Semester End Examination – July/August 2022
Scheme and Solution
Faculty-Divya Singh/Rashmi D

Sub:	**Operating System**					Sub Code:	18CS43	Branch:	ISE
Date:	14/10/2022	Duration:	3hrs	Max Marks:	100	Sem/Sec:	IV A, B & C		

		MARKS
1	**Distinguish between –** **MULTIPROGRAMMING AND MULTITASKING**	

Parameter	Multiprogramming	Multitasking
Definition	It lets multiple programs use the CPU at the same time.	Multitasking refers to the simultaneous execution of numerous programmes, processes, and threads with the certain timestamp.
Objective	It's useful for cutting down on CPU idle time and boosting throughput as much as possible.	It may be used to execute numerous tasks at once, significantly improving CPU and system throughput.
Mechanism	The context switching method is used.	Based on a time-sharing mechanism.
Time	Multiprogramming demands comparatively more time to complete tasks.	Multitasking allows you to complete tasks in less time.
Execution	In a multi-programmed system, when one job or process completes its execution or changes to an I/O task, the system temporarily suspends that process. It chooses a new process to execute from the process scheduling pool.	Multiple processes can run concurrently in this system by assigning the CPU for a fixed duration of time.
CPU Switching	The CPU shifts between processes swiftly in this environment.	The CPU shifts between the processes of several programs in a single-user environment.
CPU required	Only one CPU is needed in Multiprogramming to run the tasks.	Multiple CPUs are required for the task allocation.

Multiprogramming vs. Multitasking

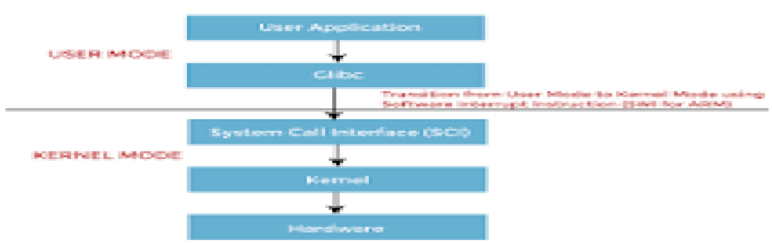
Multiprocessor systems and Clustered systems

Multiprocessor systems are single computers which have multiple CPUs. Clusters are groups of computers which work together to accomplish some specific work

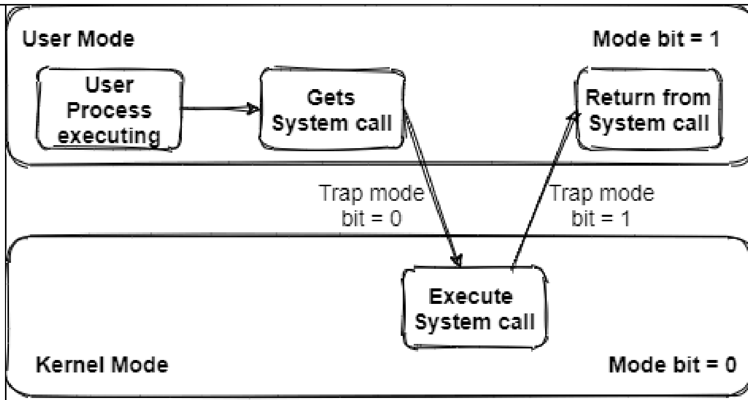
1b **What is an operating system and explain dual-mode OS?**

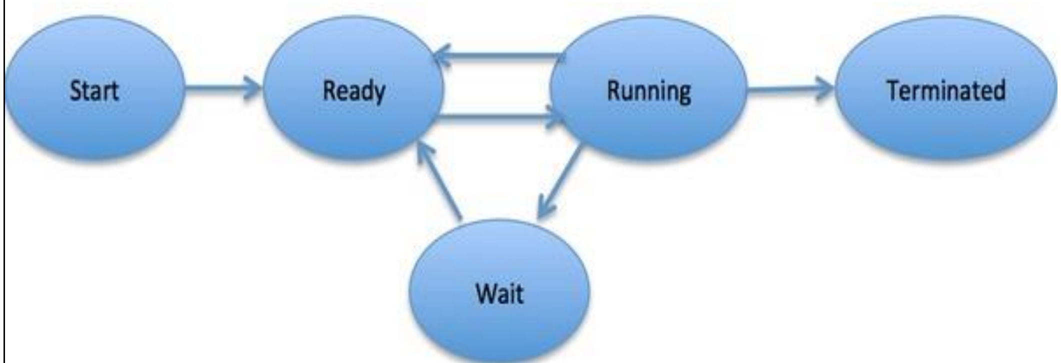
An operating system (OS) is a program that, after being initially loaded into the computer by a boot program, manages all of the other application programs in a computer. The application programs make use of the operating system by making requests for services through a defined application program interface (API). In addition, users can interact directly with the operating system through a user interface, such as a command-line interface (CLI) or a graphical UI (GUI).

