INDEX

Date: ..... Page No. 1

Assignment No.: 01

Assignment Topic: Basic functions of OS

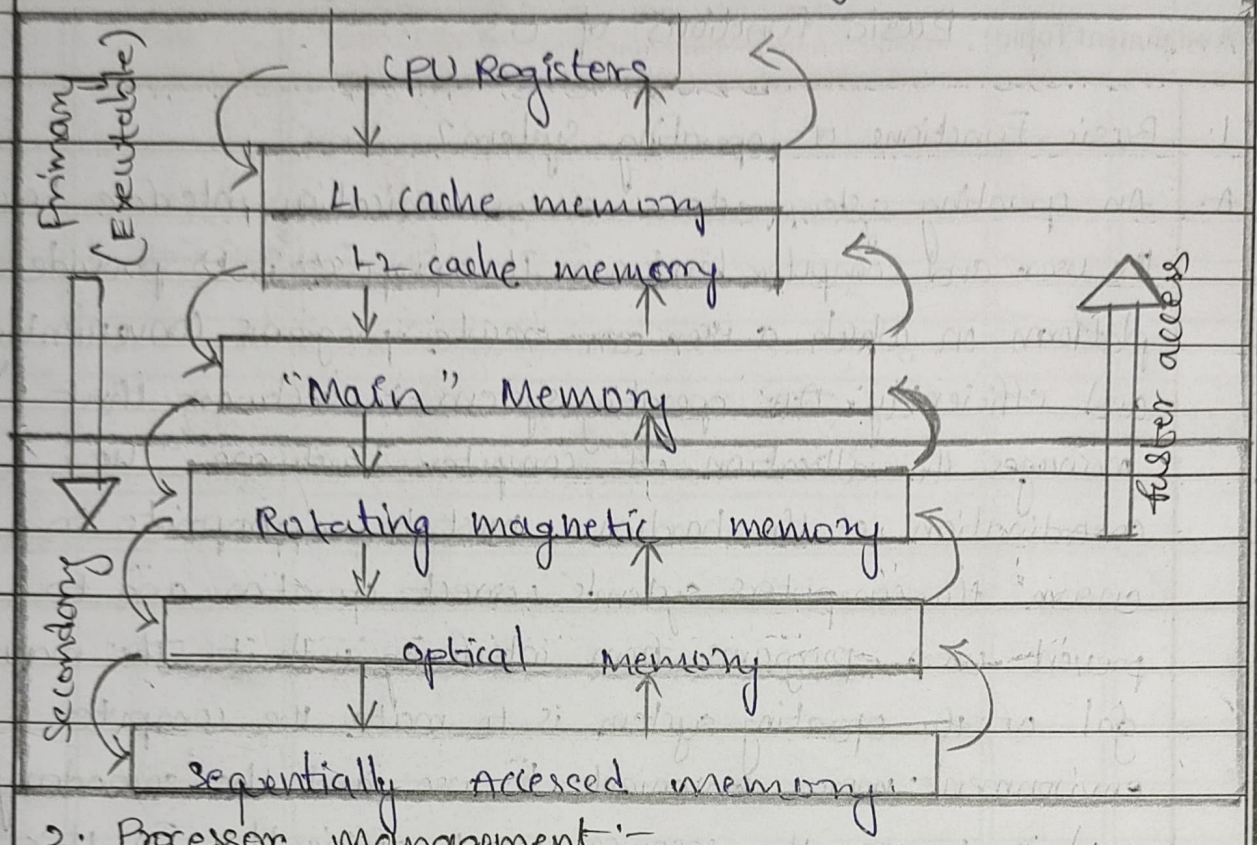
1. Basic Functions of operating System?

A. An operating system acts as a communication interface between the user and computer hardware. Its purpose is to provide a platform on which a user can execute programs conveniently and efficiently. An operating system is software that manages the allocation of computer hardware. The co-ordination of the hardware must be appropriate to ensure the computers system's correct operation and to prevent user programs from interfering with it. The main goal of the operating system is to make the computer environment more convenient to use and the secondary goal is to use the resources most efficiently. In this article we will see functions of operating system in detail.

1. Memory management :-

The operating system manages the primary memory or main memory. main memory is made up of a large of bytes of words where each byte or word is assigned a certain address. Main memory is fast storage and it can be accessed directly by the CPU. For a program to be executed, it should be first loaded in the main memory. An operating system manages the allocation and deallocation of memory to various processes and ensures that the other process does not consume the memory allocated to one process. An OS performs

The following activities for memory management

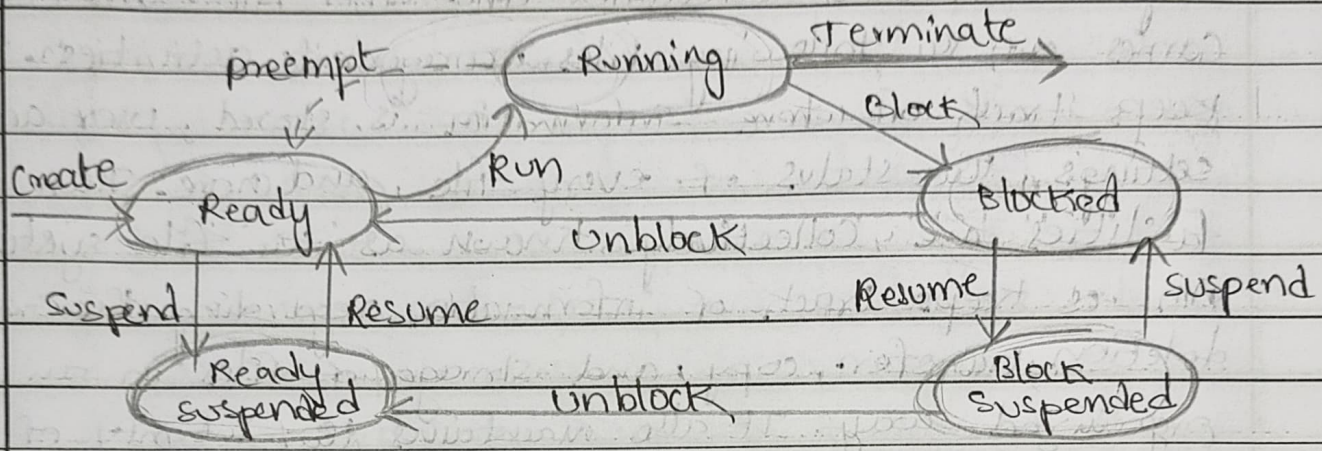


2. Processor Management :-

In a multi-programming environment, the OS decides the order in which process have access to the processor and how much processing time each process has. This function of OS is called process scheduling. An operating system performs the following activities for process management.

