuous lines using one or more **pens** controlled by the computer.

- **Drum plotter:** the paper moves back and forth on a rotating drum while the pen moves sideways.
- **Flatbed plotter:** the paper is fixed on a flat table and the pen moves in both X and Y directions.
- **Inkjet plotter:** large-format inkjet printers used today for banners and CAD prints.

Uses: CAD/CAM, architecture, civil and mechanical engineering, GIS mapping, textile design. **Disadvantage:** costly and slow compared with printers.

2.4 Other Output Devices

2.4.1 Screen Image Projector

A screen image projector (multimedia projector) projects the image of the computer screen onto a large screen or wall so that it can be seen by an audience. It is used in classrooms, conferences and cinemas.

Type	Working principle
LCD projector	Light from a lamp passes through LCD panels (red, green, blue) and is focused by a lens.
DLP projector	Uses a Digital Micromirror Device (DMD) chip with millions of tiny mirrors that reflect light to form the image.
LED / Laser projector	Uses LED or laser light sources; long life and compact size.

Key specifications: **brightness** in lumens, **contrast ratio**, **resolution** (XGA, HD, Full HD) and **throw distance**. Connected through VGA, HDMI or wirelessly.

2.4.2 Voice Response System (Audio Output)

A voice response system produces spoken output from the computer. It can work in two ways:

- **Voice reproduction (pre-recorded)**: stores recorded human speech (words and phrases) in digital form and plays them back in the required order, e.g. railway enquiry, "your account balance is...".
- **Speech synthesis (Text-to-Speech, TTS)**: converts text into artificial speech using rules of pronunciation; can speak any new text.

Applications: Interactive Voice Response (IVR) in call centres and banking, railway announcements, talking aids for the visually impaired, GPS navigation voices and virtual assistants. Output is delivered through **speakers, headphones or telephone lines**.

Quick revision: match the device

Cheque reading → **MICR** | MCQ sheets → **OMR** | Supermarket billing → **Bar code reader** | Engineering drawings → **Plotter** | Class presentation → **Projector** | Railway announcement → **Voice response system** | Map to digital data → **Digitizer**.

MODULE 3: INTRODUCTION TO NUMBER SYSTEM

Module at a glance (5 hours)

3.1 Introduction to Decimal, Binary, Octal, Hexadecimal

3.2 ASCII Code, Gray Code

3.3 BCD Code

3.1 Introduction to Number Systems

A **number system** is a method of representing numbers using a fixed set of symbols (digits). Every number system has a **base (radix)**, which equals the number of distinct digits it uses. In a **positional number system**, the value of a digit depends on the digit itself and on its position, which carries a weight equal to a power of the base.

A number N with digits $d(n-1) \dots d_1 d_0 . d(-1) d(-2)$ in base b has the value: $N = \sum d(i) \times b^i$, where i is the position of the digit (positions to the right of the point are negative).

System	Base	Digits used	Used for
Decimal	10	0 1 2 3 4 5 6 7 8 9	Everyday human counting
Binary	2	0 1	Internal working of all computers
Octal	8	0 1 2 3 4 5 6 7	Compact form of binary (groups of 3 bits)
Hexadecimal	16	0-9 and A B C D E F	Compact form of binary (groups of 4 bits); memory addresses, colour codes

Hexadecimal letters stand for: A = 10, B = 11, C = 12, D = 13, E = 14, F = 15.

3.1.1 Decimal Number System

The decimal system has base 10. Each position has a weight that is a power of 10.

Example: $5432.67 = 5 \times 10^3 + 4 \times 10^2 + 3 \times 10^1 + 2 \times 10^0 + 6 \times 10^{-1} + 7 \times 10^{-2}$

3.1.2 Binary Number System

The binary system has base 2 and only two digits, **0 and 1**, called **bits**. Computers use it because electronic circuits have two stable states (ON/OFF, high/low voltage). The rightmost bit is the **Least Significant Bit (LSB)** and the leftmost is the **Most Significant Bit (MSB)**.

Example: $(1011.01)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = 8 + 0 + 2 + 1 + 0 + 0.25 = 11.25_{10}$

Decimal	Binary	Octal	Hexadecimal
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7

Decimal	Binary	Octal	Hexadecimal
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

3.1.3 Octal Number System

Base 8, digits 0 to 7. One octal digit equals exactly **3 binary bits**.