Dual Mode OS

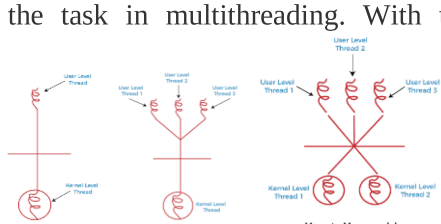


c) Explain about system calls with an example of handling a user application invoking the open () system call.

<table>
<tr><td></td><td>
</td><td></td></tr>
<tr><td>2a</td><td>OR
2 a. What is a process? Illustrate with a neat diagram the different states of a process and control block. (05 Marks)

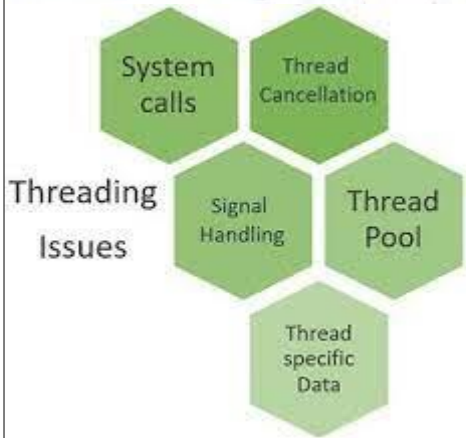
A process is defined as an entity which represents the basic unit of work to be implemented in the system.
"

</td><td></td></tr>
<tr><td>2b</td><td>Discuss the implementation of IPC using message passing systems in detail.

Inter-process communication is used for exchanging useful information between numerous threads in one or more processes (or programs)."</td><td></td></tr>
<tr><td>2c</td><td>List and explain the services provided by OS for the user and efficient operation of system. (05 Marks)

An Operating System provides services to both the users and to the programs.
It provides programs an environment to execute.
It provides users the services to execute the programs in a convenient manner.
Following are a few common services provided by an operating system −
Program execution
I/O operations
File System manipulation
Communication
Error Detection
Resource Allocation
Protection</td><td></td></tr>
<tr><td>3a</td><td>Give a brief description about multithreading and explain the different multi threading models. (05 Marks)
Multithreading allows the application to divide its task into individual threads. In multi-threads, the same process or task can be done by the number of threads, or we can say that there is more than one thread to perform the task in multithreading. With the use of multithreading, multitasking can be achieved. 
</td><td></td></tr>
</table>

3b

Discuss the issues that come with multithreaded programming.



3c

Explain CPU scheduling criteria.

The scheduling criterion is responsible for helping in the design of the good scheduler. These criteria are as follows −

CPU Utilization

The scheduling algorithm should be designed in such a way that the usage of the CPU should be as efficient as possible.

Throughput

It can be defined as the number of processes executed by the CPU in a given amount of time. It is used to find the efficiency of a CPU.

Response Time

The Response time is the time taken to start the job when the job enters the queues so that the scheduler should be able to minimize the response time.

Response time = Time at which the process gets the CPU for the first time - Arrival time

Turnaround time

Turnaround time is the total amount of time spent by the process from coming in the ready state for the first time to its completion.

Turnaround time = Burst time + Waiting time

or

Turnaround time = Exit time - Arrival time

Waiting time

The Waiting time is nothing but where there are many jobs that are competing for the execution so the Waiting time should be minimized.

