**STUDY GUIDE FOR MODULE NO. 1**

**Unit 1 – Introduction to Operating System**

**MODULE OVERVIEW**

Modern computers consists of several parts and peripherals necessary to solve problem and execute task and it makes the system complex and difficult to use. For this reason, operating system was developed, a layer of software that is equipped by computers for efficient use. Operating system provide user programs with a better, simpler, cleaner, model of the computer and to handle managing all the resources as its main purpose. This module will give you an overview of what operating systems do, components of computer system and its basic principles, evolution of operating system in different computer architecture, and types of operating system in computing environments.

**MODULE LEARNING OBJECTIVES**

At the end of this, students are expected to:

1. Define the concepts of Operating System
2. Describe the Components of Computer System
3. Discuss the importance of the evolution of Operating System and Computer Environments

**LEARNING CONTENTS (Operating System)**

**What is Operating System?**

Operating System is a program that acts as intermediary between computer user and computer hardware. Technically, it is a software that runs in kernel (kernel is a program that loads OS as the computer turned on) where operating system controls the system and performs essential functions: as an extended machine, and as a resource manager.

* **OS as an Extended Machine** – The architecture (such as instruction set, memory organization, I/O, and bus structure) of most computers at the machine-language level is primitive and difficult to program for input/output. To make it comprehensive, there are software called **drivers** that deals with the hardware and provides an interface to process tasks without getting into the details. Operating systems provide drivers for controlling I/O devices. Operating systems provide another layer of **abstraction** for using disks: files. In the utilization of abstraction is the key to managing complexity since programs can create, write, and read files, without having to deal with the messy details of how the hardware actually works. In **good abstractions**, the impossible task turns into two manageable ones: defining and implementing the abstractions, and using these abstractions to solve the problem at hand. Example abstraction is **file** that is much easier to deal with digital photo, emails, songs, and web pages than with the details of SATA disks. The purpose of the operating system is to create good abstractions and then implement and manage the abstract objects created consequently. Hide the hardware and present programs and their programmers with nice, clean, elegant, consistent, abstractions to work with is one of the major tasks of the operating system. It is important to consider that operating system’s real customers are the application programs via the application programmers (the ones who deal directly with the operating system and its abstractions). In the contrary, end users deal with the abstractions provided by the user interface, either a command line or a graphical interface.

Operating system serves as **control program** that controls the execution of programs to prevent errors and improper use of computer.

* **OS as Resource Manager** – Operating system is there to manage all the pieces of a complex system. The purpose of the operating system is a **resource allocator** (manages and allocates resources) to provide an orderly and controlled allocation of the processors, memories, and I/O devices among the various programs wanting them. Modern operating systems allow multiple programs to be in memory and run at the same time. The operating system can bring order to the potential chaos by buffering all the output destined for the printer on the disk. Operating system holds that its primary task is to keep track of which programs are using which resource, to grant resource requests, to account for usage, and to mediate conflicting requests from different programs and users. Resource management includes **multiplexing (sharing) resources** in time and in space. A resource is **time multiplexed** when different programs or users take turns using it. The task of OS in time multiplexed resource is determining “who goes next and for how long”. Example of time multiplexing is sharing the printer. A decision has to be made about which one is to be printed next when multiple print jobs are queued up for printing on a single printer. The other kind of multiplexing is **space multiplexing** where each one gets part of the resource. For example, the disk where a single disk can hold files from many users at the same time. A typical OS task is allocating disk space and keeping track of who is using which disk blocks.

**LEARNING CONTENTS (Components of Computer System)**

**Components of Computer System**

Identifying the computer system components help to understand the basic principles in the design of the computer system. It enables the programmer to develop a new hardware and applications, and an efficient utilization of computer programs for users. The computer system is a collection of hardware, software, and peopleware that are designed to receive, process, manage and present information in a meaningful format. Computer system composed of four (4) components: applications programs, application program/utilities, operating system, and the hardware. These components are classified into two categories: the hardware and software.  
  
As shown in Figure 1, a software includes applications programs (word processors, games, and web browsers), application program and/or utilities, and operating system (Microsoft Windows, MacOS, and Linux). On the other hand, a hardware category includes computer hardware components (CPU, Memory, and Input/Output Devices) as well as peripherals connected to a computer.

Applications Programs

Application Program/Utilities

Operating System

Hardware

USER

SOFTWARE

HARDWARE

Figure 1: The computer system components

**Application program** helps users to carry out specific task and solve a particular problem.

**Operating system** controls the operations of computer hardware by coordinating the activities and functions of both hardware and software.

**Utility software** helps manage, maintain and control computer resources. Operating systems contain the necessary tools this but separate utility programs can provide improved functionality.