Example: $(345)_8 = 3 \times 8^2 + 4 \times 8^1 + 5 \times 8^0 = 192 + 32 + 5 = 229_{10}$

3.1.4 Hexadecimal Number System

Base 16, digits 0 to 9 and A to F. One hexadecimal digit equals exactly **4 binary bits**.

Example: $(2AF)_{16} = 2 \times 16^2 + 10 \times 16^1 + 15 \times 16^0 = 512 + 160 + 15 = 687_{10}$

Number System Conversions

A. Any Base to Decimal

Multiply each digit by the weight of its position (power of the base) and add all the products. See the worked examples above.

B. Decimal to Any Base (Division-Remainder Method)

Integer part: divide the number repeatedly by the target base; write down the remainders; read them from **bottom to top**. **Fractional part:** multiply the fraction repeatedly by the base; the integer parts of the results, read **top to bottom**, form the fraction.

Example 1: Convert $(45)_{10}$ to binary.

Division	Quotient	Remainder
$45 \div 2$	22	1 (LSB)
$22 \div 2$	11	0
$11 \div 2$	5	1
$5 \div 2$	2	1
$2 \div 2$	1	0
$1 \div 2$	0	1 (MSB)

Reading remainders from bottom to top: $(45)_{10} = (101101)_2$

Example 2: Convert $(0.625)_{10}$ to binary.

Multiplication	Result	Integer part
0.625×2	1.25	1

Multiplication	Result	Integer part
0.25×2	0.5	0
0.5×2	1.0	1

Reading from top to bottom: $(0.625)_{10} = (0.101)_2$

Example 3: Convert $(1234)_{10}$ to octal and hexadecimal.

- Octal: $1234 \div 8 = 154 \text{ R } 2$; $154 \div 8 = 19 \text{ R } 2$; $19 \div 8 = 2 \text{ R } 3$; $2 \div 8 = 0 \text{ R } 2$. Reading upward: $(2322)_8$.
- Hexadecimal: $1234 \div 16 = 77 \text{ R } 2$; $77 \div 16 = 4 \text{ R } 13 \text{ (D)}$; $4 \div 16 = 0 \text{ R } 4$. Reading upward: $(4D2)_{16}$.

C. Binary to Octal and Hexadecimal (and back)

Binary $\rightarrow$ Octal: group the bits in threes starting from the binary point (add zeros if needed), and replace each group by its octal digit. **Binary $\rightarrow$ Hex:** group in fours and replace each group by its hex digit.

Example: $(110101101)_2 \rightarrow$ groups 110 | 101 | 101 $\rightarrow (655)_8$. Also 0001 | 1010 | 1101 $\rightarrow (1AD)_{16}$.

Octal / Hex $\rightarrow$ Binary: replace each digit by its 3-bit (octal) or 4-bit (hex) binary equivalent. *Example:* $(7B)_{16} = 0111 1011 = (1111011)_2$.

D. Octal $\leftrightarrow$ Hexadecimal

There is no direct method. Convert first to **binary** as an intermediate step. *Example:* $(257)_8 = 010 101 111 = 0 1010 1111 = (AF)_{16}$.

Binary Arithmetic (brief)

Addition	Subtraction	Multiplication
$0 + 0 = 0$	$0 - 0 = 0$	$0 \times 0 = 0$
$0 + 1 = 1$	$1 - 0 = 1$	$0 \times 1 = 0$
$1 + 0 = 1$	$1 - 1 = 0$	$1 \times 0 = 0$
$1 + 1 = 10$ (0 carry 1)	$0 - 1 = 1$ (borrow 1)	$1 \times 1 = 1$
$1 + 1 + 1 = 11$		

Example: $1011 + 0110 = 10001$ ($11 + 6 = 17$).

Signed Numbers and Complements (brief)

- 1's complement:** invert every bit (0 becomes 1, 1 becomes 0).
- 2's complement:** 1's complement + 1; used by computers to represent negative numbers and to perform subtraction by addition.
- Example:* 8-bit representation of -5: $+5 = 00000101 \rightarrow$ 1's complement $11111010 \rightarrow$ add 1 $\rightarrow 11111011$.

3.2 Computer Codes

A **code** is a systematic representation of numbers, letters and symbols in binary form so that computers can store and process them. Coding is necessary because a computer understands only 0s and 1s.

3.2.1 ASCII Code

ASCII stands for **American Standard Code for Information Interchange**. It is a character-encoding standard that assigns a unique number to each character. Standard ASCII is a **7-bit** code giving $2^7 = 128$ characters (0 to 127).

