Question 1

a)"An operating system acts like boss in computer system." Explain this statement

The statement "An operating system acts like a boss in a computer system" is a metaphorical way of describing the role and importance of an operating system (OS) in a computer system. Let's break down this metaphor and explain the statement:

1. **Central Control Authority:** In a computer system, an operating system assumes a pivotal role similar to that of a boss in an organization. Just as a boss oversees and manages the functioning of various departments and employees, an operating system manages and controls the various components and resources of a computer system.

2. **Resource Allocation and Management:** Like a boss allocates tasks, resources, and responsibilities to different employees, an operating system manages and allocates computer resources such as memory, CPU time, and storage to different applications and processes. It ensures efficient utilization of resources and prevents conflicts between different programs.

3. **Task Scheduling:** Just as a boss determines which tasks take priority and when they should be executed, an operating system employs task scheduling algorithms to determine the order in which different processes and tasks should be executed. This ensures fair and optimal utilization of the system's processing power.

4. **User Interface and Interaction:** Similar to how a boss provides a means for employees to communicate and interact with the organization's goals and objectives, an operating system provides a user interface that allows users to interact with the computer system. This can be through graphical interfaces like Windows or command-line interfaces like Unix.

5. **Overall System Control:** A boss is responsible for maintaining discipline and order within an organization. Similarly, an operating system enforces security, manages user permissions, and ensures the overall stability and integrity of the computer system.

b) Describe the following types of operating system:

Batch Systems: In batch processing, jobs are collected and executed in batches without user interaction. These systems are suitable for tasks that require little to no user input, such as large-scale data processing, where the tasks are pre-defined and run sequentially.

Distributed Systems: Distributed operating systems manage a network of interconnected computers as a single system. They allow tasks to be divided among multiple computers while providing the appearance of a single cohesive system. These systems are often used for high-performance computing and large-scale data processing.

c) Parent process creates children processes, which, In turn create other processes, forming a tree of processes then the children processes will be terminated.

Explain TWO (2) reasons for process termination by parent.

Ans:-

1. Task Completion:- One common reason for a parent process to terminate its child processes is when the parent process has completed its task or no longer requires the child processes to continue their work. Once the parent process has achieved its goal, it may terminate its children to free up system resources and ensure a clean exit.

2. Error Handling:- Another reason for process termination by a parent is error handling. If a child process encounters an unrecoverable error or behaves unexpectedly, the parent process may decide to terminate the problematic child process to prevent it from causing further issues or data corruption. This termination can help maintain system stability and reliability.

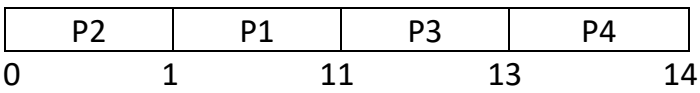
d) Consider the Information In Table 1:

Table 1

Process Name	Burst Time	Priority
P1	10	3
P2	1	1
P3	2	4
P4	1	5

i) Show the Gantt chart to illustrate the execution of the processes in Table 1 using Priority Scheduling algorithm.

Ans:- Gantt chart



ii) Identify the turnaround time and the waiting time for the processes In Table 1 using Priority Scheduling algorithm.

Ans:-

Process	AT	BT	CT	TAT	WT
P1	0	10	11	11	1
P2	0	1	1	1	0
P3	0	2	13	13	11
P4	0	1	14	14	13

e) Explain any TWO (2) conditions of deadlocks In operating system.

Ans:- Deadlocks are situations in operating systems where two or more processes are unable to proceed because they are each waiting for the other to release a resource.

Two common conditions that can lead to deadlocks are:

1. Mutual Exclusion:- This condition arises when a resource can only be used by one process at a time. If one process holds a resource and another process requests the same resource, the second process must wait until the first process releases it. If multiple processes are in a circular wait, each holding a resource while waiting for another, a deadlock can occur.

2. Circular Wait:- Circular wait occurs when a set of processes are waiting for resources in a circular chain. Process A is waiting for a resource held by Process B, which, in turn, is waiting for a resource held by Process C, and so on, until Process N is waiting for a resource held by Process A. This circular dependency can lead to a situation where no process can proceed, resulting in a deadlock.

