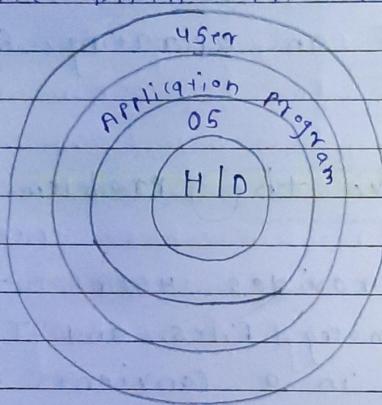


Assignment No: 1

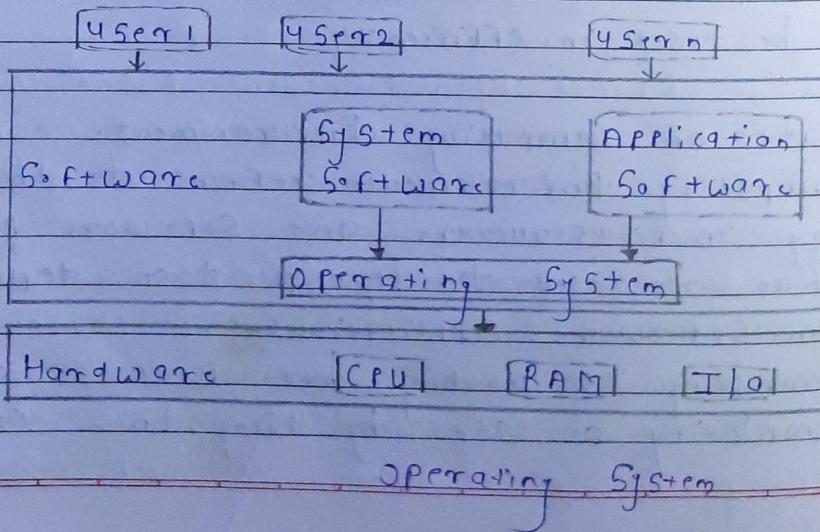
① Explain operating system in detail.

- 1) Operating System is the program that manages computer hardware, program and act as a intermediate betw Computer User and computer hardware.
- 2) The main purpose of operating system is to provide environment in which a user can easily execute their program and can interface with the computer.

3)



- 4) The operating system is very important part of almost every computer system.



There are two types of views of the operating system.

- ① User view
- ② System view

User view - The user viewpoint focuses on how the user interacts with the operating system through the usage of various application programs.

System view - An operating system can also be considered as a program running at all times in the background of a computer system known as the kernel and handling all the application programs. The operating system may also be viewed as just a resource allocator. In the system viewpoint the OS is more involved with hardware services - CPU time, memory space, I/O operation and so on.

Page No.	
Date	/ /

- ⑤ OS is a program that controls the execution of application program.
- ⑥ OS act as an interface between application and hardware.
- ⑦ OS execute user program and make solving user problem easier.
- ⑧ To make the computer system convenient to use.
- ⑨ Use all the computer hardware in an efficient manner.
- ⑩ Main object of an operating system
 - 1> Convenience
 - 2> Efficiency
 - 3> Ability to solve the problem.

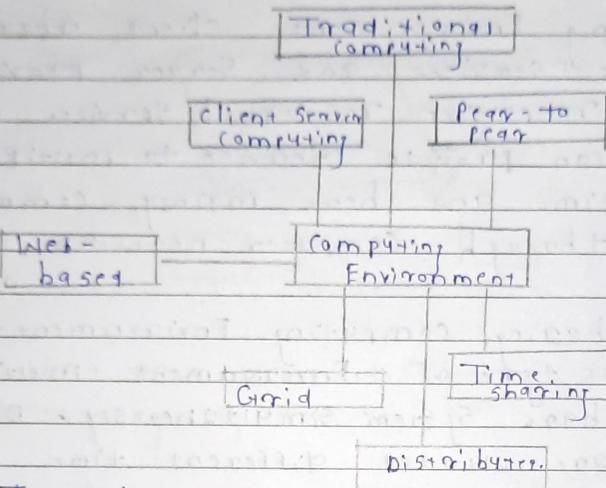
Convenience - It provides users the services (processor | Memory | Files and I/O) to execute the program in a convenient manner.

Efficiency - OS allows the system (computer resources (Hardware | Software | Data | Network)) to be used in efficiently.

