NATIONAL YOUTH COMPUTER LITERACY MISSION

RECOGNISED BY GOVT. OF INDIA

AS -ANZ accredited An

ISO 9001 : 2015 Certified Autonomous Body

A National Programme of Information TechnologyEducation & Development

About the Tutorial

An operating system (OS) is a collection of software that manages computer hardware resources and provides common services for computer programs. The operating system is a vital component of the system software in a computer system.

This tutorial will take you through step-by-step approach while learning Operating System concepts.

Audience

This tutorial has been prepared for the computer science graduates to help them understand the basic to advanced concepts related to Operating System.

Prerequisites

Before you start proceeding with this tutorial, we are making an assumption that you are already aware of basic computer concepts like what is keyboard, mouse, monitor, input, output, primary memory and secondary memory etc. If you are not well aware of these concepts, then we will suggest to go through our short tutorial on Computer Fundamentals.

Copyright & Disclaimer

© Copyright 2016 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com

Table of Contents

About the Tutorial **1**

Audience **1**

Prerequisites **1**

Copyright & Disclaimer **1**

Table of Contents **2**

1. OPERATING SYSTEM – OVERVIEW 1

Definition **1**

Memory Management **2**

Processor Management **2**

Device Management **2**

File Management **2**

Other Important Activities **3**

2. OPERATING SYSTEM – TYPES 4

Batch Operating System **4**

Time-sharing Operating Systems **4**

Distributed Operating System **5**

Network Operating System **5**

Real-Time Operating System **6**

3. OPERATING SYSTEM – SERVICES 7

Program Execution **7**

I/O Operation **7**

File System Manipulation **8**

Communication **8**

Error Handling **8**

Resource Management **9**

Protection **9**

4. OPERATING SYSTEM – PROPERTIES 10

Batch Processing **10**

Multitasking **11**

Multiprogramming **12**

Interactivity **13**

Real-Time Systems **13**

Distributed Environment **13**

Spooling **13**

5. OPERATING SYSTEM – PROCESSES 15

Process **15**

Program **16**

Process Life Cycle **16**

Process Control Block (PCB) **17**

6. OPERATING SYSTEM – PROCESS SCHEDULING 19

Definition **19**

Process Scheduling Queues **19**

Two-State Process Model **20**

Schedulers **20**

Long-Term Scheduler **20**

Short-Term Scheduler **20**

Medium-Term Scheduler **21**

Comparison among Schedulers **21**

Context Switch **21**

7. OPERATING SYSTEM – SCHEDULING ALGORITHMS 23

First Come, First Served (FCFS) **23**

Shortest Job Next (SJN) **24**

Priority Based Scheduling **25**

Shortest Remaining Time **26**

Round Robin Scheduling **26**

Multiple-Level Queues Scheduling **27**

8. OPERATING SYSTEM – MULTITHREADING 28

What is a Thread? **28**

Difference between Process and Thread **29**

Advantages of Thread **29**

Types of Thread **29**

User Level Threads **29**

Kernel Level Threads **30**

Multithreading Models **31**

Many-to-Many Model **31**

Many-to-One Model **32**

One-to-One Model **33**

Difference between User-Level & Kernel-Level Thread **34**

9. OPERATING SYSTEM – MEMORY MANAGEMENT 35

Process Address Space **35**

Static vs Dynamic Loading **36**

Static vs Dynamic Linking **36**

Swapping **36**

Memory Allocation **38**

Fragmentation **38**

Paging **39**

Segmentation **42**

10. OPERATING SYSTEM – VIRTUAL MEMORY 43

Demand Paging **44**

Page Replacement Algorithm **46**

Reference String **46**

First In First Out (FIFO) Algorithm **46**

Optimal Page Algorithm **47**

Least Recently Used (LRU) Algorithm **48**

Page Buffering Algorithm **48**

Least Frequently Used (LFU) Algorithm **48**