Q.NO. 2

a) Explain the memory compaction In memory management. Support your answer with an example.

Ans:- Memory compaction is a memory management technique used in computer systems to reduce fragmentation and optimize memory utilization. Fragmentation occurs when memory is allocated and deallocated in a way that leaves small gaps or holes between allocated memory blocks. Over time, these gaps can lead to inefficient memory usage and may result in performance degradation.

Memory compaction involves rearranging the contents of memory to consolidate free memory blocks and eliminate fragmentation. There are two main types of fragmentation: external fragmentation and internal fragmentation.

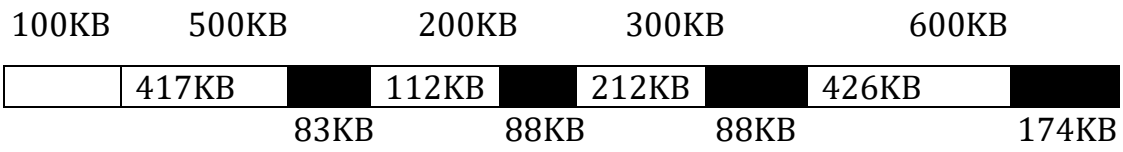
1. External Fragmentation:- This occurs when free memory blocks are scattered throughout the memory space, making it challenging to allocate contiguous memory for large processes. Memory compaction helps by moving allocated blocks and compacting them into a single contiguous block, thus reducing external fragmentation.

2. Internal Fragmentation:- This occurs when allocated memory blocks are larger than what the process actually needs, resulting in wasted memory space within those blocks. Memory compaction can also help by resizing or splitting large memory blocks to better match the actual memory requirements of processes, reducing internal fragmentation.

b) Assume that the main memory has the following five fixed partitions with the following sizes: 100KB, 500KB, 200KB, 300KB and 600KB (in order).

Show how each of the Best-fit algorithm would place processes 212KB, 417KB, 112KB and 426KB (in order).

Ans:- Best-fit algorithm

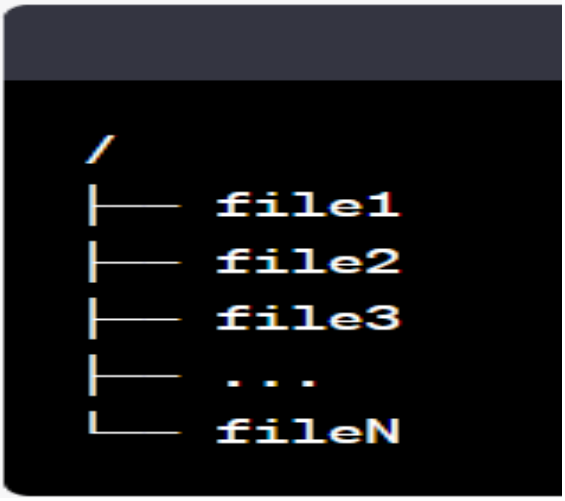


c) Describe single level and two level file structures in operating system with diagram.

Ans:- In operating systems, file structures are used to organize and manage data stored on storage devices like hard drives or SSDs. Two common file structures are the single-level file structure and the two-level file structure. Let's describe each of them along with diagrams.

1. Single-Level File Structure:- In a single-level file structure, all files are stored in a single directory or folder, without any subdirectories. This is the simplest form of file organization.

Diagram:



In the diagram above, all files (file1, file2, file3, etc.) are stored in the root directory ("/"). This structure is straightforward but can become unwieldy and challenging to manage as the number of files increases because there is no way to group or categorize them.

2. Two-Level File Structure:- A two-level file structure introduces the concept of user directories. Each user on the system has their own directory at the root level, and they can organize their files within their respective directories. This structure is more organized and provides better isolation between users' files.