Extended ASCII uses 8 bits and gives 256 characters (0 to 255).

Character group	Decimal range	Examples
Control characters	0 to 31 and 127	NUL = 0, LF = 10, CR = 13, ESC = 27, DEL = 127

Character group	Decimal range	Examples
Space and punctuation	32 to 47, 58 to 64, 91 to 96, 123 to 126	Space = 32, ! = 33, @ = 64
Digits 0 to 9	48 to 57	"0" = 48, "9" = 57
Upper-case letters A to Z	65 to 90	"A" = 65, "Z" = 90
Lower-case letters a to z	97 to 122	"a" = 97, "z" = 122

Character	Decimal	Binary (7-bit)	Hex
A	65	1000001	41
B	66	1000010	42
Z	90	1011010	5A
a	97	1100001	61
z	122	1111010	7A
0	48	0110000	30
9	57	0111001	39
Space	32	0100000	20

Useful facts

Lower-case letter = upper-case letter + 32 (e.g. "a" = 65 + 32 = 97). The digit character "5" has ASCII value 48 + 5 = 53. To write the word **CAT** in ASCII: 67 65 84 → 1000011 1000001 1010100.

3.2.2 Gray Code

Gray code (reflected binary code) is a non-weighted code in which **two successive values differ in only one bit**. This property avoids errors when a value changes, and so it is used in rotary/position encoders, analog-to-digital converters and Karnaugh maps.

Decimal	Binary	Gray code
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110

Decimal	Binary	Gray code
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Binary to Gray Conversion

1. The **MSB** of the Gray code is the same as the MSB of the binary number.
2. Each next Gray bit = XOR of the **current** binary bit and the **previous** binary bit.

Example: Binary 1011 → Gray: 1; $1 \oplus 0 = 1$; $0 \oplus 1 = 1$; $1 \oplus 1 = 0 \rightarrow 1110$.

Gray to Binary Conversion

1. The MSB of the binary number is the same as the MSB of the Gray code.
2. Each next binary bit = XOR of the **current Gray** bit and the **previous binary** bit.

Example: Gray 1110 → Binary: 1; $1 \oplus 1 = 0$; $0 \oplus 1 = 1$; $1 \oplus 0 = 1 \rightarrow 1011$.

3.3 BCD Code

BCD (Binary Coded Decimal) represents each **decimal digit** by its own 4-bit binary equivalent (8421 weights). Only the combinations 0000 to 1001 are valid; 1010 to 1111 are invalid in BCD.

Decimal digit	0	1	2	3	4	5	6	7	8	9
BCD (8421)	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

Example 1: $(47)_{10}$ in BCD = 0100 0111 (4 = 0100, 7 = 0111).

Basis	BCD	Pure binary
Conversion	Digit by digit (each decimal digit coded separately)	Whole number converted
$(25)_{10}$	0010 0101	11001
Bits needed	More bits (4 per digit)	Fewer bits
Arithmetic	More complex (needs correction)	Simple
Use	Digital clocks, calculators, meters, displays	Internal computation

Advantages: easy conversion to and from decimal; ideal for seven-segment displays and financial applications.

Disadvantages: wasteful of memory; arithmetic is more complicated.

Practice problems

1) Convert $(156)_{10}$ to binary, octal and hex. 2) Convert $(10110110)_2$ to decimal, octal and hex. 3) Write the ASCII value of "M" and "m". 4) Convert binary 1101 to Gray code. 5) Express $(359)_{10}$ in BCD. *Answers:* 1) 10011100, 234, 9C 2) 182, 266, B6 3) 77, 109 4) 1011 5) 0011 0101 1001

MODULE 4: COMPUTER SOFTWARE AND OPERATING SYSTEM

Module at a glance (7 hours)

4.1 Introduction to Software

4.2 Types of Software

4.3 Operating System and its functions

4.4 Introduction to DOS and Windows

4.5 DOS Internal and External Commands

4.1 Introduction to Software

A computer system has two parts: **hardware** (the physical, touchable components) and **software** (the intangible set of programs and data that tells the hardware what to do). Without software, hardware is useless; without hardware, software cannot run.

A **program** is a set of instructions written in a programming language to perform a specific task. A collection of related programs, along with their documentation, is called **software**. The term was popularised by John Tukey in 1958.

