**Operating system assignment**

**Part 2**

**By: Abdallah Salhab**

**21110181**

**(A)(B)(C)Task1:**

**What is virtual memory, and how does it differ from physical memory? How does the concept of paging relate to virtual memory.**

Virtual memory is a memory management method that is used by the computer to give the programs the illusion that it has a bigger memory to keep the system running quickly and efficiency, and that is done by the operating system as it transfers data between different types of memory like the main memory (RAM) and the hard disk or the SSD to extend its available memory.

The physical memory (RAM) is actual hardware that provides a temporary storage for active data, while the virtual memory is a reserved storage on the HD or the SSD if the RAM is full, the physical memory is much faster but it is smaller than the virtual memory because it is physically bounded.

Paging is a technique that helps in implementing the virtual memory by managing the transferer of pages; fixed length blocks of memory; between the physical memory and the HD or SSD, paging has of two things; first is the paging table that is a data structure that maps virtual address of the pages with their physical location in the memory or HD, SSD. The other thing is a fault handler that is a routine that handles the situation when a page is called but not found in the physical memory, so the fault handler locates the missing page on the disk, and then updates the page table accordingly.

Page replacement is when a process tries to access a page that is not in the physical memory, so the OS brings the required page into the physical memory

If there is no free space in the main memory the OS uses a page replacement algorithm to swap it.

**(D)(E)(F)Task2:**

**What is a distributed operating system? Discuss the advantages and disadvantages of distributed operating systems compared to centralized ones and provide examples of real-world applications where distributed operating systems are beneficial.**

is a type of operating system that connects several different computers and presents them to users as a single cohesive system, it connects more than one computer; with their own CPU and memory; to a single communication channel the CPUs communicate through high-speed busses on telephone lines, and they are called as loosely coupled systems. Each computer in this system is called a node and they are connected via LAN or WAN lines, and it has different configurations like client server or peer to peer...

The distributed operating systems offer advantages over the centralized system like:

-improving performance

-improving reliability

-improving scalability

-resource sharing

-cost efficient in the long run

The distributed operating systems disadvantages to the centralized system like:

-high startup cost

-security issue

-some cases of lost data between nodes

-more complex

-overload problems

real-world applications where distributed operating systems are beneficial:

1. web services and cloud computing:

For example, Amazon web services and google cloud platform;

These cloud platforms offer distributed services accessible via the internet, such as databases, machine learning, storage, and processing power.

2-social media platforms:

Like Instagram and Facebook

These platforms handle huge amounts of traffic, manage user data, and deliver real-time updates through distributed systems.

3- financial systems:

Distributed systems are used in financial markets to manage real-time data analysis and high-frequency trading.

**Task3:**

**What is concurrency and how does an operating system handle it, what are the main mechanisms it employs to ensure proper coordination among processes.**

Concurrency enables the system to execute multiple processes simultaneously or in overlapping time periods to improve performance and improve recourses utilization, this happens when there are several process threads running in parallel, and these processes\ threads can communicate with each other with massage.

To manage the concurrency the OS employs multiple techniques like:

-process scheduling: there are two types of process scheduling like preemptive scheduling that a process can be interrupted and switched to another process to make sure all processes had CPU time, and there is none preemptive scheduling that make a process that has started execution to finish it execution before switching.

-thread management: managing threads by context switching, synchronization and communication.

The running process threads communicate with each other using shared memory or massage passing.

Main mechanisms to insure coordination:

To ensure coordination among process when using concurrency the OS employs mutex, semaphores, and conditional variables:

-mutexes: are locks that protect important sections of code to ensure that only one thread can access shared resources at a time, to prevent race conditions.

- semaphores: it controls access to shared resources, they maintain a count and threads can request access by discriminating count but if the count is 0 it blocks the requests.

-conditional variable: allows threads to wait for a condition to be met before continuing the execution.