Diagram:

```
/
├── user1/
│   ├── file1
│   ├── file2
│   ├── ...
│   └── fileX
├── user2/
│   ├── fileA
│   ├── fileB
│   ├── ...
│   └── fileY
├── ...
└── userN/
    ├── fileP
    ├── fileQ
    ├── ...
    └── fileZ
```

In the two-level file structure, each user (user1, user2, userN) has their own directory, and within that directory, they can organize their files as they see fit. This organization makes it easier to manage and locate files, especially in a multi-user environment.

d) Explain how interrupt happen in input/output devices.

Ans:- Interrupts in input/output (I/O) devices are events that temporarily pause the normal execution of a computer's central processing unit (CPU) to handle a specific task or event related to an I/O device. Here's a concise explanation of how interrupts happen in I/O devices:

1. Device Request: An I/O device (e.g., keyboard, mouse, disk drive) generates an interrupt request when it needs attention from the CPU. This request can occur for various reasons, such as data arriving at a port or an I/O operation completing.

2. Interrupt Signal: The I/O device sends an interrupt signal to the CPU. This signal is usually routed through an interrupt controller.

3. Interrupt Controller: In a modern computer system, an interrupt controller, like the Programmable Interrupt Controller (PIC) or Advanced Programmable Interrupt Controller (APIC), manages interrupt requests. It prioritizes and manages multiple interrupt requests from various devices.

4. Interrupt Handling: When the CPU receives the interrupt signal, it temporarily stops executing its current program and saves its state (registers, program counter, etc.) to ensure it can resume later.

5. Interrupt Service Routine (ISR): The CPU then looks up the interrupt number or type to determine which specific code, known as an Interrupt Service Routine (ISR), should be executed to handle the device's request. Each device has a unique ISR.

6. Handling the Request: The ISR contains code to handle the I/O device's request. This could involve reading data from or writing data to the device, acknowledging input, or initiating an operation.

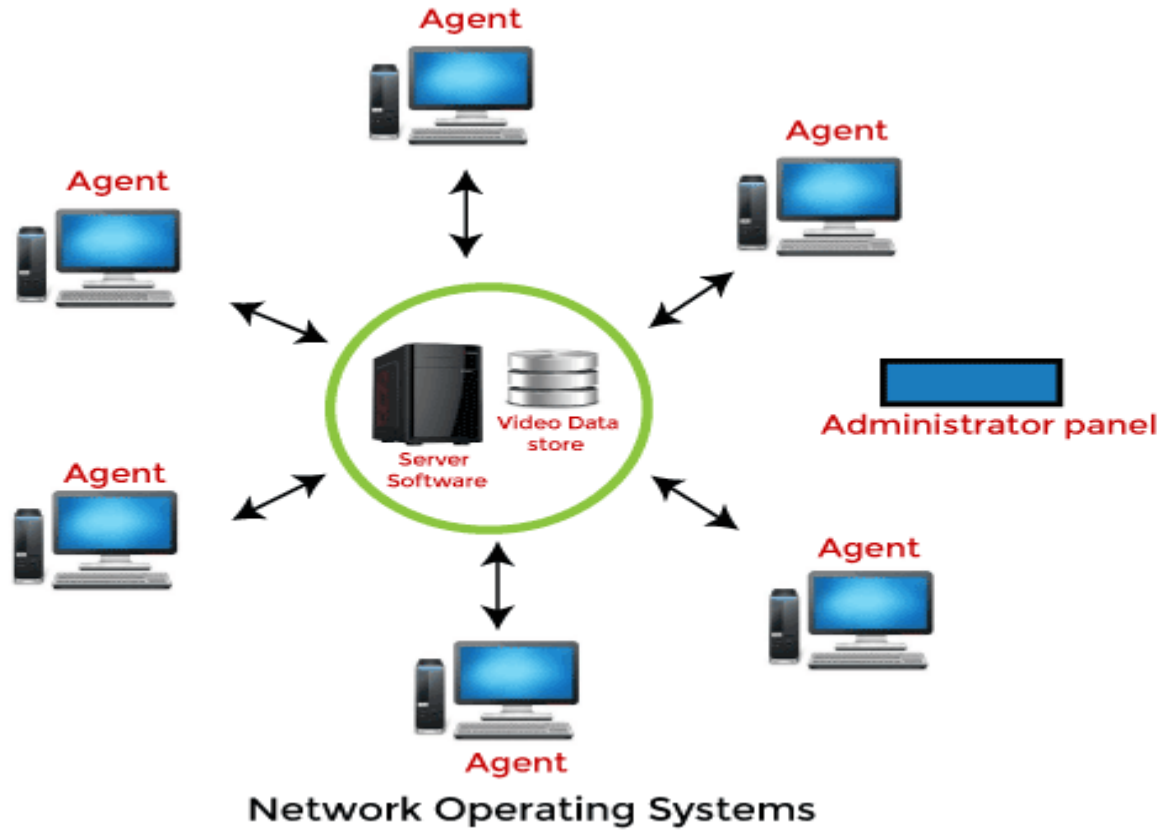
7. Completion and Resumption: After the ISR completes its task, it typically restores the CPU's saved state and returns control to the original program, allowing it to continue executing from where it left off.