Basis	Hardware	Software
Nature	Physical components	Set of programs / instructions
Touch	Can be touched	Cannot be touched
Failure causes	Wear, dust, heat, damage	Bugs, viruses, corruption
Replacement	Replaced by another part	Re-installed or updated
Examples	CPU, RAM, keyboard	Windows, MS Word, Chrome

Software Characteristics

- Software is **developed or engineered**, it does not wear out physically.
- It is **logical** in nature and is stored on storage media.
- It needs hardware to run and is easily copied and modified.

4.2 Types of Software

Software is broadly divided into **System Software** and **Application Software**; a third group, **Utility Software**, is sometimes treated separately.

Category	Sub-types / Examples
System software	Operating systems, language translators (compiler, interpreter, assembler), device drivers, firmware, linker and loader
Application software	General purpose (packaged) and special purpose (tailored/bespoke)
Utility software	Antivirus, disk defragmenter, backup tools, file compression

4.2.1 System Software

System software manages the computer hardware and provides a platform on which application software runs. The user usually does not interact with it directly.

(a) Operating System

The most important system software; it manages hardware and software resources. Examples: MS-DOS, Windows, Linux, macOS, Android, iOS. (Detailed in 4.3.)

(b) Language Translators

Computers understand only machine language (0s and 1s). A **translator** converts a program written in another language into machine code.

Translator	Function	Features
Assembler	Converts assembly language to machine code.	One-to-one translation; used for low-level programs.
Compiler	Translates the entire high-level program at once into machine code.	Reports all errors after compilation; fast execution; e.g. C, C++ compilers.
Interpreter	Translates and executes the program line by line.	Stops at the first error; slower; e.g. Python, BASIC.

(c) Device Drivers

A device driver is a small program that enables the operating system to communicate with a hardware device such as a printer, scanner or graphics card. Each device needs its own driver.

(d) Firmware

Permanent software stored in ROM/flash memory (e.g. BIOS), supplied with hardware devices like routers and washing machines.

4.2.2 Application Software

Application software is designed to perform specific tasks for the user.

Type	Description	Examples
General purpose (packaged)	Ready-made software for common tasks; sold to many users.	MS Word, Excel, PowerPoint, Tally, Photoshop, web browsers
Special purpose (customised)	Developed for the specific needs of one organisation.	Hospital management, payroll, railway reservation systems

Common categories of application software:

- **Word processors:** MS Word, LibreOffice Writer.
- **Spreadsheets:** MS Excel, Google Sheets.
- **Presentation software:** MS PowerPoint.
- **Database software:** MS Access, MySQL, Oracle.
- **Graphics / multimedia:** CorelDRAW, Photoshop, VLC.
- **Web browsers:** Chrome, Firefox, Edge.
- **Educational and entertainment software.**

4.2.3 Utility Software

Utility programs perform maintenance and support tasks: **antivirus** (detects and removes viruses), **disk cleanup and defragmenter**, **file compression** (WinZip, 7-Zip), **backup and restore** and **disk partitioning** tools.

Basis	System software	Application software
Purpose	Operates and manages the computer	Performs user tasks

Basis	System software	Application software
Dependence	Runs independently of applications	Needs system software to run
Installation	Installed with the computer	Installed by the user as required
Language	Usually low-level (C, assembly)	Usually high-level
Examples	Windows, Linux, device drivers	MS Word, Tally, VLC

Open Source, Freeware, Shareware and Proprietary Software

Kind	Meaning	Example
Proprietary	Owned by a company; source code hidden; licence fee required.	MS Office, Windows
Open source	Source code freely available to view, modify and share.	Linux, LibreOffice, Firefox
Freeware	Free to use but source code not provided.	Adobe Reader, Skype
Shareware	Free to try for a limited time, then payment is needed.	WinRAR, trial antivirus

4.3 Operating System and its Functions

An **Operating System (OS)** is a system software that acts as an **interface between the user and the computer hardware**. It manages all hardware and software resources and provides services to application programs. It is the first program loaded into memory (**booting**) when the computer starts, and it stays there until shut-down.

Layer (top to bottom)	Description
User	Person using the computer
Application programs	Word processor, browser, games
Operating system	Manages resources and provides services
Hardware	CPU, memory, I/O devices

Main Functions of an Operating System

1. **Process management:** creates, schedules and terminates processes; allocates CPU time (CPU scheduling) and handles multitasking and deadlocks.
2. **Memory management:** keeps track of which parts of memory are in use, allocates and de-allocates memory to programs, and uses virtual memory (paging/swapping) when RAM is insufficient.
3. **File management:** creates, deletes, copies, renames and organises files and directories; controls access permissions.
4. **Device (I/O) management:** controls peripheral devices through device drivers, using buffering and spooling (e.g. print spooler).
5. **Secondary storage management:** allocates disk space, manages free space and schedules disk access.
6. **Security and protection:** passwords, user accounts, access rights, firewalls, protection against unauthorised access.
7. **User interface:** provides a Command Line Interface (CLI) or a Graphical User Interface (GUI).
8. **Error detection and handling:** detects hardware/software errors and takes corrective action.
9. **Resource allocation and accounting:** allocates resources among users and records their usage.