**Hardware** refers to the physical component, tangible equipment and devices connected to the computer, which provide support for major functions such as input, processing (internal storage, computation and control), output, secondary storage (for data and programs), and communication.

**LEARNING CONTENTS (Evolution of OS)**

**Evolution of Operating System**

Operating systems have been evolving through generations. The architecture of computers was anchored to the history of operating systems since they run on different generations of computer architectures. Mapping of operating system generations to computer generations provide some structure in the evolution process. The progression shown in Figure 2 is largely chronological, but each development did not wait until the previous one nicely finished before getting started. It will serve as a guide.

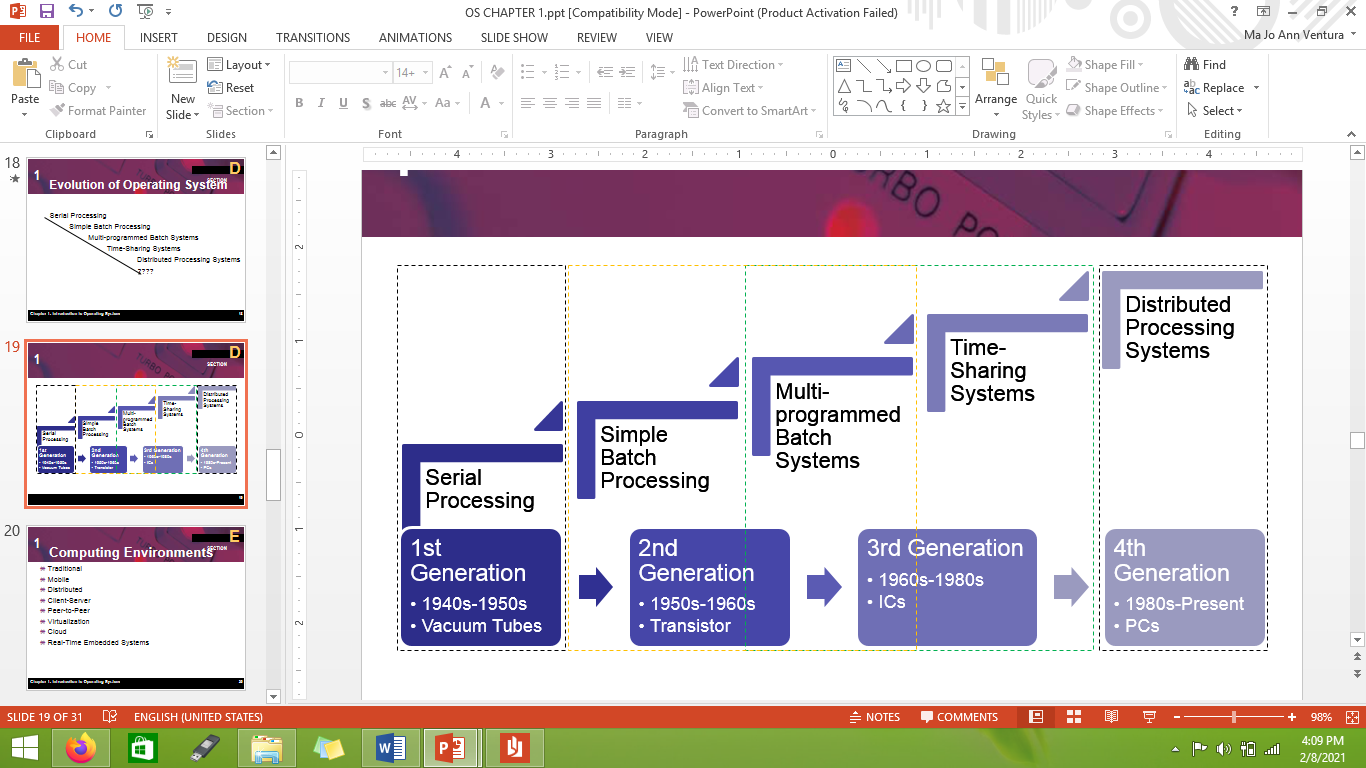


Figure 2: Evolution of Operating Systems

**Serial processing** perform instructions in a sequence manner (First In, First Out) using program counter to execute all the instructions. It is being used on vacuum tubes architecture.

**Simple batch processing** technique is similar to serial processing with automatic job sequencing. Batch system reads the first job from tape and ran it where output was written onto a second tape instead of being printed. Operating system automatically read the next job from tape every after each job is finished. It was used during 2nd generation and upgraded until 3rd generations.

**Multi-programmed batch system** was used during 3rd generation of computer architecture. This technique uses memory and CPUs to execute multiple programs on the system at a time. It will store several jobs in main memory and the CPU multiplexed among them.