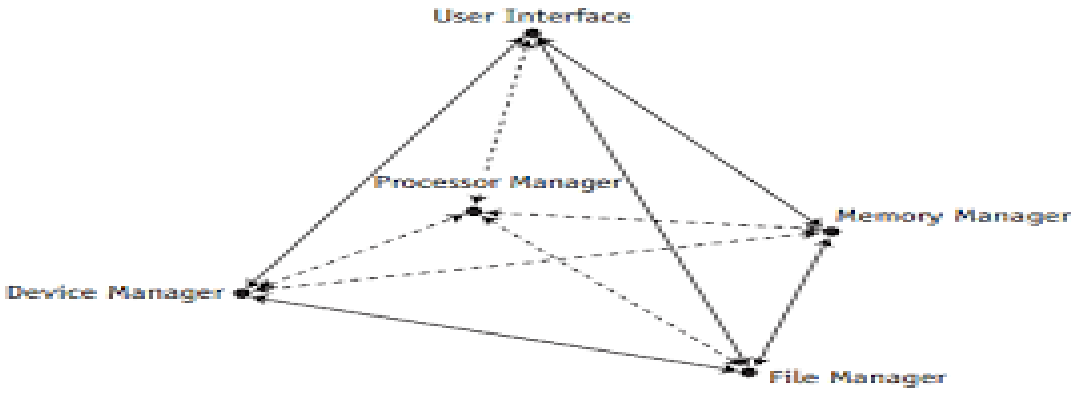
8. Priority and Nesting: Interrupts can have different priorities, and some systems support interrupt nesting, where a higher-priority interrupt can interrupt the handling of a lower-priority one. This ensures critical tasks are addressed promptly.

Q.NO. 3

a) Model a diagram showing the relationship of networked and non-networked operating system with their resource categories.

Ans:-





non-networked operating system

b) The distributed systems work in Wide Area Network (WAN) that Involved more communication as compared to computation. Each node Is a complete computer and users have an impression that they are working on single machine. Differentiate between transparency issues in distributed systems.

Ans:- Transparency in distributed systems refers to the degree to which the distributed nature of the system is hidden from users and applications. There are several types of transparency issues:

1. Access Transparency: This type of transparency ensures that users can access remote resources (e.g., files, services) in a manner similar to accessing local resources. It hides the details of where and how resources are located. Issues related to access transparency include authentication, naming, and location transparency.

2. Location Transparency: Location transparency hides the physical location of resources from users. It allows users to access resources without needing to know their specific physical addresses. This is essential in distributed systems where resources can be distributed across multiple locations.

3. Concurrency Transparency: Concurrency transparency ensures that multiple users or processes can access shared resources without conflicts. It manages concurrent access and synchronization without users needing to explicitly handle it.

4. Replication Transparency: In systems where data or services are replicated across multiple nodes for fault tolerance or load balancing, replication transparency hides the fact that there are multiple copies of the same resource. Users can access the resource without being aware of its replication.

5. Failure Transparency: Failure transparency hides the effects of failures in the system from users and applications. It includes mechanisms for fault tolerance, fault detection, and error recovery so that users perceive a reliable system despite potential failures.