10. **Networking and communication:** supports data exchange between computers.

Types of Operating Systems

Type	Description	Example
Batch OS	Similar jobs are collected into batches and run one after another with no user interaction.	Early IBM mainframe systems
Single-user, single-tasking	One user, one program at a time.	MS-DOS
Single-user, multitasking	One user runs several programs together.	Windows, macOS
Multi-user OS	Many users access the system at the same time through terminals.	UNIX, Linux
Time-sharing OS	CPU time is divided into small slices among users/jobs.	UNIX
Real-time OS (RTOS)	Guarantees response within a strict time limit.	Air-traffic control, missile systems, VxWorks
Distributed OS	Many computers appear to the user as one system.	LOCUS, cluster systems
Network OS	Provides file/printer sharing across a network.	Windows Server, Novell NetWare
Mobile OS	Designed for smartphones and tablets.	Android, iOS

Popular Operating Systems

- **MS-DOS** (1981): command-line, single-user.
- **Windows** (Microsoft): GUI-based; Windows 95, XP, 7, 10, 11.
- **UNIX / Linux**: multi-user, multitasking, secure; Linux is open source (Ubuntu, Red Hat, Fedora).
- **macOS** (Apple), **Android** (Google), **iOS** (Apple).

Basis	CLI (Command Line Interface)	GUI (Graphical User Interface)
Interaction	Typing commands	Clicking icons, windows, menus
Ease of use	Difficult; commands must be remembered	Easy and user-friendly
Resources	Very little memory/CPU needed	More memory and processing power
Example	MS-DOS, Linux shell	Windows, macOS

4.4 Introduction to DOS and Windows

4.4.1 Introduction to DOS

DOS stands for **Disk Operating System**. **MS-DOS** (Microsoft Disk Operating System) was developed by Microsoft and released in 1981 for the IBM PC. It is a **single-user, single-tasking, command-line** operating system: the user types commands at a prompt (such as **C:\>**) to run programs and manage files.

Features of DOS

- Command-driven (text-based) interface; no mouse needed.
- Small in size; runs on very low-end hardware.
- Supports files and directories (folders) in a tree structure.
- Allows batch files (*.BAT) to automate a series of commands.

Important DOS System Files

File	Role
IO.SYS (or IBMBIO.COM)	Hidden system file that contains basic input/output drivers; loaded first at boot.
MSDOS.SYS (or IBMDOS.COM)	Hidden system file: the DOS kernel that manages files and memory.
COMMAND.COM	Command interpreter (shell): reads and executes the commands typed by the user; contains the internal commands.
CONFIG.SYS	Configuration file that loads device drivers and sets system parameters at start-up.
AUTOEXEC.BAT	Batch file executed automatically at boot to run start-up commands.

Files, Directories and Wildcards

- A **file name** consists of a name (up to 8 characters) and an extension (up to 3 characters), e.g. **REPORT.TXT**. Extensions: .COM, .EXE (executable), .BAT (batch), .TXT (text), .DOC.
- A **directory** (folder) holds files and sub-directories; the top is the **root directory** (\).
- A **path** gives the location of a file, e.g. C:\COLLEGE\BSC\NOTES.TXT.
- **Wildcard characters**: ** matches any number of characters; *?* matches any single character. Example: .TXT (all text files), A?C.DOC.

DOS Boot Process

1. Power-on self-test (POST) by the BIOS checks the hardware.
2. The bootstrap loader reads the boot sector of the disk.
3. IO.SYS and MSDOS.SYS are loaded into memory.
4. CONFIG.SYS is processed, then COMMAND.COM is loaded.
5. AUTOEXEC.BAT is executed and the prompt **C:\>** appears.

Cold boot means starting the computer from the power-off state; **warm boot** means restarting a running computer (Ctrl + Alt + Del).

4.4.2 Introduction to Windows

Microsoft Windows is a family of graphical operating systems developed by Microsoft. It started in 1985 as a GUI shell on top of MS-DOS (Windows 1.0) and later became a complete operating system with **Windows 95**. Later versions: Windows 98, 2000, XP, Vista, 7, 8, 10 and 11.