- Q.2 Note on Computing Environment.
- (1) Computing Environment refers to the technology in infrastructure and software platforms that are used to develop, test, deploy and run software applications.
 - (2) Computing is nothing but process of completing a task by using this computer

technology and it may involve computer hardware and software.

③ Types of Computing Environment.



Traditional
client - Server

Peer-to Peer

Web-based

Grid

Time sharing
Distributed

① Traditional computing -

In traditional computing the user can use a traditional method like static memory allocation and it is mainly used in single user operating system.

It is like one task performed by CPU at a given time and CPU utilizes the memory that is used only for one task.

Page No.	/ /
Date	

② Client Server Computing -

In client server computing environment two machine are involved i.e. client machine and server machine. In this computing Environment client request resource / service and server provides that respective resource / service. A server can provide service to multiple client at a time and here mainly communication happen through computer network.

③ Time-Sharing Computing Environment -

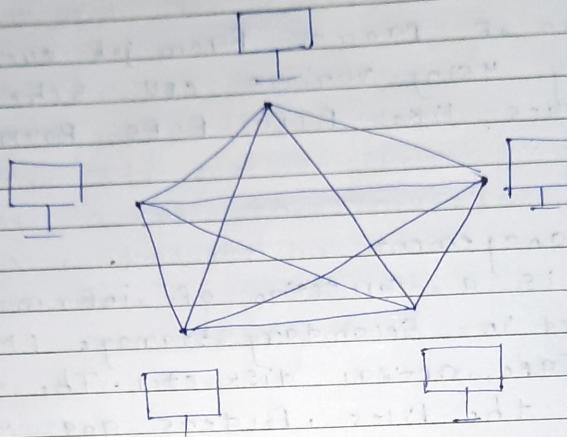
In this type of Environment multiple user share system simultaneously. Different users are allocated different time slice and processor switches rapidly according to the user, ex- Student listening to music while coding something in python.

④ Distributed Computing Environment -

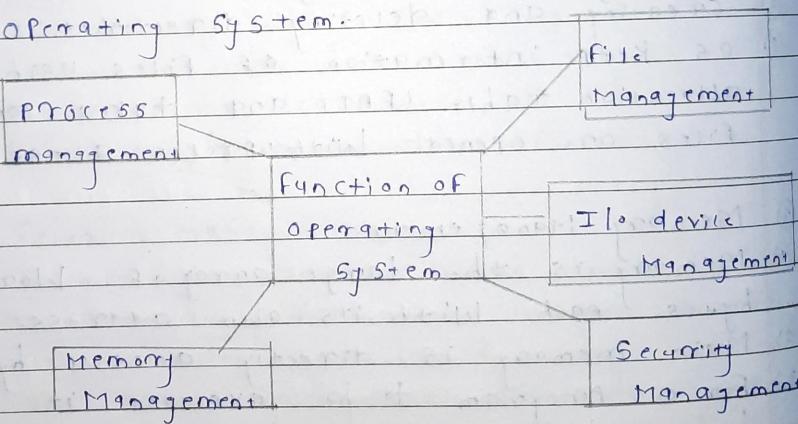
In this Environment multiple nodes are connected together using network but physically they are separated. A single task is performed by different functional units of different nodes of distributed unit.

⑤ Peer-to-Peer Computing Environment -

The peer-to-peer computing architecture contains nodes that are equal participant in data sharing. All the task are equally divided between all the nodes.



a.3 What are the different functions of operating system.



- ① Process management
- ii) In process management OS do the management of CPU.
- 2> The OS takes care of the allotment of CPU to different processes.
- 3> When CPU is free, OS selects a process from job queue and allocates the CPU to the process.

4) Selection of process from job queue is done by using various CPU scheduling techniques like FTFO, FCFS, ROUND ROBIN etc.

② File Management

- 1) A file is a collection of information. File is stored in Secondary storage (Magnetic disk, Tape, Optical disk etc.). The OS manages the files, folders and directory system on computer. Like-
 - ① Creating and deleting files
 - ② Creating and deleting directories
 - ③ OS keeps information of files using file allocation table (FAT) and takes care that files are opened with proper access rights

③ Memory Management

- 1) Memory is the large array of words or bytes each with its own address.
- 2) Main memory is directly accessed by CPU.
- 3) For a program to be executed it must be in the main memory.
- 4) Some activities are as follows
 - ① Keeps track of primary memory using free space management.
 - ② Allocates the memory to process when it requests it.
 - ③ De-allocates the memory when a process terminates and the same area is

allocated to another process.