6. Migration Transparency: Migration transparency allows resources to be moved from one location to another without disrupting user access. Users are unaware of resource migration and continue to access them as if nothing has changed.

c) Differentiate between network operating systems and distributed operating systems. Include the tasks performed in both operating systems.

Ans:-

Network Operating System (NOS)	**Distributed Operating System (DOS)**
NOS primarily focuses on managing and administering network resources within a local area network (LAN).	DOS is designed to manage resources and provide services across a network of interconnected computers, often in a wide area network (WAN).
It manages file and print services, user authentication, access control, and network protocols within a single LAN.	It handles process coordination, resource sharing, fault tolerance, and communication in a distributed environment.
Novell NetWare, Windows Server (in its NOS mode), and older versions of AppleShare.	Distributed versions of Unix (e.g., Sun Microsystems' SunOS, which evolved into Solaris), modern Windows Server (when configured as a distributed system), and various research-oriented distributed OS projects.

Q.NO. 4

a) Consider the following file structure (Figure 1):

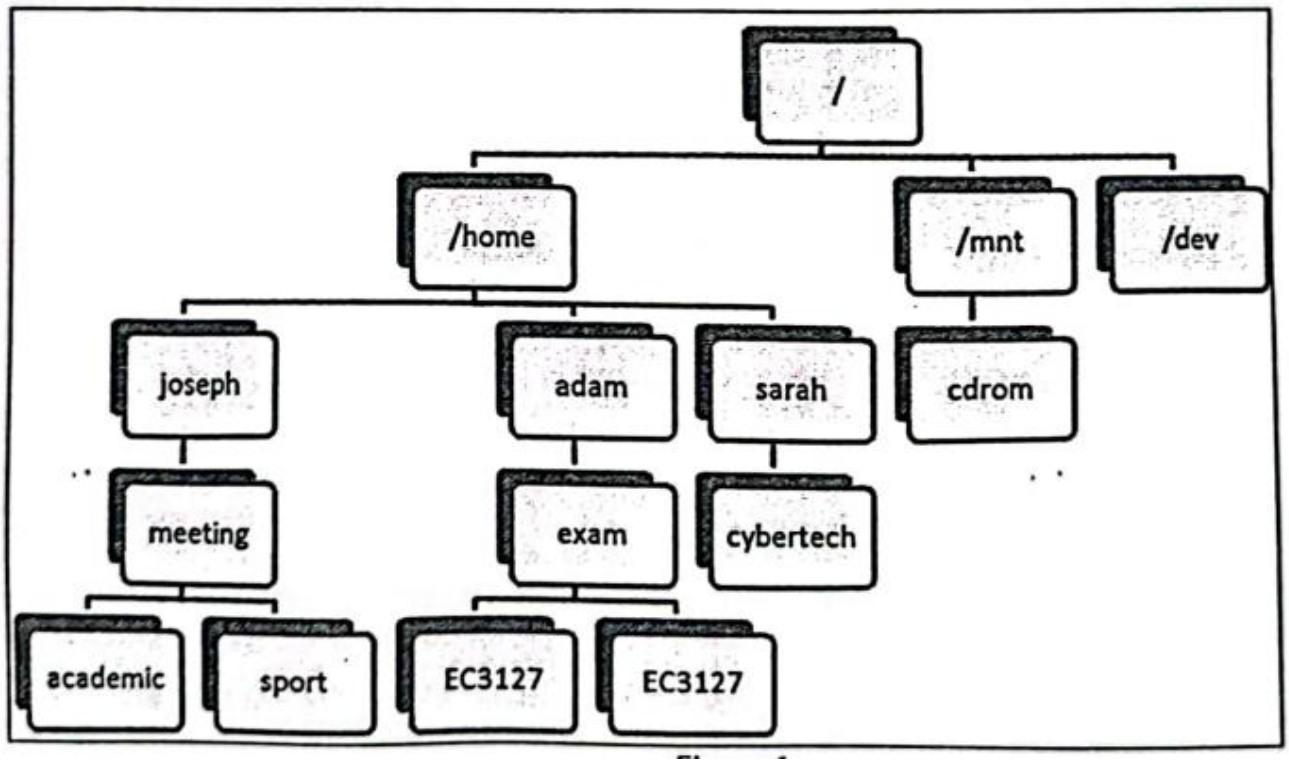


Figure 1

Write a single Linux command for following statements:

i) Assume the current working directory Is /home/joseph. Change from current working directory to directory academic.

