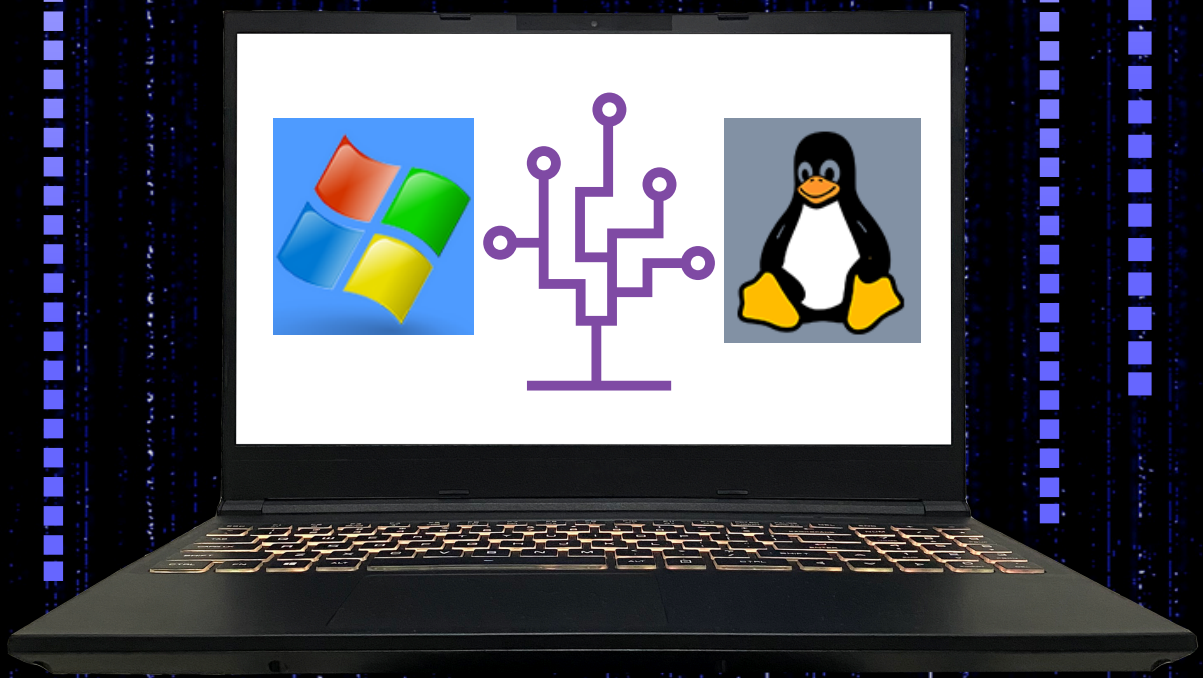
INTRODUCTION TO OPERATING SYSTEMS



Written By
JACEY MARIADASS
NORHASLINDA ABDUL KARIM

TOPIC 1
INTRODUCTION TO OPERATING SYSTEMS

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY
POLITEKNIK UNGKU OMAR

DEPARTMENT OF POLYTECHNIC EDUCATION AND COMMUNITY COLLEGES
MINISTRY OF HIGHER EDUCATION

PREFACE

This e-book is meant to serve as your guide to navigate the wide world of operating system, with the emphasize on comprehending the context in which they are functioning. The introduction of the e-book begin with the breakdown of operating system environment, continue with the analogies and explanations to help for concept understanding. Next, the e-book also provide with the practical examples that will close the gap between theoretical knowledge and hands on application. Finally, the impact of the operating system environment on user experiences is covered in this e-book, starting with basic functions of operating system and ending with system calls.

DECLARATION & DISCLAIMER

e-ISBN: 978-629-7635-39-2

INTRODUCTION TO OPERATING SYSTEM

WRITER:

JACEY MARIADASS [jacey@puo.edu.my]
NORHASLINDA ABDUL KARIM [nhaslinda@puo.edu.my]

PUBLISHER:
POLITEKNIK UNGKU OMAR
JALAN RAJA MUSA MAHADI
31400 IPOH
PERAK DARUL RIDZUAN
TEL : 1-300-88-1969
https://www.puo.edu.my

Writer

JACEY MARIADASS
NORHASLINDA ABDUL KARIM

Editor

JACEY MARIADASS
NORHASLINDA ABDUL KARIM

Illustrator

JACEY MARIADASS

WRITER'S DETAILS

JACEY MARIADASS
SENIOR LECTURER
DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY
POLITEKNIK UNGKU OMAR
jacey@puo.edu.my
012-5250603

Jacey Mariadass holds a Bachelor's Degree in Information Technology with Hons from Northern University of Malaysia (UUM) and a Master's Degree in Cyber Security from National University of Malaysia (UKM). She started her career as a lecturer in 2007 and is currently teaching in Politeknik Ungku Omar. Her subject interest revolves around Operating System, Security Audit, Information Security and etc.

NORHASLINDA ABDUL KARIM
LECTURER
DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY
POLITEKNIK UNGKU OMAR
nhaslinda@puo.edu.my
016-5417115

Norhaslinda Abdul Karim holds a Bachelor's Degree in Computer Science from University of Technology Malaysia (UTM). She started her career as a lecturer in 2010 and currently is teaching in Politeknik Ungku Omar. Her subject interest revolves around Operating System, Information Security, Security Audit and etc.

ACKNOWLEDGMENTS

We would like to express our gratitude to the following individuals for their invaluable contributions to the creation of this interactive eBook:

CONTENT REVIEWER

- Dr Nur Syadhila binti Che Lah
  Lecturer
  Computer & Information Science Department
  Universiti Teknologi PETRONAS

- Nurizah binti Mahmor
  Senior Lecturer
  Department of Information and Communication Technology
  Politeknik Ungku Omar

LANGUAGE REVIEWER

- Jeyahar Mark a/l Johnson Sundrasehgar
  Lecturer
  Department of General Studies
  Politeknik Ungku Omar

- Nishantini a/p Ganesan
  Lecturer
  Department of General Studies
  Politeknik Ungku Omar

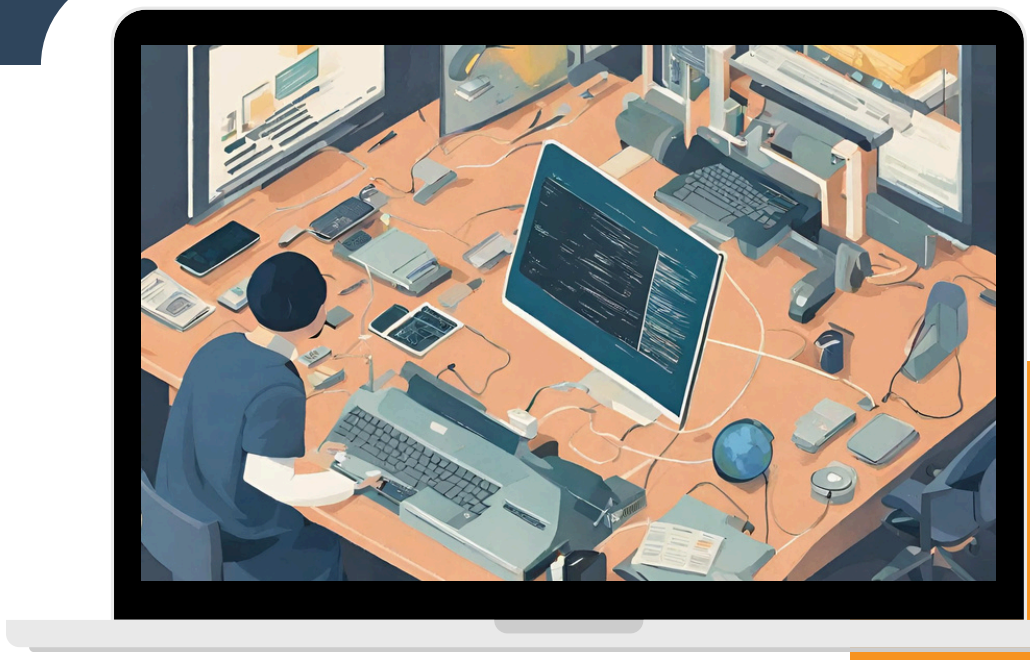
GRAPHIC REVIEWER

- Norami Mastura Binti Abdul Halim
  Lecturer
  Tunku Abdul Rahman University of Management and Technology (TAR UMT), Perak Branch

- Nurliyana Amirah Binti Nadzari
  Designer
  Unit of Instructional and Multimedia Development
  Politeknik Ungku Omar

TABLE OF CONTENTS

DEFINITION OF OPERATING SYSTEM

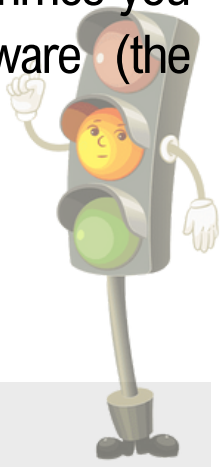


BACK TO CONTENTS PAGE

WHAT ARE OPERATING SYSTEMS

An operating system is a type of **system software that manages the way various programs are executed on a computer and serves as an interface between the user and the hardware.**

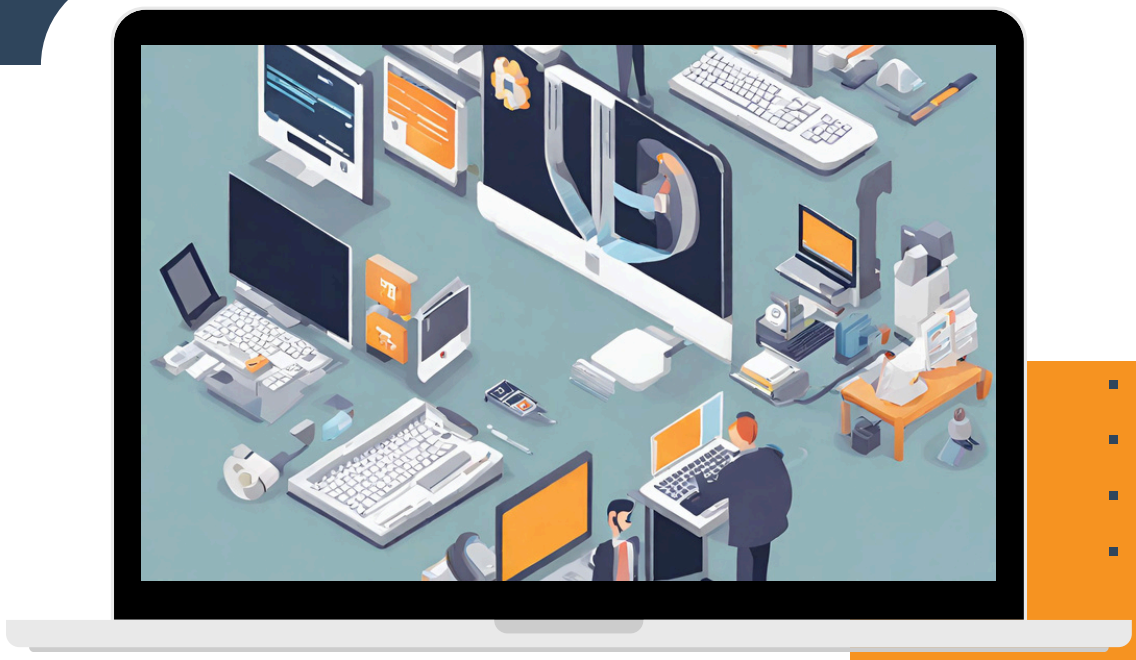
It manages the information exchange between the software (the programmes you use) and the hardware (the CPU and memory).