④ I/O Device Management:

- 1) OS manages I/O device and makes the I/O process effective.
- 2) OS accept the input from the input device (Keyboard, mouse etc.) stores in the main memory, ask the CPU to process it and finally provides the result to the device (Screen | Printer) for output.
- 3) OS track all devices with the help of I/O controller.
- 4) OS decides which process gets the device when and for how much time.
- 5) Allocation and deallocation of the device which is happen in the efficient way.

⑤ Security Management

- 1) There are various virus threats which can interfere with the normal operation of computer and can be very harmful and result in loss of data and system crashes. So there are no of built-in tool including virus scanning utilities and setting up a firewall to block suspicious network activity. It also helps to prevent unauthorized access to program and set password in user login etc..

Q.4 What are the different services provided by OS?

→ ① Operating system is the program that manage computer hardware program and act as a intermediate between computer user and computer hardware.

②

User interface

System calls

Program execution	I/O Operations	File System	Communication
-------------------	----------------	-------------	---------------

Resource allocation	Accounting	Error detection	Protection and security
---------------------	------------	-----------------	-------------------------

Services

Operating System

hardware

Operating system services are:

③ Program Execution-

The system must be able to load program into memory and to run that program.

④ I/O operation - A running programs may required I/O. It allocates and interacts of

Various I/O devices taken difficult programs are being executed.

- ③ Error detection - The OS needs to be aware of possible errors. Errors may occur in the CPU memory, I/O, I/O device and in user programs. It is the duty of OS that appropriate error message whenever an error occurs and takes the suitable action for this error.
- ④ Resource allocation - When multiple users log on the system or multiple jobs are running on the same time then resources must be allocated to each of them.
- ⑤ File Management - This is the service or function of the OS to keeping the files on various storage devices and move all this files from one device to another.
- ⑥ Memory Management - It includes the assignment of main memory and other storage areas to the system program, user program and data.
- ⑦ Protection and Security :-

Page No.	
Date	11

- Q.5 What is system call? Explain in detail.
- When a program is in user mode requires access to RAM or hardware resources. It must ask the kernel to provide access to that resource this is done via something called System call.
 - ② When a program makes a system call, the mode is switch from user mode to kernel mode this is called switch context.
 - ③ A system call is a way for program to interact with the operating system. A computer program makes a system call when it makes a request to the operating system's kernel.
 - ④ System call provides the services of the OS to the user program via application program interface (API).
 - ⑤ A user program can interact with the operating system using a system call. A number of services are requested by the program, and the OS responds by launching a number of system's calls to fulfill the request.
 - ⑥ A system call can be written in high-level language like C or Pascal or in assembly language.
 - ⑦ If a high-level language is used, the OS may directly invoke system calls, which are predefined function.
 - ⑧

User program

User Interface

System call

Program
control

I/O

file
System

comms

Error
management

Resources

Auditing

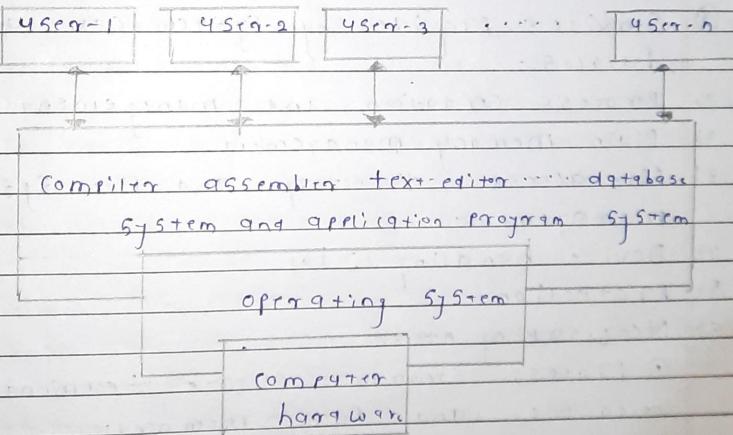
Security

Hardware

- ⑨ Services provided by system call are as follows:-
- 1> Process creation and management
 - 2> Main memory management
 - 3> file access, directory, and file system management.
 - 4> Device handling (I/O)
 - 5> protection
 - 6> Networking etc..
- ① Process control - create, terminate, allocate and free memory
 - ② File management - create, open, close, delete, read files etc..
 - ③ Device management
 - ④ Information maintenance
 - ⑤ Communication.