Waiting time = Turnaround time - Burst time

Fairness

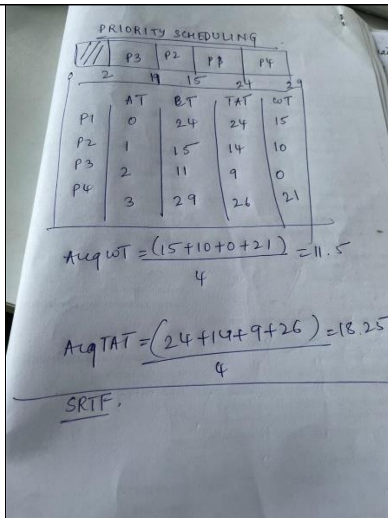
For schedulers, there should be fairness in making sure that the processes get a fair share of chances to be executed.

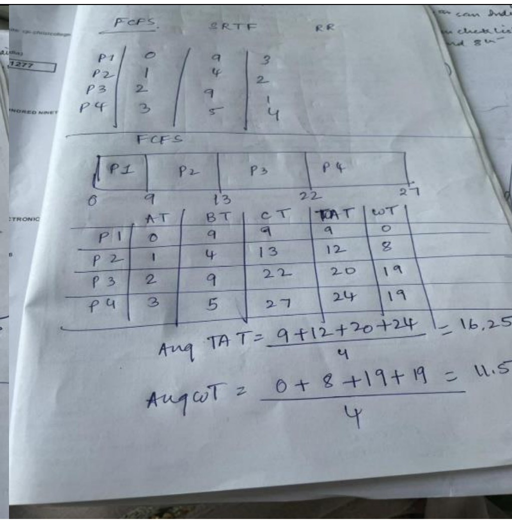
4a

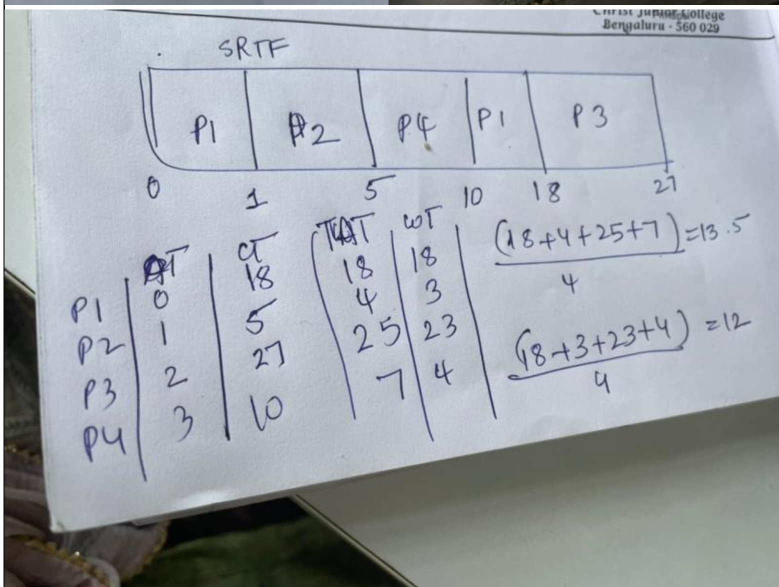
4 a. Calculate the average waiting time and the average turnaround time by drawing the Gantt chart using FCFS, SRTF, RR (q = 2ms) and priority algorithms. Lower priority number represents higher priority.

Process	Arrival Time	Burst Time	Priority
P_1	0	9	3
P_2	1	4	2
P_3	2	9	1
P_4	3	5	4

(12 Marks)



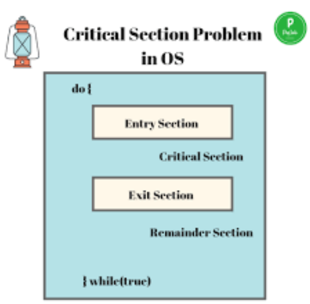




4b

b. What is critical section problem? What are the requirements for the solution to critical section problem? Explain Peterson's solution. **(08 Marks)**

The critical section problem is **used to design a protocol followed by a group of processes**, so that when one process has entered its critical section, no other process is allowed to execute in its critical section.



Peterson's algorithm (or Peterson's solution) is **a concurrent programming algorithm for mutual exclusion that allows two or more processes to share a single-use resource without conflict, using only shared memory for communication**. It was formulated by Gary L. Peterson in 1981.

