Enrollment No:- 220101103007 20/07/2023

Practical 1

1] what is operating system ?

**Operating System** lies in the category of system software. It basically manages all the resources of the computer. An operating system acts as an interface between the software and different parts of the computer or the computer hardware. The operating system is designed in such a way that it can manage the overall resources and operations of the computer.

Operating System is a fully integrated set of specialized programs that handle all the operations of the computer. It controls and monitors the execution of all other programs that reside in the computer, which also includes application programs and other system software of the computer. Examples of Operating Systems are Windows, Linux, Mac OS, etc.

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 the most important type of system software in a computer system.

**2)** Example of operating system?

Examples of Operating Systems

Microsoft Windows. Without doubt the most popular of the OS, although it really is a set of distributions (an operating environment) built to provide older Operating Systems (such as MS-DOS) with a supporting graphical interface and a set of software tools. Its first version appeared in 1985 And since then it has not stopped updating in more powerful and diverse versions, as Microsoft, its mother company, prevails in the market of digital technologies.

microsoft windows 10

GNU / Linux. This term refers to the combined use of the kernel free from the Unix family called “Linux”, along with the GNU distribution, also free. The result is one of the main protagonists in the development of free software, whose source code can be freely used, modified and redistributed.

UNIX. This portable, multi-tasking, multi-user operating system was developed early in 1969, and over the years its rights to copyright they have passed from one company to another. In reality it is a family of similar OS, many of which have become commercial and others are free format, all from the Linux kernel.

Fedora. It is essentially a general-purpose Linux distribution, which emerged after the discontinuation of Red Hat Linux, with which he is closely linked but which emerged as a community project. It is another indispensable name when it comes to talking about free software and open source, in its three main versions: Workstation, Cloud and Server.

Ubuntu. Based on GNU / Linux, this free and open source Operating System takes its name from the South African philosophy focused on the loyalty of man to the rest of the species. In this sense, Ubuntu is oriented towards ease and freedom of use, although Canonical, the British company that owns its rights, subsists on the basis of technical services linked to the program.

MacOS. The Machintosh operating system, also known as OSX or Mac OS X, whose environment is based on Unix and has been developed and sold as part of Apple-brand computers since 2002. Part of this family of software was released by Apple as an open and free source operating system called Darwin, to which they later added components such as Aqua and the Finder, to obtain the interface on which Mac OS X, its most recent version, is based

**3) operating system important?**

An operating system, or OS for short is the most basic program on your computer. It interfaces with hardware and helps you run programs that are stored on it to do tasks like write letters or play games.

The three most important aspects of an operating system are its ability to manage resources, establish the interface for users and services that run on it. Additionally, an effective OS must be fast enough so as not to interrupt what you’re doing when loading programs or switching between them in order not to get caught with unexpected errors.

Importance

The need for an operating system is clear. The importance of one can not be overstated, but the importance is here,

The operating system is a clever and efficient resource manager. It utilizes the computer in an economical manner, keeping track of different jobs as they come along with their results stored where appropriate so that it can return them when requested by other processes or users on your device.

Job control is the key to keeping jobs running smoothly in a multi-user environment. It schedules jobs according to their priority, passing control from one program or task group to another until they are all completed successfully without interruption.

Operating system is the main link between your computer and all of its programs. It not only helps run applications, but also ensures that you get good output for them.

The operating system can fetch programs from memory when required and not all at once. This gives the user more space to work in one package, rather than having multiple packages loading on top of each other.

The operating system helps the user in file management, making directions and saving files to them. This is a very good feature provided by an OS that can organize data according to your needs.