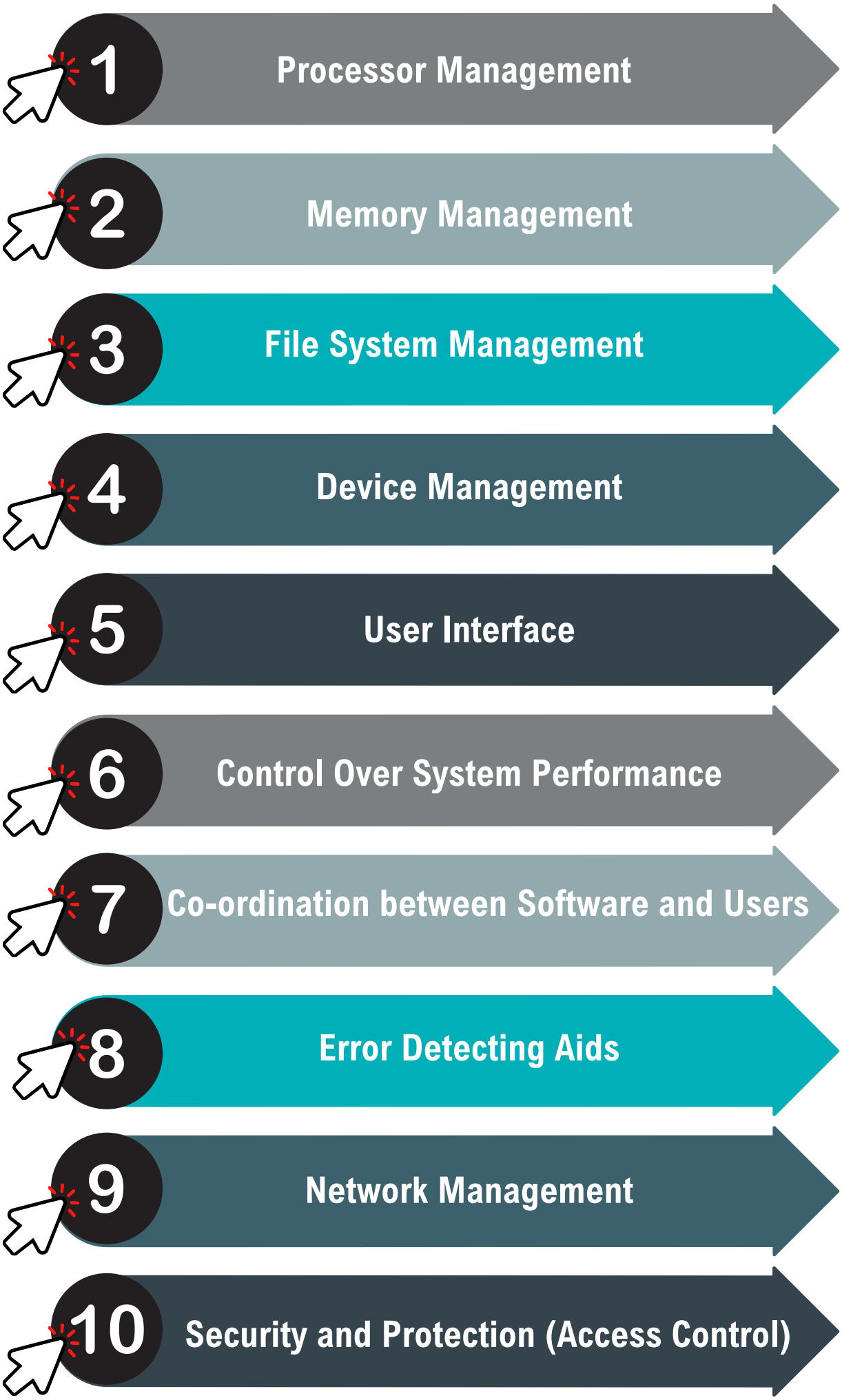
Analogy:

Just like a traffic controller keeps cars moving smoothly on the road, the operating system guarantees that everything in your computer works together effortlessly, allowing you to run applications, access the internet, and accomplish various tasks on your device.

FUNCTIONS OF OPERATING SYSTEM



BACK TO CONTENTS PAGE


1
Processor Management
2
Memory Management
3
File System Management
4
Device Management
5
User Interface
6
Control Over System Performance
7
Co-ordination between Software and Users
8
Error Detecting Aids
9
Network Management
10
Security and Protection (Access Control)

1 Processor Management

To ensure every operating will run smoothly and it determines which task or programme gets to use the CPU and for how long. Consider the operating system to be a traffic policeman directing cars at a junction.

2 Memory Management

It guarantees that each programme has sufficient space to access and storing data. Similar to how a librarian arranges and assigns shelves for books, your computer's operating system controls memory.

3 File System Management

It facilitates the creation, access, and organisation of files, images, and other types of data on your computer. The operating system arranges and maintains track of all of your files, much like a filing cabinet.

4 Device Management

The operating system interacts with a variety of devices (such as printers, keyboards, and mouse) to make sure they are compatible with your computer, functioning as a kind of worldwide remote control.

5 User Interface

Imagine your computer's user interface as a helpful tour guide. It offers a method for you to communicate with the system using either text commands or graphical icons.

6 Control Over System Performance

Consider your operating system to be the performance traffic management for your computer. Assuring that no one uses up all the resources, it organises and guides the flow of tasks. The operating system makes sure that your computer's power is used properly and efficiently to prevent lags and jams, the same way traffic lights keep cars moving smoothly.

7 Co-ordination between Software and Users

Consider the operating system as a programme translator. It facilitates communication between software programmes and hardware elements, guaranteeing an uninterrupted information flow.

8 Error Detecting Aids

The operating system detects and handles faults, much like a helpful assistant. In addition to trying to avoid crashes, it gives you alternatives or information when something goes wrong to help you resolve the problem.

9 Network Management

Consider the operating system on your computer as a helpful air traffic controller for your network connections. It guarantees that, like aeroplanes at an airport, data can take off and land without incident. The operating system assists your computer in navigating the wide network airspace while sending emails or browsing the web, ensuring that information arrives safely at its destination in a timely manner safely.

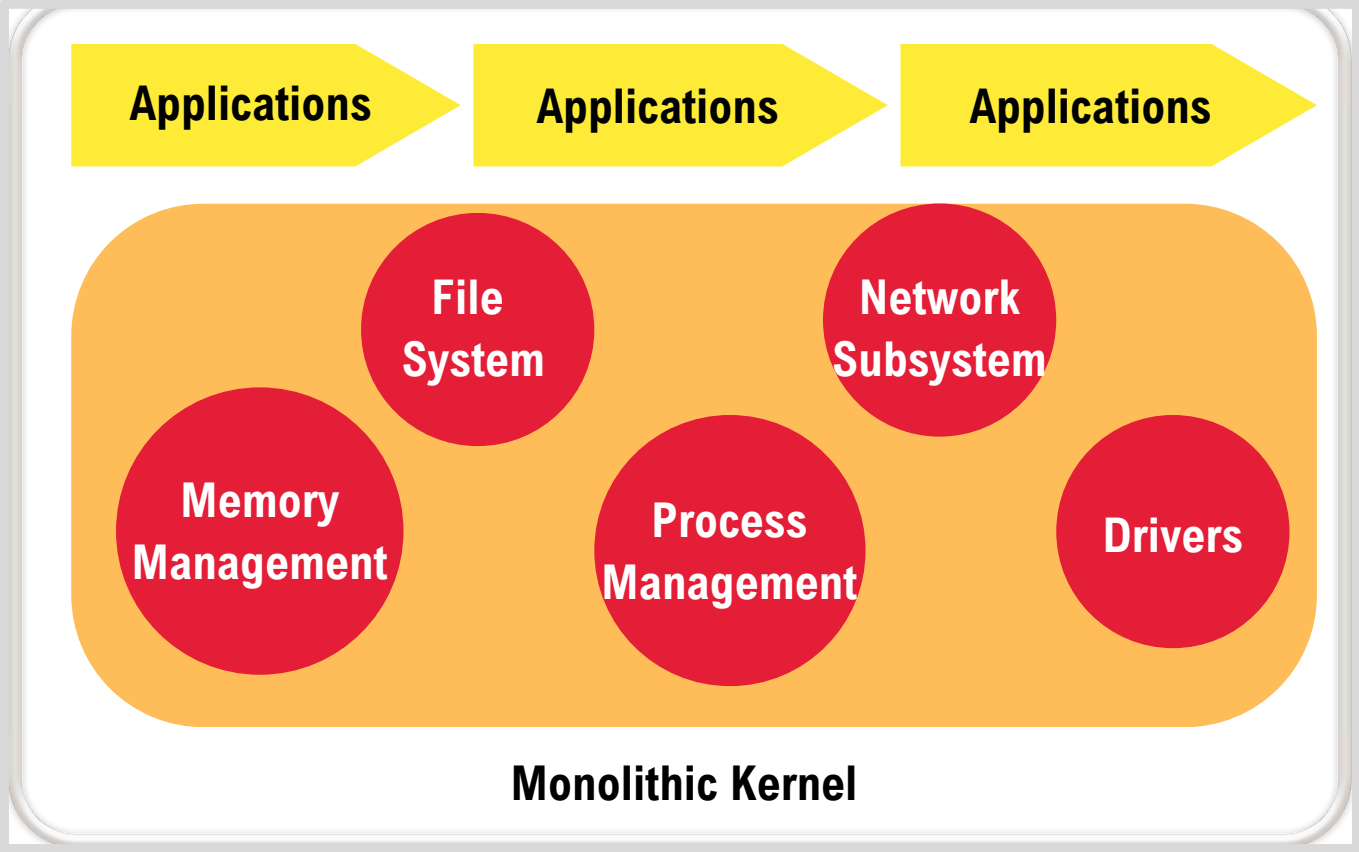
10 Security and Protection

Consider the operating system as a building's security guard. It restricts access to certain areas of the computer, making sure that only users with permission can see or alter particular files and settings.

TYPES OF OPERATING SYSTEM STRUCTURE

BACK TO CONTENTS PAGE

Monolithic Structure



- The earliest OS structure.
- In the monolithic approach, the entire operating system functions as a single program in kernel mode.
- Composed as a set of operations that are connected to form a single, sizable binary executable program.
- Any process within the system is able to call on any other procedure if the latter offers a helpful computation that the former requires.
- It is efficient to call any procedure you choose, but having thousands of procedures that are free to call each other could also result in a complicated and confusing system.
- The operating system will crash if any of these processes crashes.
- The operating system's calls can be requested with some structure by placing the arguments at a predetermined location on the stack and then carrying out a trap instruction.
- Examples of monolithic-based operating systems: Windows, Linux, MacOS.

Monolithic Structure

Advantages of Monolithic Structure

Fast Performance:
It provide better process scheduling, memory management, file management, etc.

Direct Interaction between Components:
All the components and the kernel can directly interact. It also helps in getting a better speed.

Simple and Easy:
Every part is posited in the same address's space to provide a simple and easy structure.

Better Completion of Smaller Tasks:
Works better for handling smaller tasks.

Disadvantages of Monolithic Structure

Prone to Errors:
It can generate errors and problem in the system. It is because kernel and user programs use the same address space.

Update Difficulties:
The OS code is all contained in a single, large block, providing the challenge in adding
or removing the features.

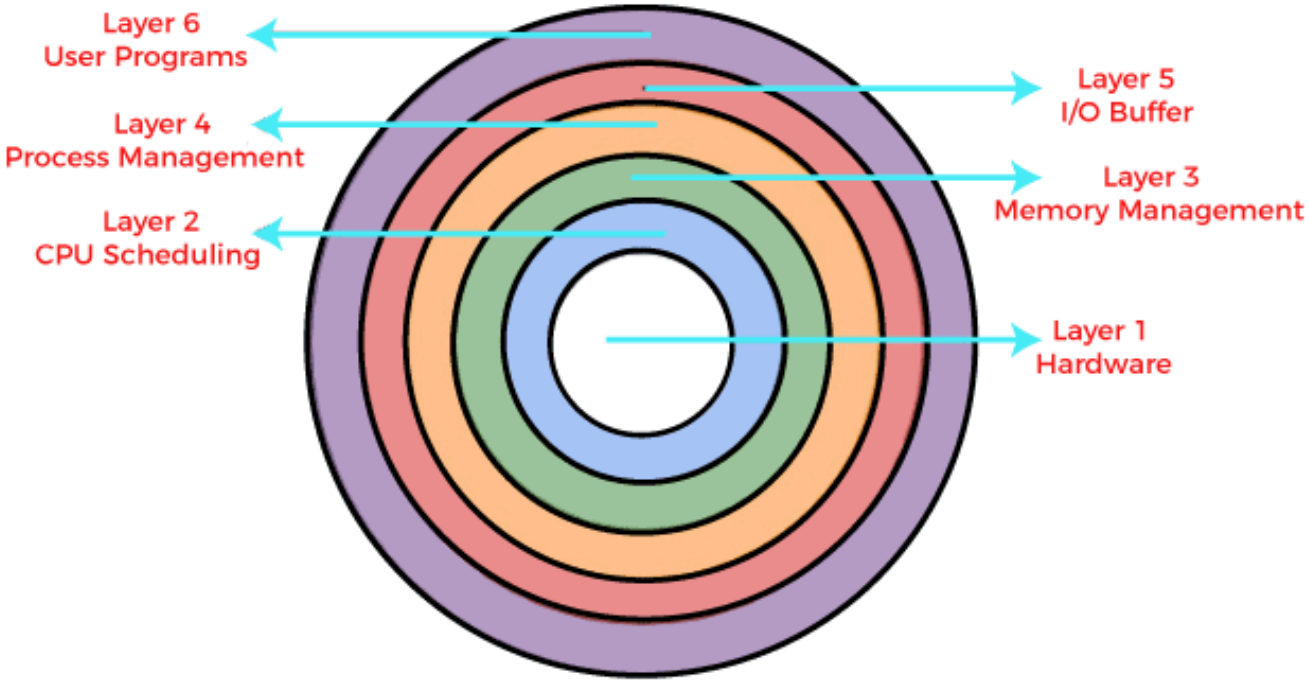
None portable:
Difficult to carry with or transfer to another system: due to the code works in a big chunk only, which requires a complete removal.