Ans:- cd academic

ii) Change directory from academic to directory EC3239.

Ans:- cd ../EC3239

iii) Change directory from EC3239 to directory EC3127.

Ans:- cd EC3127

iv) Change directory from EC3127 to directory cdrom.

Ans:- cd ../../mnt/cdrom

v) View content of directory /dev from directory cdrom.

Ans:- ls /dev

b) Write a single Linux command for following statements:

i) Deborah has a file named recipe in her home directory and she wants to rename the file to myrecipe.

Ans:- mv ~/recipe ~/myrecipe

ii) Jackie wants to set a password to newJackle123.

Ans:- sudo passwd Jackie

iii) Janusha want to add a new group with the group ID Is 1567 and group name is Galaxy.

Ans:- sudo groupadd -g 1567 Galaxy

iv) Misbun wants to create a directory called Holiday.

Ans:- mkdir ~/Holiday

v) You have a hard link file called hardlinkmember.conf. and want to remove the link to the original file.

Ans:- rm hardlinkmember.conf

vi) Sanlah wants to add a new user Aqillah with user ID 3241 and username aqillah.

Ans:- sudo useradd -u 3241 -m -s /bin/bash aqillah

vii) Jasmine wants to delete a user account Justin from system.

Ans:- sudo userdel Justin

viii) Jessica wants to modify a group name from Galaxy to Neptune.

Ans:- sudo groupmod -n Neptune Galaxy

ix) Marvin wants to switch account to root account.

Ans:- sudo su

x) You want to display all files with permission in directory /tmp from your current working directory, EC3114.

Ans:- ls -l /tmp

xi) Cinta wants to change the group of user Aqillah from group 1567 to 6565.

Ans:- sudo usermod -g 6565 Aqillah

xii) Ace wants to copy file Exercise in her home directory to directory Amp.

Ans:- cp ~/Exercise ~/Amp/

c) Consider the Information given In Figure 2:

```
ISuSE tmp 12981$ Is -1
total 32
-rwx--  1 rosemary 1213 1483 2019-03-03 11:56 tutorial.txt
-rwx-- 1 rosemary 1213 8382 2019-03-12 1156 order
-rw- 1 rosemary 1213 222 2019.03-09 11:56 deduction.c
-rw-- 1 rosemary 1213 10760 2019-03-01 11:50 abchardlink.conf
-rw— 1 rosemary 1213 1020 2019-03.01 1030 LabTest.sh
drw-rw-rw- 1 rosemary 1213 1020 2019-03-01 10:50 MinutesMeeting
```

Figure 2

Write a single Linux command you would use to accomplish each of the following goals, in a minimal fashion. There should be no other rights granted. Assume the commands will be entered in the current working directory whose contents are shown above:

i) Allow the owner of file to read, write, execute, and the member of group 1213 to view the contents of file tutorlat.txt.

Ans:- chmod u=rwx,g=r,o= tutorial.txt

ii) Change the ownership of file order from rosemary to micheal.

Ans:- chown michael order

iii) Allow all members of group 1213 and other users to view and modify the file deduction.c.

Ans:- chmod g+rw,o+rw deduction.c

iv) Allow the owner to read, modify and execute, the group and other members to view and modify the file abchardlink.conf.

Ans:- chmod u=rwx,g=rw,o=rw abchardlink.conf

v) Allow the member and non-member of group 1213 to view, modify and execute the file LabTest.sh

Ans:- chmod ug=rwx,o=rx LabTest.sh

vi) Change the group ownership of file LabTest.sh to group CC3115.

Ans:- chown :CC3115 LabTest.sh

vii) Change the ownership of file LabTest.sh from rosemary to Jennifer.

Ans:- chown Jennifer LabTest.sh

viii) Remove the view and modify permission for other users to directory MinutesMeeting.

Ans:- chmod o= MinutesMeeting