An operating system manages the processors work by allocating various jobs to it and ensuring that each process receives enough time from the processor to function properly. Keeps tracks of the status of processes. The program which performs this task is known as a traffic Controller. Allocates the CPU that is a processor to a process. De-allocates

processon when a process is no longer required.



3. Device Management :-

An OS manages device communication via its respective drives. It performs the following activities for device management.

- Keeps track of all devices connected to the system. Designates a program responding for every device known as the input/output controller.
- Decide which process gets access to a certain device and for how long.
- Allocates devices effectively and efficiently. Deallocates devices when they are no longer required.
- There are various input and output devices. An OS controls the working of these input-output devices.
- It receives the requests from these devices, performs a specific task, and communicates back to the

requesting process.

4. File management :-

A file system is organized into directories for efficient or easy navigation and usage. These directories may contain other directories and other files. An OS carries out the following files management activities. It keeps track of where information is stored, user access settings, the status of every file, and more. These facilities are collectively known as the file system. An OS keeps track of information regarding the creation, deletion, transfer, copy, and storage of files in an organized way. It also maintains the integrity of the data stored in these files, including the file directory structure, by protecting against unauthorized access.

5. I/O management :-

I/O management is the important function of OS refers to how the OS handles input and output operations between the computer and external devices, such as keyboards, mice, printers, hard drives, and monitors.

User Interface or command interpreter :-

The user interacts with the Computer system through the OS. Hence OS acts as an interface between the user and the computer hardware. This user interface is offered through a set of commands or a graphical user interface. Through this interface, the user mates interacts with the applications and the machine hardware.

Date : ..... Page No. 5

Assignment No. : 02

Assignment Topic : unix OS

2. Overview of unix os & functions of OS.

A. Introduction to Unix system :-

Unix is an innovative or groundbreaking os which was developed in the 1970's by ken Thompson, Dennis Ritchie, and mdy Others at AT & T Laboratories. It is like a backbone for many modern operating systems like Ubuntu, Solaris, Kali Linux, Arch Linux, and also POSIX. orginally, It was designed for developers only, UNIX played a most important role in the development and creation of the software and computing environment. Its distribution to government and academic institutions led to its widespread adoption across various types of hardware components. The core part of the UNIX system lies in its base kernel, which is integral to its architecture, structure, and key functionality, making it the heart of the operating system.

The basic design philosophy of UNIX is to provide simple, powerful tools that can be combined to perform complex tasks. It features a command-lines interfaced that allows users to interact with the system through a series of commands, rather than through a graphical user interface.

Some of the key features of UNIX include.

1. Multiusers support :- UNIX allows multiple users to simultaneously access the same system and

Shares resources.

2. Multitasking :- UNIX provides a powerful scripting languages that allows users to automate tasks.

3. Shell scripting :- UNIX is capable of running multiple process at the same time.

4. Security :- UNIX has a robust security model that includes file permissions, user accounts, and network security features.

5. Portability :- UNIX can run on a wide variety of hardware platforms, from small embedded systems to large mainframe computers.

6. Communication :- UNIX support communication methods using the write command, mail command, etc...

7. Process tracking :- UNIX maintains a record of the jobs that the user creates. This functions improves System performance by monitoring CPU usage. It also allows you to keep track of how much disk space each user uses, and the use that information to regulate disk space.

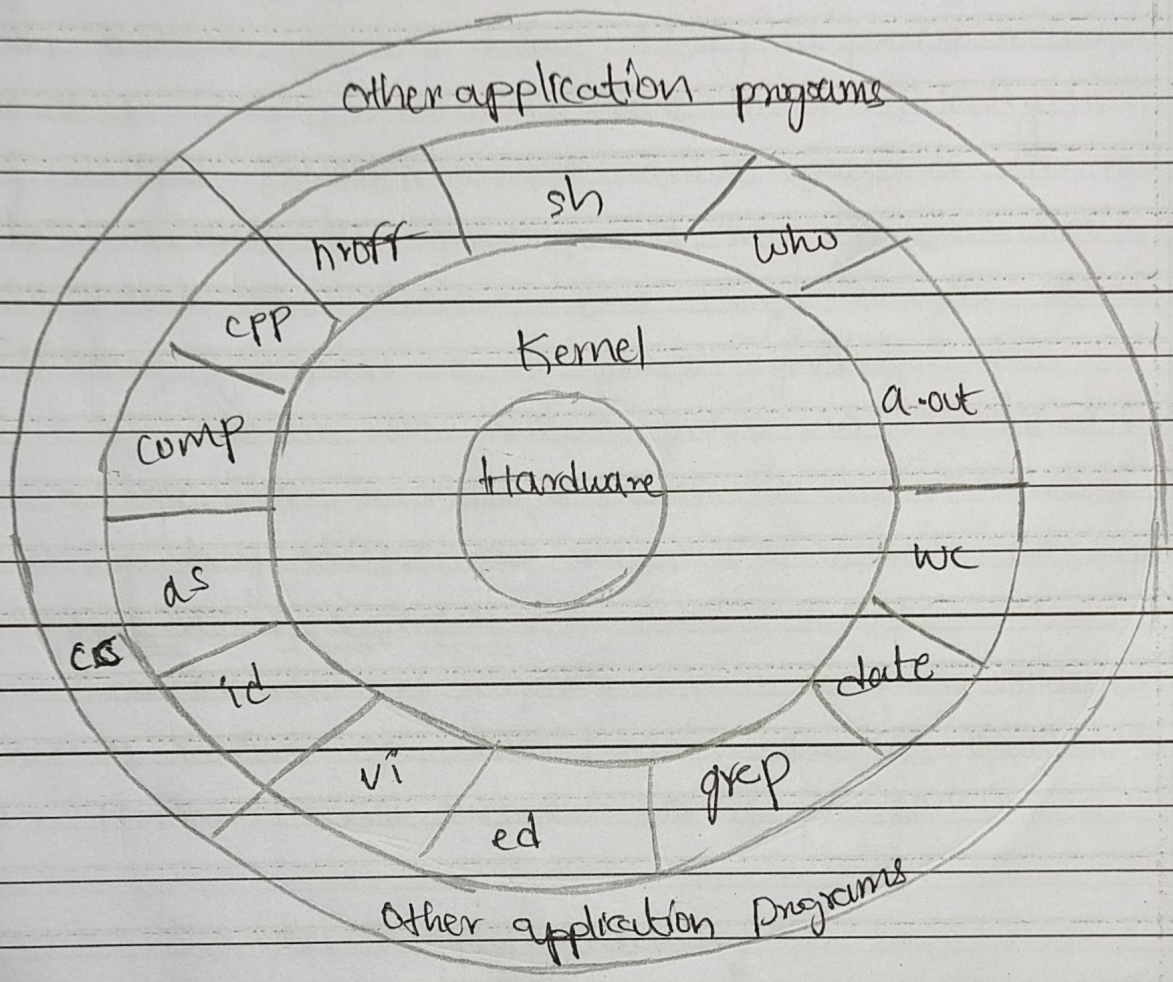
Today, UNIX is widely used in enterprise-level computing, scientific research, and web servers. Many modern OS, including Linux and macOS, are based on UNIX or its variants.