Q.6 Explain OS Structure in detail.

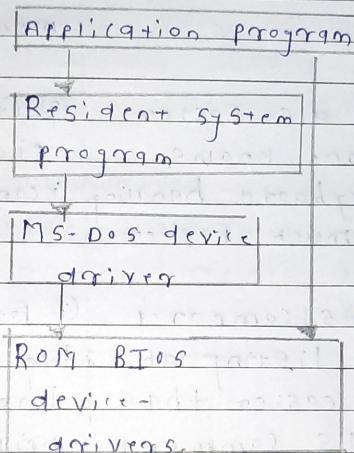
- ① OS is the program that manage computer hardware program and act as an intermediate b/w computer hardware and computer software.
- ② OS can be implemented with the help of various structure. The structure of the OS depends mainly on how the various common component of the OS are interconnected and melded into the kernel.
- ③ Depending on this we have following structures of the operating system.



OS Structure.

- ④ Such operating system do not have well define structure and are small, simple and limited structured system. The interface

and levels of functionality are not well separated. MS-DOS is an example of such OS. In MS-DOS application program are able to access the basic I/O routines. These types of operating system cause the entire system to crash if one of the user program fails.



⑤ Advantages -

- ① It delivers better application performance bcz of the few interface b/w the application program and the hardware.
- ② Easy for kernel developers to develop such an OS.

⑥ Disadvantages -

- ① The structure is very complicated as no clear boundaries exist b/w modules.
- ② It does not ^{enforce} interface data hiding in the OS.

Q.7 DIFF betw command interpreter and graphical user interface.

CLI

GUI

- | | |
|--------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| ① We have to remember command to perform any specific task. | ① GUI is easy for user hence it is user friendly. |
| ② It consumes low memory. | ② It consumes more memory. |
| ③ CLI is also faster if any one knows the keyboard handling pretty much faster. | ③ GUI is faster in picking and clicking icons. |
| ④ For ex- As compared to GUI listing of files is easier that is by 15 command and display list of all files. | ④ For ex- In GUI if we have to search for the files in the appropriate directory we must know about the directory location. |
| ⑤ In CLI os needs only keyboard. | ⑤ In GUI os needs both mouse and keyboard. |
| ⑥ While CLI appearance cannot be modify or changed. | ⑥ GUI appearance can be modify or changed. |
| ⑦ In CLI input can be entered only at command prompt. | ⑦ In GUI input can be entered anywhere on the screen. |

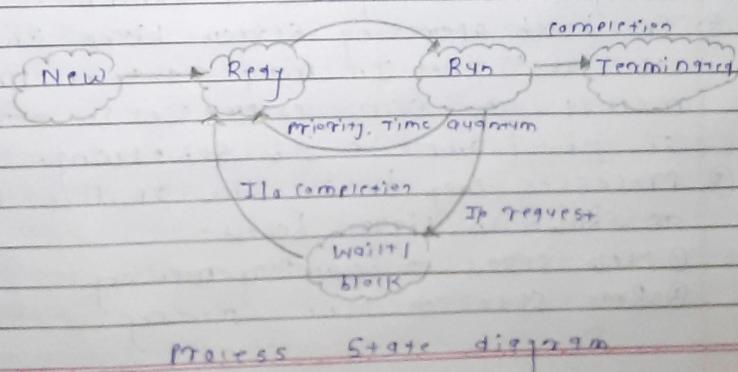
- ① In CLT info is presented to the user in plain text and files.
- ② In GUT the info is presented to the user in any form such as plain text, video, image, etc.
- ③ In CLT there are no menus provided.
- ④ In GUT menus are provided.
- ⑤ There are no graphics in CLT.
- ⑥ In GUT graphics are used.
- ⑦ CLT does not use any pointing devices.
- ⑧ It uses pointing device for selecting and choosing an item.