- An **analogy** would be to picture a large, single-story home with all the utilities, appliances, furnishings, and rooms contained within the same building.

- Rearranging furniture in a huge packed room is similar to the considerable alteration required to change or update one component of system.

Device drivers, file systems, and the kernel are all housed in the same address space as a monolithic operating system.

Layered Structure



- Operating system that groups the related functionality together and separates those unrelated.
- Each layer (level) of the operating system is constructed on top of a lower layer. The user interface is layer N, whereas the hardware is layer 0, at the bottom.
- Layers are chosen with modularity, only lower-level layers' functions and services are used by each layer.
- Hiding information at each layer.
- Guidelines for applying the layers are as follows:

  i. user interface layer will be the outermost layer.

  ii. the hardware layer will be the innermost layer.

  iii. certain layer has access to every layer below it, but having no access to any layer above it.
- Examples of layered-based operating systems: UNIX, THEOS, VMS

Layered Structure

Advantages of Layered Architecture

Modularity:
Promotes modularity because each layer only does its assigned duties.

Easy debugging:
The unique nature of the layers provides easy debugging steps. System developer only need to troubleshoot to the identified CPU scheduling layer.

Easy update:
A change made to one layer will have no effect on the additional layers.

No direct access to the hardware:
User will be able to use the hardware services but have no direct access or modification to the hardware. The hardware layer is the innermost layer of the design.

Abstraction:
Every layer is focused on its own set of tasks. As a result, the other layers' functions and implementations are abstract.

Disadvantages of Layered Architecture

Complex and detail implementation:
The layers must be carefully arranged, significant modularity leads to complicated implementation.

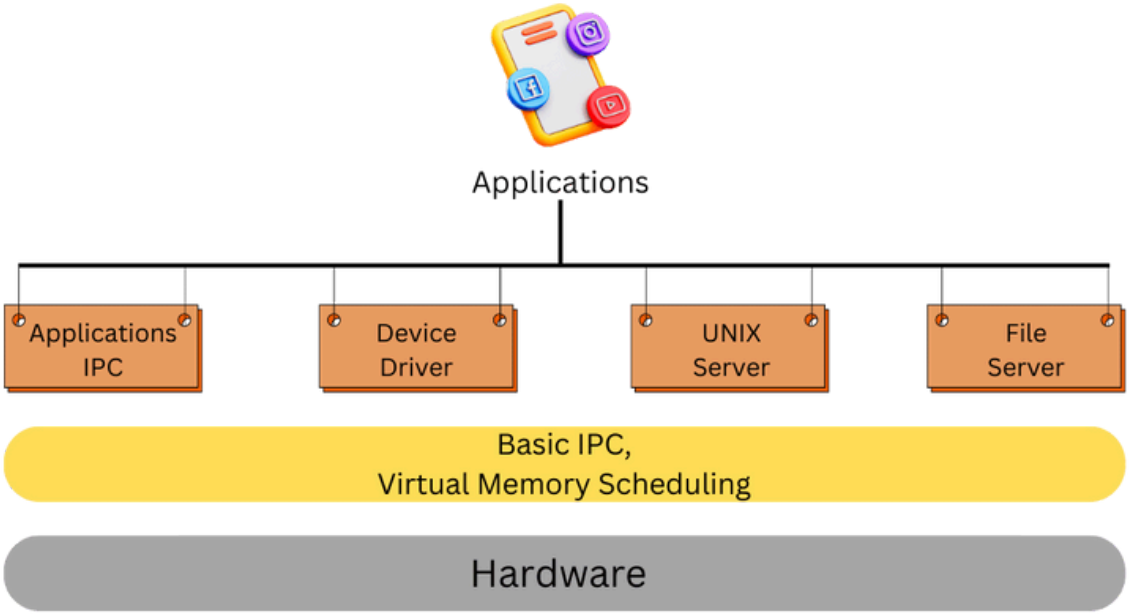
Slower execution:
Communication between the layers will need a sending request that has to pass through two levels for the completion step. This will lead to ineffective design due the additional layers to the structure.

- **Analogy** : The visualization is similar to a sandwich with variety of ingredients layers which placed on top of each other in order to produce the mouthwatering sandwiches.

- In the layered operating system, user will be able to modify or replacing each layers without impacting each other, just similar to customizing the sandwich by altering the ingredient layers.

Similar to how each component of a sandwich adds to the overall flavor and texture, each layer interacts with the layers above and below it.

Microkernel Structure



- Divide the OS into several process which implements a single set of services each.
- Clearly defined modules, only one of which the microkernel runs in kernel mode and the others as comparatively helpless regular user processes.
- The microkernel architecture aims to achieve high reliability.
- The file system and device drivers are independent user processes, a flaw in one of them can only crash the relevant component and not the system as a whole.
- Message passing, as opposed to shared memory, is commonly used for communication between various operating system components.
- More dependable and safe method of exchanging data across the processes, which help to prevent the problems and mistakes from spreading throughout the entire system
- New services and drivers may be added to the user space without altering the kernel itself, make it simpler to add new functionality to the operating system.
- More secured since there is a lower chances for user space code flaws and vulnerabilities that will impact the kernel.
- Examples of microkernel structure operating systems: MINIX, QNX, L4 microkernel and GNU Hurd.

Microkernel Structure

Advantages of Microkernel Structure

Performance: Microkernel architecture is compact and separated, it can function more effectively.

Security: Only affected components will be included that might otherwise interfere with the system's operation, microkernels are safe.

Scalable: It is easily expandable as compared to a monolithic kernel.

Modularity: With the modularity features, the Kernel will remains unaffected even when different modules are added, removed, or reloaded. System crashes is lower with distributed systems compared with monolithic. Modular system architecture can be implemented with the help of Microkernel interface.

Disadvantages of Microkernel Structure

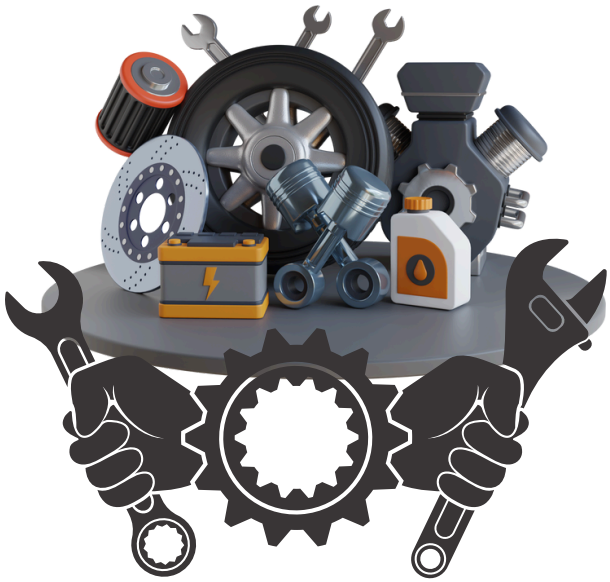
Performance overhead: Performance overhead arises due to the need for inter-process communication between kernel modules. If the drivers are implemented as procedures or processes, then a context switch or a function call is needed.

Limited hardware support: Microkernel architecture is more difficult to port to new hardware platforms.

Higher cost: Compared to a monolithic system, providing services in a microkernel system are costly.

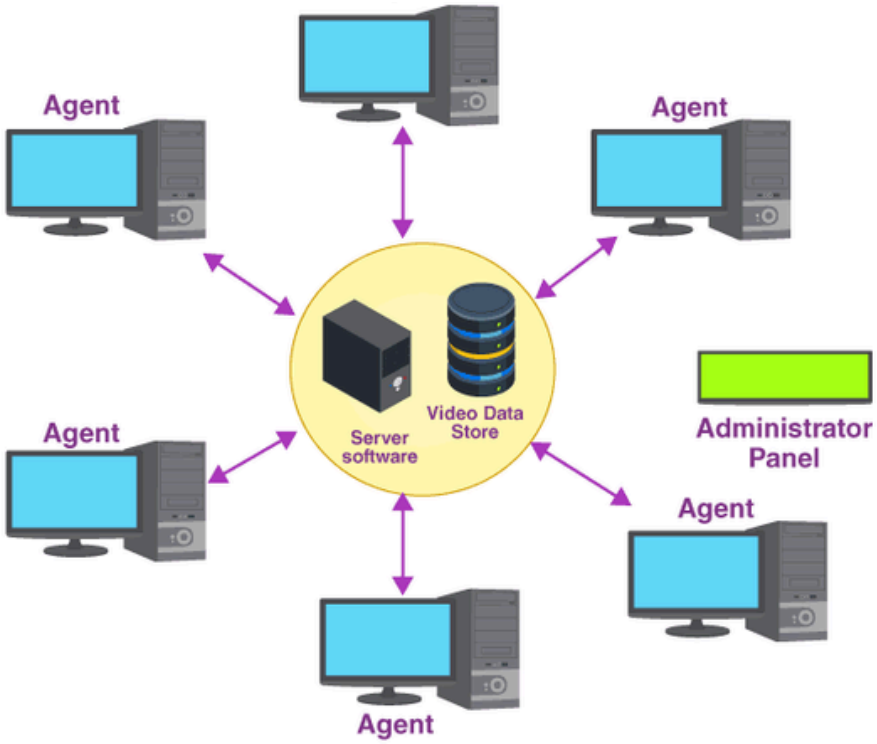
- **Analogy** : Imagine an automobile that is modular and easily adaptable. Individual parts like the engine, transmission, or entertainment system can be swapped out.

- Essential functions like memory management and scheduling are kept to a minimum in microkernel operating system. Extra features like file systems and device drivers are implemented as distinct user-space processes.



More flexibility and simpler maintenance are made possible by this structure, which is comparable to changing individual automotive parts without having to replace the complete vehicle.

Network Structure



- Includes methods and software enabling network-based communication with other independent computers.
- Practical and economical.
- Shared devices, such as hard discs, printers, and other peripherals can be shared between computers.
- Network operating system is installed on top of each individual machine in the network, each of which runs its own operating system.
- Each machine's operating system permits it to access resources from other computers and each machine needs to log in to another machine with the correct password.
- Stops processes from transferring across machines and from interacting.
- Transmission control protocol is a widely used network protocol.
- Operating systems for networks can be customized to functions such as:
    a. Peer-to-Peer System
    b. Client-Server System

Peer-to-Peer

- Every node is identical to every other node in terms of functionality and functioning.
- Each has its own local memory and resources, and they can all carry out comparable tasks.
- They are able to interact, connect, and exchange resources and data with one another.
- By utilizing the network's authentication capability, a node can converse and exchange information with a distant node.
- Using a switch or hub, O.S. nodes are directly connected to one another within the network.

Client-Server System

- Has several client PCs connected to a single server.
- Operates on the client computer, and the server computer is equipped with the Network Operating System.
- Every client's computer is concentrated around the server machine.
- Requests for data or resources are created by the client computers and sent to the server computer.
- Server machine responds to the client machine by securely giving it the necessary services. It is an extremely strong computer that can handle complex computations and operations.
- Possess the capacity to manage the resources and the entire network.
- Can be multiprocessing in nature, which can process multiple client requests at the same time.

Network Structure

Advantages of Network Structure

- High stability due to centralized server.
- Provides good security.
- Upgradation of new technology and hardware can be easily implemented in the network.
- Provides remote access to servers from different locations.

Disadvantages of Network Structure

- Dependent on the central location to perform the operations.
- High cost of buying a server.
- Regular updates and maintenance are required.

ARCHITECTURE OF OPERATING SYSTEM

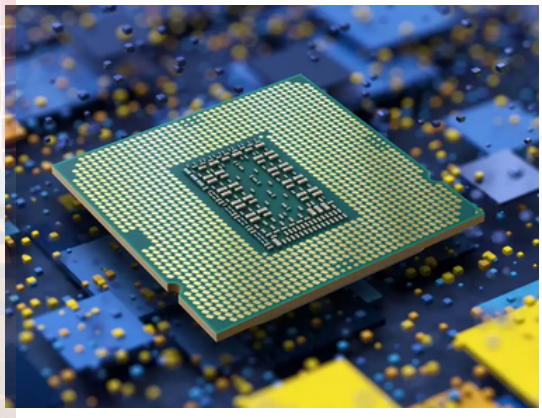
BACK TO CONTENTS PAGE

Single Processor System

- A single processor.
- Known as a Central Processing Unit (CPU).
- Hardware component within a computer system.
- Responsible for executing instructions.
- Primary unit for carrying out the majority of tasks within a computer.
- Only one CPU handles all computational tasks, including executing software instructions, managing data processing, and interacting with other hardware components.
- Common in personal computers, laptops, and many other computing devices.