Date : .....

Page No. 7

Assignment No. : .....

Assignment Topic : .....



Date : .....

Page No. 9

Assignment No. : 03

Assignment Topic : operations on process

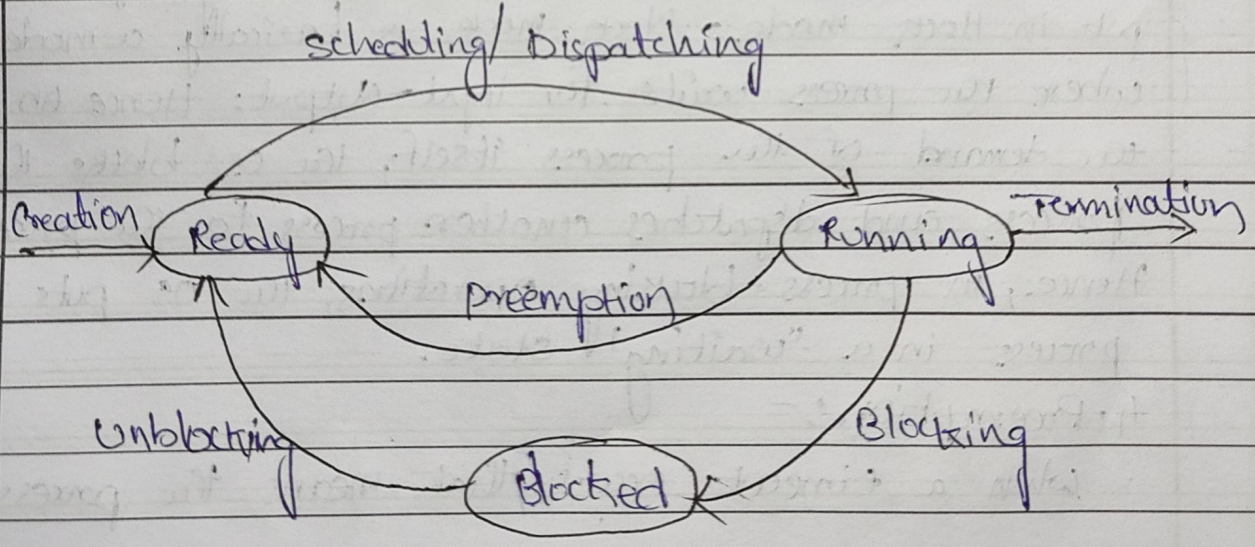
3. Process management operations on process?

A. Process management for a single tasking or batch processing system is easy as only one process is active at a time. With multiple processes being active, the process management becomes complex as a CPU needs to be efficiently utilized by multiple processes. Multiple active processes can may share resources like memory and may communicate with each other. This further makes things complex as an operating system has to do process synchronization.

We can run multiple process in interleaved manner on a single CPU. For example, when the current process is getting busy with IO, we assign CPU to some other process.

Operation on a process :-

These execution of a process is a complex activity. It involves various operations. Following are the operations that are performed while execution of a process.



1- Creation :-
This is the initial step of the process execution activity. Process creation means the construction of a new process for execution. This might be performed by the system, the user, or the old process itself. There are several events that lead to the process creation.

2. Scheduling / Dispatching :-
The event or activity in which the state of the process is changed from ready to run. It means the operating system puts the process from the ready state into the running state. Dispatching is done by the operating system when the resources are free or the process has higher priority than the ongoing process. There are various other cases in which the process in the running state is preempted and the process in the ready state is dispatched by the operating system.

3. Blocking :-
When a process invokes an input-output system call that blocks the process, and operating system is put in block mode. Block mode is basically a mode where the process waits for input-output. Hence on the demand of the process itself, the OS blocks the process and dispatches another process to the processor. Hence, in process-blocking operations, the OS puts the process in a "waiting" state.

4. Preemption :-
When a timeout occurs that means the process

hadn't been terminated in the alloted time interval and the next process is ready to execute, then the operating system preempts to process. This operation is only valid where CPU scheduling support preemption. Basically, this happens in priority scheduling where on the incoming of high priority process the ongoing process is preempted.

5. Process Terminated :-

process termination is the activity of ending the process. In other words, process termination is the relaxation of computer resources taken by the process for the execution.

- The OS itself terminates the process due to service errors.
- There may be a problem in hardware that terminates the process.

Date : .....

Page No. 13

Assignment No. : 04

Assignment Topic : Scheduling

Q. Process scheduling in operating System?

A. A process is the instance of a computer program in execution.

- scheduling is important in OS with multiprogramming as multiple process might be eligible for running at a time.
- one of the key responsibility of an OS is to decide which programs will execute on the CPU.
- Process scheduling is the activity of the process manager that handles the removal of the running process from the CPU and the selection of another process based on a particular Strategy. Throughout its lifetime, a process moves between various scheduling queues, such as the ready queue, waiting queue, or devices queue.

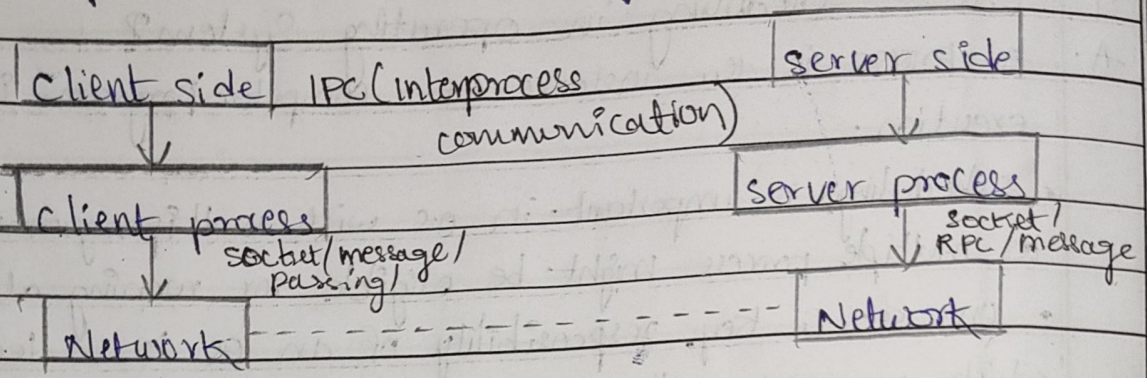
Categories of scheduling :-

Scheduling falls into one of two categories :

- Non-preemptive : In this case, a process's resource cannot be taken before the process has finished running. When a running process finishes and transactions to a waiting state, resources are switched.
- Preemptive :- In this case, the OS can switch a process from running state to ready state. This switching happens because the CPU may give other processes priority and substitute the currently active process for the higher priority process.