**Task 4: What is a remote procedure call (RPC), and how does it facilitate communication between processes in distributed systems, what is marshalling process and what it is used for, finally, what is the difference between RPC and remote method invocation.**

It is a way of communication used by a program to make a request for another program to utilize its service on a network with unknown details, procedure call is also called subroutine or function call.

It facilitates communication by:

Client-server model: the client makes a request to the server, and what process it wants to make, and the server services it over the network.

Abstraction of network communication: he developers can call a procedure remotely as if it was a local function, with out knowing the network details.

Subs: the client sub works as a proxy, and it converts the procedure call into a network massage to send to the server. And the server sub receives it, unpacks it and call it on the server side.

Marshalling is the packing of arguments or parameters into a massage packet.

Unmarshalling is the unpacking of arguments or parameters received from the call packet.

RPC supports procedural programming, supports various formats and limited to data types and the RMI supports OOP or object-oriented programming, java serialization, full object manipulation and can manage state and refrances.

**Task 5: Cloud computing is the on-demand availability of computer system resources, especially data storage (cloud storage) and computing power, without direct active management by the user. Large clouds often have functions distributed over multiple locations, each of which is a data center.**

**There are three standard service models in Cloud Computing:**

**• Infrastructure as a Service (IaaS)**

**• Platform as a Service (PaaS)**

**• Software as a Service (SaaS)**

**Discuss the service models of cloud computing and describe how functionalities and operations of modern and Distributed OS implemented in Cloud Computing Service models.**

**• Infrastructure as a Service (IaaS)**

It provides a virtual computing resource over the internet; like a vertual machine or storage or networks.

The IAAS features:

Computing; virtual machine, load balancing, auto scaling

Storage.

Networking: virtual network, VPN.

Its implementation in a modern distributed system:

Virtualization using hypervisors.

Resource allocation to allocate recourses on demand.

Scalability because it supports horizontal scaling.

Fault tolerance by the redundant system and automatic failover mechanisms.

**• Platform as a Service (PaaS)**

It allows customers to develop and run and manage applications without dealing with infrastructure by providing a platform.

PAAS features:

Templates for components and services to make development easier.

Database management

Integration services and massage queues and managing API

Its implementation in a distributed OS:

It integrates CI/CD or continuous integration and continuous deployment pipelines to make the build and test and deployment automated.

Scaling down the applications and services into smaller parts, so that they can be individually deployed and scaled.

It allows the applications to run in isolated container to insure consistency by using Docker and Kubernetes.

**• Software as a Service (SaaS)**

It gives access to software applications over the internet via a subscription.

SAAS features:

Maintenance free

Scalability

Subscription based

Easily accessible over any device on the internet

Its implementation:

Managing the user access by integrating authentication and authorization services

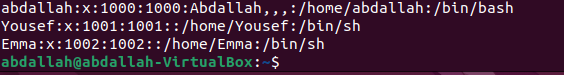
It exposes APIs to integrate with other services and platforms

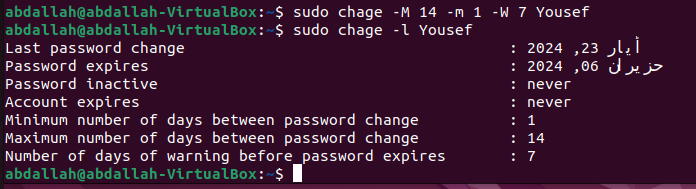
Multi tenancy

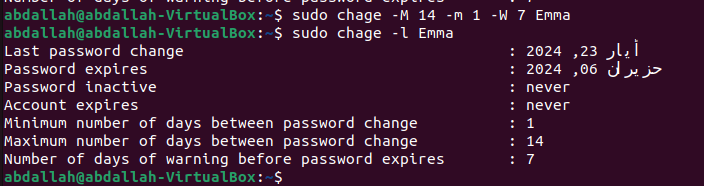
Managment of the distributed data.

