Ch-1 Computer System Overview

1. Introduction to Computer System: A **computer system** is an integrated setup of hardware and software that performs data processing tasks to generate meaningful results.

Key Components:

- **Input**: Data entered by the user (e.g., keyboard).
- **Processing**: Performed by CPU.
- **Storage**: Temporarily or permanently holds data.
- **Output**: Displays the results (e.g., monitor).

2. Hardware: Hardware refers to the **physical components** of a computer system that you can touch.

Examples:

- Keyboard, Mouse
- Monitor, Printer
- CPU, RAM, Hard Disk

3. Software: Software is a set of **instructions or programs** that tell the computer what to do.

Types of Software:

Type	Description	Examples
System Software	Manages hardware and system resources	Windows, Linux
Application Software	Helps user perform specific tasks	MS Word, Excel, Photoshop

4. Input Devices: **Input devices** are used to **enter data** into a computer.

Common Input Devices:

Device	Function
Keyboard	Enter text and commands
Mouse	Pointing and selection
Scanner	Converts images to digital data
Microphone	Captures sound

5. Output Devices: **Output devices** are used to **display or present** the processed data.

Common Output Devices:

Device	Function
Monitor	Displays visual output
Printer	Produces hard copies
Speaker	Outputs audio signals
Projector	Projects visuals to a larger screen

6. Central Processing Unit (CPU): The **CPU** is the **brain of the computer**. It processes instructions and manages the flow of data.

Components of CPU:

Component	Description
ALU	(Arithmetic Logic Unit) Performs calculations
CU	(Control Unit) Manages instructions and data flow
Registers	Temporary high-speed storage inside the CPU

7. Memory: Memory stores data and instructions temporarily or permanently.

Types of Memory:

1. Primary Memory:

- Also called **main memory**.
- Directly accessible by CPU.
- Volatile (data lost when power off).

- Includes **RAM** (Read/Write) and **ROM** (Read-Only).

2. Cache Memory:

- High-speed memory between CPU and RAM.
- Stores frequently used data/instructions.
- Improves performance.

3. Secondary Memory:

- Non-volatile storage.
- Used for long-term data storage.
- Examples: HDD, SSD, USB drives, CDs.

Memory Type	Speed	Volatility	Used For
Primary (RAM/ROM)	High	RAM: Yes	Temporary data storage
Cache	Very High	Yes	Quick access by CPU
Secondary	Low	No	Permanent storage (files, OS)

8. Units of Memory: Memory is measured in units that represent **data capacity**.

Unit	Abbreviation	Size (in Bytes)
Bit	b	Smallest unit (0 or 1)
Byte	B	8 bits
Kilobyte	KB	1,024 Bytes
Megabyte	MB	1,024 KB = 1,048,576 B
Gigabyte	GB	1,024 MB = 1,073,741,824 B
Terabyte	TB	1,024 GB
Petabyte	PB	1,024 TB

Types of Software

Software is a set of programs that govern the operations of a computer system. It can be broadly classified into:

1. **System Software**
2. **Programming Tools & Language Translators**
3. **Application Software**

1. System Software: System software helps run the computer hardware and provides a platform for application software to function.

a) Operating Systems: is software that manages computer hardware and software resources.

Functions of OS:

- Process management
- Memory management
- File system management
- Device management
- User interface

Operating System	Type	Examples
Single-user	Desktop use	Windows 10, macOS
Multi-user	Servers	UNIX, Linux
Real-time	Embedded	RTOS, VxWorks
Mobile OS	Smartphones	Android, iOS

b) System Utilities: These are small programs that help in **system maintenance and optimization**.

Utility Software	Function
Antivirus	Detects and removes viruses
Disk Cleanup	Frees up disk space
File Management Tools	Organize and manage files
Backup Software	Creates copies of data for recovery

Defragmentation Tool	Rearranges fragmented data on disk

c) Device Drivers: Device Drivers are programs that enable communication between the operating system and hardware devices.

Device	**Driver Function**
Printer	Converts data into printer-understandable form
Graphics Card	Enables display rendering
Sound Card	Manages audio input/output
USB Devices	Manages data transfer via USB ports

2. Programming Tools & Language Translators: These are tools used to write, translate, and debug programs written in programming languages.

a) Assembler

- Converts **assembly language** into **machine code**.
- Specific to a particular CPU architecture.