Q. 8 What is process? and Explain its state in detail.

- ① A process is an instance of a program running in a computer.
- ② In batch operating system work in terms of "jobs" used. In many advanced process method are still expressed in terms of job. Example is job scheduling.
- ③ Processes may be in one of five states as shown in figure below-
 - ① New state \rightarrow Secondary memory
 - ② Ready state \rightarrow Main memory

- ③ Running State → Main memory
- ④ Waiting or Block State → Main memory
- ⑤ Terminated → -

- 1) NEW - The process is in the stage of being created.
- 2) Ready State - In First Stage, a process moves from New State to Ready State after it is loaded into the main memory and this process is ready for execution.
- 3) Running State - A process moves from Ready State to Run State after it is assigned the CPU for each process for execution.
- 4) Waiting or Block State - The process cannot run at the time because process is waiting for some I/O resources or some event to occur.
- 5) Terminated - The process has completed.



a.g) Write a short note on Process Control Block.

→ ① Process control block or (PCB) is a data structure that stores all information about a particular each process. This all information about a particular each process is required by the CPU while executing the process time.

② The PCB diagram of process look like -

Process ID
Program counter
Process State
Priority
General purpose Register
List of open files
List of open device
Protection

③ Process ID

- Process ID is a unique ID for each process that identifies process of the system uniquely.
- A process ID is assigned to each process during its creation of process.

④ Program Counter

- Before process execution, PC is initialized

With the address of the first instruction of the program.

- After executing a first instruction value of program counter incremented to point to the next instruction of process and this process repeats till the end of the program.

⑤ Process State - • Each process goes through different state of process state, running state, waiting state etc. during its lifetime.
 • process state specifies the current state of the process.

⑥ Priority

- process with the very highest priority is allocated the CPU first among all the other process.

⑦ General purpose registers

- These are mainly used to hold the data of process generated during its execution time.
- Each process has its own set of registers which are maintained by its process control block (PCB).

⑧ List of open files

- Each process requires some important files which must be present in the main memory during its execution time.

- PCB maintains a list of files used by the process during its execution.

① List of Open Files -

PCB maintains a list of open devices used by the each process during its execution.

Q.10 Describe the diff b/w short-term, medium term and long-term.

→ There are three types of Scheduler-

- ① Long Term Scheduler
- ② Short Term Scheduler
- ③ Medium-Term Scheduler

1. Long-Term Scheduler

- Long-Term Scheduler is also called as Job Scheduler.
- In LTS, it selects a balanced mix of I/O bound and CPU bound process from the Secondary memory such as new file.
- Then, it loads the selected process into the main memory (Ready State) for time or execution.

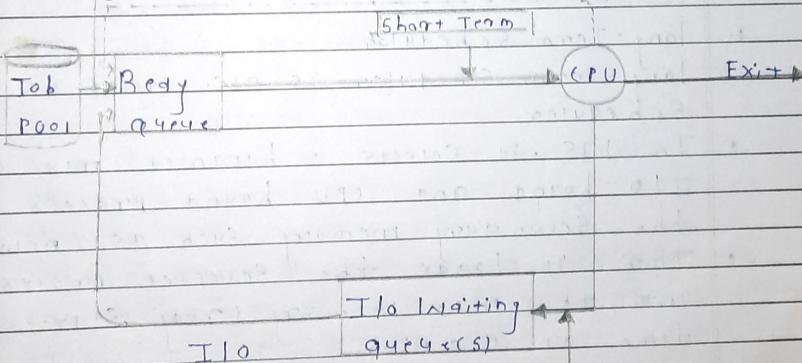
2. Short-term Scheduler (STS)

- It is also called as CPU Scheduler.
- It decide which individual process to execute next from the ready queue.

② Medium Term Scheduler (MTS)

- Medium Term Scheduler processes swaps out from main memory to secondary memory to free up the main memory when it requires.
- It mainly reduces the degree of multiprogramming.
- After some time when main memory becomes available to use, MTS swaps in and swapped out process to the main memory and its execution is continued from where it left off.
- Swapping technique may also be required to improve the process mix.

Long-Term
Scheduler



Medium
Term Scheduler

Comparison of Schedulers.

Long-Term	Medium-Term	Short-Term
It is a Job Scheduler.	It is a process-swapping Scheduler.	It is a CPU Scheduler.
It controls the degree of multiprogramming.	It requires the degree of multiprogramming.	It provides less control over degree of multiprogramming.
Speed is lesser than short-term scheduler.	Speed is in between the long-term and short-term scheduler.	Speed is faster among the other two.
It is minimal or almost absent in time sharing system.	It is a part of time sharing system.	It is also minimal in time sharing system.
It selects processes from new states and loads them into ready state.	It swaps-out processes from main-memory to secondary memory and later swaps-in.	It selects processes from the ready state and assigns to the CPU.
Operates less frequently since processes are not rapidly created.	Operates more frequently than long-term scheduler but less frequently than Short-term Scheduler.	Operates very frequently to match the speed of CPU, so it rapidly switches from one process to another.

a.ii Explain operation on process.