**Task6:**

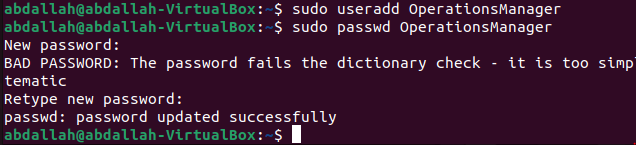
1. Create the employees “Yousef" and "Emma" accounts with a strong password and with heightened security configurations.







2. Create a manager account named "OperationsManager" to oversee the DevOps project.



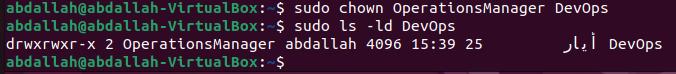
3. Form a new project team named "DevOps Team" containing "Yousef" and "Emma."



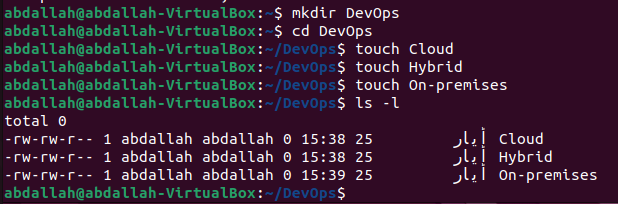




4. Assign "OperationsManager" as the manager of the DevOps project to facilitate efficient collaboration between team members.

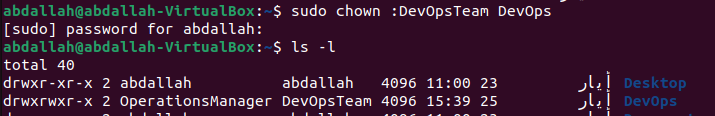


5. Prepare a dedicated directory called 'DevOps' for the project, and it must include three files: 'Cloud', 'Hybrid', and 'On-premises' within the directory.



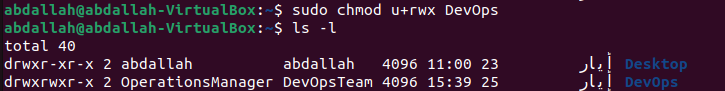
6. Verify and set correct group ownership for the "DevOps" directory and its contents.

7. Ensure the "DevOps Team" group has necessary access for seamless collaboration while maintaining security standards.

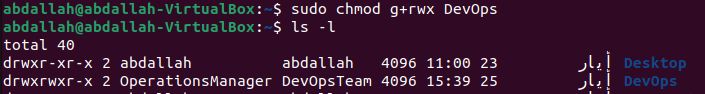


8.

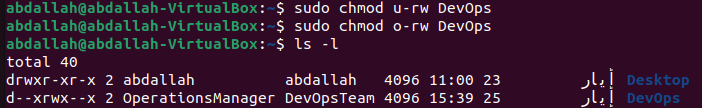
a. Grant comprehensive permissions to “OperationsManager” for full control over files and directories.



b. Grant comprehensive permissions to Yousef and Emma for full read and write capabilities on files and directories.

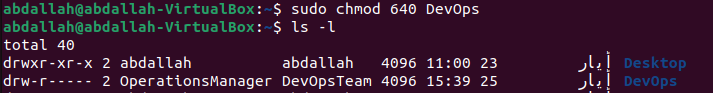


9. Implement a restrictive access model for the manager and others, allowing only execution permissions.

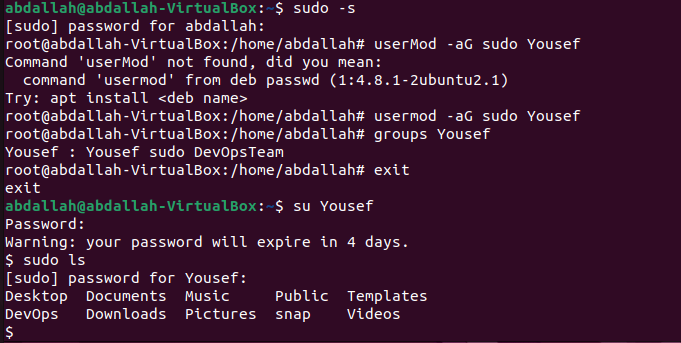


10. Configure file permissions within the "DevOps" directory to afford employees read and write capabilities, configure group permissions to read-only, and impose restricted access for others.

