Applied Operating System
Study Guide

Module 1
INTRODUCTION TO OPERATING SYSTEM

SUBTOPIC 1: Operating System Overview

What is an Operating System?
An Operating System (OS) is a program or system software that acts as an interface between the user and the computer hardware and controls the execution of all kinds of programs.

Goals of an Operating System?
- Execute user programs and make solving user problems easier.
- Make the computer system convenient to use.
- Use the computer hardware in an efficient manner.

Components of a Computer System
- **Computer hardware** – CPU, memory and I/O devices, provides the basic computing resources.
- **Application programs** – are used to solve the computing problems of the users such as word processors, games and business programs.
- **Users** – who utilize a computer or network service trying to solve different problems.
- **Operating System** – controls and coordinates the use of the hardware among the various application programs for the various users.

What is a Kernel?
Kernel is the central part of an OS which manages system resources and is always resident in memory. It also acts like a bridge between application and hardware of the computer. It is also the first program that loads after the bootloader.

Bootloader is a program that loads and starts the boot time tasks and processes of an OS. It also places the OS of a computer into memory.

Common Services Offered By Almost All Operating Systems:
- User Interface
- Program Execution
- File system manipulation
- Input / Output Operations
- Communication
- Resource Allocation
- Error Detection
- Accounting
- Security and protection

1. **User Interface (UI)** refers to the part of an OS, or device that allows a user to enter and receive information.
   Types of UI:
   - Command line interface
   - Batch based interface
   - Graphical User Interface

2. **Program Execution.** The OS must have the capability to load a program into memory and execute that program.
3. **File system manipulation.** Programs need has to be read and then write them as files and directories. File handling portion of OS also allows users to create and delete files by specific name along with extension, search for a given file and / or list file information.
4. **Input / Output Operations.** A program which is currently executing may require I/O, which may involve file or other I/O device. The OS is responsible for reading and/or writing data from I/O devices such as disks, tapes, printers, keyboards, etc.
5. **Communication.** Process needs to swap over information with other process. Processes executing on same computer system or on different computer systems can communicate using operating system support.
6. **Resource Allocation** The OS manages the different computer resources such as CPU time, memory space, file storage space, I/O devices, etc. and allocates them to different application programs and users.
7. **Error Detection** The operating system should be able to detect errors within the computer system (CPU, memory, I/O, or user program) and take the appropriate action.
8. **Job Accounting.** OS keeps track of time and resources used by various tasks and users, this information can be used to track resource usage for a particular user or group of user.
9. **Security and Protection.** *Protection* is any mechanism for controlling access of processes or users to resources defined by the OS. *Security* is a defense of the system against internal and external attacks (denial-of-service, worms, viruses, identity theft, theft of service)

Types of Operating System

- Batch operating system
- Time-sharing operating systems
- Distributed operating system
- Network operating system
- Real-time operating system
- Handheld operating system

Batch Operating System (BOS)

The user of a BOS never directly interacts with the computer.
Every user prepares his or her job on an offline device like a punch card and submit it to the computer operator.

- **Examples:** Payroll System, Bank Statements etc.

Time-sharing Operating System
Time-sharing or **multitasking** is logical extension in which CPU switches jobs so frequently that users can interact with each job while it is running, creating **interactive computing.**

- **Examples:** Unix, Linux, Multics and Windows

Distributed Operating System
Distributed systems use multiple central processors to serve multiple real-time applications and multiple users. Data processing jobs are distributed among the processors accordingly.

- **Examples:** Telecom Network, WWW, Cloud Computing, etc.

Network Operating System (NOS)
A NOS runs on a server and provides the server the capability to manage data, users, groups, security, applications, and other networking functions.

- **Examples:** Microsoft Windows Server 2003, Microsoft Windows Server 2008, UNIX, Linux, Mac OS X Server, Novell NetWare, and BSD/OS (Berkeley Software Design)

Real-time Operating System (RTOS)
RTOS is an operating system intended to serve real-time systems/applications that process data as it comes in, mostly without buffer delay.