→ Theory

- ① A process is an instance of program running in a computer.
- ② There are many types of operation that can be performed on a process.
Some of the process operations such as-
 - 1> Process Creation
 - 2> Process Termination
 - 3> Inter Process Communication

③ Process creation-

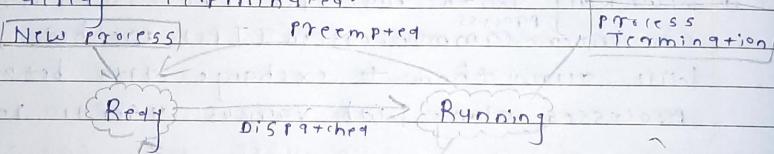
- Process essential to be created in the system for different operations. This can be done by the following events-
- User request for new process creation.
 - Automatic System initialization.
 - Batch job initialization.

A process may be created by another new process using fork() system call.

The method creating process is called the Parent process and already created process is the child process. A child process can have only one parent and but a parent process may have many children. Both the parent process and child process have the same memory image, open files and environment setting.

④ Process Termination

- After a process has successfully completed the execution task of its last instruction it is terminated.
- A child process can be terminated by its parent process if its task is no longer used. The child process sends current status information to the parent process before it terminates. Also when parent process is terminated, its child process terminates automatically as well as the child process cannot run if the parent process has already terminated.



Event occurred
Waiting on an Event
Blocked

Process Termination
diagram.

⑤ Inter process communication

- Working with multiple process communication (TCP) method which will allow them to exchange data between multiple processes along with various useful info.

There are two primary models of inter process communication.

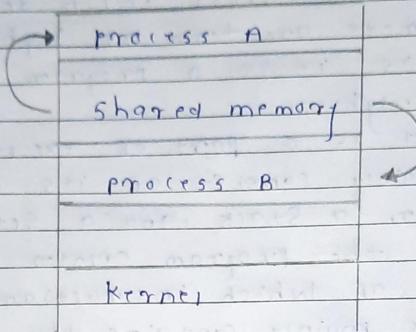
- ① Shared memory and
- ② Message passing.

Shared memory allows co-operating processes to access the same piece of data concurrently, whereas Message Passing System are mainly support a minimum System call for purpose of send msg and receive msg.

Q.12 Note on Interprocess communication

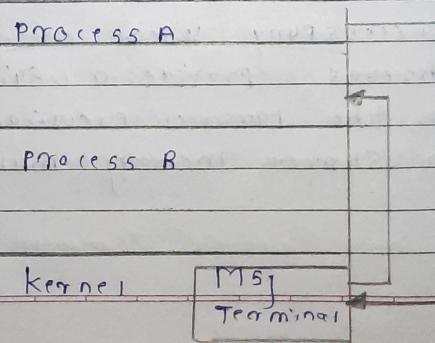
- ① Working with multiple processes requires an inter-process communication (IPC) method which will allow them to exchange data between multiple processes along with various useful info.
- ② There are two primary models of inter-process communication:
 - 1> Shared memory and
 - 2> Message passing
- ③ Shared memory
 - In shared-memory model of inter-process communication, a region of memory which is shared by cooperating processes gets established.
 - process can be then able to exchange all info by writing and reading all the data to the shared region.
 - Typically, a shared-memory region resides within the address space of any process

Creating the shared memory Segment



④ Message passing process.

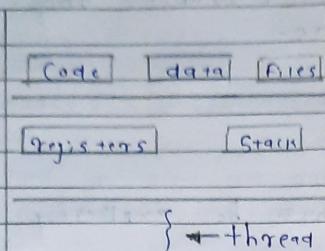
- Message passing system are mainly supported in a minimum system (OS) for purpose of send msg and receive msg.
- A communication link must be developed betn the cooperating process before msg can be sent.
- There are three key issues to be resolved in msg passing process.
 - ① direct or indirect communication.
 - ② synchronous and asynchronous communication
 - ③ automatic or explicit buffering.