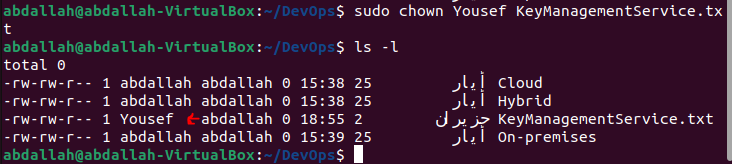
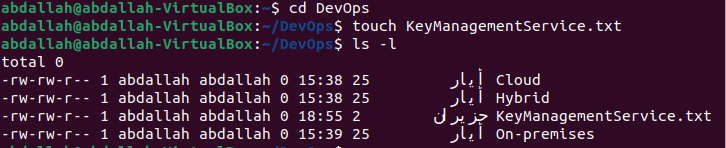
11. Validate implemented permissions to ensure alignment with team's operational requirements.



12. As "Yousef," verify access to files in the "DevOps" directory according to established permissions.



13. Ensure "Yousef" has access to a restricted file named " KeyManagementService.txt" in the "DevOps" directory without changing its access rights or revealing its contents to the group owner.



14. Ensure employees can create and modify their own files within the "DevOps" directory while preventing them from deleting files owned by other employees.



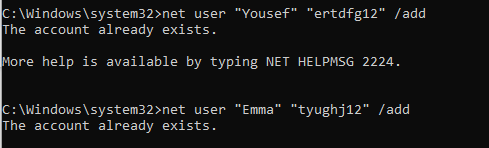
15. Remove the accounts of "Yousef," "Emma," and " OperationsManager" along with their associated directories upon completion of the project.





Task 7: Repeat parts from 1 to 5 of the previous task (Task 6) under windows using command prompt or MacOS and provide a screenshot for each part.