Types of process schedulers :-

OS ~~and it~~ includes sharing memory, message queues, semaphores and pipes among others.



Distributed file systems :-

Distributed file system provide access to files multiple machines in network. client can access and manipulate files stored on Remote servers through standard inter face Example ~~network~~ file system and server message Block.

1. long term or job scheduler :-

long term scheduler loads a process from disk to main memory for execution. The new process to the 'Ready state'.

- It mainly processes from job queue to Ready Queue.
- It controls the Degree of multi-programming, i.e., the no of processes present in a ready state or in main memory at any point in time.
- slowest among the three.

2. Short-Term or CPU scheduler.

CPU scheduler is responsible for selecting one process from the ready state for running

- It picks from ready queue.

- It mainly calls dispatcher.
- Fastest among the three.

3. Medium-Term scheduler :-

Medium Term scheduler is responsible for moving a process from memory to disk

- It reduces the degree of multiprogramming
- When needed, it brings process back into memory and pick up right where it left off.
- It is faster than long term and slower than short term.

Date : .....

Page No. 17

Assignment No. : 05

Assignment Topic : Synchronization

5. Process of synchronization & semaphores.

A. Synchronization refers to one of two distinct but related concepts: synchronization of process, and synchronization of data. Process synchronization refers to the idea that multiple processes are to join up or handshake at a certain point, in order to reach an agreement or commit to a certain sequence of action. Data synchronization refers to the idea of keeping multiple copies of a dataset in coherence with one another, or to maintain data integrity. Process synchronization primitives are commonly used to implement data synchronization.

The critical section problem:-

Critical section is the part of a program which tries to access shared resources. That resource may be any resource in a computer like a memory location, Data structure, CPU or any IO device.

The critical section can not be executed by more than one process at the same time; operating system faces the difficulties in allowing and disallowing the processes from entering the critical section.

The critical section problem is used to design a set of protocols which can ensure that the Race condition among the processes will never again.

A diagram that demonstrate the critical section is as follows:

do {

Entry section

critical section

Exit section

Remainder section

} while (TRUE);

In the above diagram, the entry section handles the entry into the critical section. It acquires the resources needed for execution by the process. The exit section handles the exit from the critical section. It releases the resources and also informs the other processes that the critical section is free.

Semaphores :-

Usage & Implementation :-

Semaphores are integer variables that are used to solve the critical section problem by using two atomic operations wait and signal that are used for process synchronization.

The definitions of wait and signal are as follows

- wait

The wait operation decrements the value of its argument S, if it is positive. If S is negative or zero, then no operation is performed.

```
wait (S)
{
while (S<=0);
  S--;
}
```

- Signal :-

The signal operation increments the value of its argument S.

```
signal (S)
{
  S++;
}
```

Types of semaphores :-

There are two main types of semaphores i.e. counting semaphores and binary semaphores. Details about these are given as follows-

- counting semaphores :-

These are integer value seamaphores and have an unrestricted value domain. These semaphores are used to coordinate the resource access, where the semaphore count is the no. of available resources. If the resources are added, semaphores count automatically incremented and if the resources are removed, the count is decremented.

- Binary semaphores :-

The binary semaphores are like counting semaphores but their value is restricted to 0 and 1. The wait operation only works when the seamaphores is 1 and the signal operation succeeds when semaphores is 0. It is somethings easier to implement binary semaphores than counting semaphores.

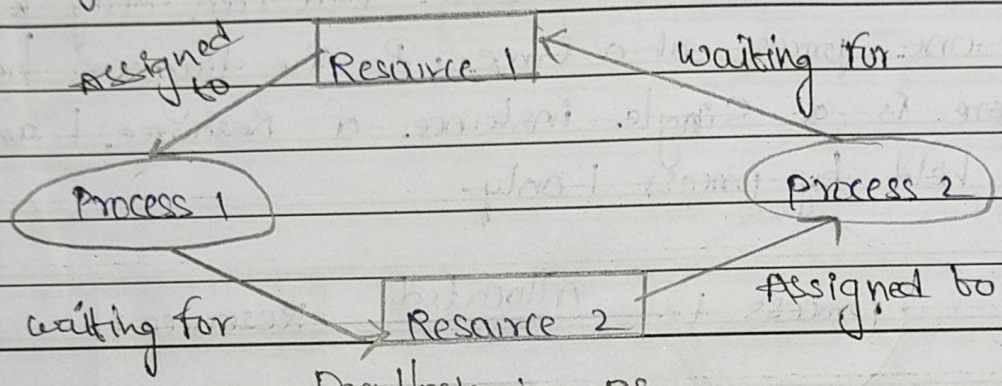
Date: ..... Page No. 21

Assignment No.: 06

Assignment Topic: Dead locks

6. Dead locks & Dead locks characterization?

A A deadlock occurs when a set of processes is stalled because each process is holding a resources and waiting for another process to acquire another resource. In the diagram below, for example, Process 1 is holding Resource 1 while Process 2 acquires Resource 2, and process 2 is waiting for resource 1.



Deadlock in OS

System Model:-

- For the purposes of deadlock discussion, a system can be modeled as a collection of limited resources that can be divided into different categories and allocated to a variety of processes, each with different requirement.
- Memory, printers, CPU's, open files, tape drivers, CD-Roms, and other resources are examples of resource categories
- Some categories may only have one resource.
- When every process in a set is waiting for a

resource that is currently assigned to another process in the set, the set is said to be deadlocked.