Analogy:

- Imagine a small kitchen with just one chef preparing meals.
- The chef handles all the cooking tasks, from chopping vegetables to grilling the meat and plating dishes.
- Only one central processing unit (CPU) is responsible for executing instructions and managing system resources.

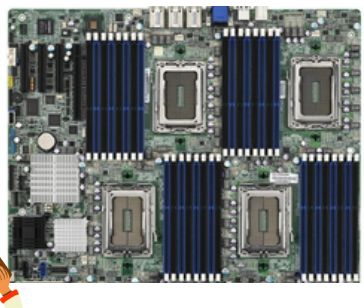


Multiprocessor System

- Known as a parallel processing system.
- Contains more than one central processing unit (CPU) working together to execute tasks and process data simultaneously.
- Multiprocessor systems distribute tasks across multiple CPUs.
- Allowing for increased computational power and efficiency.
- Each CPU typically operates independently.
- They can communicate with each other and share resources such as memory, input/output devices, and system buses.
- Tasks can be divided among the processors, with each CPU handles a portion of the workload.
- Perform tasks faster and handle more complex computations than single processor systems.
- Commonly used in servers, high-performance computing clusters, supercomputers, and large-scale computational power is required.

Analogy:

- Imagine a large restaurant kitchen with several chefs working together.
- Each chef specializes in a particular cuisine or task, such as chopping vegetables, grilling meat, or preparing desserts.
- They collaborate and coordinate their efforts to efficiently prepare multiple dishes simultaneously.
- Multiple CPUs or processor cores are working together to execute tasks and manage system resources.

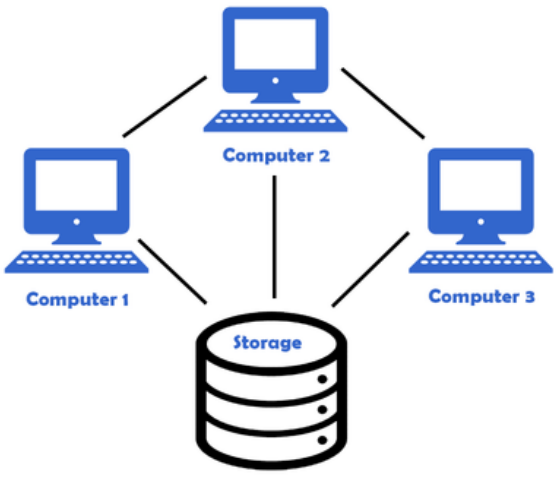


Clustered System

- Multiple individual computer systems or processing units (which could be single processors or multiprocessor systems) are interconnected to work together as a single unit.
- Provides fault tolerance and high availability.
- Often used in high-performance computing (HPC) environments, scientific computing, and data-intensive applications.
- Can involve parallel processing, distributed computing, or both.
- Uses specialized software for task distribution, load balancing, and fault tolerance.
- Can be implemented using various interconnection technologies such as Ethernet, InfiniBand, or other high-speed networking solutions.

Analogy:

- Consider a food truck festival where several food trucks are parked together, each serving a different types of cuisine, such as Mexican, Italian, or Asian.
- While each food truck operates independently, they collaborate to provide a diverse range of food options to customers.
- Similarly, in a clustered system, multiple independent computers (nodes) work together to provide a unified computing environment, with each node contributing its resources to process tasks and provide services to users.



Clustered System

PRODUCT OF OPERATING SYSTEM

BACK TO CONTENTS PAGE

Closed Source System

more security threats due to widespread used

user code kept secret

easy to use

requires a paid license

e.g :
Microsoft Windows

Open Source System

better security with great community support

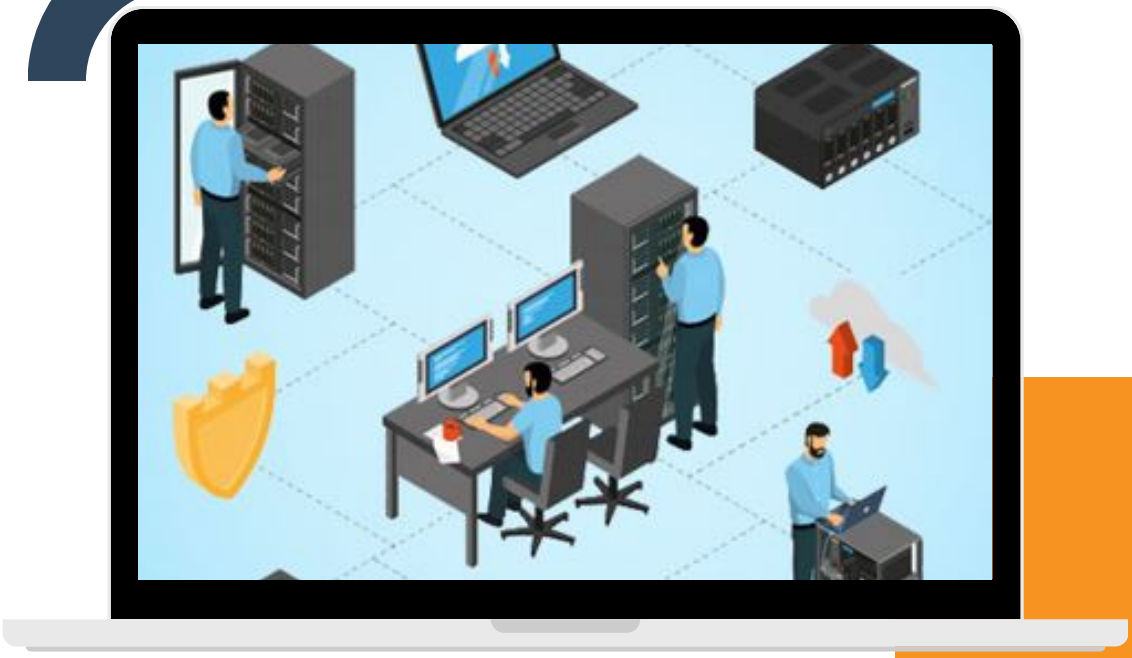
user code is freely-distributed

requires extensive knowledge

free to use

e.g :
Linux

CONCEPTS IN RELATION TO OPERATING SYSTEM



BACK TO CONTENTS PAGE

Multitasking

Description

The central processing unit (CPU) of a single computer can execute several tasks or processes in one time when it is assigned in multiple relation. Users can switch between tasks almost simultaneously because the operating system splits CPU time across multiple processes.

Analogy

Imagine the role of a chef in a restaurant. Managing several cooking activities at once is similar to multitasking within an operating system. The soup is simmering and the cake is baking in the oven while you chop the vegetables. The chef moves between duties with efficiency, ensuring that everything goes according to plan.

Multiprogramming

Description

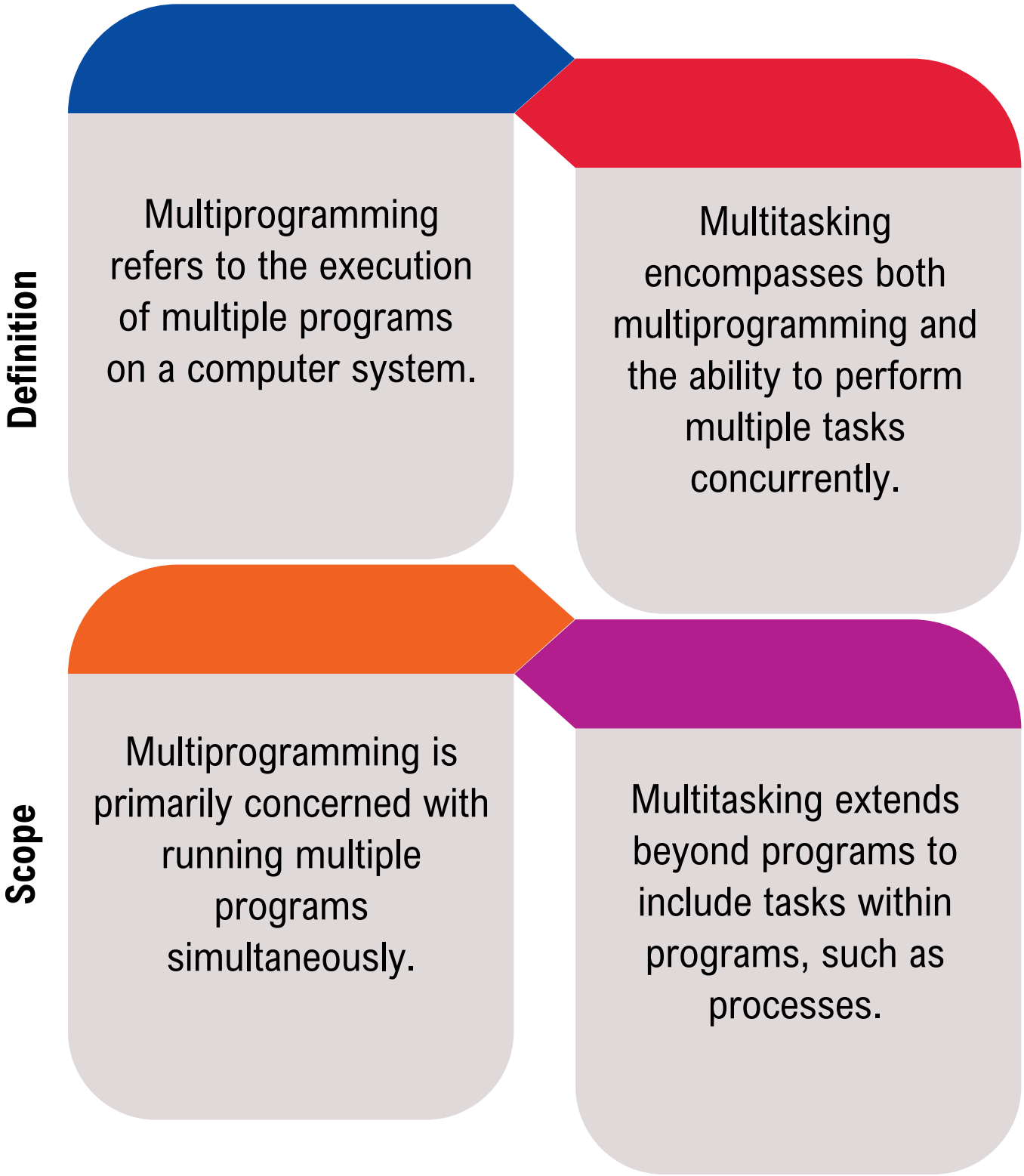
The continuous loading programmes into the computer's memory and sharing the CPU during execution is known as multiprogramming. The operating system controls how these programmes are executed, making effective use of the resources available to the system.

WATCH VIDEO

Analogy

Multiprogramming is similar to a student managing different homework assignments at the same time. Instead of completing one assignment before starting the next, the student might switch between subjects, making progress on each task. This approach allows the student to stay productive and utilize their study time efficiently.

Key differences



Time Sharing

Description

A time sharing system allows for the simultaneous operation of many different apps. As a result, it would appear that they are all using the CPU simultaneously.

Multi-user :

One way is to allow many individuals to engage with a computer system at once. The operating system assigns one particular user to each of the tiny time slices that make up the CPU time. This makes it appear as though each user is using the CPU concurrently, even though it is shared.

Single-user :

Individual utilising a single computer. The user will be executing several programmes simultaneously. Again, time slicing is used by the CPU.