**Time-sharing system** also known as real time system and interactive computing that follows the multiprogramming rapidly during 3rd generation. The CPU is multiplexed among several jobs that are kept in memory and on disk where the CPU is allocated to a job only if the job is in memory. A job swapped in and out of memory to the disk. On-line communication between the user and the system is necessary and must be available for users to access data and code because when the operating system finishes the execution of one command, it seeks the next “control statement” from the user’s input device.

**Distributed processing system/Distributed system** takes place during 4th generation. It uses network to distribute the computation among several physical processors. Data can be stored and processed on multiple locations.

**LEARNING CONTENTS (Computer Environments)**

**Computing Environments**

A variety of operating systems have been developed for being around in century and some variety are not widely known. Every computing environments consists of computers, computational devices, software, and networks that solve problem task and support processing and sharing information demanded by software solution.

**Traditional** computing environment refers to stand-alone general purpose machines that uses direct cable connectionsas most systems interconnect with others (i.e. the Internet)**.** Portals provide web access to internal systemsand network computers (thin clients) are like Web terminals.However, mobile computers interconnect via wireless networks and theyuse firewalls to protect computers from Internet attacks even home systems since networking becoming ubiquitous.

**Mobile** computing environments are handheld smartphones, tablets, and the like, led by Apple iOS and Google Android. It has more OS features (such as GPS and gyroscope) and it allow new types of apps like augmented reality.This type of computing uses IEEE 802.11 wireless, or cellular data networks for connectivity.

**Distributed** computing is a collection of separate, possibly heterogeneous, systems networked together where network as communication path use the most common TCP/IP that includes: Local Area Network (LAN), Wide Area Network (WAN), Metropolitan Area Network (MAN), and Personal Area Network (PAN). **Network Operating System** provides features between systems across network. It is a communication scheme that allows systems to exchange messages and provide an illusion of a single system.

**Client-Server Computing** are dumb terminals supplanted by smart PCs**.** Many systems now are **servers**, responding to requests generated by **clients.** It consists of **compute-server system** that provide an interface to client to request services (i.e., database)and **file-server system** toprovide interface for clients to store and retrieve files.

**Virtualization** allows operating systems to run applications within other OSs**.** Avast and growing industry that provide **Emulation** utility which used when source CPU type different from target type (i.e. PowerPC to Intel x86)and **Virtualization** where OS natively compiled for CPU, running **guest** OSes also natively compiled considering VMware running WinXP guests, each running applications, all on native WinXP **host** OSvia **VMM** that provides virtualization services.

**Cloud Computing** delivers computing, storage, even apps as a service across a network. It is a logical extension of virtualization as based on virtualization such as Amazon **EC2** that has thousands of servers, millions of VMs, PBs of storage available across the Internet, pay based on usage. There are many types of cloud computing it includes the following: **Public cloud** – available via Internet to anyone willing to pay, **Private cloud** – run by a company for the company’s own use, **Hybrid cloud** – includes both public and private cloud components, Software as a Service (**SaaS**) – one or more applications available via the Internet (i.e. word processor), Platform as a Service (**PaaS**) – software stack ready for application use via the Internet (i.e a database server), and Infrastructure as a Service (**IaaS**) – servers or storage available over Internet (i.e. storage available for backup use).

**LEARNING ACTIVITY 1**

1. Different computing environments of operating systems are described. Give a list of applications for each of these systems (one per computing environments type).
2. What is the difference between timesharing and multiprogramming systems?

**SUMMARY**

* The goal of Operating System is to execute user programs, easier solving of user problems, and make the computer system convenient to use by utilizing abstraction that controls program in execution, and resource manager (resource allocator) to allocate resources properly for efficient use.
* Computer system composed of four (4) components: applications programs, application program/utilities, operating system, and the hardware. These components are classified into two categories: the hardware and software.
* In the evolution of operating system the architecture of computers was anchored to its history since they run on different generations of computer architectures. Serial processing was used in the 1st generation that execute job in a sequence manner (First In, First Out). Simple batch processing is similar to serial processing with automatic job sequencing and was developed and used during 2nd generation. In the 3rd generation, the operating system uses multi-programmed batch system and rapidly added the time-sharing system. Distributed system was then used during 4th generation up to present.
* Every computing environments consists of computers, computational devices, software, and networks that solve problem task and support processing and sharing information demanded by software solution. It includes traditional, mobile, distributed, client-server, virtualization, and cloud computing.

 **REFERENCES**

**­Books/E-Books:**

Tanenbaum, A. S., Bos, H. Modern Operating Systems. 4th Edition. Pearson Education, Inc. 2015

Silberschatz, Abraham., Galvin., Gagne Greg., Operating System Concepts. 10th Edition. John Wiley and Sons, Inc., 2018

**E-Sources:**

<https://www.geeksforgeeks.org/computing-environments/>