Summary of Chapter 1: Operating System Concepts

1 Introduction

This chapter provides an overview of operating systems (OS), explaining what they do and their importance in managing hardware and software resources.

2 Objectives

The key goals of this chapter are:

- To describe the organization of computer systems.
- Provide a high-level tour of major OS components.
- Explore various computing environments.
- Introduce open-source OS.

3 What is an Operating System?

An OS is an intermediary between users and computer hardware. The main goals are:

- To execute user programs.
- Make the system convenient for users.
- Use hardware efficiently.

4 Computer System Structure

A computer system has four main components:

- **Hardware**: CPU, memory, I/O devices.
- **Operating System**: Manages hardware.
- **Application Programs**: Define how resources are used.
- **Users**: Individuals or machines.

5 What Operating Systems Do

The OS acts as a resource allocator and control program, optimizing hardware use and managing user program execution.

6 Operating System Definition

The OS has no universal definition, but it is generally described as:

- **Kernel**: A program that runs at all times, managing hardware.
- **System programs**: Essential programs that accompany the OS.
- **Application programs**: Installed by users for tasks like word processing or web browsing.

7 Computer Startup

The *bootstrap program* runs at startup, initializing the system and loading the OS kernel.

8 Computer-System Organization

Multiple CPUs and device controllers share memory, enabling concurrent execution and efficient resource management.

9 Interrupts and Handling

Interrupts transfer control to the OS to handle device or software signals. There are mechanisms like *polling* and *vectored interrupts* to manage various types of interruptions.

10 I/O Structure and Storage

- **I/O Structure**: Devices operate concurrently, managed through interrupts. Data is moved via device controllers and the OS handles I/O through system calls.
- **Storage Structure**: Main memory is volatile, while secondary storage (e.g., hard disks) provides persistent data storage.

11 Storage Hierarchy

The storage hierarchy is defined by speed, cost, and volatility, with caching being a key principle to optimize access times.

12 Direct Memory Access (DMA)

Direct memory access (DMA) allows devices to transfer data to memory without CPU intervention, improving efficiency.

13 Computer-System Architecture

Most systems use multiprocessors, where multiple CPUs increase throughput, reliability, and fault tolerance. Symmetric multiprocessing (SMP) allows processors to share tasks equally.

14 Types of Operating Systems

Types include:

- **Batch systems**
- **Multi-programmed OS**
- **Real-time OS**
- **Embedded OS**

Distributed and clustered systems enhance performance and availability through cooperation between multiple machines.

15 OS Operations

OS operations are driven by interrupts. Modern OSes provide **dual-mode operations** (user and kernel mode) to protect the system from misuse.

16 Process and Memory Management

- **Process Management**: OS handles process creation, synchronization, and communication.
- **Memory Management**: OS keeps track of memory use and manages allocation and deallocation.

17 Storage Management

- **File Systems**: OS provides a logical view of data through files, which are organized into directories and controlled via access permissions.

18 I/O Subsystem

The OS abstracts hardware peculiarities, managing buffering, caching, and spooling to ensure efficient I/O handling.

19 Protection and Security

The OS ensures **protection** (access control) and **security** (defense against attacks). User IDs and group IDs manage access rights.