Analogy

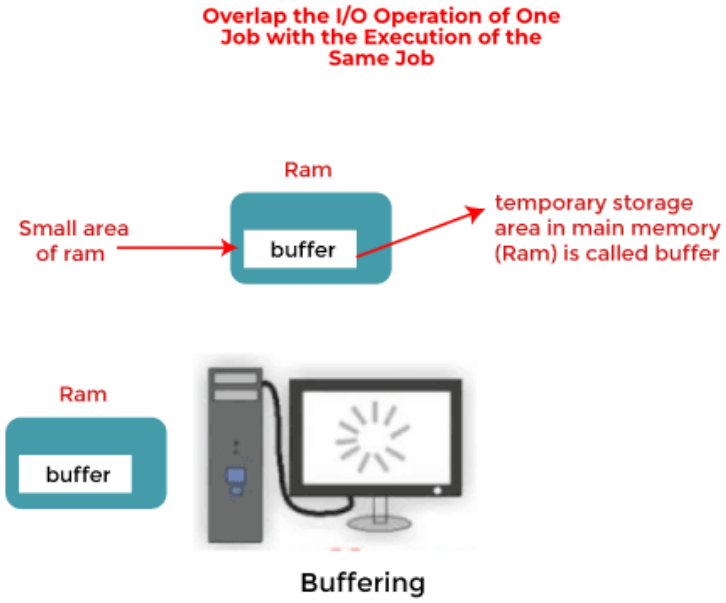
Imagine a common playground. An operating system's time-sharing mechanism is similar to multiple kids enjoying the swings, slides, and other play areas in turns. To guarantee that every child has an equal opportunity to utilise the resources, each child is given a fair quantity of playtime.

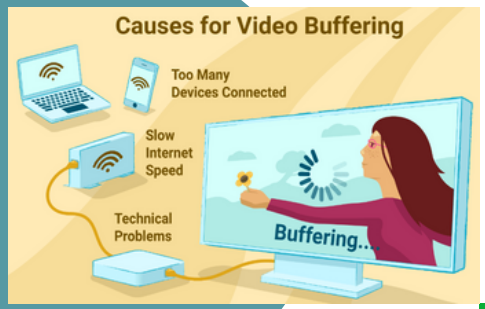

Multitasking is like a person handling different tasks at the same time.
Multiprogramming is about managing multiple programs concurrently.
Time sharing is about allocating time slices fairly among users.

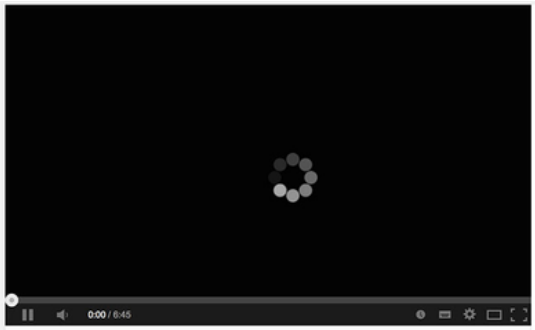
Buffering

Description

Buffering involves temporarily storing data in a reserved memory area, known as a buffer. This ensures an ongoing and uninterrupted flow of data by allowing for differences in speed between devices.







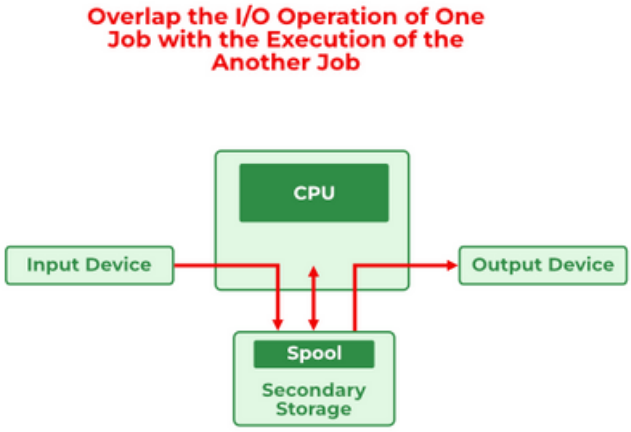
Analogy

When you watch YouTube videos or live streams, you experience buffering. A buffer in a video stream is the quantity of data that needs to be downloaded before the viewer can see the video in real time.

Spooling

Description

Spooling functions similarly to a computer assistant. Because it arranges jobs in a queue, your computer will be preparing for the next task while it is working on one. It helps to keep everything organised and runs more smoothly, much like a to-do list.



Analogy

Imagine a secretary managing a queue of documents to be typed. Spooling is similar to the secretary placing each document in a queue (spool) and then sending them to the printer one at a time. This guarantees that the printer will continue to function when the secretary is ready to create the next document.

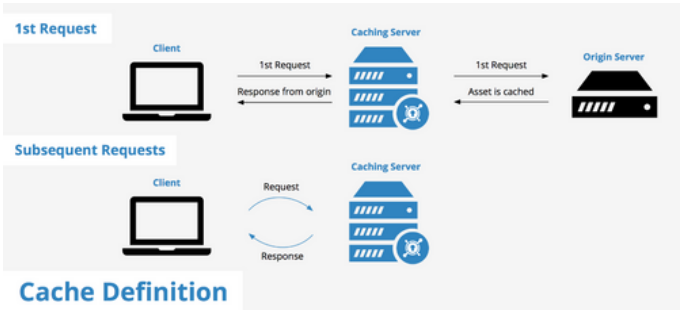
Buffering vs Spooling

Buffering	Spooling
The buffer is a limited area in the main memory.	Spool uses the disk as a huge buffer.
Buffering handles I/O of one job along with its computation.	Spooling can handle the I/O of one job along with the computation of another job at the same time.

Caching

Description

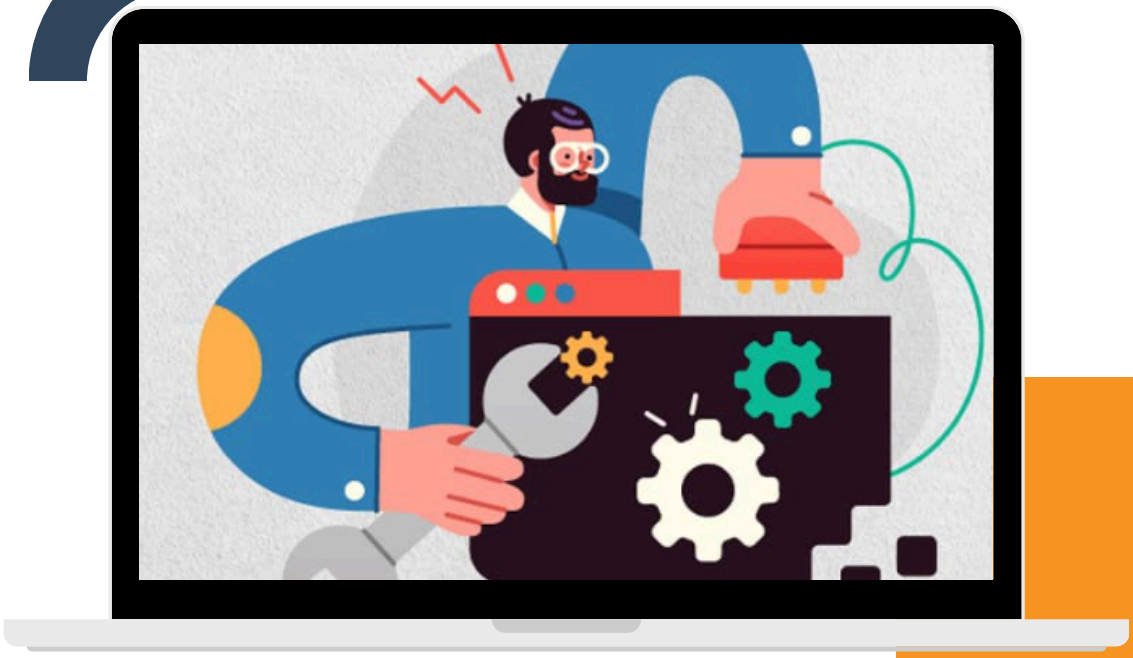
Caching is a way to store frequently used information in a fast-access place. When the computer needs that data, it checks the cache first for quick access. If the data is in the cache, it's retrieved faster; otherwise, the computer fetches it from the original source.



Analogy

Caching is like keeping your favorite snacks on a nearby shelf for quick access. This will enable you to grab your favorite snack easily from the nearby shelf instead of going to the store every time.

COMPONENTS OF OPERATING SYSTEM

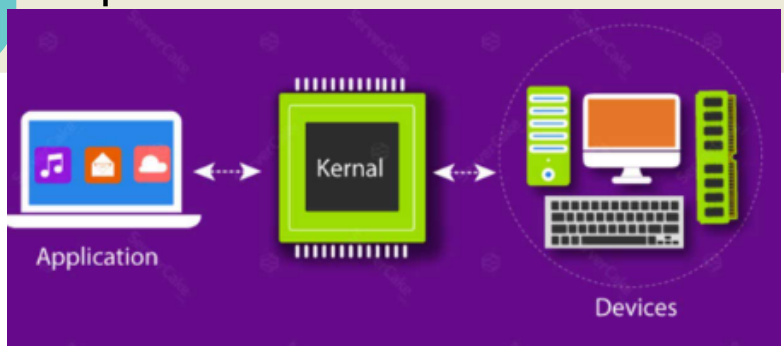


BACK TO CONTENTS PAGE

Kernel

Explanation

The kernel functions similarly to the brain of the operating system. It manages important operations and has direct communication with the computer hardware.



Analogy

Consider the kernel to be the orchestra's conductor. It leads and organises all of the musicians (hardware and software) to ensure that they play in harmony and achieve the intended outcome.

Shell

Explanation

The shell functions as a user-kernel interface. It transforms user commands into actions that the kernel can understand by interpreting them.

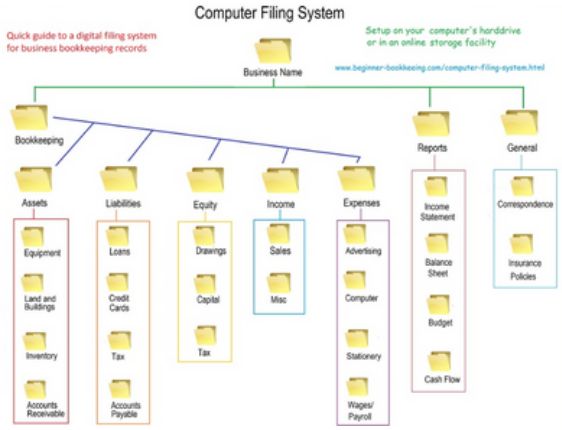
Analogy

Consider the shell to be a hotel receptionist. In order to meet the requirements of the users, users (guests) communicate with the receptionist, giving instructions or requests, which the receptionist then forwards to the relevant departments (kernel).

File system

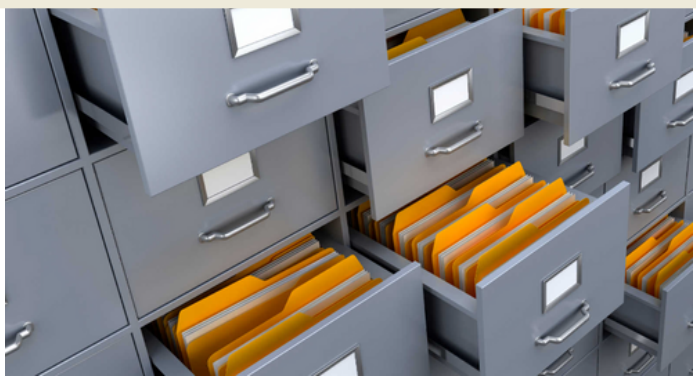
Explanation

The file system functions similarly to the organisational structure that maintains and tracks files on a computer. It defines how data is stored, retrieved, and organised on storage devices.