Features of Windows

- **GUI**: windows, icons, menus, pointer (the WIMP interface).
- **Multitasking**: several programs can run at the same time.
- **Plug and Play**: hardware is detected and configured automatically.
- **Long file names** (up to 255 characters).
- **Networking and Internet** support; security (user accounts, firewall, Defender).
- **Object Linking and Embedding (OLE)** and **drag-and-drop** support; support for multimedia.

Basic Windows Elements

Element	Description
Desktop	The main screen displayed after logging in; holds icons and the taskbar.
Icon	Small picture representing a file, folder or program.

Element	Description
Taskbar	Bar at the bottom of the screen showing the Start button, running programs and the system tray (clock, volume).
Start menu	Gateway to programs, settings, files and shut-down.
Window	Rectangular area displaying a program; has title bar, menu bar, minimise / maximise / close buttons, scroll bars.
My Computer / This PC	Shows drives and storage devices.
Recycle Bin	Holds deleted files temporarily so they can be restored.
File Explorer	Tool to browse, copy, move and delete files and folders.
Control Panel / Settings	Used to configure hardware, software, display and users.

Useful Windows Shortcut Keys

Shortcut	Action	Shortcut	Action
Ctrl + C	Copy	Ctrl + V	Paste
Ctrl + X	Cut	Ctrl + Z	Undo
Ctrl + A	Select all	Ctrl + S	Save
Alt + Tab	Switch between programs	Alt + F4	Close the active window
Ctrl + Alt + Del	Open the security / Task Manager screen	Win + D	Show the desktop

Basis	MS-DOS	Windows
Interface	Command line (CLI)	Graphical (GUI)
Tasking	Single-tasking	Multitasking
User	Single user	Single or multi-user
Input device	Keyboard only	Keyboard and mouse / touch
Ease of use	Need to remember commands	Easy, icon based
Hardware need	Very low	Higher

4.5 DOS Internal and External Commands

DOS commands are of two kinds:

- **Internal commands** are built into **COMMAND.COM** and are loaded into memory at boot time. They can be executed at any time without any extra file. Examples: DIR, CD, COPY, DEL.
- **External commands** are separate program files (with .COM or .EXE extension) stored on the disk. The file must be available in the current directory or in the PATH. Examples: FORMAT, XCOPY, CHKDSK.

Basis	Internal commands	External commands
Location	Inside COMMAND.COM	Separate files on disk
Memory	Always in memory after boot	Loaded only when used
Execution	Faster	Slightly slower

Basis	Internal commands	External commands
Needs the disk file?	No	Yes
Examples	DIR, CLS, COPY, DEL, REN, MD, CD, RD, DATE, TIME, VER, TYPE	FORMAT, CHKDSK, XCOPY, DISKCOPY, ATTRIB, SCANDISK, DELTREE, EDIT, MORE, SORT

4.5.1 DOS Internal Commands

Command	Syntax	Purpose and example
DIR	DIR [path] [/P] [/W]	Lists files and sub-directories. /P pauses after each screen; /W gives wide format. Example: DIR *.TXT
CLS	CLS	Clears the screen.
MD / MKDIR	MD dirname	Makes (creates) a new directory. Example: MD NOTES
CD / CHDIR	CD dirname CD .. CD \	Changes the current directory; CD .. goes up one level; CD \ goes to the root.
RD / RMDIR	RD dirname	Removes an empty directory.
COPY	COPY source destination	Copies one or more files. Example: COPY A.TXT B.TXT or COPY *.TXT D:\BACKUP
COPY CON	COPY CON filename	Creates a text file from the keyboard; press F6 or Ctrl+Z, then Enter to save.
DEL / ERASE	DEL filename	Deletes files. Example: DEL *.BAK
REN / RENAME	REN oldname newname	Renames a file. Example: REN A.TXT NEW.TXT
TYPE	TYPE filename	Displays the contents of a text file.
DATE	DATE	Displays or changes the system date.
TIME	TIME	Displays or changes the system time.
VER	VER	Displays the DOS version.
VOL	VOL [drive:]	Displays the disk volume label and serial number.
PATH	PATH [dirs]	Sets the search path for executable files.
PROMPT	PROMPT \$P\$G	Changes the DOS prompt appearance.
LABEL	LABEL [drive:]	Creates or changes the volume label.
EXIT	EXIT	Quits the command interpreter.