Peterson's solution

```
                turn = 0;
                flag[0] = false;
                flag[1] = false;
P0                                      P1
while(true){                            while(true){
  flag[0] = true;                         flag[1] = true;
  turn =1;                                turn = 0;
  while (flag[1] and turn ==1) do no-op;    while (flag[0] and turn ==0) do no-op;
    CS                                      CS
  flag[0] = false;                        flag[1] = false;
  remainder section;                      remainder section;
}                                       }
```

Module-3

5a What is a deadlock? What are the necessary conditions for the deadlock to occur? (

Deadlock is a situation where two or more processes are waiting for each other. For example, let us assume, we have two processes P1 and P2. Now, process P1 is holding the resource R1 and is waiting for the resource R2. At the same time, the process P2 is having the resource R2 and is waiting for the resource R1. So, the process P1 is waiting for process P2 to release its resource and at the same time, the process P2 is waiting for process P1 to release its resource. And no one is releasing any resource. So, both are waiting for each other to release the resource. This leads to infinite waiting and no work is done here. This is called Deadlock.

Necessary conditions-

1. Mutual exclusiom
2. No pre-emption
3. Hold and wait
4. Circular wait

5b How to prevent the occurrence of deadlock, explain in detail.

Banker's algorithms

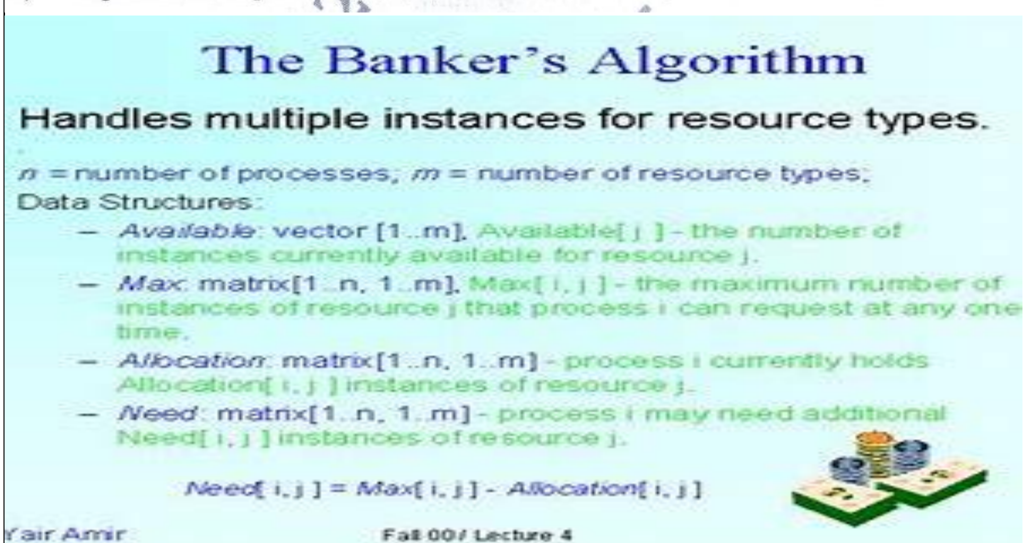
5c Consider the following snapshot of a system:

Process	Allocation				Max				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P_0	2	0	0	1	4	2	1	2	3	3	2	1
P_1	3	1	2	1	5	2	5	2				
P_2	2	1	0	3	2	3	1	6				
P_3	1	3	1	2	1	4	2	4				
P_4	1	4	3	2	3	6	6	5				

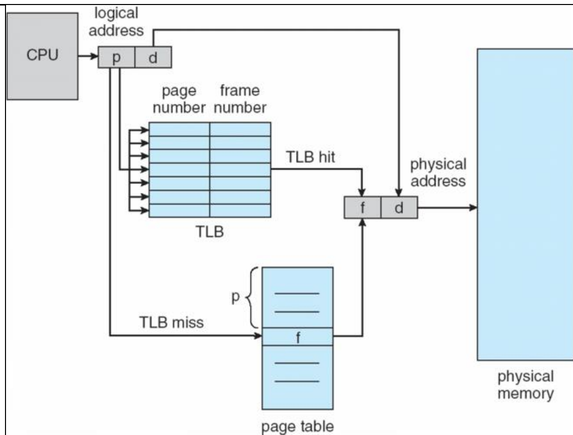
Answer the following using Banker's algorithm.