Most Frequently Used (MFU) Algorithm **48**

11. OPERATING SYSTEM – I/O HARDWARE 49

Device Controllers **49**

Synchronous vs Asynchronous I/O **50**

Communication to I/O Devices **50**

Polling vs Interrupts I/O **52**

12. OPERATING SYSTEM – I/O SOFTWARE 54

Device Drivers **55**

Interrupt handlers **55**

Device-Independent I/O Software **55**

User-Space I/O Software **56**

Kernel I/O Subsystem **56**

13. OPERATING SYSTEM – FILE SYSTEM 57

File **57**

File Structure **57**

File Type **57**

File Access Mechanisms **58**

Space Allocation **58**

14. OPERATING SYSTEM – SECURITY 60

Authentication **60**

One Time passwords **60**

Program Threats **61**

System Threats **61**

Computer Security Classifications **62**

15. OPERATING SYSTEM – LINUX 63

Components of Linux System **63**

Kernel Mode vs. User Mode **64**

Basic Features **64**

Architecture **65**

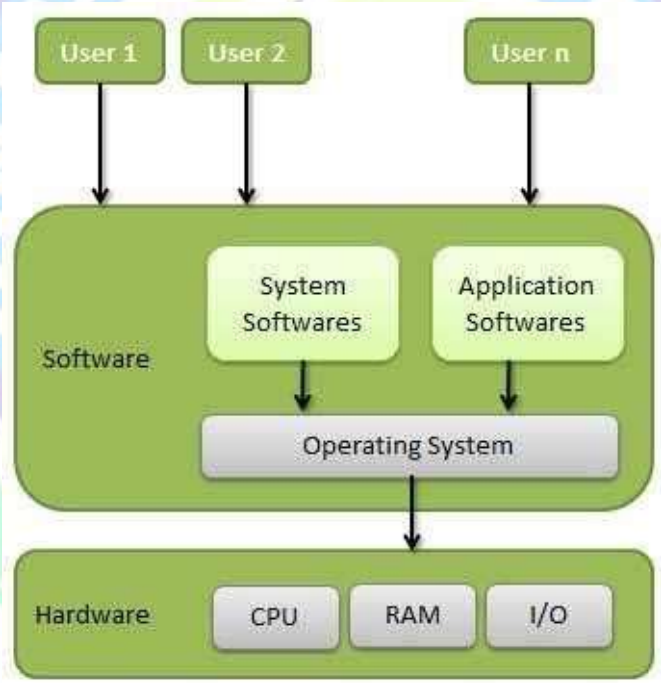
1. Operating System – Overview

An Operating System (OS) is an interface between a computer user and computer hardware. An operating system is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.

Some popular Operating Systems include Linux Operating System, Windows Operating System, VMS, OS/400, AIX, z/OS, etc.

Definition

An operating system is a program that acts as an interface between the user and the computer hardware and controls the execution of all kinds of programs.



Following are some of important functions of an operating System.

- Memory Management
- Processor Management
- Device Management
- File Management
- Security
- Control over system performance
- Job accounting
- Error detecting aids
- Coordination between other software and users

Memory Management

Memory management refers to management of Primary Memory or Main Memory. Main memory is a large array of words or bytes where each word or byte has its own address.

Main memory provides a fast storage that can be accessed directly by the CPU. For a program to be executed, it must in the main memory. An Operating System does the following activities for memory management:

- Keeps tracks of primary memory, i.e., what part of it are in use by whom, what part are not in use.
- In multiprogramming, the OS decides which process will get memory when and how much.
- Allocates the memory when a process requests it to do so.
- De-allocates the memory when a process no longer needs it or has been terminated.

Processor Management

In multiprogramming environment, the OS decides which process gets the processor when and for how much time. This function is called **process scheduling**. An Operating System does the following activities for processor management:

- Keeps tracks of processor and status of process. The program responsible for this task is known as **traffic controller**.
- Allocates the processor (CPU) to a process.
- De-allocates processor when a process is no longer required.

Device Management

An Operating System manages device communication via their respective drivers. It does the following activities for device management:

- Keeps tracks of all devices. The program responsible for this task is known as the **I/O controller**.
- Decides which process gets the device when and for how much time.
- Allocates the device in the most efficient way.
- De-allocates devices.

File Management

A file system is normally organized into directories for easy navigation and usage. These directories may contain files and other directions.

An Operating System does the following activities for file management:

- Keeps track of information, location, uses, status etc. The collective facilities are often known as **file system**.
- Decides who gets the resources.
- Allocates the resources.
- De-allocates the resources.