1. Create the employees “Yousef" and "Emma" accounts with a strong password and with heightened security configurations.

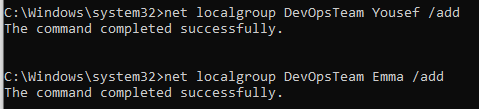


2. Create a manager account named "OperationsManager" to oversee the DevOps project.

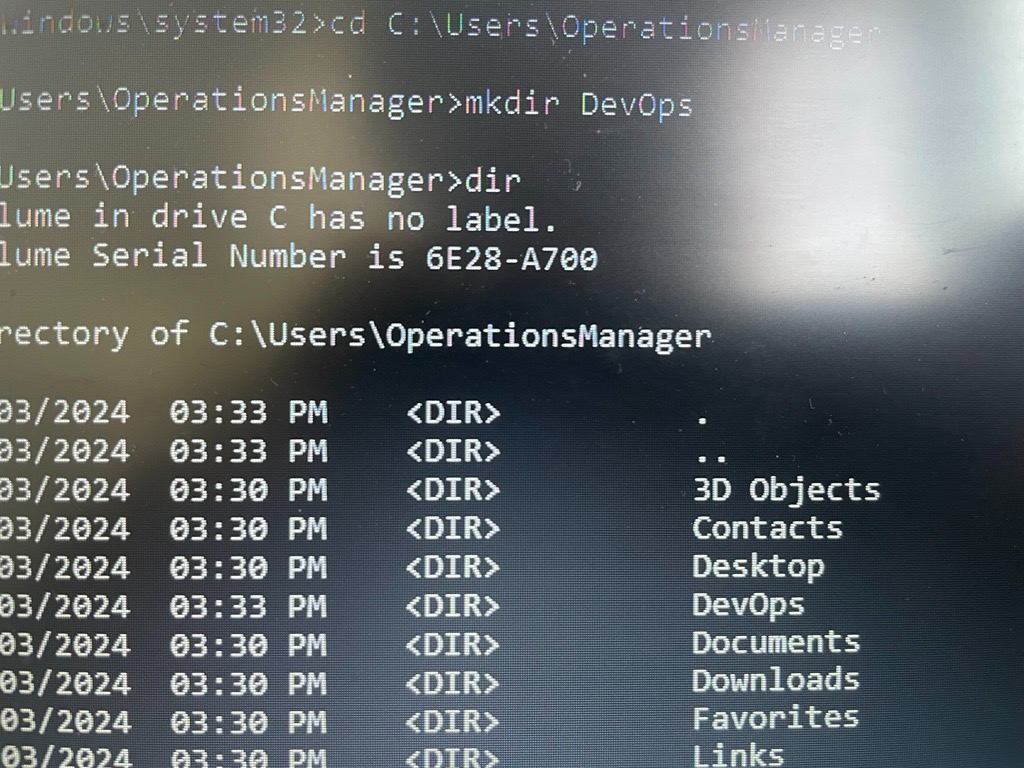


3. Form a new project team named "DevOps Team" containing "Yousef" and "Emma."

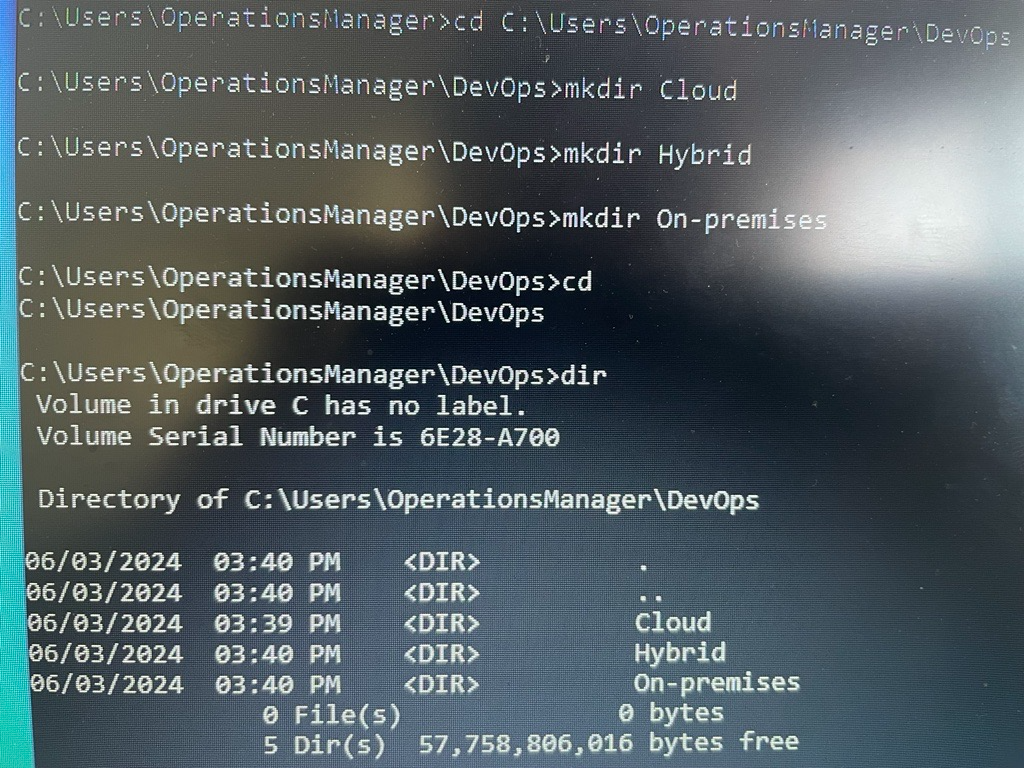




4. Assign "OperationsManager" as the manager of the DevOps project to facilitate efficient collaboration between team members.



5. Prepare a dedicated directory called 'DevOps' for the project, and it must include three files: 'Cloud', 'Hybrid', and 'On-premises' within the directory.



