



Computer Organization & Architecture

Cybersecurity Department

Course Code: CBS219

Lecture 8: Operating System

Halal Abdulrahman Ahmed

Lecture Outline

- Introduction to Operating Systems
- OS Structure and Components
- OS Goals and Functions
- Resource and I/O Management
- Types and Applications of Operating Systems

Lecture Outcomes

By the end of this lecture, students will be able to:

- Define an operating system and explain its purpose.
- Describe how the OS acts as an interface between users, applications, and hardware.
- Identify the main goals of an operating system.
- Distinguish between the kernel and the shell.
- Explain the major functions of an operating system.
- Identify common types of operating systems.
- Explain how the OS manages hardware resources.
- Describe the role of the OS in I/O and memory management.

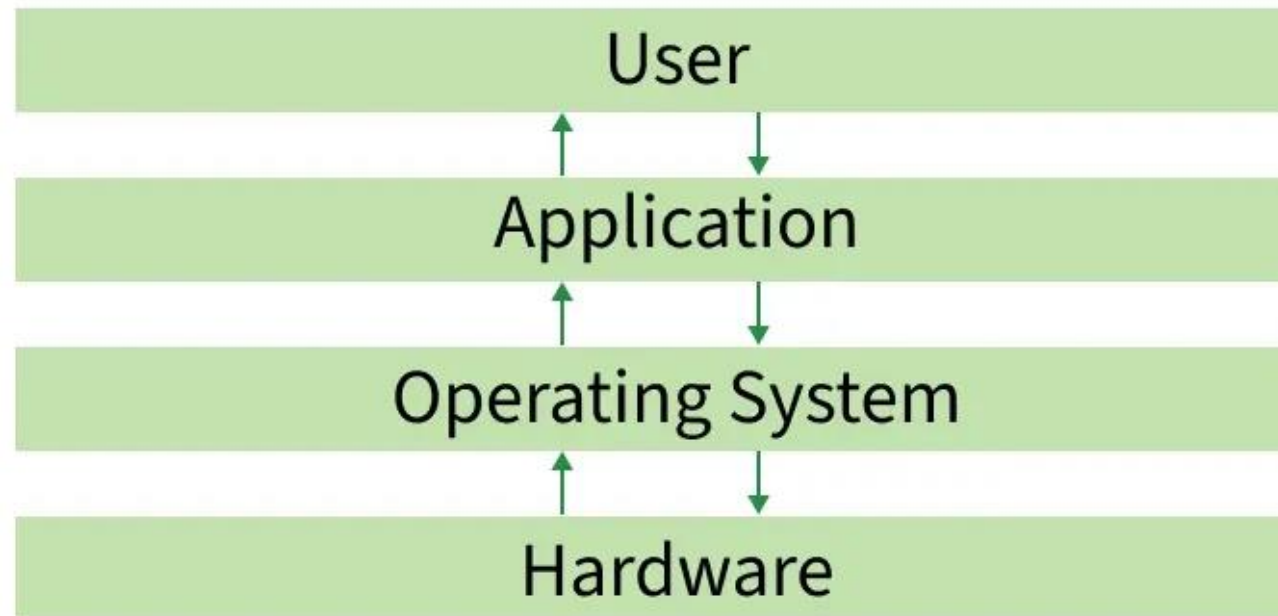
Introduction to Operating System

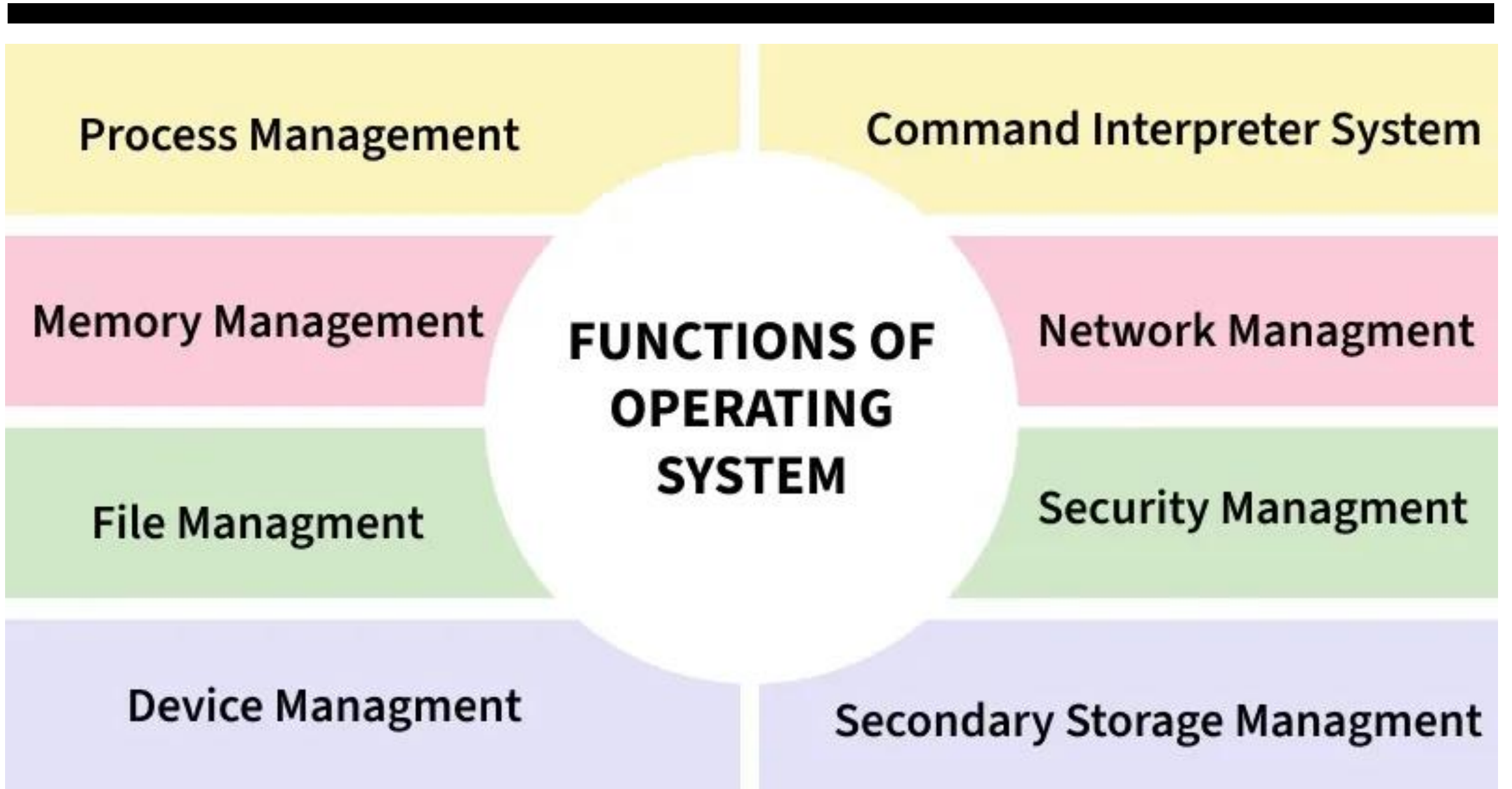
- An operating system acts as an intermediary between the computer hardware and the user. In short, it is an interface between computer hardware and the user.
- The purpose of an operating system is to provide an environment in which a user can execute programs conveniently and efficiently.
- The operating system (OS) is a program that runs at all times on a computer. All other programs, including application programs, run on top of the operating system.
- It assigns resources such as memory, processors, and input/output devices to processes that need them. The assignment of resources has to be fair and secure.

Introduction to Operating System

- The **Operating System (OS)** is a large **software program** that acts as the **middle layer** between **you, your applications**, and the **hardware** of the computer.

Control flow





Operating System and User Interaction

- Every general-purpose computer consists of hardware, an operating system, system programs and application programs.
- The hardware includes CPU, ALU, memory, I/O devices, peripheral devices and storage devices.
- The system programs include compilers, loaders, editors and the OS itself.
- The application programs are user-level software that perform specific tasks.

Users (User 1, User 2, User 3, User n)

- Each user runs different programs:
- User 1 → Text editor
- User 2 → Compiler
- User 3 → Database System
- User n → Any application
- **Users do not talk directly to hardware.**

Users talk to *applications*, and applications talk to the *Operating System*.

