

GOVERNMENT POLYTECHNIC, DEBAGADA

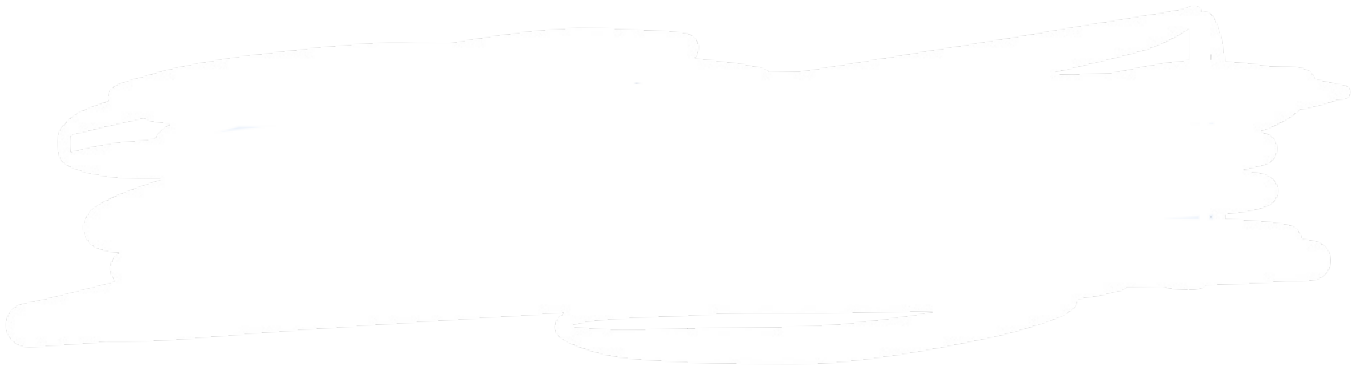
AT/PO- PURUNAGARH, DIST-DEOGARH, ODISHA, 768119

www.gpdeogarh.org.in Mail- principaldeogarh@gmail.com

DEPARTMENT OF MATH AND SCIENCE

COMPUTER & AI

Lecture Notes 2026-27



Unit- 2 - Computer Software

Prepared by

Abhilipsa Mishra

Guest Faculty of Computer Application

GOVERNMENT POLYTECHNIC, DEBAGADA

1. Computer Software

Definition & Overview

- Computer software comprises a set of instructions that tell a computer to perform a specific task to serve user needs.
- Software is not a physical object like hardware. Instead, it enables the computer hardware to function according to specific user requirements.

Classification of Software

I. System Software

- System software provides the basic functionality required to operate a computer by interacting directly with its hardware.
- It controls internal computer operations and manages peripherals such as monitors, printers, and storage devices.
- Acts as an essential interface between hardware components and user application software to facilitate communication.

Types of System Software:

1. Operating System (OS):

- The primary program of a computer system.
- When the computer turns on, the operating system is the first software loaded into memory.
- Manages all hardware resources including computer memory, CPU, hard disk, and printers.
- *Popular Examples:* Windows, Linux, Macintosh, Ubuntu, Fedora, Android, iOS.

2. Device Drivers:

- Acts as an interface between external devices and the operating system.
- Controls specific hardware devices and assists them in performing their functions.

3. Programming Language Processor:

- Converts human-readable programming language into machine code (binary) and vice-versa.

- Translates programs written in high-level programming languages (e.g., Java, C, C++, Python) into machine instructions.

II. Application Software

- Software designed to perform special functions or offer features beyond basic computer operations.
- Developed as a product or program specifically to fulfill end-user requirements.
- *Examples:* Spreadsheets, word processors.

Types of Application Software:

1. General Purpose Software:

- Used for a wide variety of tasks according to general end-user requirements.
- *Example:* Spreadsheet tools can be used by any user to perform calculations or create accounting ledgers.

2. Customised Software:

- Tailored specifically to perform specific functions or tasks based on unique individual or organizational needs.
- *Examples:* Websites, school management software, accounting software.

III. Utility Software

- Software dedicated to maintaining, configuring, and optimizing a computer system.
- Some utilities come bundled with the operating system, while others are acquired separately.
 - *Bundled Utilities:* Disk defragmentation tools, disk formatting utilities.
 - *Third-Party Utilities:* Antivirus software, disk cleaners, disk compression software.

Types of Utility Software:

- **Antivirus Software:** Designed to detect and remove viruses, malware, and other malicious programs.
- **Disk Cleaners:** Tools that scan the hard drive for unnecessary files and safely delete them to free up storage space.

- **Backup and Recovery Software:** Allows users to create duplicate copies of data and restore them in the event of system failure or data loss.
- **System Optimizers:** Tools that enhance computer performance by tuning system settings, managing system resources, and removing unwanted files/programs.
- **Disk Defragmenters:** Reorganizes file blocks on a hard drive to improve file access speed and overall performance.
- **File Compression Software:** Compresses files and folders to reduce storage footprint and simplify internet file transfers.
- **Disk Encryption Software:** Encrypts drive data to protect sensitive information from unauthorized access.