**Task 9: Research the evolution of the operating system scheduler, specifically in Multilevel Queue and Multi level feedback queue, what are their benefits over the traditional scheduling algorithm.**

The operating system schedulers evolution is very important to enhance the responsiveness, speed, efficiency and the overall performance of the computer.

The first scheduling algorithm was the FIFO or first come first serve, and in it when a process enters the ready queue its PCB is linked into the tail of the queue, every time the CPU is free it is allocated to the process at the head of the queue, and this process is then removed from the queue.

After that the Shortest Job First algorithm and in this algorithm is associated with each process, the length of the processes next CPU burst, and when the CPU is available, it is assigned to the process that has the smallest next CPU burst, and if two processes have the same next CPU burst, FIFO is assigned to choose how goes first.

After that, came the round robin that is designed for time sharing systems, it is similar to FIFS but preemption is added to switch between the processes.

Then, came the shortest remaining time first, it is a preemptive version of Shortest Job First, the process with the shortest remaining time to completion is always selected.

Next, priority scheduling, where a priority is associated with each process and the CPU is allocated to the process with the highest priority.

After that:

Multilevel Queue

It divides the ready queue into separate queues, and each one has its own scheduling algorithm, the processes are assigned to queue based on its characteristics such as priority or memory size.

Advantages:

-each queue can be specialized in a specific type of processes.

-priority management

Multilevel feedback queue:

It is the same as Multilevel Queue but it improves on it by allowing the processes to move between the queues.

- the processes are assigned to queue based on its characteristics

- if a process uses too much CPU time it will be moved to a lower priority queue

- the process that waits too long in a lower priority queue may be moved to a higher priority one

It has parameters to:

The number of queues

The algorithm for each queue

The methods used to determine when to upgrade a process or demote to a deferent priority queue

Benefits over traditional scheduling algorithms:  
-improved responsiveness:

They improve responsiveness over traditional algorithms by prioritizing interactive and short tasks, to make sure the system is fast to user inputs and high priority tasks.

-enhance resources utilization:

By tailoring scheduling policies to deferent types of processes and adjusting the priorities based on the process behavior.

-reduce starvation:

Ensuring that all processes eventually receive CPU time through dynamic priority adjustment.

-versatility:

They can accommodate to both batch processing and interactive applications effectively.

**Task 10: Explore what an operating system is by briefly explain the following:**

**1. - what is Degree of Multi Programming in Operating Systems and what its performance impact.**

The degree of multi programming describes the maximum number of processes that a single processor system can accommodate efficiently.

The number of processes that are kept in the memory at the same time allowing the CPU to switch between them, high degree means that more processes are kept in the memory.

The degree of multi programing effect the performance of the operating system, including CPU utilization, memory utilization, throughput, response time and the system’s efficiency increasing the degree enhances the CPU utilization and the throughput, it can also have negative impact like increasing the context switching and resource contention and thrashing, so the optimal degree should have mode benefits and less drawbacks.

**2. - Process management.**

Process management is an important task of the operating system because it handles the execution of processes.

Tasks of the process management:

-processes scheduling: it decides in what order the processes run, managing the CPU time through scheduling algorithms.

-the creation and termination of processes

-managing the processes states: the process goes through deferent states like ready, running, waiting and termination and the process manager manages these states.

-process synchronization: ensuring that the process gets executed without conflicting.

- inner process communication (IPC): enable communication between processes through pipes, shared memory...

[**https://www.shiksha.com/online-courses/articles/process-management-in-operating-system/#:~:text=Process%20management%20is%20a%20critical%20function%20of%20the%20operating%20system,programs%20interact%20with%20each%20other**](https://www.shiksha.com/online-courses/articles/process-management-in-operating-system/#:~:text=Process%20management%20is%20a%20critical%20function%20of%20the%20operating%20system,programs%20interact%20with%20each%20other)**.**

**3. - Resource management.**

The job of the resource management in the OS is to manage all the resources efficiently like the CPU, memory, I/O devices, file system, network management.

-CPU management: Allocation CPU time to deferent processes according to scheduling processes.

-I/O management: handling the I/O operations, like device drivers, buffering, I/O requests.