Application Programs

This layer includes all the **software programs** you use in your computer:

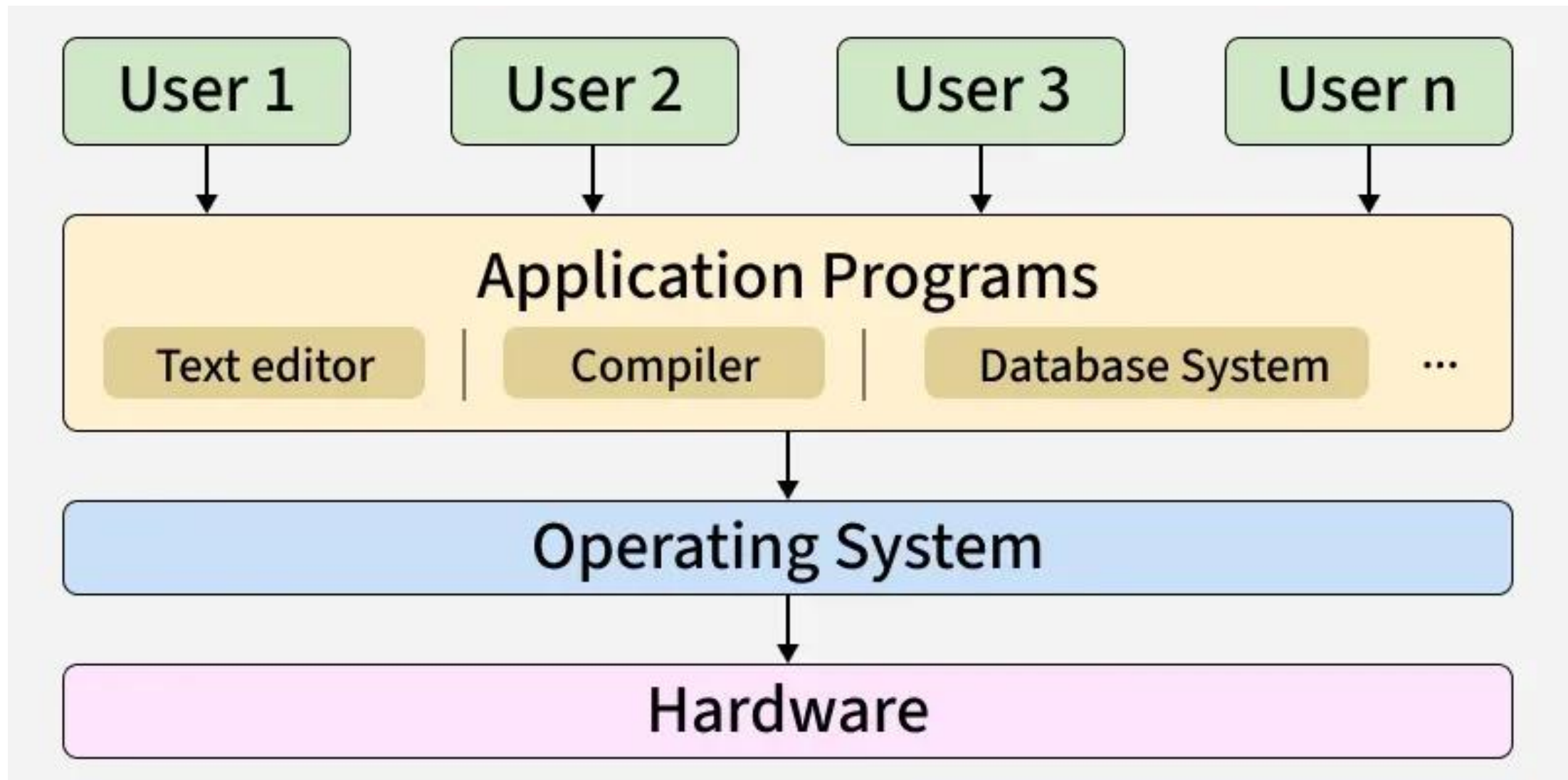
- Text editors (Notepad, Word)
- Compilers (Java compiler, C compiler)
- Database systems (MySQL, Oracle)
- Web browsers (Chrome, Firefox)
- Any application...
- These programs **cannot access hardware directly**.
- Instead, they send a request to the **Operating System**.

Operating System (OS)

- The **middle layer**. The OS acts as a **manager** or **bridge** between:
- **Users ↔ Applications ↔ Hardware**
- The OS controls and manages:
 - CPU scheduling
 - Memory allocation
 - Disk access
 - Input/Output (keyboard, mouse, printer)
 - Security
 - Files and directories
 - Communication between programs
- **Applications cannot use hardware without the OS.** The OS gives them controlled, safe, organized access.