2. Operating Systems (OS)

Overview

- An Operating System manages all resources of a computer, including hardware (CPU, RAM, hard disk, network, I/O devices) and software (applications, device drivers, access control, and security).
- *Popular OS Examples:* Windows, Linux, Android, Macintosh.
- **Execution Flow:**
 1. When an application needs to run, the OS loads the program into memory and allocates it to the CPU for execution.
 2. If multiple application programs need to run simultaneously, the OS decides their execution order.
 3. Provides a user interface to receive user inputs/commands and process them.

Core Functions of an Operating System

1. Process Management

A process is a program in execution. Its lifecycle spans from execution start to completion.

- **Scheduling:** Decides which process gets CPU time next.
- **Synchronization:** Ensures orderly execution using locks or monitors to prevent race conditions.

- **Deadlock Handling:** Detects and prevents situations where multiple processes wait infinitely for resources held by each other.
- **Inter-Process Communication (IPC):** Facilitates data exchange between processes using shared memory or message passing.

2. Memory Management

Controls how data is organized and stored across main memory (RAM) and secondary storage.

- **Allocation & Deallocation:** Assigns memory when needed and frees it up when no longer in use.
- **Protection:** Prevents one process from interfering with the memory allocated to another process.
- **Virtual Memory:** Utilizes secondary disk space as extended memory to run large programs.
- **Fragmentation Handling:** Minimizes wasted memory space through compaction techniques.
- **Disk Management:** Coordinates file storage, free space tracking, disk scheduling, and backups.

3. File System Management

Organizes and controls how data is stored, named, and accessed on storage devices.

- **File Attributes:** Manages characteristics such as file name, file type, file size, and access permissions.
- **File Types:** Identifies categories like text, binary, and executable files based on content/format.
- **Operations:** Executes creation, reading, writing, and deletion of files.
- **Access Methods:** Supports sequential access, direct access, and indexed access.

4. Device Management (I/O System)

Manages communication between the operating system and hardware peripherals.

- **Device Drivers:** Serves as software interfaces between hardware components and the OS.

- **Buffering & Caching:** Holds temporary data to handle speed differences between devices and optimize system performance.
- **Spooling:** Manages queues of data waiting to be processed (e.g., placing print jobs in a spool so the CPU can execute other tasks while the printer finishes printing).

5. Protection and Security

Safeguards system resources against unauthorized access.

- **Access Control:** Limits user and process permissions.
- **Authentication:** Validates user identity via login credentials.
- **Resource Protection:** Prevents misuse of files, system memory, or devices.
- **Security:** Guards against malware, Denial-of-Service (DoS) attacks, and data theft.

Types of Operating Systems & Interfaces

A. Graphical User Interface (GUI)

Allows users to give instructions using visual icons, menus, pointers, and windows rather than typing commands. Interaction typically uses a mouse and keyboard.

- *Examples:* Microsoft Windows, Ubuntu, Fedora, Macintosh.

GUI Component	Function / Description
Icons	Small image representations of files, folders, or actions.
Pointers	Visual cursor symbols used to select elements on-screen.
Pointing Tools	Hardware tools (mouse, trackpad) used to control cursor movements.
Menus	Lists of selectable choices for system navigation.
Buttons	Clickable user interface elements that trigger execution.
Scrollbars	Elements used to navigate content extending beyond the screen view.
Window	Container frames holding application or document interfaces.

GUI Component	Function / Description
Tab	Divides content or functionality within a single active window.
Text Fields	Entry boxes where users enter text inputs or search terms.
Toolbars	Strip rows containing shortcuts to common commands.

B. Character User Interface (CUI / CLI)

Also known as Command Line Interface (CLI), users interact with system programs via text-based commands. Users must memorize command names and syntaxes. It is less interactive and generally handles a single execution at a time.

Components of CUI:

- **Shell:** The intermediate layer between the user and OS that interprets entered commands (e.g., Bash, PowerShell, Command Prompt).
- **Commands:** Explicit instructions typed into the shell to perform tasks.
- **Syntax:** Structural rules determining how commands and parameters must be written.
- **Output:** Results returned after command execution, indicating success or an error.

C. Multi-programmed Operating System

Allows execution of multiple jobs simultaneously on a single processor to maximize CPU utilization.

How it Works:

1. Incoming jobs enter the system and reside in a **job pool**.
 2. The OS loads a subset of these jobs into main memory and begins execution.
 3. If an active job pauses to wait for an I/O operation, the OS switches execution immediately to another job.
 4. When the paused job completes its I/O wait, it gets back in line for CPU time.
 5. As long as at least one job requires execution, the CPU is never idle.
- *Examples:* UNIX, Windows NT, Linux, macOS.
 - **Advantages:**

- Increases overall resource utilization.
- Enhances system responsiveness.
- Improves overall reliability and system stability.
- **Disadvantages:**
 - Long-term jobs may experience extended wait times if job queues are large.
 - Requires complex memory management schemes.
 - Excessive multiprogramming can lead to system overheating issues.