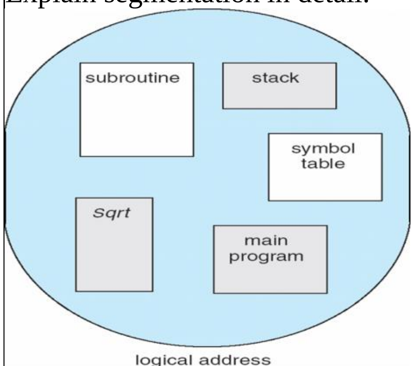
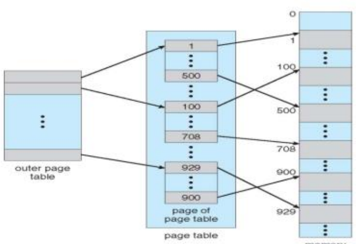
i) Is the system in safe state? If so, give the safe sequence.

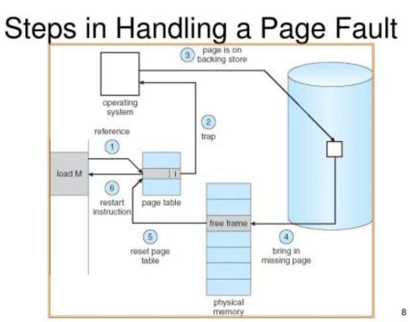
ii) If process P_2 requests (0, 1, 1, 3) resources can it be granted immediately? (



6a. Explain paging hardware with TLB 5m

<table>
<tr><td></td><td>

paging hardware with TLB. Explanation of figure given.</td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>6b</td><td>Explain segmentation in detail.

</td><td>5m</td></tr>
<tr><td>6c</td><td>Discuss structure of page table with suitable diagrams


structure of page table and brief about page table</td><td>10m</td></tr>
<tr><td></td><td>Module 4</td><td></td></tr>
<tr><td>7a</td><td>Describe the steps in handling page faults


Brief each step with the diagram given above.</td><td>6m</td></tr>
<tr><td>7b</td><td>Consider the page reference string for a memory with 3 frames. Determine the number of page faults using FIFO, optimal and LRU replacement algorithms.</td><td>5</td></tr>
</table>

FIFO FIRST IN FIRST OUT

PAGE DEMAND		6	2	3	4	2	1	5	6	2	1	2	3	7	6	3	4	1	2	7	6
	f1	6	6	6	6	6	1	1	1	1	1	1	3	3	3	3	3	3	2	2	2
ory	f2		2	2	2	2	2	5	5	5	5	5	5	7	7	7	7	7	7	7	6
	f3			3	3	3	3	3	6	6	6	6	6	6	6	6	4	4	4	4	4
	f4				4	4	4	4	4	2	2	2	2	2	2	2	2	1	1	1	1
		fault	fault	fault	fault	hit	fault	fault	fault	fault	hit	hit	fault	fault	hit	hit	fault	fault	fault	hit	fault

REFERENCE STRING 20
PAGE FAULTS 14
PAGE HITS

OPTIMAL PAGE REPLACEMENT

REFERENCE STRING		6	2	3	4	2	1	5	6	2	1	2	3	7	6	3	4	1	2	7	6
	f1	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
MM	f2		2	2	2	2	2	2	2	2	2	2	2	7	7	7	7	7	7	7	7
	f3			3	3	3	3	5	5	5	5	5	3	3	3	3	4	4	2	2	2
	f4				4	4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
		fault	fault	fault	fault	hit	fault	fault	hit	hit	hit	hit	fault	fault	hit	hit	fault	hit	fault	hit	hit

PAGE FAULT 10
PAGE HIT 10

LRU LEAST RECENTLY USED

		6	2	3	4	2	1	5	6	2	1	2	3	7	6	3	4	1	2	7	6
	f1	6	6	6	6	6	1	1	1	1	1	1	1	1	6	6	6	6	2	2	2
	f2		2	2	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	4	6
	f3			3	3	3	3	5	5	5	5	5	3	3	3	3	3	3	3	7	7
	f4				4	4	4	4	6	6	6	6	6	7	7	7	7	1	1	1	1
		fault	fault	fault	fault	hit	fault	fault	fault	hit	hit	hit	fault	fault	fault	hit	fault	fault	fault	fault	fault