-file system management: managing the files creation, delete, access control and the organization of files.

-network management: managing network resources like, protocols, connections, data transfer between devices.

-device management: managing the devices hardware and software.

[https://www.ge**eksforgeeks.org/resource-management-in-operating-system/**](https://www.geeksforgeeks.org/resource-management-in-operating-system/)

**4. - Memory management.**

Managing the computer main memory (RAM) and ensuring efficient memory utilization:

Memory management tasks:

-memory allocation: assigning memory space to processes.

-using virtual memory:

-memory protection: making sure that no process interferes with another process’s memory space.

-making memory paging: dividing main memory into fixed size blocks called pages, to recue fragmentation and to allow noncontiguous memory allocation.

-cache management: storing frequently used data in the cash to speed up memory access.

(A)<https://www.indeed.com/career-advice/career-development/virtual-memory#:~:text=Virtual%20memory%20is%20a%20method,or%20solid%2Dstate%20disk%20storage>.

(B)<https://www.lenovo.com/us/en/glossary/physical-memory/#:~:text=Virtual%20memory%20and%20physical%20memory%20serve%20distinct%20roles%20in%20a,when%20physical%20memory%20is%20exhausted>.

(C)<https://www.linkedin.com/advice/0/how-do-you-manage-limited-physical-memory-using#:~:text=Paging%20is%20a%20mechanism%20that,locations%20in%20memory%20or%20disk>.

(D)<https://www.geeksforgeeks.org/advantages-and-disadvantages-of-distributed-systems/>

(E)<https://www.shiksha.com/online-courses/articles/distributed-operating-system/##Disadvantages-of-distributed-operating-system>

(F)<https://www.geeksforgeeks.org/examples-and-applications-of-distributed-systems-in-real-life/>

For task 3:

<https://eng.libretexts.org/Courses/Delta_College/Operating_System%3A_The_Basics/05%3A_Process_Synchronization/5.1%3A_Introduction_to_Concurrency#:~:text=in%20Operating%20System-,Concurrency%20in%20Operating%20System,memory%20or%20through%20message%20passing>.

<https://www.geeksforgeeks.org/concurrency-in-operating-system/>

<https://www.harrisonclarke.com/blog/mastering-concurrency-a-guide-for-software-engineers#:~:text=Ensuring%20proper%20coordination%20and%20synchronization,%2C%20semaphores%2C%20and%20condition%20variables>.

For task 4:

<https://www.quora.com/What-is-the-meaning-of-marshalling-and-unmarshalling-in-relation-to-RPC#:~:text=In%20RPC%2C%20Marshalling%20and%20Unmarshalling,received%20from%20the%20call%20packet>.

<https://www.geeksforgeeks.org/what-is-rpc-mechanism-in-distributed-system/>

<https://stackoverflow.com/questions/65646238/what-is-the-difference-between-rpc-and-rmi#:~:text=RPC%20and%20RMI%20both%20are,RMI%20supports%20object%2Doriented%20programming>.

For task 9 and 10:

<https://testbook.com/operating-system/multilevel-queue-scheduling#:~:text=A%20Multilevel%20queue%20is%20where,size%2C%20priority%20settings%2C%20etc>.

<https://www.geeksforgeeks.org/multilevel-feedback-queue-scheduling-mlfq-cpu-scheduling/>

<https://www.researchgate.net/publication/365100564_Comparison_of_CPU_Scheduling_Algorithms_FCFS_SJF_SRTF_Round_Robin_Priority_Based_and_Multilevel_Queuing>

<https://www.quora.com/What-happens-when-the-degree-of-multiprogramming-is-very-low#:~:text=degree%20of%20multiprogramming.-,The%20degree%20of%20multiprogramming%20describes%20the%20maximum%20number%20of%20processes,be%20allocated%20to%20executing%20processes>.

<https://datatrained.com/post/process-management-os/#:~:text=In%20an%20operating%20system%2C%20process,executing%20on%20a%20computer%20system>.