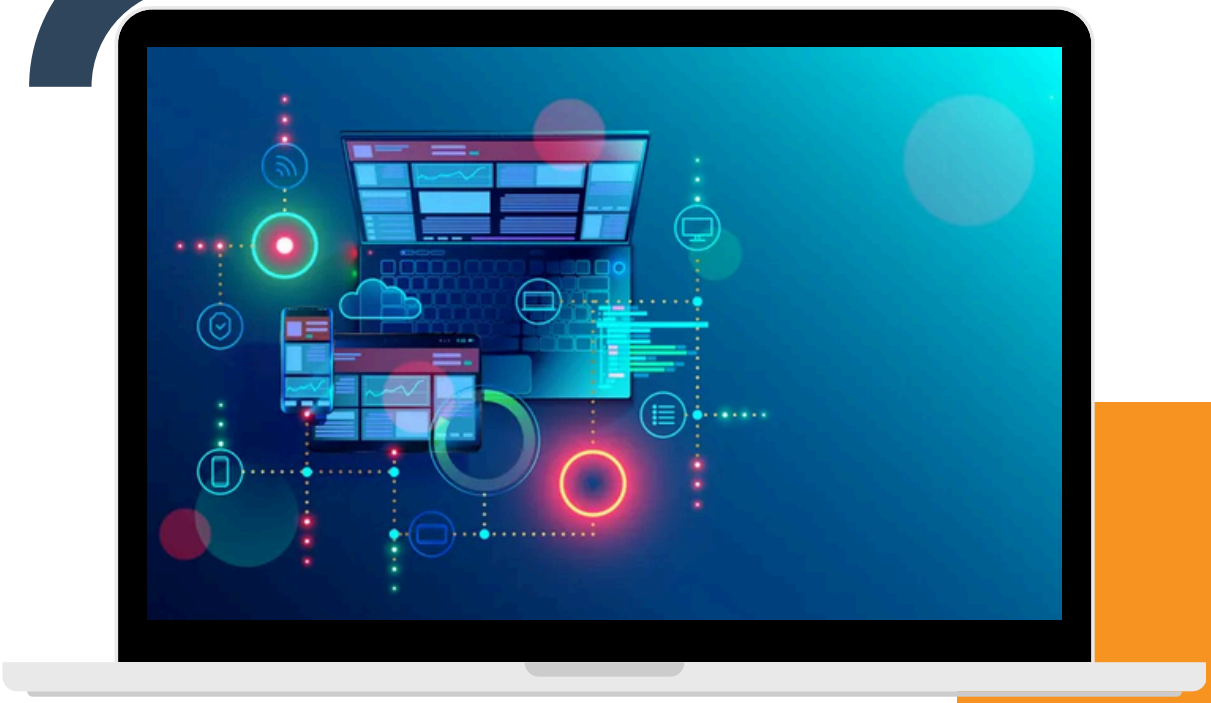


Analogy

Consider the file system to be an office filing cabinet. Files are arranged alphabetically by folders inside each drawer, which serves as an illustration for a storage unit. Locating, accessing, and effectively managing your papers are guaranteed by the file system.

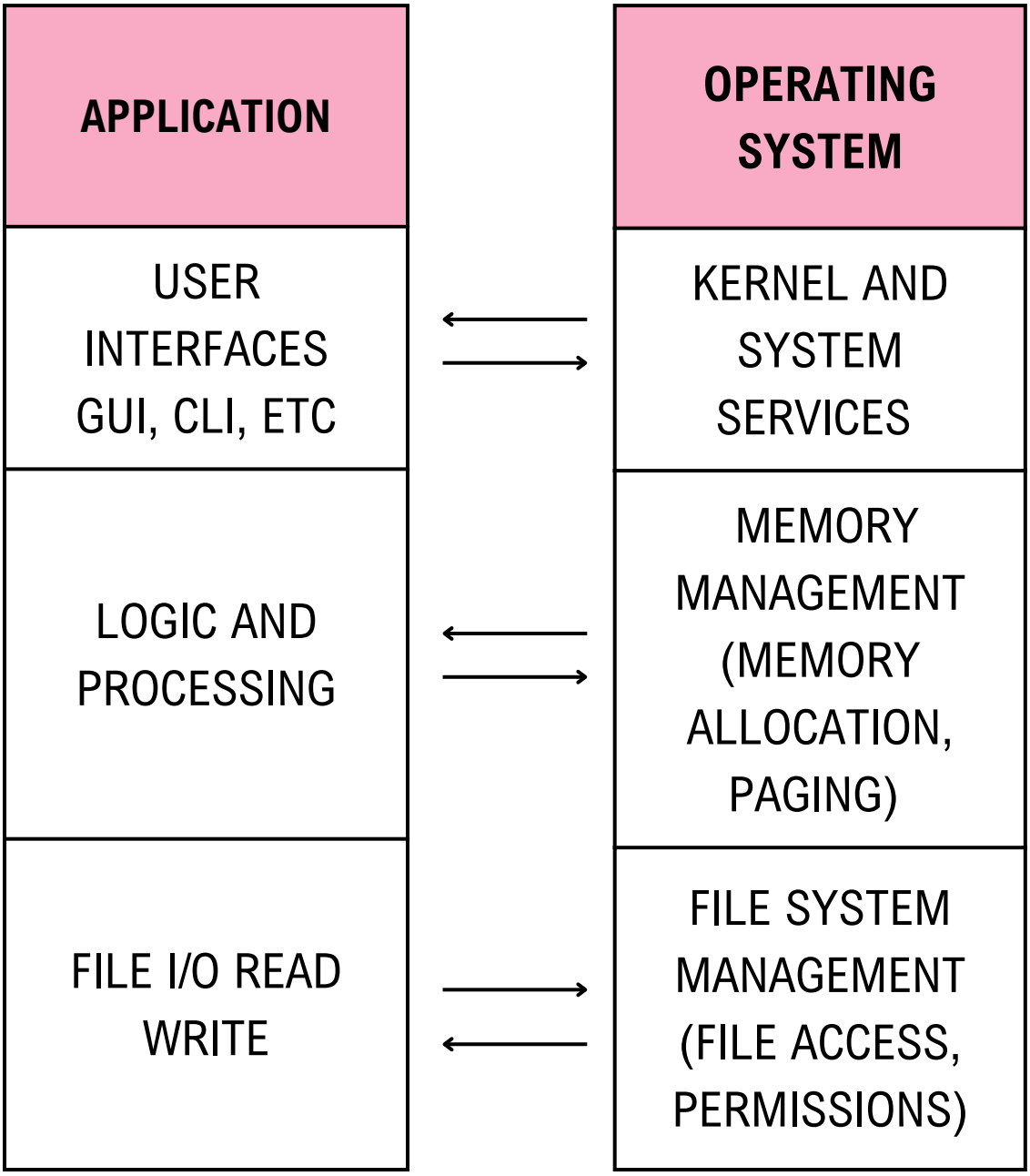


INTERACTION BETWEEN APPLICATION & THE OPERATING SYSTEM



BACK TO CONTENTS PAGE

Application & Operating System



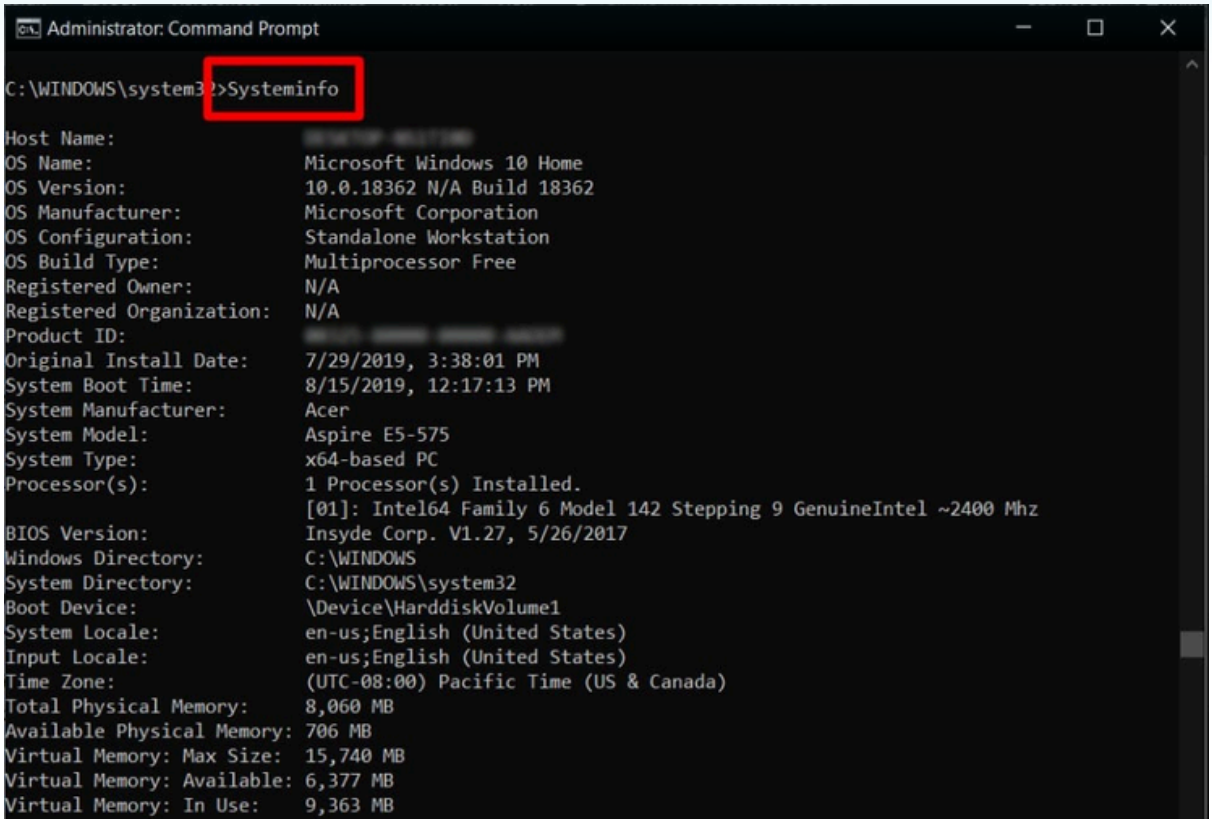
The interaction between applications and the operating system involves various system calls, API invocations, and service requests initiated by applications to perform tasks such as memory allocation, file I/O operations, process management, and more. The operating system, in turn, manages these requests, enforces security policies, and provides the necessary resources and services for applications to execute effectively.

INTERFACES OF OPERATING SYSTEM

BACK TO CONTENTS PAGE

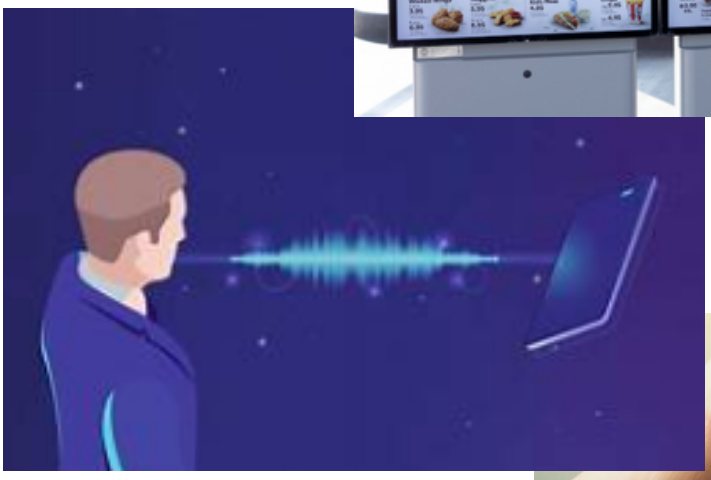
Command Line

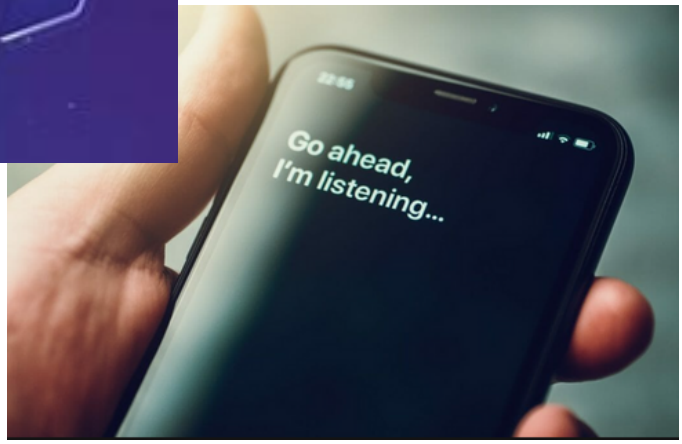
- The command line is a text-based interface where users interact with a computer by typing commands.
- It is a straightforward and effective method of communicating with the operating system or applications.



Voice-Activated

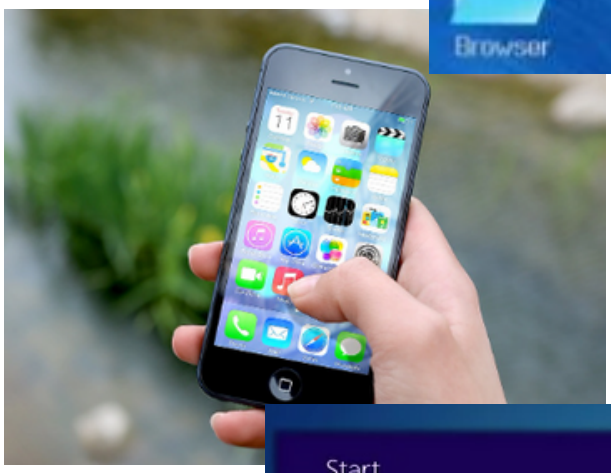
- Voice-activated systems enable users to give commands to control devices.
- Voice recognition technology processes spoken input and performs the appropriate actions.





Graphical User Interface (GUI)

- GUI enables users to interact with software or a computer using graphical elements like windows, buttons, and icons.
- It offers a simple and easy approach to navigate and carry out tasks.