D. Time-Sharing Operating System

A logical extension of multiprogramming that allows multiple users to share a computer system simultaneously. The CPU switches between tasks so rapidly that users can interact with each program while it runs.

- **Time Quantum:** Executions are allocated short time intervals called time slices or quanta (typically 10 to 100 milliseconds).
- Requires sophisticated memory management and security capabilities.

Process States in Time-Sharing:

1. **Active State:** The user program is currently under active CPU execution (only one program at a time).
2. **Ready State:** The program is ready to run and waiting for its turn to be allocated CPU time.
3. **Waiting State:** The program is paused awaiting an I/O operation completion.

- **Advantages:**
 - Equal CPU execution opportunities across tasks.
 - Minimizes software duplication.
 - Reduces overall CPU idle time.
- **Disadvantages:**
 - Requires high reliability.

- High security and data integrity management required for user programs.
- Potential data communication bottlenecks.

E. Real-Time Operating System (RTOS)

Specifically engineered for real-time applications that require precise task execution within rigid time limits. Used in spacecraft, robotics, industrial processes, and scientific equipment.

Types of RTOS:

- **Hard Real-Time OS:** Completing tasks within strict deadlines is mandatory; failing a deadline leads to catastrophic failure.
- **Soft Real-Time OS:** Deadlines are important, but occasional execution delays beyond allocated times can be tolerated.
- **Advantages:**
 - Well-suited for designing and deploying compact real-time applications.
 - Highly compact; requires low memory footprints.
- **Disadvantages:**
 - Executes very few tasks concurrently to avoid timing errors.
 - Low-priority tasks can suffer from starvation.
 - Requires specialized device drivers and interrupt handling configurations.

3. Computer Viruses & Antivirus Software

Computer Viruses

- A computer virus is a malicious software program designed to replicate itself by modifying other software and inserting its code.
- Affected software components are referred to as "infected".
- **Spread Vectors:** Network connections, infected disks, file-sharing platforms, and malicious email attachments.
- **Impact:** Corrupts data, consumes system resources, and damages software.

Common Types of Viruses

1. **Boot Sector Virus:**

- Infects the Master Boot Record (MBR) or boot sector of storage drives.
- Loads into memory before the operating system boots up, making detection and removal difficult.
- *Examples:* Polyboot B, AntiEXE.

2. **Macro Virus:**

- Targets files containing embedded macros (e.g., .doc, .xls, .pps, .mdb).
- Hides inside documents shared over networks or via email.
- *Examples:* Relax, Melissa.A, Bablas.

3. **Memory Resident Virus:**

- Resides directly in system RAM after initial execution.
- Remains in memory to infect any files opened, closed, copied, or renamed.
- *Examples:* Randex, CMJ, Meve, MrKlunky.

4. **Polymorphic Virus:**

- Encrypts or alters its signature every time it infects a system.
- Evades traditional signature-based detection algorithms.
- *Examples:* Elkern, Marburg, Satan Bug, Tuareg.

5. **File Infector Virus:**

- Attaches malicious code directly to executable binaries (.exe, .com, .bin, .sys, .dll).
- Remains dormant until the host executable is run.

Antivirus Software

Software designed to prevent, scan, detect, and remove malicious threats including viruses, Trojans, worms, rootkits, keyloggers, ransomware, and spyware.

Scanning Modes:

- **On-Demand Scan:** Manually triggered by the user or run on a set schedule.
- **Real-time Protection:** Background monitor that scans active memory continuously during operation.
- **Smart Scan:** Selectively targets high-risk system areas to balance protection with performance.

Detection and Prevention Guidelines

1. **Email Caution:** Never open email attachments received from unknown, unverified, or suspicious senders.
2. **Sender Verification:** Confirm the legitimacy of unexpected attachments even if sent by known contacts.
3. **Spam Handling:** Immediately delete unsolicited spam and chain emails without opening or forwarding them.
4. **Download Integrity:** Download files only from trusted sites and verify they are scanned by antivirus software.
5. **Regular Updates:** Regularly update antivirus definition libraries and apply OS security patches.
6. **Data Backups:** Keep offline secondary backups of critical data in a separate location.
7. **Avoid High-Risk Websites:** Stay away from torrenting, unauthorized file-sharing, online gaming, or gambling sites loaded with malware risks.
8. **Handling Pop-ups:** Do not click pop-ups warning about fake infections (e.g., *Win 7 Security 2011*, *XP Antivirus*). Use **Alt + F4** to close pop-ups safely, or restart the system immediately.