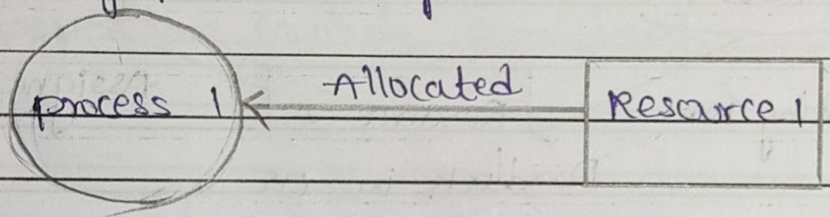
Deadlock characterization :-

A deadlock happens in operating system when two or more processes need some resource to complete their execution that is held by the other process.

A deadlock occurs if the four coffman Gotteitor conditions hold true. But these conditions are not mutually exclusive.

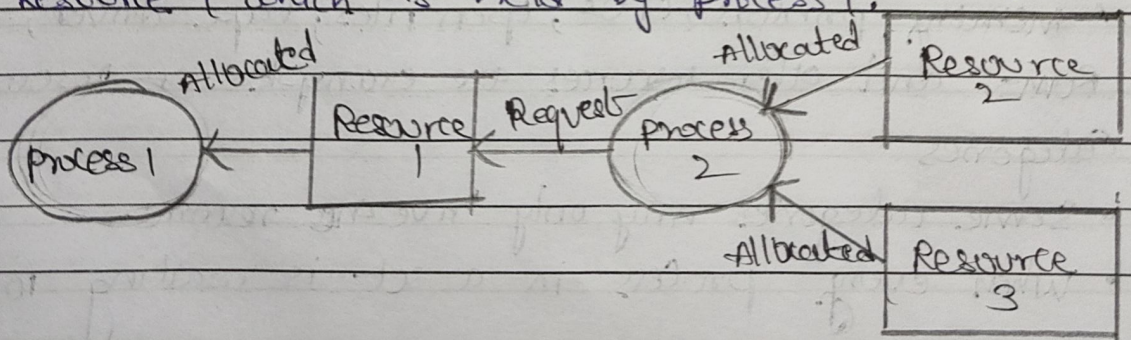
Mutual Exclusion.

There should be a resource that can only be held by one process at a time. In the diagram below, there is a single instance of Resource 1 and it is held by process 1 only.



Hold and wait :-

A process can hold multiple resources and still request more resources from other processes which are holding them. In the diagram given below, process 2 holds Resource 2 and Resource 3 and is requesting the Resource 1 which is held by process 1.



Date : .....

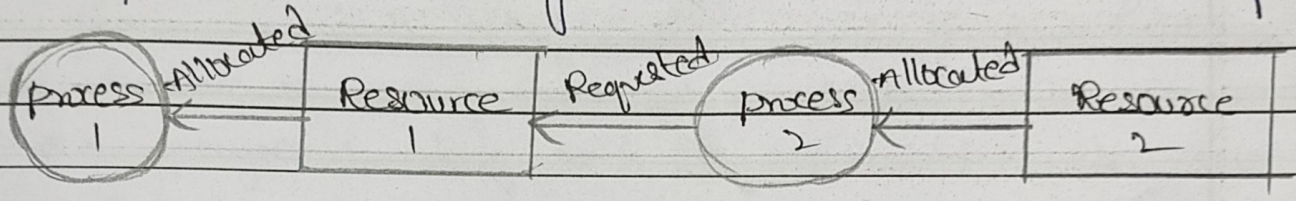
Page No. 23

Assignment No. : .....

Assignment Topic : .....

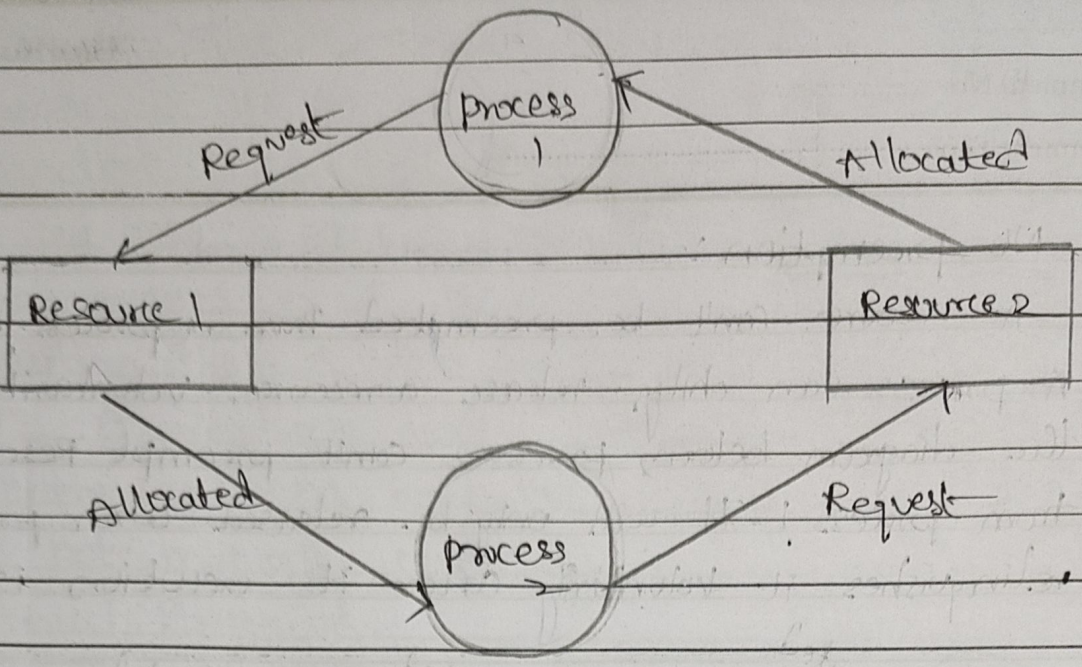
No preemption :-

A resource can't be preempted from a process by force. A process can only release a resource voluntarily. In the diagram below, process 2 can't preempt Resource 1 from process 1. It will only be released when process 1 relinquishes it voluntarily after its execution is complete.



Circular wait :-

A process is waiting for the resource held by the second process, which is waiting for the resource held by the third process and so, on till last process is waiting for a resource held by the first process. This forms a circular chain. For example : process 1 is allocated Resource 2 and it is requesting Resource 1. similarly, process 2 is allocated Resource 1 and it is requesting Resource 2. This forms a Circular wait loop.


Process 1
Request
Allocated
Resource 1
Resource 2
Allocated
Request
Process 2

Date : .....