Web Form

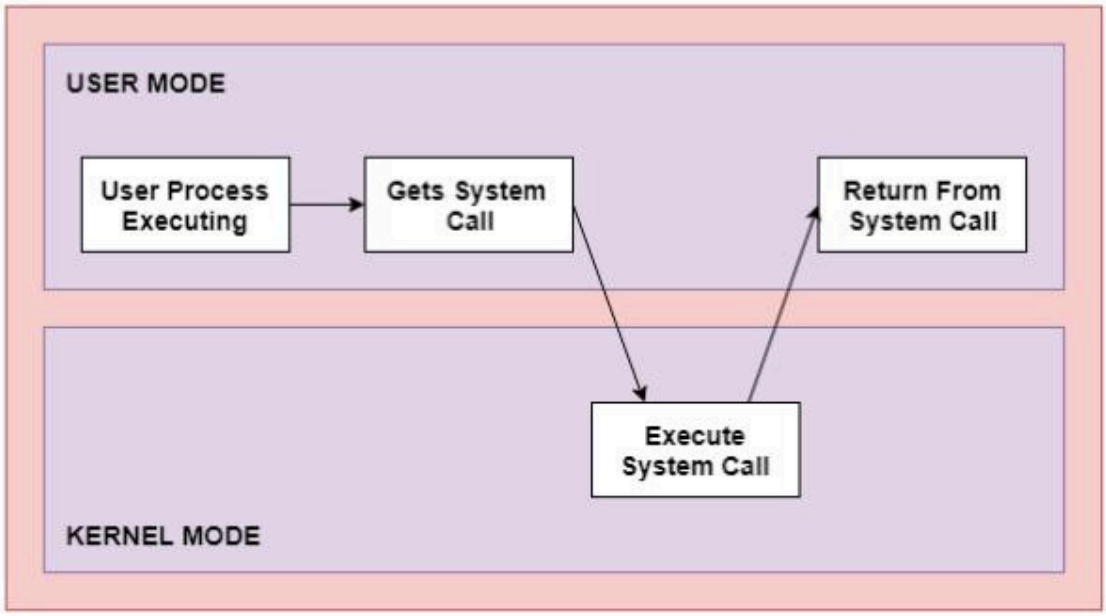
- The interface on a webpage that lets users enter data or submit information is called a web form.
- Fields, checkboxes, and buttons are usually included to allow users to submit data to a server.

Laman Web Rasmi Politeknik Ungku Omar | Sistem ePortal

RELATIONSHIP BETWEEN SYSTEM CALLS AND APPLICATION PROGRAMMING INTERFACE (API)

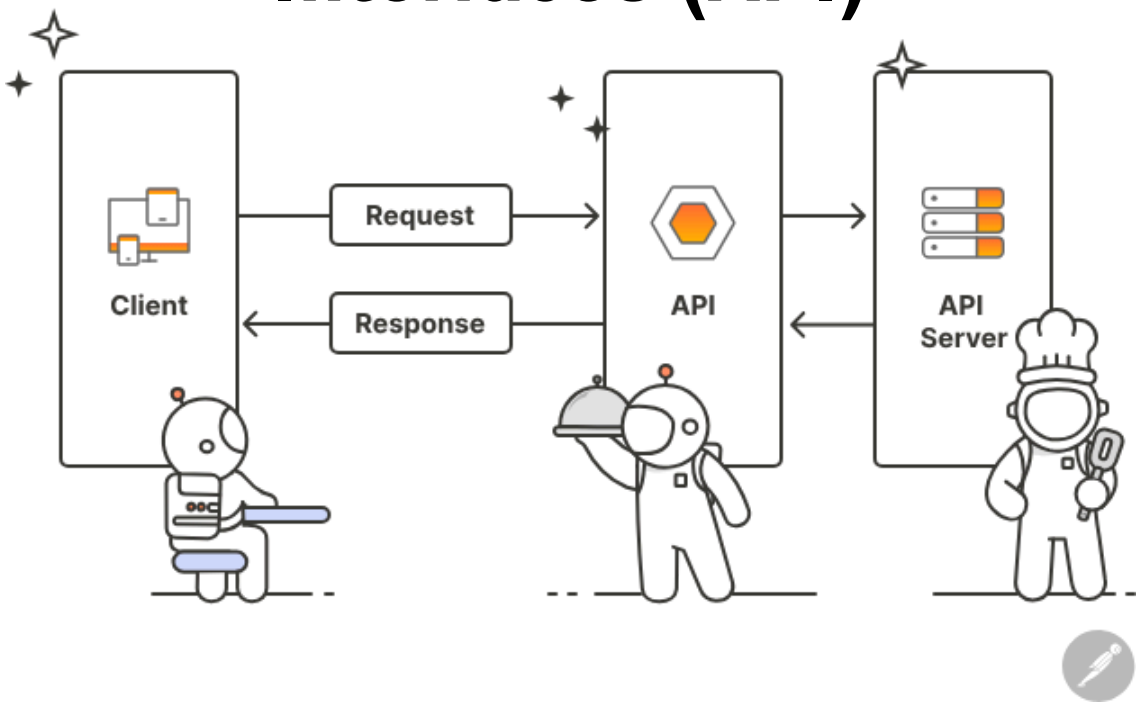
BACK TO CONTENTS PAGE

System Calls



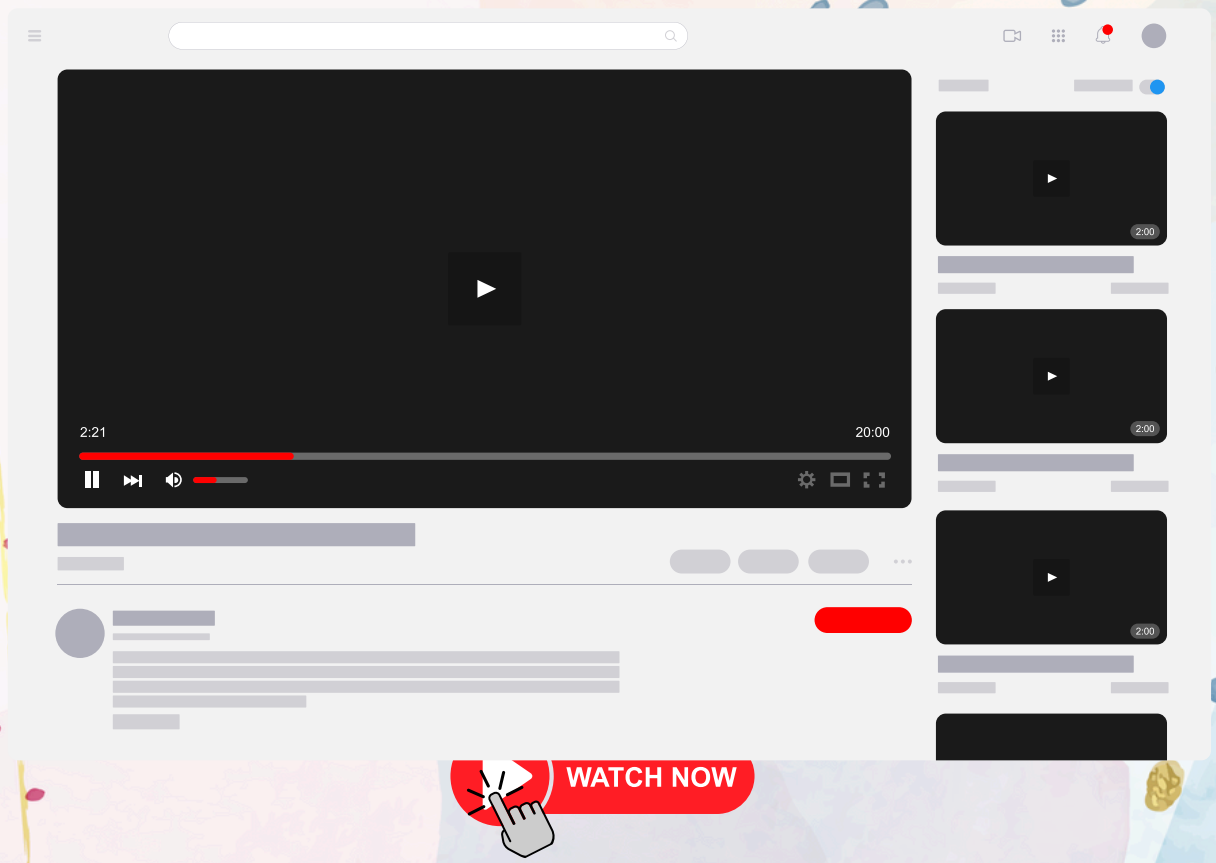
- Interfaces provided by the operating system to allow user-level processes (applications) to request services from the kernel.
- Involves operations that require privileged access or interaction with hardware resources, such as file I/O, process management, networking, and memory management.
- Provides a way for user-level processes to communicate with the operating system kernel and request actions or resources that are managed by the operating system.

Application Programming Interfaces (API)



- Set of defined methods, functions, and protocols that allow one software application to interact with another software component, such as an operating system, library, or service.
- Abstracts away the underlying implementation details and provides a standardized interface for developers to use when building software applications.
- Can be provided by various entities, including operating systems, programming languages, libraries, frameworks, and web services.

WHAT IS APPLICATION PROGRAMMING INTERFACES (API)?



Relationship between System Calls and API

- System calls are the mechanism through which user-level applications interact with the operating system kernel.
- APIs often include bindings or wrappers around system calls to provide higher-level abstractions and make it easier for developers to work with the underlying system functionality.
- APIs abstract the complexity of system calls and provide a more user-friendly interface for developers to utilize operating system services.
- When developers use an API provided by the operating system or a library, behind the scenes, the API may make one or more system calls to the operating system kernel to perform the requested operations.
- System calls and APIs work together to facilitate communication and interaction between user-level applications and the underlying operating system or software components.
- System calls provide the low-level interface for accessing operating system services, while APIs offer higher-level abstractions and convenience for developers when building software applications.

An **analogy** for the relationship between system calls and an Application Programming Interface (API) is when a user uses a smartphone app to communicate with a smart home system.

1 OS - SHA

An Operating System is represented by the Smart Home Automation (SHA), which is responsible for managing and controlling the system's different resources and functionalities.

2 API

The smartphone app represents the API, which serves as an intermediary that allows users (applications) to interact with the Smart Home Automation (operating system) in a standardized and simplified manner.

3 SYSTEM CALLS

When the user interacts with the app by tapping buttons or issuing commands, the app translates these actions into specific requests or instructions (system calls) that the smart home system can understand and execute.

4 SYSTEM CALLS - API

Similar to this, when an application uses the API, it makes calls or requests to particular operating system functions or services (called system calls) in order to carry out operations like file input/output, process management, or network connection.

Relationship between System Calls and API

WATCH NOW

TUTORIAL TIME!

Instructions: Answer all questions.

Activity outcomes | The basic functions of operating system.

Question : List **FIVE** (**5**) operating system functions.

i. _____

ii. _____

iii. _____

iv. _____

v. _____

Activity Outcomes | Identify the different interfaces of operating systems.

Question: Complete the table given.

Operating System Interfaces	How it works?	Example
a) Command line		
b) Voice actuated		
c) Graphical user interface		
d) Web form		

Activity Outcomes | Describe the following concepts in relation to the operating system.

Question : Match all the given concepts with the correct definitions.

Concepts	Definitions
Multitasking	The sharing of a computing resource among many users by means of multiprogramming and multitasking.
Multiprogramming	Send data that is intended for printing or processing on a peripheral device to an intermediate storing.
Time sharing	The concept of loading many programs at one time to share a single CPU.
Buffering	Is a process where a part of RAM used for temporary storage of data that is waiting to be sent to a device; when transferring data between devices or programs operating at different speeds.
Spooling	The ability of a computer system to time share it's (at least one) CPU with more than one program at once.
Caching	Is a process where the component transparently stores data so that future requests for that data can be served faster.

Activity Outcomes | The various product of operating system.

Question : List **THREE** (**3**) differences between a closed source system and an open source system.