Other Important Activities

Following are some of the important activities that an Operating System performs:

- **Security** -- By means of password and similar other techniques, it prevents unauthorized access to programs and data.
- **Control over system performance** -- Recording delays between request for a service and response from the system.
- **Job accounting** -- Keeping track of time and resources used by various jobs and users.
- **Error detecting aids** -- Production of dumps, traces, error messages, and other debugging and error detecting aids.
- **Coordination between other software and users** -- Coordination and assignment of compilers, interpreters, assemblers and other software to the various users of the computer systems.

2. Operating System – Types

Operating systems are there from the very first computer generation and they keep evolving with time. In this chapter, we will discuss some of the important types of operating systems which are most commonly used.

Batch Operating System

The users of a batch operating system do not interact with the computer directly. Each user prepares his job on an off-line device like punch cards and submits it to the computer operator. To speed up processing, jobs with similar needs are batched together and run as a group. The programmers leave their programs with the operator and the operator then sorts the programs with similar requirements into batches.

The problems with Batch Systems are as follows:

- Lack of interaction between the user and the job.
- CPU is often idle, because the speed of the mechanical I/O devices is slower than the CPU.
- Difficult to provide the desired priority.

Time-sharing Operating Systems

Time-sharing is a technique which enables many people, located at various terminals, to use a particular computer system at the same time. Time-sharing or multitasking is a logical extension of multiprogramming. Processor's time which is shared among multiple users simultaneously is termed as time-sharing.

The main difference between Multiprogrammed Batch Systems and Time-Sharing Systems is that in case of Multiprogrammed batch systems, the objective is to maximize processor use, whereas in Time-Sharing Systems, the objective is to minimize response time.

Multiple jobs are executed by the CPU by switching between them, but the switches occur so frequently. Thus, the user can receive an immediate response. For example, in a transaction processing, the processor executes each user program in a short burst or quantum of computation. That is, if **n** users are present, then each user can get a time quantum. When the user submits the command, the response time is in few seconds at most.

The operating system uses CPU scheduling and multiprogramming to provide each user with a small portion of a time. Computer systems that were designed primarily as batch systems have been modified to time-sharing systems.

Advantages of Timesharing operating systems are as follows:

- Provides the advantage of quick response
- Avoids duplication of software
- Reduces CPU idle time

Disadvantages of Time-sharing operating systems are as follows:

- Problem of reliability
- Question of security and integrity of user programs and data
- Problem of data communication

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.

The processors communicate with one another through various communication lines (such as high-speed buses or telephone lines). These are referred as **loosely coupled systems** or distributed systems. Processors in a distributed system may vary in size and function. These processors are referred as sites, nodes, computers, and so on.

The advantages of distributed systems are as follows:

- With resource sharing facility, a user at one site may be able to use the resources available at another.
- Speedup the exchange of data with one another via electronic mail.
- If one site fails in a distributed system, the remaining sites can potentially continue operating.
- Better service to the customers.
- Reduction of the load on the host computer.
- Reduction of delays in data processing.

Network Operating System

A Network Operating System runs on a server and provides the server the capability to manage data, users, groups, security, applications, and other networking functions. The primary purpose of the network operating system is to allow shared file and printer access among multiple computers in a network, typically a local area network (LAN), a private network or to other networks.

Examples of network operating systems include Microsoft Windows Server 2003, Microsoft Windows Server 2008, UNIX, Linux, Mac OS X, Novell NetWare, and BSD.

The advantages of network operating systems are as follows:

- Centralized servers are highly stable.
- Security is server managed.
- Upgrades to new technologies and hardware can be easily integrated into the system.

- Remote access to servers is possible from different locations and types of systems.

The disadvantages of network operating systems are as follows:

- High cost of buying and running a server.
- Dependency on a central location for most operations.
- Regular maintenance and updates are required.

Real-Time Operating System

A real-time system is defined as a data processing system in which the time interval required to process and respond to inputs is so small that it controls the environment. The time taken by the system to respond to an input and display of required updated information is termed as the **response time**. So in this method, the response time is very less as compared to online processing.

