Question 1(16 marks)

Explain

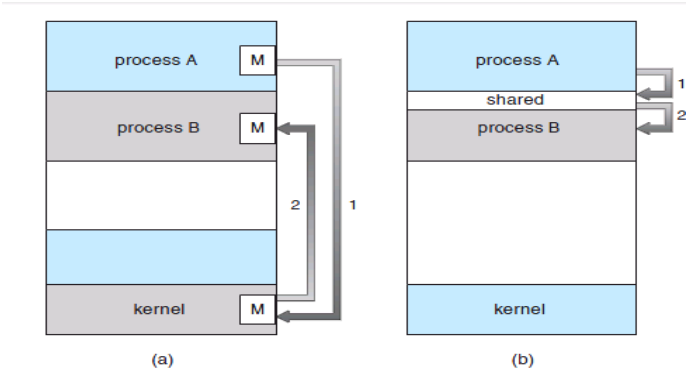
a) The models of interprocess communication.

b) Simple structure approach and layered approach of operating system

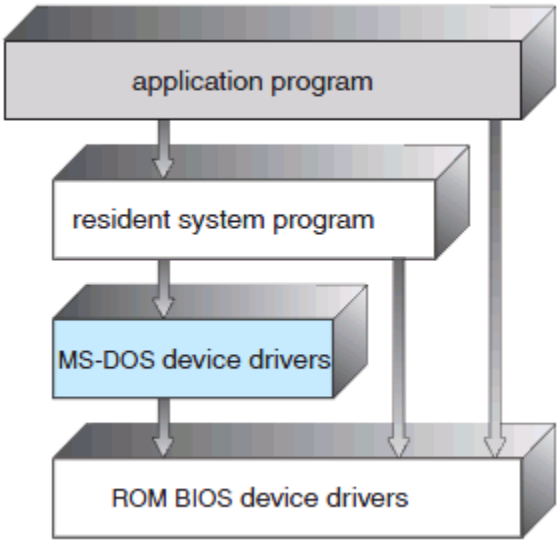
c) PCB and Context Switch

d) The difference between clustered systems and multiprocessor systems

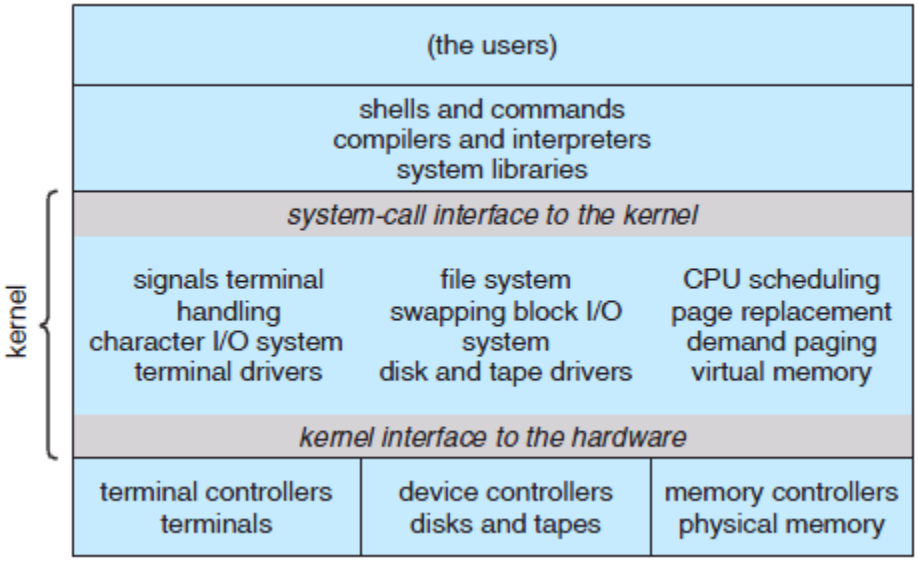
a. The two models of interprocess communication are message passing model and the shared-memory model.

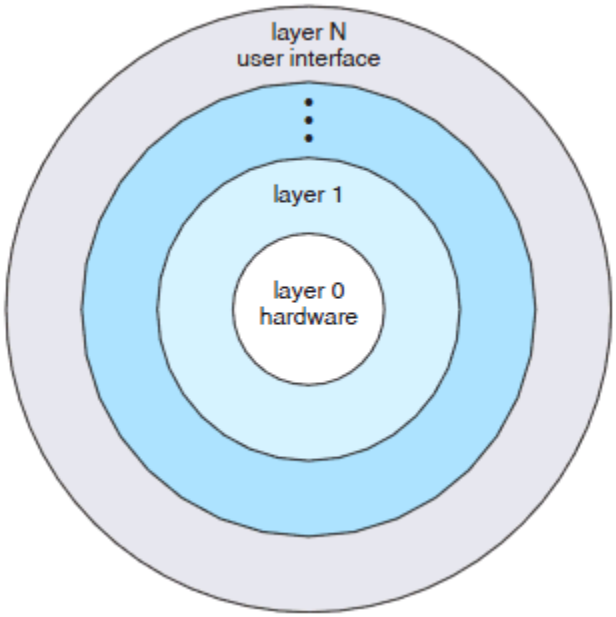


Communications models. (a) Message passing. (b) Shared mem


application program
resident system program
MS-DOS device drivers
ROM BIOS device drivers

b.