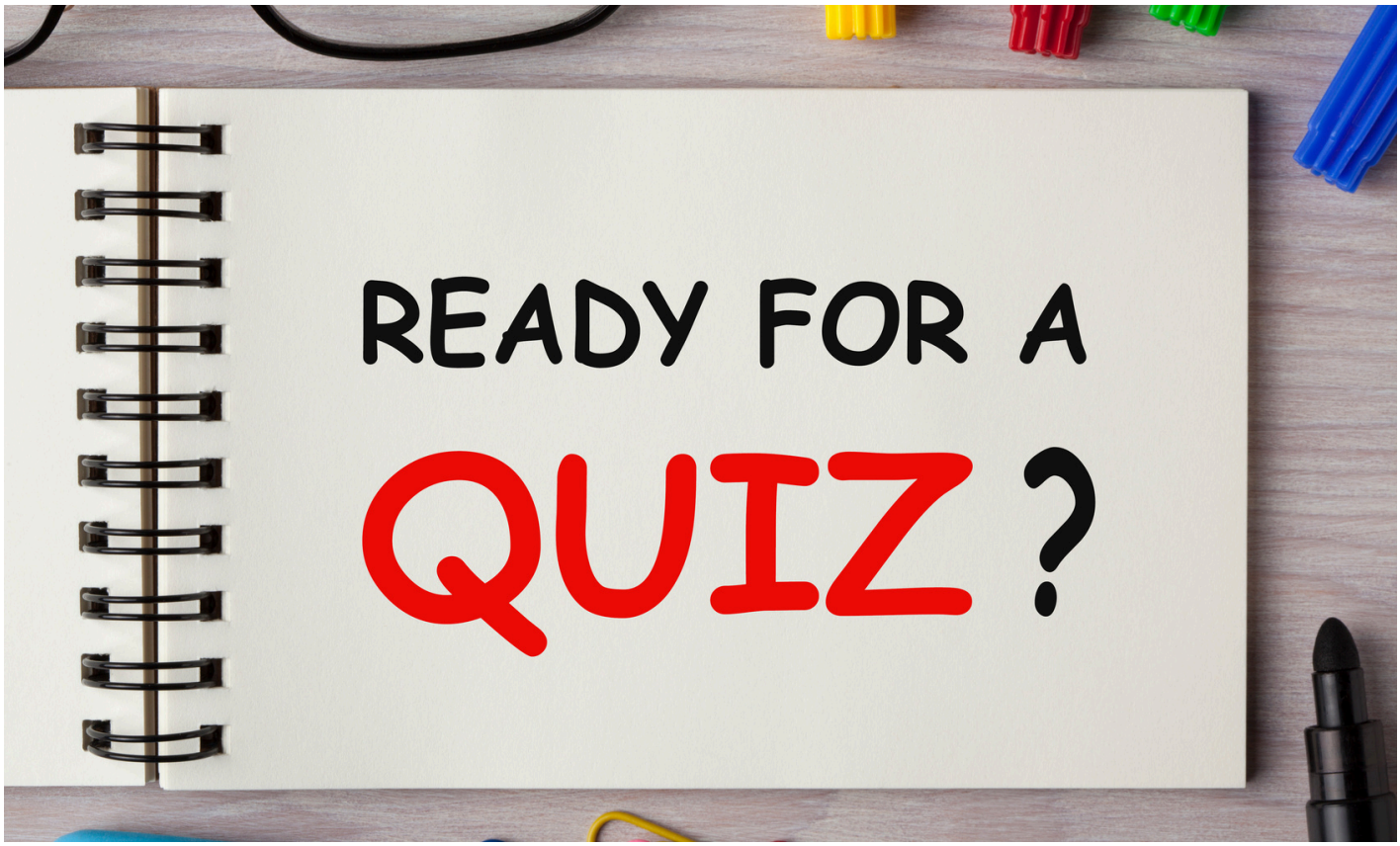
Closed source system	Open source system

https://bit.ly/OS_T1_TutorialAnswer

TO GET YOUR TUTORIAL

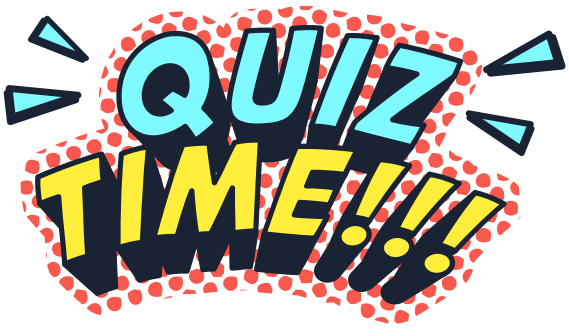
E-QUIZ

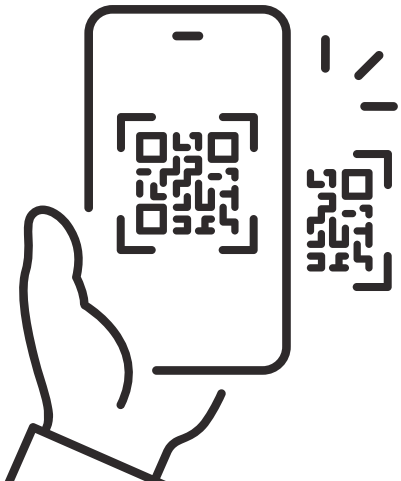
TOPIC 1
INTRODUCTION TO OPERATING SYSTEMS



E-certificate will be provided if the quiz score is above than 80%

SCAN ME





STRUCTURED QUESTIONS

QUESTION 1

Describe **ONE** (**1**) of the operating system architecture.

[3 marks]

QUESTION 2

List **THREE** (**3**) components of an operating system.

[3 marks]

QUESTION 3

State **THREE** (**3**) interfaces of an operating system.

[3 marks]

QUESTION 4

Identify **THREE** (**3**) differences between a closed source operating system and an open source operating system.

[6 marks]

QUESTION 5

Identify the concept in operating system for X, Y and Z based on the situations given.

Situation 1 : A streaming video is loading because of different Internet speed rate of client PC and server.
Concept : _____

Situation 2 :

A technique developed to speed up the input and output process among different rates of devices such as between printers and computers.
Concept : _____

Situation 3 :

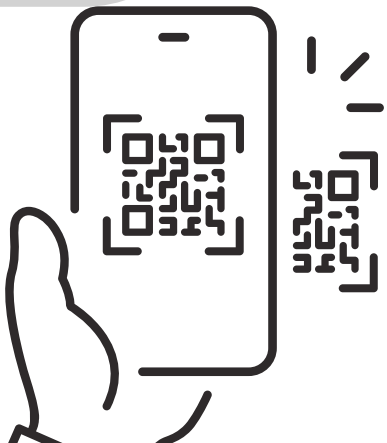
A process which stores data so that future requests for that data can be served faster.
Concept : _____

[3 marks]

End of Questions

BACK TO CONTENTS PAGE


SCAN ME
ANSWER
QUICK
TIPS



SELECTED YouTube VIDEO CORNER

Understanding OS

SCAN ME!

What is OS?

SCAN ME!

Microkernel vs Monolithickernel

SCAN ME!

Multitasking

SCAN ME!

Single vs Multiprocessor

SCAN ME!

Book Citations

Silberschatz, Abraham. & Peter, Galvin., B.. (2021). Operating system concepts. Wiley. (ISBN 978-1119800361).

Meyers, M. & Hutz, H. (2023). Comptia a+ certification all-in-one exam guide, eleventh edition (exams 220-1101 & 220-1102). Mcgraw Hill Professional. (ISBN 978-1264609901).

YouTube Citations

_____ (8 February 2022). Microkernel vs Monolithic kernel. Downloaded from https://www.youtube.com/watch?v=ZOP0RUrxYWM

_____ (1 September 2020). Computer Basics: Understanding Operating Systems. Downloaded from https://www.youtube.com/watch?v=fkGCLIQx1MI

Prasad (25 Jun 2020). Multitasking Operating System | Animation | Simple Example | Easy Explanation. [Video]. https://www.youtube.com/watch?v=fhhc_bjgiuA

YouTube Citations

Prasad (19 Jun 2020). Multi-Programming Operating System | Easy Explanation using Animation. [Video]. https://www.youtube.com/watch?v=EphIwdKYsu8

_____ (19 Jun 2020). What is Multi Core CPU? E learning animation video. Downloaded from https://www.youtube.com/watch?v=Pr5yosuGZDc

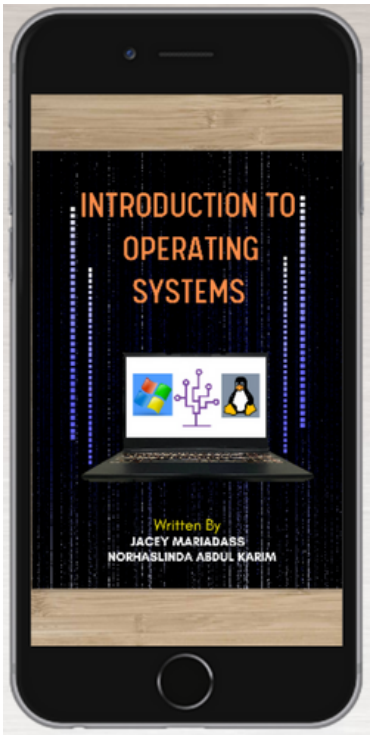
Prasad (6 Jun 2020). What is an Operating System? Goals & Functions of Operating System | Concept Simplified by Animation. [Video]. https://www.youtube.com/watch?v=ACsLvXuaKxw

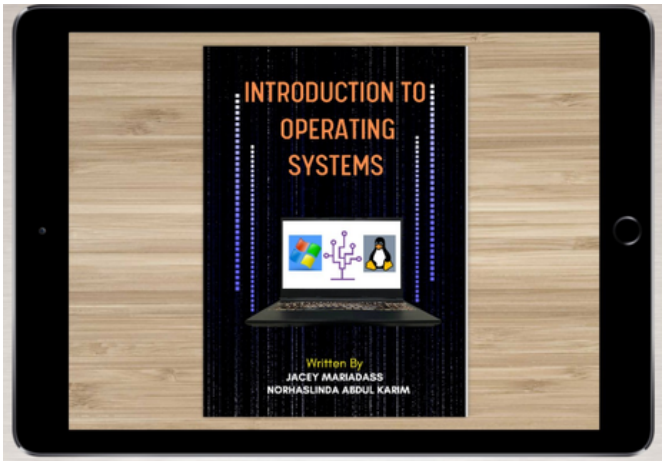
_____ (18 October 2017). Anatomy of a System Call - Operating Systems. Downloaded from https://www.youtube.com/watch?v=FVng3SqpU-Y

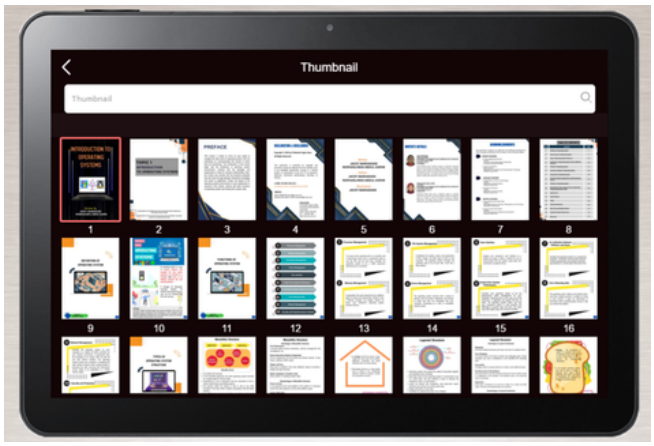
_____ (20 Jun 2015). What is an API? Downloaded from https://www.youtube.com/watch?v=s7wmiS2mSXY

SCAN THE QRCODE TO GET YOUR FLIPNOTES VERSION OF THIS INTERACTIVE EBOOK

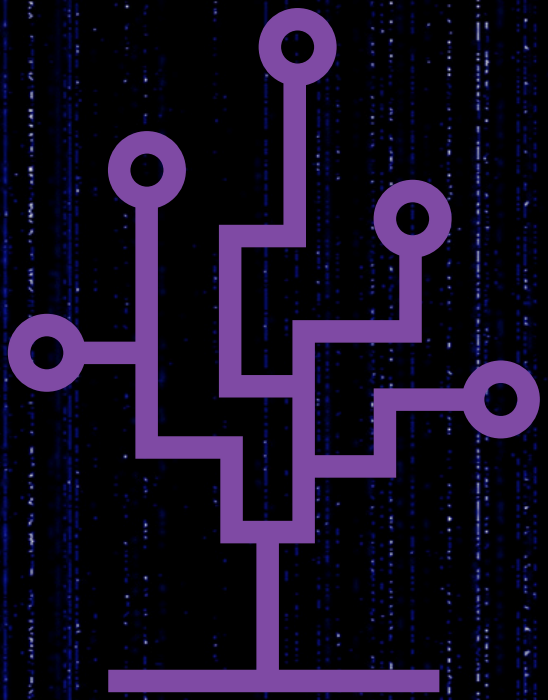
SCAN ME







INTRODUCTION TO OPERATING SYSTEMS



e ISBN 978-629-7635-39-2

9 786297 635392

POLITEKNIK UNGKU OMAR
(online)