Assignment No. : 07

Page No. 25

Assignment Topic : Memory Management

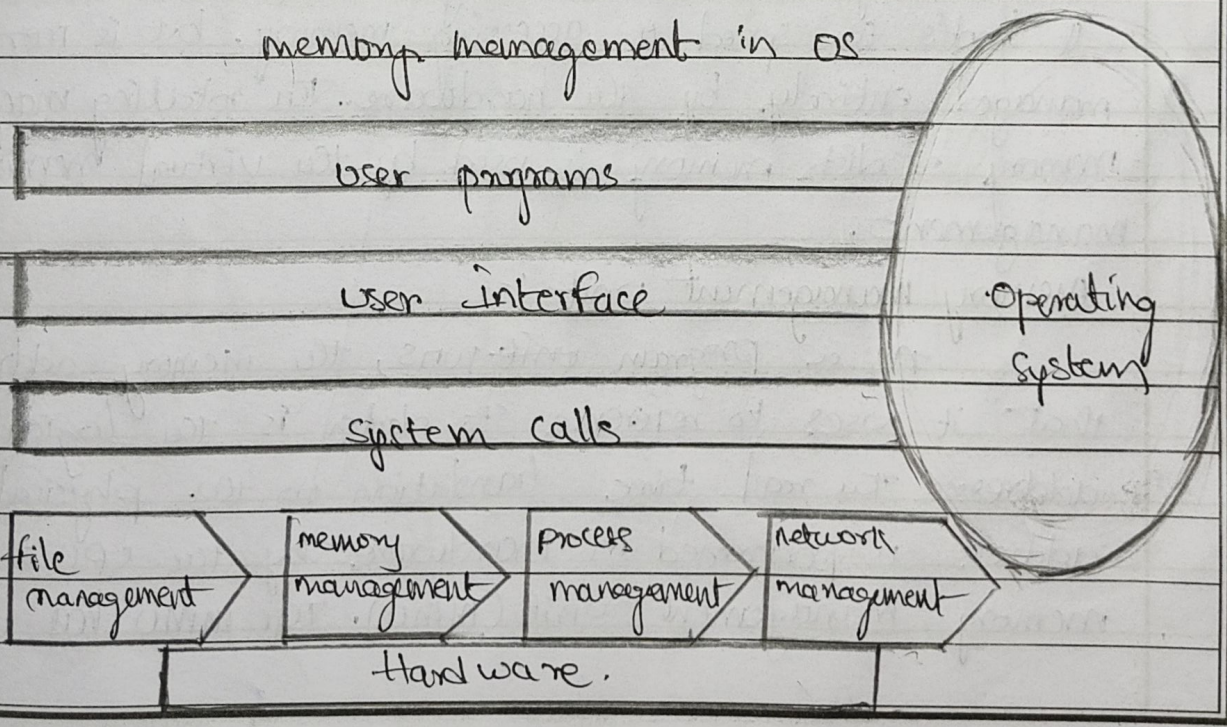
T. Explain the overview of memory Measurement?

A. **Memory management :-**

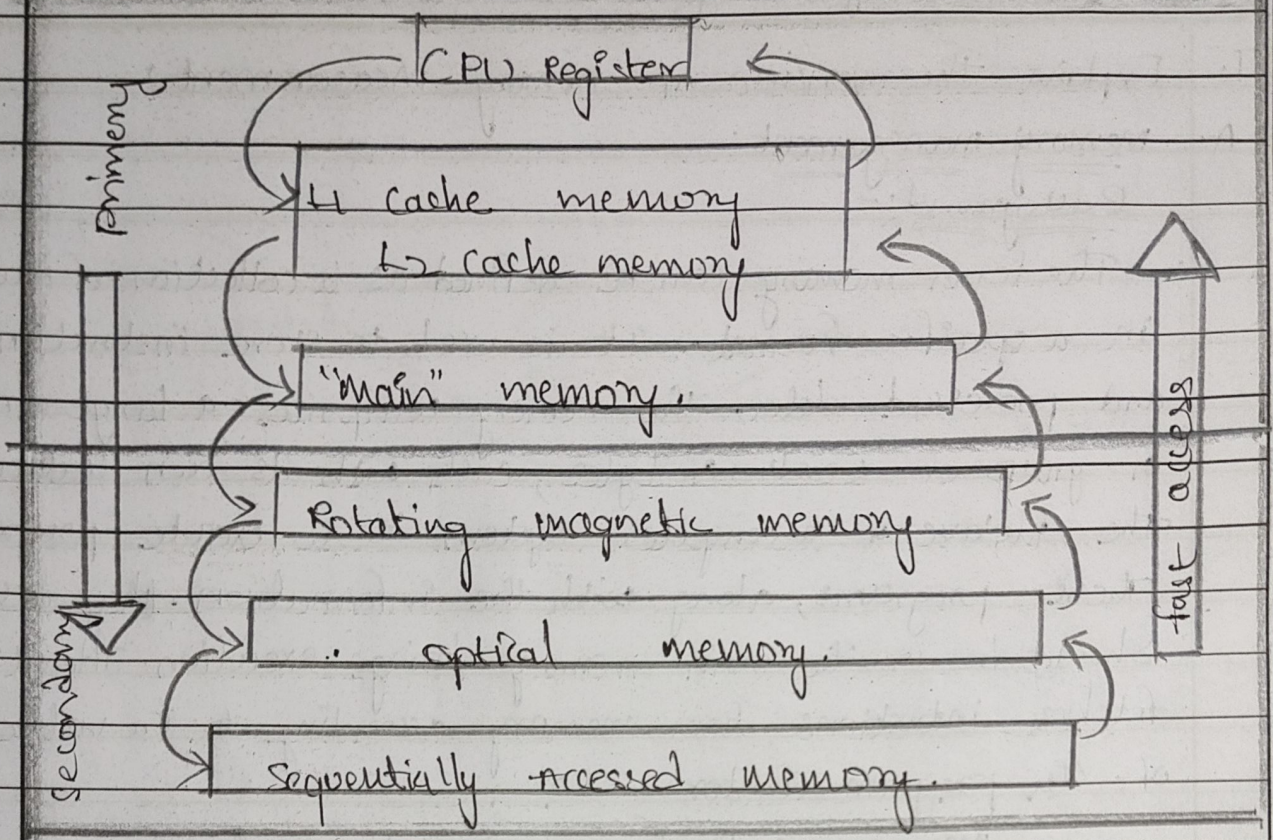
Background :-

The term memory can be defined as a collection of data in a specific format. It is used to store instructions and processed data. The memory comprises a large array or group of words or bytes, each with its own location. The motive of a computer system is to execute programs. These programs, along with the information they access, should be in the main memory during execution. The CPU fetches instructions from memory according to the value of the program counter.

memory is the important part of the computer that is used to store the data.