(the users)
shells and commands
compilers and interpreters
system libraries
system-call interface to the kernel
signals terminal handling
character I/O system
terminal drivers
file system
swapping block I/O system
disk and tape drivers
CPU scheduling
page replacement
demand paging
virtual memory
kernel interface to the hardware
terminal controllers
terminals
device controllers
disks and tapes
memory controllers
physical memory
kernel



C.

S.N.	Information & Description
1	**Process State** The current state of the process i.e., whether it is ready, running, waiting, or whatever.
2	**Process privileges** This is required to allow/disallow access to system resources.
3	**Process ID** Unique identification for each of the process in the operating system.
4	**Pointer**

	A pointer to parent process.
5	**Program Counter** Program Counter is a pointer to the address of the next instruction to be executed for this process.
6	**CPU registers** Various CPU registers where process need to be stored for execution for running state.
7	**CPU Scheduling Information** Process priority and other scheduling information which is required to schedule the process.
8	**Memory management information** This includes the information of page table, memory limits, Segment table depending on memory used by the operating system.
9	**Accounting information** This includes the amount of CPU used for process execution, time limits, execution ID etc.
10	**IO status information** This includes a list of I/O devices allocated to the process.

Context Switch

A context switch is the mechanism to store and restore the state or context of a CPU in Process Control block so that a process execution can be resumed from the same point at a later time.

d)

Answer: Clustered systems are typically constructed by combining multiple computers into a single system to perform a computational task distributed across the cluster. Multiprocessor systems on the other hand could be a single physical entity comprising of multiple CPUs. A clustered system is less tightly coupled than a multiprocessor system. Clustered systems communicate using messages, while processors in a multiprocessor system could communicate using shared memory. In order for two machines to provide a highly available service, the state on the two machines should be replicated and should be consistently updated. When one of the machines fail, the other could then take-over the functionality of the failed machine.

Questin2(***16 marks***)
Describe the difference between

a) Long term scheduler and short term scheduler.
b) Producer-consumer problem
c) OPT and LRU
d) User-Level & Kernel-Level Thread

a)

S.N.	Long-Term Scheduler	Short-Term Scheduler
1	It is a job scheduler	It is a CPU scheduler
2	Speed is lesser than short term scheduler	Speed is fastest among other two
3	It controls the degree of multiprogramming	It provides lesser control over degree of multiprogramming

4	It is almost absent or minimal in time sharing system	It is also minimal in time sharing system
5	It selects processes from pool and loads them into memory for execution	It selects those processes which are ready to execute

b)

c)

Least Recently Used (LRU) algorithm

- Page which has not been used for the longest time in main memory is the one which will be selected for replacement.
- Easy to implement, keep a list, replace pages by looking back into time.

d)

S.N.	User-Level Threads	Kernel-Level Thread
1	User-level threads are faster to create and manage.	Kernel-level threads are slower to create and manage.
2	Implementation is by a thread library at the user level.	Operating system supports creation of Kernel threads.
3	User-level thread is generic and can run on any operating system.	Kernel-level thread is specific to the operating system.
4	Multi-threaded applications cannot take advantage of multiprocessing.	Kernel routines themselves can be multithreaded.

Question3 (***20 marks***)

Process	Burst Time	Priority
P_1	10	3
P_2	1	1
P_3	2	3
P_4	1	4
P_5	5	2

The processes are assumed to have arrived in the order P1, P2, P3, P4, P5, all at time 0.

a) What is the turnaround time of each process for the scheduling algorithms FCFS, RR (Quantum=1), SJF, and Priority

b) What is the waiting time of each process for each of the scheduling algorithms in part a?

c) Which of the schedules in part a results in the minimal average waiting time (over all processes)?

b. Turnaround time

	FCFS	RR	SJF	Priority
P_1	10	19	19	16
P_2	11	2	1	1
P_3	13	7	4	18
P_4	14	4	2	19
P_5	19	14	9	6

c. Waiting time (turnaround time minus burst time)

	FCFS	RR	SJF	Priority
P_1	0	9	9	6
P_2	10	1	0	0
P_3	11	5	2	16
P_4	13	3	1	18
P_5	14	9	4	1

d. Shortest Job First

Question4 (***18 marks***)

a) Describe the Producer/Consumer problem and Show a possible solution to this problem.