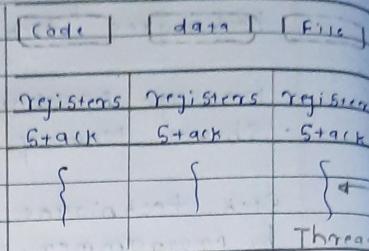
Content of

(a) Explain the threads of Lin in detail.

- ① A threads are term as light+weight process. that is it is a program by+ it is light+weight.
- ② Thread is a part of execution unit that are mainly consist of its own program counter, a stack, and a set of register where the program counter mainly keep track of which instruction to execute next.
- ③ A stack mainly contains the history of execution and a set of registers mainly hold its current working variables.
- ④ Threads are also called as light+weight process method.
- ⑤ Threads are very popular way to improve the performance of an application through parallelism.
- ⑥ Threads are mainly used to represent a software approach in order to increase the performance of OS.
- ⑦ Each thread goes to exactly one process and outside a process no. threads exists.
- ⑧ In the implementation of Network servers and Web servers threads have been successfully used.
- ⑨ Threads provide a solid foundation for the parallel execution of application on shared memory multiprocessors.
- ⑩



Single Thread
process



Multi-threaded
process

Q.13 What are the benefits of multithreading programming.

→ The benefits of multi-threaded programming can be broken down into four major categories.

① Responsiveness -

• Multithreading is an interactive application may allow a program to continue running even if a part of it is blocked or is performing a lengthy operation, thereby increasing responsiveness to the user.

• For ex - a multi-threaded Web browser allows user interaction in one thread while an video is being loaded in another thread. So instead of waiting for the whole web-page to load the user can continue viewing some portion of Web-Page.

② Resource Sharing -

process my share resources only through techniques such as -

- Message Passing
- Shared Memory

Such techniques must be explicitly organized by programmer.

The benefit of sharing code and data is that it allows an application to have several threads of activity within same address space.

③ Economy -

Allocating memory and resources for process creation is a costly job in terms of times and space.

Since, threads share memory with the process it belongs, it is more economical to create and context switch threads.

Generally much more time is consumed in creating and managing processes than in threads.

④ Scalability

The benefits of multiprogramming greatly increase in case of multiprocessor architecture, where threads may be running parallel on multiple processors.

⑤ It has better communication

Q.15 Explain the various types of multi-threading models.

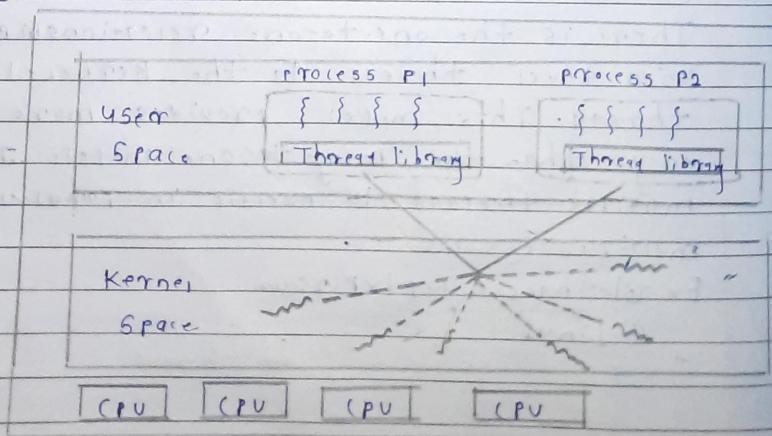
→ Multi-threading Models are Three types.

- (1) Many-to-many relationship
- (2) Many-to-one relationship
- (3) One-to-one relationship

1) Many-to-many Model

- The many-to-many model multiplexes any number of the user threads onto an equal or smaller number of kernel threads.

The following diagram shows the many-to-many threading model where six user level threads are multiplexed with six kernel level threads. In this model, developers can create many user threads as per necessary and the corresponding kernel threads can run in parallel on a multiprocessor machine.



① Many-to-one Model

When threads makes a blocking system call, the entire process will be blocked. When threads make a blocking system call, the entire process will be blocked automatically. Only one thread can access the kernel at a time. So multiple threads are unable to run in parallel on multiprocessors.

Ex - ① Solaris Green Threads

② GNU Portable Threads

③ One-to-one model

There is the one-to-one relationship of user-level threads to the kernel-level threads. This model provides more concurrency than the many-to-one model. It supports multiple threads to execute in parallel on multiprocessors.

Ex - ① Windows NT / XP / 2000

② Linux