Computer have several different types of memory. This memory is often viewed as a hierarchy as shown below.



Our main concern here will be the computer's main or RAM memory. The cache memory is important because it boosts the speed of accessing memory, but is mostly managed entirely by the hardware. The rotating magnetic memory or disk memory is used by the virtual memory management.

Memory management unit :-

As a program runs, the memory addresses that it uses to reference its data is the logical address. The real time translation to the physical address is performed in hardware by the CPU's memory management unit (MMU). The MMU has two

Date : .....

Page No. 27

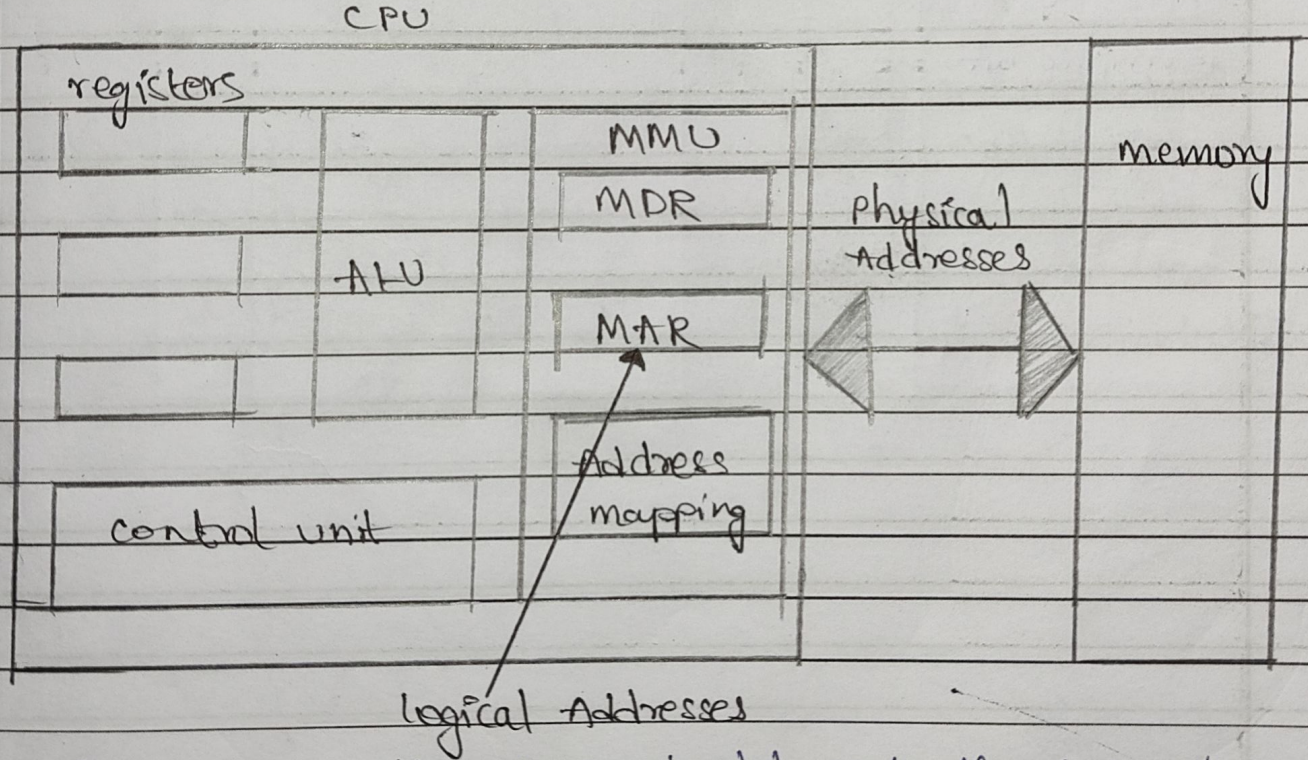
Assignment No. : .....

Assignment Topic : .....

special registers that are accessed by the CPU's control unit. Data to sent to main memory or retrieved from memory is stored in the memory Data Register (MDR). The desired logical memory address is stored in the memory Address Register (MAR).

Note :-

The job of the OS is to load the appropriate data into the MMU when a process is started and to respond to the occasional page faults by loading the needed memory and updating the memory map.



Before memory addresses are loaded on to the system bus, they are translated to physical addresses by MMU.

Date : .....

Page No. 29

Assignment No. : 08

Assignment Topic : Paging

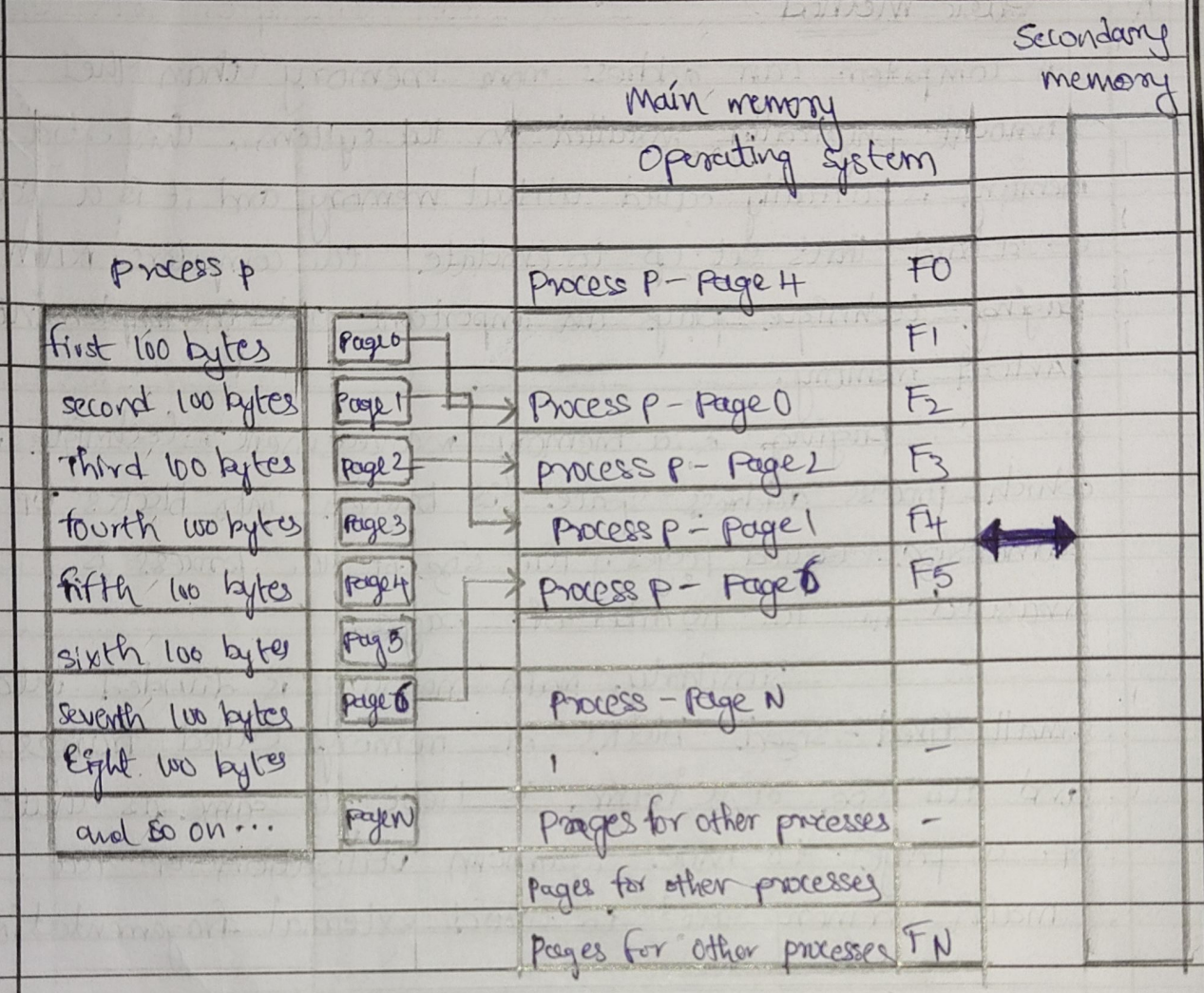
Q. Explain about paging & it's example?