Hardware

- This is the physical part of the computer:
- CPU
- RAM
- Hard disk / SSD
- Keyboard / mouse
- Screen
- Network card
- Printer



An operating system is always running in the background to manage and coordinate the use of hardware resources among system and application programs for multiple users. The OS itself is not only a user interface. Instead, it provides an interface for user interaction, typically through:

- Command-Line Interface (CLI), (e.g., Bash, PowerShell)
- Graphical User Interface (GUI), (e.g., Windows desktop, macOS Finder)

Goals of Operating System

Primary Goals

- The primary goals of an operating system (OS) are to provide an easy to use and convenient environment for executing user programs.
- **User Convenience:** It should be easy to use, providing a user-friendly interface and making it simple to interact with the system.
- **Program Execution:** It facilitates the execution of user programs, providing the necessary environment and services for them to run.
- **Resource Management:** The OS manages and allocates the computer's resources, including the CPU, memory, disk storage and input/output devices, to ensure fair utilization.
- **Security:** The OS protects the system and user data from unauthorized access, ensuring confidentiality, integrity and availability of information.

Secondary Goals

- **Efficient Resource Utilization:** It should aim to maximize the performance and utilization of computer resources like CPU, Memory and I/O devices, ensuring that the system runs smoothly and efficiently.
- **Reliability:** It should be robust and reliable, able to handle errors and exceptions gracefully, ensuring that the system continues to operate smoothly. It should be modular in design and easy to debug.

Components of an Operating System

- There are two basic components of an Operating System.
- [Shell](#) is the outermost layer of the Operating System and handles user interaction. It interprets input for the OS and handles the output from the OS.
- [Kernel](#) is the core component of the operating system. The kernel is the primary interface between the Operating system and Hardware.

List of Common Operating Systems

There are multiple types of operating systems each having its own unique features:

- **Windows OS** : Developed by Microsoft. Used in personal computing, business environment, gaming.
- **macOS** : Developed by Apple. Used for creative industries (design, video editing, music production), Personal computing and Professional environments.
- **Linux** : Developed by Linus Torvalds and maintained by the open-source community (Linux Foundation, Red Hat, Canonical, etc.). Used in servers and data centers, development and programming, Personal computing for tech enthusiasts.
- **Unix** : Developed originally AT&T Bell Labs, various commercial and open-source versions available. Used in servers and workstations, Development environments, Research and academic settings



Windows



macOS



Linux



ChromeOS

Examples of OS



Android



iOS

UNIX[®]
An Open Group Standard

UNIX

BSD

BSD

Applications of Operating System

- **Platform for Application Programs:** It provides a platform, on top of which, other programs, called application programs can run.
- **Managing Input-Output Unit:** It also allows the computer to manage its own resources such as memory, monitor, keyboard, printer, etc. Management of these resources is required for effective and fair utilization.
- **Multitasking:** It manages memory and allows multiple programs to run in their own space and even communicate with each other through shared memory.
- **Manages Memory and Files:** It manages the computer's main memory and secondary storage. Additionally, it allows and deallocates memory to all tasks and applications.
- **Provides Security:** It helps to maintain the system and applications safe through the authorization process. Thus, the OS provides security to the system.

Why to Choose the Operating System?

There are so many factors to be considered while choosing the best Operating System for our use. These factors are mentioned below.

- **Price Factor:** Price is one factor to consider when choosing an operating system, as some OSs are free, like Linux, but there is some more OS that is paid like Windows and macOS.
- **Accessibility Factor:** Some Operating Systems are easy to use like macOS and iOS, but some OS are a little bit complex to understand like Linux. So, you must choose the Operating System in which you are more accessible.
- **Compatibility factor:** Some Operating Systems support few applications whereas some Operating Systems supports more application. You must choose the OS, which supports the applications which are required by you.
- **Security Factor:** The security Factor is also a factor in choosing the correct OS, as macOS provides some additional security while Windows has little fewer security features.

References

- Fox, C. (2022). Computer architecture: From the Stone Age to the quantum age. No Starch Press.
- Stallings, W. (2022). Computer Organization and Architecture: Designing for Performance (11th ed.). Pearson.
- Take U Forward. (n.d.). *Page fault*. Take U Forward. Retrieved November 27, 2025, from <https://takeuforward.org/operating-system/page-fault>

Any

Question