4.5.2 DOS External Commands

Command	Syntax	Purpose and example
FORMAT	FORMAT drive: [/S] [/Q]	Prepares a disk for use by creating tracks, sectors and the file system. /S copies system files (bootable); /Q quick format. Example: FORMAT A: /S
CHKDSK	CHKDSK [drive:] [/F]	Checks a disk for errors and shows disk and memory status; /F fixes errors.

Command	Syntax	Purpose and example
XCOPY	XCOPY source dest [/S] [/E]	Copies files and entire directory trees including sub-directories.
DISKCOPY	DISKCOPY A: B:	Makes an exact copy of one floppy disk on another.
ATTRIB	ATTRIB +R +H filename	Sets or removes file attributes: R (read-only), A (archive), S (system), H (hidden).
SCANDISK	SCANDISK [drive:]	Checks and repairs disk errors.
DELTREE	DELTREE dirname	Deletes a directory with all its files and sub-directories.
MOVE	MOVE source destination	Moves files or renames a directory.
TREE	TREE [drive:]	Displays the directory structure graphically.
EDIT	EDIT filename	Starts the full-screen DOS text editor.
MORE	TYPE file MORE	Displays output one screen at a time.
SORT	SORT < file	Sorts lines of text in alphabetical order.
FDISK	FDISK	Creates and manages hard-disk partitions.
MEM	MEM	Displays the used and free memory.
SYS	SYS A:	Copies the system files to make a disk bootable.
UNDELETE	UNDELETE filename	Recovers deleted files, if not overwritten.
DOSKEY	DOSKEY	Recalls and edits previously typed commands.

Sample DOS Session

```

C:\> MD COLLEGE
C:\> CD COLLEGE
C:\COLLEGE> COPY CON NOTES.TXT
Computer Science notes (press Ctrl+Z, then Enter)
C:\COLLEGE> DIR
C:\COLLEGE> COPY NOTES.TXT NOTES.BAK
C:\COLLEGE> REN NOTES.BAK OLD.TXT
C:\COLLEGE> TYPE NOTES.TXT
C:\COLLEGE> DEL OLD.TXT
C:\COLLEGE> CD ..
C:\> FORMAT A: /Q

```

Redirection, Pipes and Batch Files

- **Redirection (>):** sends output to a file instead of the screen, e.g. DIR > LIST.TXT. >> appends to a file.
- **Pipe (|):** passes the output of one command to another, e.g. DIR | SORT.
- **Batch file:** a text file with extension .BAT containing a series of DOS commands executed together.

Remember

Internal commands live in COMMAND.COM (**DIR, COPY, DEL, REN, MD, CD, RD, CLS, TYPE, DATE, TIME, VER**). Everything that needs its own file on disk is external (**FORMAT, CHKDSK, XCOPY, ATTRIB, DISKCOPY, SCANDISK**).

QUESTION BANK AND QUICK REVISION

A. Choose the Correct Option (1 mark each)

1. The brain of the computer is: (a) Memory (b) CPU (c) Monitor (d) Keyboard. **Answer: (b)**
2. Which unit directs and coordinates all operations? (a) ALU (b) CU (c) RAM (d) ROM. **Answer: (b)**
3. Which memory is volatile? (a) ROM (b) PROM (c) RAM (d) EPROM. **Answer: (c)**
4. The symbol used for a decision in a flowchart is: (a) Rectangle (b) Diamond (c) Oval (d) Parallelogram. **Answer: (b)**
5. Which device reads cheques in banks? (a) OMR (b) OCR (c) MICR (d) Bar code reader. **Answer: (c)**
6. Which printer is an impact printer? (a) Laser (b) Inkjet (c) Dot-matrix (d) Thermal. **Answer: (c)**
7. A plotter is used to produce: (a) Sound (b) Large vector drawings (c) Scanned text (d) Video. **Answer: (b)**
8. The base of the hexadecimal system is: (a) 2 (b) 8 (c) 10 (d) 16. **Answer: (d)**
9. Standard ASCII uses how many bits? (a) 5 (b) 7 (c) 8 (d) 16. **Answer: (b)**
10. In Gray code, successive numbers differ in: (a) one bit (b) two bits (c) all bits (d) no bits. **Answer: (a)**
11. Which is system software? (a) MS Word (b) Tally (c) Operating system (d) Photoshop. **Answer: (c)**
12. Which of the following is an internal DOS command? (a) FORMAT (b) CHKDSK (c) DIR (d) XCOPY. **Answer: (c)**
13. COMMAND.COM is the: (a) kernel (b) command interpreter (c) boot file (d) driver. **Answer: (b)**
14. Decimal 10 in binary is: (a) 1000 (b) 1010 (c) 1100 (d) 1001. **Answer: (b)**
15. BCD code of decimal 9 is: (a) 1001 (b) 1010 (c) 0101 (d) 1111. **Answer: (a)**

