1.1 ROLE OF OS IN MODERN COMPUTING: SERVERS, MOBILE, EMBEDDED, CLOUD

Operating System is a program that acts as an interface between a user of a computer and the computer hardware. Eg. Windows (Microsoft), iOS (Apple), Android (Google), Linux/Ubuntu (open source)

Goals of an Operating System (Simplified)

- To run user programs and help users solve problems easily.
- To make the computer easy and comfortable to use.
- To use the computer hardware efficiently and effectively.

1. Servers

Operating systems in servers manage high-volume requests, multi-user access, security, and resource allocation.

Role includes:

- Handling concurrent client connections
  - Server OS manages hundreds or thousands of simultaneous requests.
  - Example: A web server serving webpages to many users at once.
- Ensuring system stability and uptime
- Managing storage, networks, and memory efficiently
  - Allocates CPU, RAM, and storage efficiently among background services like databases, mail servers, file servers, etc.
  - Ensures that no single process consumes entire system resources.
- Supporting virtualization and container environments (e.g., Docker, Kubernetes)
  - Allows running multiple virtual machines on one physical server using KVM, Hyper-V, or VMware.
  - Improves resource utilization and scalability.

Examples: Linux (Ubuntu Server, CentOS), Windows Server.

2. Mobile Devices

Mobile OS is designed to support energy efficiency, touch interfaces, mobile applications, and wireless connectivity.

Role includes:

- Managing app execution and background tasks
  - Supports installation, update, and uninstallation of apps.
  - Runs apps inside sandboxes to prevent them from accessing other apps’ data.
  - Manages app permissions (camera, location, microphone, storage).
- Ensuring battery optimization
  - Most important function in mobiles.
  - OS controls:
    - CPU frequency scaling
    - Background app restrictions
    - Battery saving modes
    - Charging optimization
  - Ensures longer battery life.
- Handling cellular, Wi-Fi, and Bluetooth communication
- Enforcing security through sandboxing

Examples: Android, iOS.

3. Embedded Systems

Embedded OS runs on hardware with limited resources (microcontrollers, appliances, automotive systems).

Role includes:

- Real-time task scheduling and control
  - Key requirement for many embedded systems.
  - OS provides deterministic (predictable) response time.
  - Used in robotics, medical devices, automotive systems, industrial machines.
- Efficient memory use with low overhead
- Device-specific hardware control

- High reliability for continuous operation
    - Embedded systems must run continuously without failure.
    - OS avoids crashes and ensures safe operation.

Examples: FreeRTOS, VxWorks, Embedded Linux.

4. Cloud Computing

In cloud systems, OS plays a major role in virtualization, resource pooling, scalability, and service deployments.

Role includes:

- Managing virtual machines and containers
    - Cloud OS enables creation of:
        - Virtual Machines (VMs)
        - Containers (Docker)
        - Microservices
    - Allows sharing hardware among many users.
- Allocating compute, storage, and network resources dynamically
- Supporting distributed applications and remote execution
- Ensuring security, multi-tenancy, and isolation
    - Different organizations share common cloud hardware but stay isolated.
    - OS enforces strong isolation and data privacy.

Examples: Linux (most common), Windows Azure OS, OpenStack environments.