Real-time systems are used when there are rigid time requirements on the operation of a processor or the flow of data and real-time systems can be used as a control device in a dedicated application. A real-time operating system must have well-defined, fixed time constraints, otherwise the system will fail. For example, Scientific experiments, medical imaging systems, industrial control systems, weapon systems, robots, air traffic control systems, etc.

There are two types of real-time operating systems.

Hard real-time systems

Hard real-time systems guarantee that critical tasks complete on time. In hard real-time systems, secondary storage is limited or missing and the data is stored in ROM. In these systems, virtual memory is almost never found.

Soft real-time systems

Soft real-time systems are less restrictive. A critical real-time task gets priority over other tasks and retains the priority until it completes. Soft real-time systems have limited utility than hard real-time systems. For example, multimedia, virtual reality, Advanced Scientific Projects like undersea exploration and planetary rovers, etc.

3. Operating System – Services

An Operating System provides services to both the users and to the programs.

- It provides programs an environment to execute.
- It provides users the services to execute the programs in a convenient manner.

Following are a few common services provided by an operating system:

- Program execution
- I/O operations
- File System manipulation
- Communication
- Error Detection
- Resource Allocation
- Protection

Program Execution

Operating systems handle many kinds of activities from user programs to system programs like printer spooler, name servers, file server, etc. Each of these activities is encapsulated as a process.

A process includes the complete execution context (code to execute, data to manipulate, registers, OS resources in use). Following are the major activities of an operating system with respect to program management:

- Loads a program into memory
- Executes the program
- Handles program's execution
- Provides a mechanism for process synchronization
- Provides a mechanism for process communication
- Provides a mechanism for deadlock handling

I/O Operation

An I/O subsystem comprises of I/O devices and their corresponding driver software. Drivers hide the peculiarities of specific hardware devices from the users.

An Operating System manages the communication between user and device drivers.

- I/O operation means read or write operation with any file or any specific I/O device.
- Operating system provides the access to the required I/O device when required.

File System Manipulation

A file represents a collection of related information. Computers can store files on the disk (secondary storage), for long-term storage purpose. Examples of storage media include magnetic tape, magnetic disk and optical disk drives like CD, DVD. Each of these media has its own properties like speed, capacity, data transfer rate and data access methods.

A file system is normally organized into directories for easy navigation and usage. These directories may contain files and other directions. Following are the major activities of an operating system with respect to file management:

- Program needs to read a file or write a file.
- The operating system gives the permission to the program for operation on file.
- Permission varies from read-only, read-write, denied, and so on.
- Operating System provides an interface to the user to create/delete files.
- Operating System provides an interface to the user to create/delete directories.
- Operating System provides an interface to create the backup of file system.

Communication

In case of distributed systems which are a collection of processors that do not share memory, peripheral devices, or a clock, the operating system manages communications between all the processes. Multiple processes communicate with one another through communication lines in the network.

The OS handles routing and connection strategies, and the problems of contention and security. Following are the major activities of an operating system with respect to communication:

- Two processes often require data to be transferred between them.
- Both the processes can be on one computer or on different computers, but are connected through a computer network.
- Communication may be implemented by two methods, either by Shared Memory or by Message Passing.

Error Handling

Errors can occur anytime and anywhere. An error may occur in CPU, in I/O devices or in the memory hardware. Following are the major activities of an operating system with respect to error handling:

- The OS constantly checks for possible errors.
- The OS takes an appropriate action to ensure correct and consistent computing.

Resource Management

In case of multi-user or multi-tasking environment, resources such as main memory, CPU cycles and files storage are to be allocated to each user or job. Following are the major activities of an operating system with respect to resource management:

- The OS manages all kinds of resources using schedulers.
- CPU scheduling algorithms are used for better utilization of CPU.

Protection

Considering a computer system having multiple users and concurrent execution of multiple processes, the various processes must be protected from each other's activities.

Protection refers to a mechanism or a way to control the access of programs, processes, or users to the resources defined by a computer system. Following are the major activities of an operating system with respect to protection:

- The OS ensures that all access to system resources is controlled.
- The OS ensures that external I/O devices are protected from invalid access attempts.
- The OS provides authentication features for each user by means of passwords.

End of ebook preview

If you liked what you saw…

Buy it from our store @ **https://store.tutorialspoint.com**