b) Explain the process using **fork( )** system call

a) A **producer** process produces information that is consumed by a **consumer** process. For example, a compiler may produce assembly code, which is consumed by an assembler. The assembler, in turn, may produce object modules, which are consumed by the loader. The producer–consumer problem also provides a useful metaphor for the client–server paradigm. We generally think of a server as a producer and a client as a consumer. For example, a Web server produces (that is, provides) HTML files and images, which are

consumed (that is, read) by the client Web browser requesting the resource. One solution to the producer–consumer problem uses shared memory. To allow producer and consumer processes to run concurrently, we must have available a buffer of items that can be filled by the producer and emptied by the consumer. This buffer will reside in a region of memory that is shared by the producer and consumer processes. A producer can produce one item while the consumer is consuming another item. The producer and consumer must be synchronized, so that the consumer does not try to consume an item that has not yet been produced.

```
while (true) {
     /* produce an item in nextProduced */
     while (((in + 1) % BUFFER_SIZE) == out)
        ; /* do nothing */
     buffer[in] = nextProduced;
     in = (in + 1) % BUFFER_SIZE;
}
```

c) The producer process.

```
item nextConsumed;

while (true) {
     while (in == out)
        ; // do nothing

     nextConsumed = buffer[out];
     out = (out + 1) % BUFFER_SIZE;
     /* consume the item in nextConsumed */
}
```

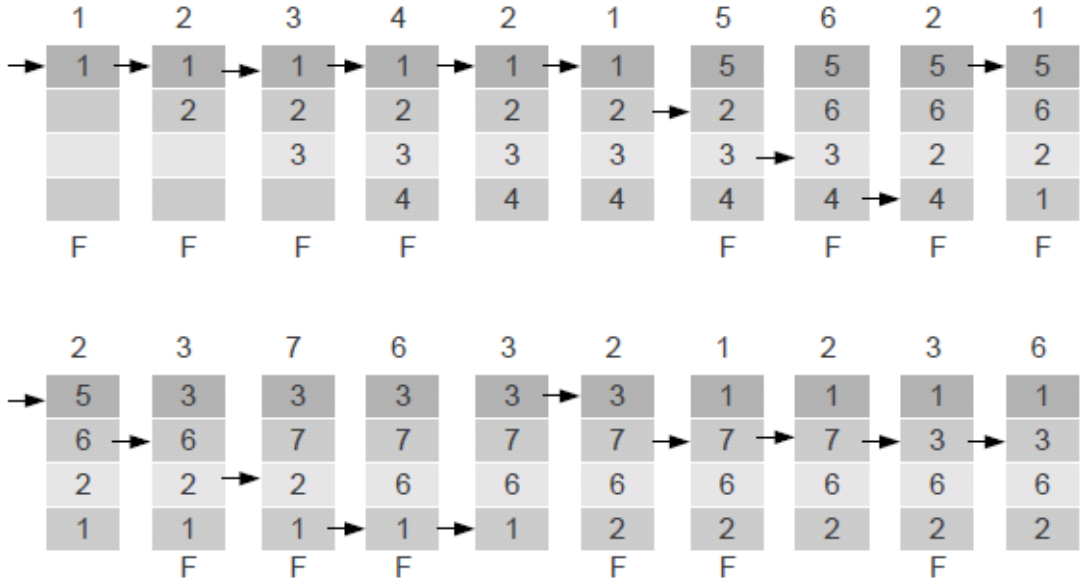
d) The consumer process.

e)

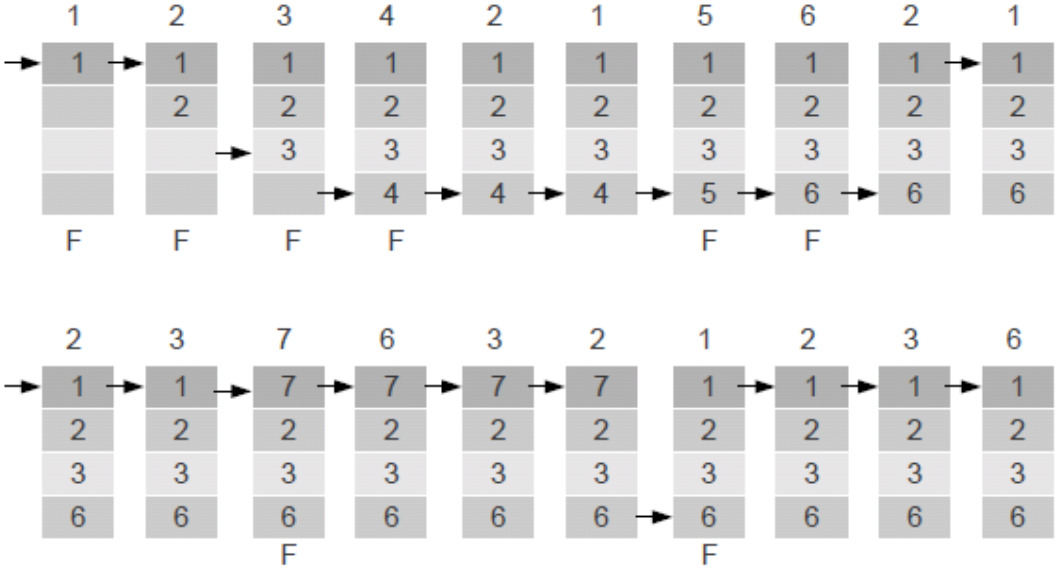
Question5 (***20 marks***)

What is the result of FIFO and Optimal page replacement algorithm for the following reference string with four page frames 1,2,3,4,2,1,5,6,2,1,2,3,7,6,3,2,1,2,3,6

FIFO



OPT



Good Luck