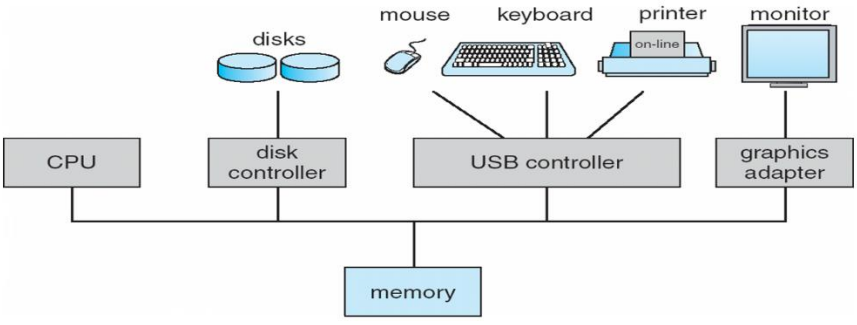
- **Examples:** LynxOS, OSE, QNX, RTLinux, VxWorks, Windows CE

Handheld Operating System
It is also known as **Mobile OS** which is built exclusively for a mobile device, such as a smartphone, personal digital assistant (PDA), tablet, wearable devices or other embedded mobile OS.

- **Examples:** Android, Symbian, iOS, BlackBerry OS and Windows Mobile.

SUBTOPIC 2: Computer System Organization, Architectures and Computing Environment

A computer system is made up of various components. The components can be hardware or software. Because these systems are so massively complex, the components are organized in layers.



The important points about the previous figure displaying Computer System Organization is:

- One or more CPUs, device controllers connect through common bus providing access to shared memory.
- The I/O devices and the CPU both execute concurrently.

- Some of the processes are scheduled for the CPU and at the same time, some are undergoing input/output operations.
- There are multiple device controllers, each in charge of a particular device such as keyboard, mouse, printer etc.
- There is buffer available for each of the devices. The input and output data can be stored in these buffers. Buffer is a region of memory used to temporarily hold data while it is being moved from one place to another.
- The data is moved from memory to the respective device buffers by the CPU for I/O operations and then this data is moved back from the buffers to memory.
- The device controllers use an interrupt to inform the CPU that I/O operation is completed.

What is an Interrupt?

- An operating system is interrupt driven.
- Interrupt is a signal emitted by hardware or software when a process or an event needs immediate attention.
- It alerts the processor temporarily to a high priority process requiring interruption of the current working process and then return to its previous task.

Types of Interrupts:

- Hardware interrupt
- Software interrupt

Hardware interrupt is a signal created and sent to the CPU that is caused by some action taken by a hardware device.
Example: When a key is pressed or when the mouse is moved.

Software Interrupt arises due to illegal and erroneous use of an instruction or data. It often occurs when an application software terminates or when it requests the operating system for some service.
Example: stack overflow, division by zero, invalid opcode, etc. These are also called **traps**.

Interrupt Handling

The operating system preserves the state of the CPU by storing registers and the program counter
Determines which type of interrupt has occurred:

- Polling – operating system sends signal to each devices asking if they have a request
- Vectored interrupt system – requesting device sends interrupt to the operating system.

Separate segments of code determine what action should be taken for each type of interrupt.

Operating System Operations

- Dual-mode operation allows OS to protect itself and other system components
- **User mode** and **kernel mode**
- Mode bit provided by hardware
- Provides ability to distinguish when system is running user code(1) or kernel code(0)
- Some instructions designated as privileged, only executable in kernel mode
- System call changes mode to kernel, return from call resets it to user

A **system call** is a way for programs to interact with the OS. A computer program makes a system call when it makes a request to the OS‘s kernel.

Single-Processor System
There is one main CPU capable of executing a general-purpose instruction set, including instructions from user processes.

Multiprocessor System
Also known as **parallel-system** or **multicore**.
First appeared in servers and now in smartphones and tablet computers.