PAGE FAULT 15
PAGE HIT 5

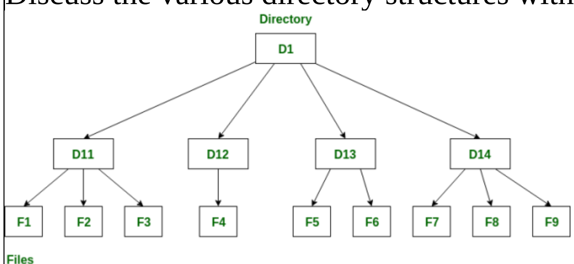
8a Explain the different allocation methods — 10

Contiguous Allocation

Linked Allocation

Indexed Allocation

8b Discuss the various directory structures with required diagrams. — 10



Directory structures expalantion

Module 5

9a Explain the access matrix method of system protection with domain as objects and its implementation — 5+5

domain \ object	F_1	F_2	F_3	laser printer	D_1	D_2	D_3	D_4
D_1	read		read			switch		
D_2				print			switch	switch
D_3		read	execute					
D_4	read write		read write		switch			

Access matrix explanation along with the matrix given.

9b A drive has 5000 cylinders,numbered 0 to 4999.The drive is currently Serving a request at cylinder 143,and the previous request was at cylinder 125. The queue of pending requests in FIFO order is 86,1470,913,1774,948,1509,1022,1750, 130.Starting from current head position,what is a total distance(in cylinders) that the Disk arm move to satisfy all a pending requests using FCFS, SSTF, LOOK AND C-LOOK Algorithms? — 10m

FCFS

The FCFS algorithm just follows the order of the requests given.

The FCFS schedule is:

143, 86, 1470, 913, 1774, 948, 1509, 1022, 1750, 130

The total distance is: 7,081

SSTF

The SSTF algorithm starts a cylinder 143 and from there successively selects the shortest request from its current location.

The SSTF schedule is:

143, 130, 86, 913, 948, 1022, 1470, 1509, 1750, 1774

The total distance is: 1,745

LOOK

The LOOK algorithm is just like the SCAN algorithm, except the disk head only goes as far as the last request in each direction.

The LOOK schedule is:

143, 913, 948, 1022, 1470, 1509, 1750, 1774, 130, 86

The total distance is: 3,319

C-LOOK

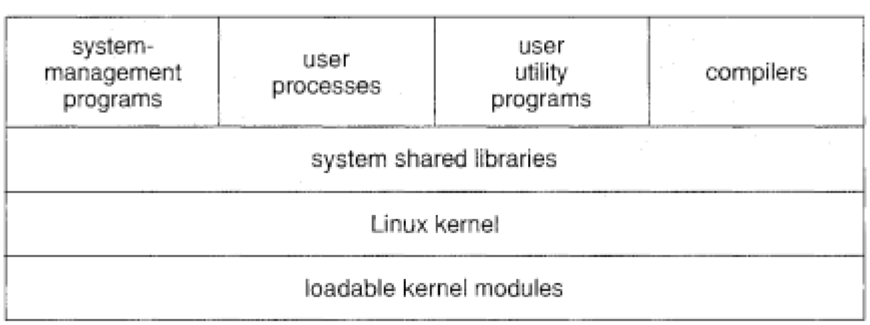
The C-LOOK algorithm is a circular version of the LOOK algorithm. It doesn't scan on the way back to the beginning of the disk, rather operates in a circular fashion.

The C-LOOK schedule is:

143, 913, 948, 1022, 1470, 1509, 1750, 1774, 86, 130

The total distance is: 3,363

10a Write a neat diagram, explain the components of a Linux system 8m



Kernel. The kernel is responsible for maintaining all the important abstractions of the operating system, including such things as virtual memory and processes.

System libraries. The system libraries define a standard set of functions through which applications can interact with the kernel. These functions implement much of the operating-system functionality that does not need the full privileges of kernel code.

	System utilities. The system utilities are programs that perform individual, specialized management tasks. Some system utilities may be invoked just once to initialize and configure some aspect of the system; othersknown as daemons in	
10b	Explain the different IPC mechanisms available in Linux. • Synchronization and Signals • Passing of Data Among Processes	6m
10c	Discuss about scheduling in Linux • Process Scheduling • Kernel Synchronization • Symmetric Multiprocessing	6m