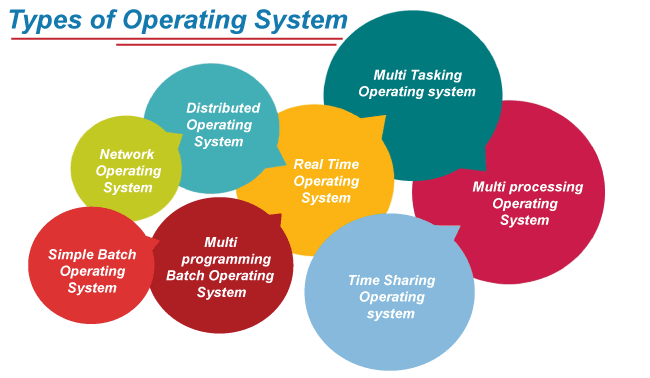
Multiprogramming is a very important feature of the operating system. It schedules and controls the running of several programs at once so you can get more done, faster.

The program editor allows users to modify and update their programs with ease.

The operating system provides helpful debugging aids to the user that help them detect errors in programs.

The operating system uses diskette checks to make sure the data on your disks are valid.

**4)Types of operating system**



**5) List out operating system reference book of operating system?**

1. Operating Systems

1."Operating System Concepts" by Avi Silberschatz and Peter Galvin

Book Review: In this book, you will gain a clear understanding of how operating systems work. It covers important concepts and uses examples from popular operating systems like UNIX, Solaris, Windows NT, Apple, IBM, and Linux. The book also explores the impact of Java technology on operating systems, illustrating key concepts. You will learn about multitasking, CPU scheduling, process synchronization, deadlock, and security using Java. There are plenty of examples from operating systems such as Solaris, Windows NT, and Linux. A dedicated chapter is included to help you understand how the operating system schedules threads, enhancing your knowledge in this area.

2."Operating Systems: Internals and Design Principles" by William Stallings

Book Review: This book offers a thorough and easy-to-understand introduction to operating systems. It covers important topics and focuses on both design issues and fundamental principles found in modern operating systems. The author provides a strong foundation by explaining key structures and mechanisms in operating systems. They also explore trade-offs and practical decisions that impact design, performance, and security. The book uses real-world examples to reinforce design concepts and their practical applications. Additionally, it includes information on the latest trends and advancements in operating systems, ensuring readers stay up-to-date with the field’s developments.

3."Operating Systems: A Concept-Based Approach" by D M Dhamdhere

Book Review: This book is perfect for students learning about operating systems as it provides easy-to-understand definitions and explanations of the basic concepts. It covers a wide range of topics including techniques, case studies, and design and implementation details. The book offers clear explanations of each concept in operating systems, covering both the fundamentals and the latest technologies like encryption and security. Additionally, it includes real-world case studies that connect the concepts to practical experiences in operating systems. With its comprehensive coverage and practical approach, this book is an excellent resource for anyone studying operating systems.

4."Operating System: A Design-oriented Approach" by Charles Crowley

“Operating System: A Design-oriented Approach” Book Review: This book explains the fundamental concepts of Operating Systems and how they have developed over time. It includes code examples and case studies from various Operating Systems like Mach, Windows NT, OS/2, MacOS, MS/DOS, and different versions of UNIX. The author also discusses design tools and important techniques for creating OS data structures. While the book mainly focuses on UNIX, it also explains how to apply these concepts in other major Operating Systems. To help students and instructors, a simulator is provided to run the code that implements the operating system.

5."Operating Systems: A Modern Perspective" by Gary J Nutt

Book Review: This book combines the principles of operating system design with real-world implementation. It thoroughly explains the concepts of operating systems and includes practical code examples, algorithms, and discussions on implementation issues. The book also offers lab exercises for students to practice using Linux, Unix, and Windows. It helps students understand the distinction between operating system fundamentals and coding examples. Moreover, the book highlights the relevance of operating system components in relation to real-world constraints.

6."Design of the Unix Operating Systems" by Maurice Bach

“Design of the Unix Operating Systems” Book Review: This book simplifies the core principles of internal algorithms and clarifies how the operating system’s kernel connects with the programmer interface. Programmers can use this knowledge to write more efficient code or compare algorithms used in UNIX with those in other operating systems. The book includes exercises that range from deepening understanding to investigative tasks, with varying levels of difficulty. Some challenging exercises are marked with an asterisk. With 13 chapters, the author also explores the evolution of operating systems in an easily understandable manner.