Computer System Architecture

- Most systems use a single general-purpose processor (PDAs through mainframes)
- Most systems have special-purpose processors as well
- Multiprocessors systems growing in use and importance
- Also known as parallel systems or tightly-coupled systems

Advantages:

1. **Increased throughput**. Increasing the number of processor, expect to get more work done in less time.
2. **Economy of scale**. It can cost less than equivalent multiple single-processor systems because they can share peripherals, mass storage and power supplies.
3. **Increased reliability**. Functions can be distributed properly among several processors. If one processor fails, the other processor can pick-up the task.

Two types of Multiprocessing:

- Asymmetric Multiprocessing
- Symmetric Multiprocessing

1. **Asymmetric multiprocessing**. Each processor is assigned a specific task. A boss processor controls the system and the other processors either look to the boss for instructions or have predefined tasks. **Boss-worker relationship**.
2. **Symmetric multiprocessing(SMP)**. The most commonly used. In which each processor performs all tasks within the operating system. **All processors are peers** and **no boss-worker relationship.**

The difference between symmetric and asymmetric may result from either hardware or software.

A recent trend in CPU design is to include multiple computing cores on a single chip. Such multiprocessor systems are termed multicore. They can be more efficient than multiple chips with single core.
A dual-core design with two cores on the same chip. Each core has its own register set as well as its own local cache.

Clustered System

- Like multiprocessor systems, but multiple systems working together
- Usually sharing storage via a storage-area network (SAN)
- Provides a high-availability service which survives failures
- Asymmetric clustering has one machine in hot-standby mode

- Symmetric clustering has multiple nodes running applications, monitoring each other
- Some clusters are for high-performance computing (HPC).

Computing Environment

- **Office computing environment**
  - PCs connected to a network, terminals attached to mainframe or minicomputers providing batch and timesharing
  - Now portals allowing networked and remote systems access to same resources

- **Home computing environment**
  - Used to be single system, then modems
  - Now firewalled, networked

- **Mobile computing**
  - Refers to computing on handheld smartphones and tablet computers.

- **Distributed system**
  - It is a collection of physically separate, possibly heterogeneous computer systems that are networked to provide users with access to the various resources that the system maintains.
  - Network operating system is an operating system that provides services across the network.

Types of Distributed System:

- **Client-Server Computing**
  - Dumb terminals succeeded by smart PCs
  - Many systems now **servers**, responding to requests generated by **clients**
    - **Compute-server** provides an interface to client to request services (i.e. database)
    - **File-server** provides interface for clients to store and retrieve files
- **P2P** does not distinguish clients and servers
  - Instead **all nodes are considered peers**
  - May each act as client, server or both
  - Node must join P2P network
    - Registers its service with central lookup service on network, or
    - Broadcast request for service and respond to requests for service via **discovery protocol**
  - **Examples:** *Napster* and *BitTorrent*

- **Virtualization**
  - It is a technology that allows operating systems to run as applications within other operating system.
  - Virtualization plays a major role in cloud computing as it provides a virtual storage and computing services to the cloud clients which is only possible through virtualization.

- It is one member of the class software that also includes emulation. Emulation is used when the source CPU type is different from the target CPU type.
  Example: virtual machine, OracleVirtualBox

Cloud Computing

- It is a type of computing that delivers computing, storage and even applications as a service across a network.
- It is a logical extension of virtualization.

Types of Cloud

- **Public cloud** – cloud available via the Internet
- **Private cloud** – cloud run by a company for that company's own use
- **Hybrid cloud** – cloud that includes both public and private

Cloud Computing Service Models

- **Software as a service(SaaS)** – one or more applications available via the Internet
- **Platform as a service(PaaS)** – software stack ready for application use via the Internet
- **Infrastructure as a service(IaaS)** – servers or storage available over the Internet

Open-Source Operating System

- **Open Source operating systems** are released under a license where the copyright holder allows others to study, change as well as distribute the software to other people.
- Counter to the copy protection and **Digital Rights Management (DRM)** movement
- Started by **Free Software Foundation (FSF),** which has **"copyleft"** GNU Public License (GPL)
- **Examples:** GNU(**GNU's N**ot **U**nix)/Linux, BSD UNIX (including core of Mac OS X), and Sun Solaris