Feature	**Description**
Input	Assembly Language
Output	Machine Language (Binary)
Example	NASM (Netwide Assembler), MASM

b) Compiler: Translates the **entire program** from a **high-level language** to **machine code** in one go.

- Fast execution after compilation.

Feature	**Description**
Input	High-Level Language (C, C++, Java)
Output	Executable Machine Code
Example	GCC, Turbo C++, javac

c) Interpreter: Translates **and executes code line by line**, slower than compilers but easier for debugging.

Feature	**Description**
Input	High-Level Language (Python, JavaScript)
Output	Immediate Execution
Example	Python Interpreter, Node.js

3. Application Software: Application software is designed for **end-users** to perform specific tasks.

Type	**Description**	**Examples**
Word Processing	Creates text documents	MS Word, Google Docs
Spreadsheet	Organizes data in rows/columns	MS Excel, Google Sheets
Presentation	Creates slide shows	MS PowerPoint, Google Slides
Web Browsers	Accesses websites	Chrome, Firefox, Edge
Media Players	Plays audio/video	VLC, Windows Media Player
Database Management	Stores and manages data	MS Access, MySQL
Graphics Software	Edits images and graphics	Photoshop, GIMP

Summary Table

Category	**Subtypes**	**Purpose**
System Software	OS, Utilities, Drivers	Controls and manages system hardware
Programming Tools	Assembler, Compiler, Interpreter	Translates and develops software
Application Software	Word processor, Browser, Media player	Performs specific user-related tasks

Operating System (OS): is system software that acts as an **intermediary between the user and the computer hardware**. It manages hardware resources and provides services for computer programs.

1. Functions of the Operating System

The OS performs several key tasks to ensure the computer system runs efficiently and user interactions are seamless.

a) Process Management

- Handles creation, scheduling, and termination of processes.
- Allocates CPU time to different processes.

Feature	Description
Multitasking	Runs multiple applications simultaneously
Scheduling	Decides which process runs next

b) Memory Management

- Manages the system's primary memory (RAM).
- Allocates and deallocates memory spaces as needed.

Task	Role of OS
Allocation	Assigns memory to processes
Deallocation	Frees memory after use
Virtual Memory	Uses secondary storage as additional RAM

c) File Management

- Organizes, stores, retrieves, and secures data files.

File Operation	OS Role
Create/Delete	Manages file life cycle
Read/Write	Provides access to file contents
Access Control	Ensures only authorized access

d) Device Management

- Manages communication between hardware and software via device drivers.

Device Type	OS Task
Input/Output	Coordinates data flow to/from devices
Drivers	Interface software for hardware

e) Security and Access Control

- Protects data and system from unauthorized access or threats.

Feature	Description
User Authentication	Verifies identity of users
Permissions	Restricts access to files and settings

f) User Interface Management

- Provides ways for users to interact with the computer.
- Covered in detail in the next section.

2. OS User Interface

The **User Interface (UI)** is the point where users interact with the OS. It can be:

a) Command-Line Interface (CLI)

- Text-based interface.
- Users type commands using a keyboard.
- Requires knowledge of command syntax.

Feature	Description
Input Method	Typed commands
Efficiency	Fast for experienced users
Example OS	MS-DOS, Linux Shell (Bash)

b) Graphical User Interface (GUI)

- Visual interface using windows, icons, menus, and pointers.
- Easier for beginners and general users.

Feature	Description
Input Method	Mouse, keyboard (point-and-click)
User-Friendly	Easy to navigate
Example OS	Windows, macOS, Ubuntu

c) Touch-Based Interface

- Designed for touch input using fingers or stylus.
- Common in smartphones and tablets.

Feature	Description
Input Method	Touch gestures (tap, swipe, pinch)
Devices Used	Smartphones, tablets
Example OS	Android, iOS

Summary Table

Function Area	Description
Process Management	Manages running applications
Memory Management	Allocates and frees memory
File Management	Handles file storage and retrieval
Device Management	Controls peripheral devices
Security Management	Ensures authorized system access
User Interface	Allows user to interact with the system