B. Short Answer Questions (2 to 3 marks each)

1. Define computer. List any four characteristics of a computer.
2. Distinguish between data and information.
3. What are the functions of the Control Unit and the ALU?
4. What is a port? Name any four ports with their uses.
5. Differentiate between RAM and ROM.
6. What is cache memory? Why is it used?
7. Define algorithm. State its characteristics.
8. Name any six flowchart symbols with their purposes.
9. What is OMR? State two applications.
10. Write short notes on bar code reader.
11. Distinguish between impact and non-impact printers.
12. What is a voice response system?
13. Convert $(101101)_2$ to decimal and $(255)_{10}$ to hexadecimal.
14. What is Gray code? Where is it used?
15. Differentiate between system software and application software.
16. What is an operating system? Give four examples.
17. Distinguish between internal and external commands of DOS.
18. Explain the wildcard characters used in DOS.

C. Long Answer Questions (5 to 8 marks each)

1. Draw and explain the block diagram of a computer system. Describe the functions of CPU components.
2. Explain the memory hierarchy. Describe main memory and secondary memory with examples.

3. What is an algorithm? Write an algorithm and draw a flowchart to find the largest of three numbers.
4. Describe the different data scanning devices (OCR, OMR, MICR, bar code reader, image scanner).
5. Explain the working and types of printers. Compare dot-matrix, inkjet and laser printers.
6. Describe types of monitors and explain resolution, refresh rate and dot pitch.
7. Explain the number systems used in computers. Convert $(1234)_{10}$ to binary, octal and hexadecimal.
8. Explain ASCII, Gray and BCD codes with suitable examples.
9. What is an operating system? Explain its functions and types.
10. Explain DOS: system files, boot process and any ten internal and five external commands with syntax.
11. Compare MS-DOS and Windows. Explain the main features of Windows.

D. Quick Revision: Key Points and Formulae

Topic	Must remember
CPU	ALU (arithmetic and logic) + CU (control) + MU (registers). Fetch → Decode → Execute.
Memory order (fast → slow)	Registers → Cache → RAM → Secondary storage
Volatile / non-volatile	RAM, cache, registers are volatile; ROM and secondary memory are non-volatile.
Units	8 bits = 1 byte; 1 KB = 1024 bytes; 1 MB = 1024 KB; 1 GB = 1024 MB; 1 TB = 1024 GB
Flowchart	Oval (start/stop), parallelogram (I/O), rectangle (process), diamond (decision), arrow (flow)
Scanning devices	OMR = marks; OCR = characters; MICR = magnetic ink cheques; bar code = lines
Printers	Impact: dot-matrix, daisy-wheel, line. Non-impact: inkjet, laser, thermal.
Bases	Binary 2, Octal 8, Decimal 10, Hexadecimal 16. Octal = 3 bits; Hex = 4 bits.
ASCII	7-bit, 128 characters. A = 65, a = 97, 0 = 48.
Gray code	Only one bit changes between consecutive numbers; $G(\text{msb}) = B(\text{msb})$, then XOR.
BCD	4 bits per decimal digit; valid 0000 to 1001.
Software	System (OS, translators, drivers) and Application (general, special purpose).
DOS internal	DIR, CLS, COPY, DEL, REN, MD, CD, RD, TYPE, DATE, TIME, VER
DOS external	FORMAT, CHKDSK, XCOPY, DISKCOPY, ATTRIB, SCANDISK, DELTREE, EDIT

Wishing you success!

Revise the tables and examples regularly, practise number-system conversions and DOS commands, and attempt the question bank without looking at the notes.

References

1. V. Rajaraman, *Fundamentals of Computers*, PHI Learning.
2. R. Hunt and Shelly Y., *Computers and Common Sense*, Prentice Hall of India.
3. P. K. Sinha, *Computer Fundamentals* (5th Edition), BPB Publications.

4. Syllabus of B.Sc. F.Y. Computer Science (Semester I), SCSCCT1101, Swami Ramanand Teertha Marathwada University, Nanded.