A. Basic method :-

A computer can address more memory than the amount physically installed on the system. This extra memory is actually called virtual memory and it is a section of a hard that's set up to emulate the computers RAM. paging technique plays an important role in implementing virtual memory.

paging is a memory management technique in which process address space is broken into blocks of the same size called pages. The size of the process is measured in the number of pages.

Similarly, main memory is divided into small fixed-sized blocks of memory called frames and the size of a frame is kept the same as that of a page to have optimum utilization of the main memory and to avoid external fragementation.


Secondary memory
Main memory
Operating system
Process P
first 100 bytes
second 100 bytes
third 100 bytes
fourth 100 bytes
fifth 100 bytes
sixth 100 bytes
seventh 100 bytes
Eight 100 bytes
and so on...
Page 0
Page 1
Page 2
Page 3
Page 4
Page 5
Page 6
Page N
Process P - Page 4
F0
F1
Process P - Page 0
F2
process P - Page 2
F3
Process P - Page 1
F4
Process P - Page 6
F5
Process - Page N
Pages for other processes
Pages for other processes
Pages for other processes
FN

Hardware Support in Paging :-

Most OS's allocate a page table for each process. A pointer to the page table is stored with the other register values in the process control block. When the dispatcher is told to start a process, it must reload the user registers and define the correct hardware page table values from the stored user page table.

Some TLB's store address space identifiers (ASIDS) in each TLB entry. An ASID uniquely identifies each process and it used to provide address space protection for that process. An ASID allows the TLB to contain entries for several different processes simultaneously. If the TLB does not support seperate ASIDS, then every time a new page table is selected, the TLB must be flushed or erased to ensure that the next executing process does not use the wrong translation information.

Protection :-

Memory protection is a paged environment is accomplished by protection bits associated with each frame. These bits are kept in the page table. One bit can define a page to be read - write or read only. Every reference to memory goes through the page table to find the correct frame number. When the physical address is

being computed, the protection bits can be checked to verify that no writes are being made to a read-only page. An attempt to write to a read-only page causes a hardware trap to the OS.

Some systems provide hardware in the form of a page table length register (PTLR) to indicate the size of the page table. This value is checked against every logical address to verify that the address is in the valid range for the process. Failure of this test causes an error trap to the OS.

Date : .....

Page No. : 33

Assignment No. : 9

Assignment Topic : Operations on files

Q. Explain about basic operation on file?

A. Basic operation on File :-

A file is a collection of logically related data that is recorded on the secondary storage in the form of sequence of operations. The content of the files are defined by its creator who is creating the files. The various operations which can be implemented on a files such as read, write, open and close etc., are called file operations. These operations are performed by the user by using the ~~manipulator~~ commands provided by the operating system. some common operations are as follows:

1. Create operations :-

This operation is used to create a file in the file system. It is the most widely used operation performed on the file system. To create a new file of a particular type the associated application programs calls the file system. This file system allocates space to the file. As the file system knows the format of directory structure, So entry of this new file is made into the appropriate directory.

2. Open operation :-

This operation is a common operation performed on the file. Once the file is

created, it must be opened before performing the file processing operations. When the user wants to open a file, it provides a file name to open the particular file in the file system. It tells the operating system to invoke the open system call and passes the file name to the file system.

3. Write program operation:-

This operation is used to write the information into a file. A system call write is issued that specifies the name of the file and the length of the data has to be written to the file. Whenever the file length is increased by specified value and the file pointer is repositioned after the last byte written.

4. Read operation:-

This operation reads the contents from a file. A read pointer is maintained by the OS, pointing to the position up to which the data has been read.

5. Re-position or seek operation:-

The seek system call re-positions the file pointer from the current position to a specific place in the file i.e. forward or backward depending upon the user's requirement. This operation is generally performed with those file management system that

support direct access files.

6. Delete operation :-

Delete the file will not only delete all the data stored inside the file it is also used so that disk space occupied by it is freed. In order to delete the specified file the directory is searched. When the directory entry is located, all the associated file space and the directory entry is released.

7. Truncate operation :-

Truncating is simply deleting the file except deleting attributes. The file is not completely deleted although the information stored inside the file gets replaced.

8. Close operation :-

When the processing of the file is complete, it should be closed so that all the changes made permanent and all the resources occupied should be released. On closing it deallocated all the internal descriptors that were created when the file was opened.

Changing permission modes :-

To change the file or the directory permissions, you use the chmod command. There are two ways to use chmod. The symbolic mode and the absolute mode.

Using chmod in symbolic mode

The easiest way for a beginner to modify file or directory permissions is to use the symbolic mode. With symbolic permissions you can add, delete, or specify the permission set you want by using the operators in the following

Chmod operator & Description :-

1. + ⇒ Adds the designated permission to a file or directory.
2. - ⇒ Removes the designated permissions from a file.
3. = ⇒ Sets the designated permissions.