Department of CS&E, MIT, Manipal

I Test

V Sem B Tech(CSE)

Operating Systems and Linux (CSE 309)

TIME : 1 HOUR 13-09-2014 MAX.MARKS : 20

Note: Answer all the questions

Evaluation Scheme

1. Multiple Choice Questions

1A. A system is a multi-programming system because
   a. There are multiple devices connected to the system
   b. There are multiple clients who are using the system.
   c. There are multiple memory resident ready-to-run programs in the system
   d. There are multiple processors in the system

1B. ______ selects processes from a pool of processes at mass storage device and loads them into memory
   a. Long-term scheduler
   b. Short-term scheduler
   c. Medium-term scheduler
   d. Process Control Block

1C. Including the initial parent process, how many processes will be created by the following program?
   int main() { fork(); fork(); fork(); fork(); return 0; }
   a. 5 b. 16 c. 12 d. 7

1D. State which of the following is true:
   a. Dispatcher gives control of CPU to the process selected by medium-term scheduler
   b. In a real system, CPU utilization should range from 80 to 100 percent
   c. A major problem with priority scheduling algorithms is definite blocking
   d. In RR scheduling, average turnaround time can be improved, if most processes finish their next CPU burst in a single time quantum

1E. State which of the following is false:
   a. All unsafe states are deadlocks
   b. In general, we cannot prevent deadlocks by denying the mutual-exclusion condition
   c. If resource preemption is used to deal with deadlocks, then it must address starvation
   d. Deadlock can be prevented by allowing a process to request resources only when it has none (1X5)

Ans: 1A. c 1B. a 1C. b 1D. d 1E. a

2A. Name certain services provided by the operating system services that are not for helping the user but for ensuring efficient operations of system. List the advantages and disadvantages of layered architecture of OS design. 2 Marks

Ans: 1) Resource Allocation
2) Accounting
3) Protection
4) Security **½ Mark**

Advantages:

i) Simplicity of construction and debugging.
ii) Each layer is implemented with only those operations provided by lower level layers.
iii) A layer does not need to know how these operations are implemented; it needs to know only what these operations do.
iv) Each uses functions (operations) and services of only lower-level layers. This approach simplifies debugging.
v) Each layer hides the existence of certain data structures, operations, and hardware from higher-level layers.

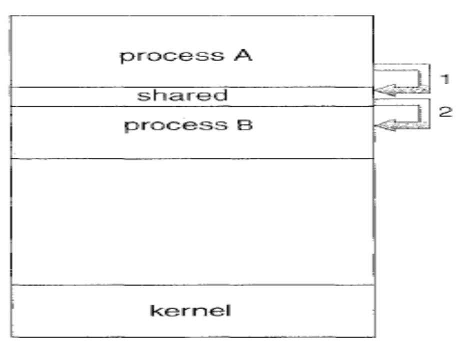
1 Mark

Diadvantages:

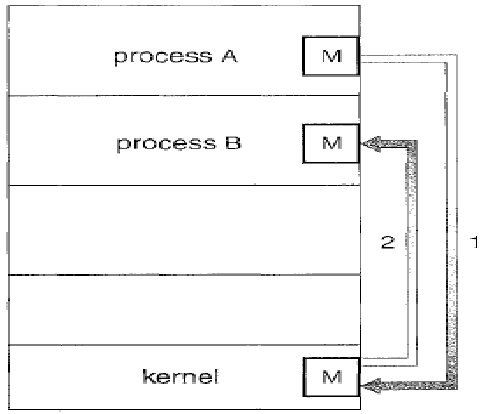
i) Defining the various layers. Careful planning is necessary because a layer can use only lower-level layers.
ii) Layered approach tend to be less efficient than other types. **½ Mark**

2B. What are the fundamental models of IPC mechanism that will allow cooperating processes to exchange data? Explain with diagram. List the pros and cons among them. Also explain the process scheduling with queuing diagram. 3 Marks

Ans: Cooperating processes require an interprocess communication (IPC) mechanism that will allow them to exchange data and information. There are two fundamental models of IPC are
(1) shared memory and
(2) message passing.



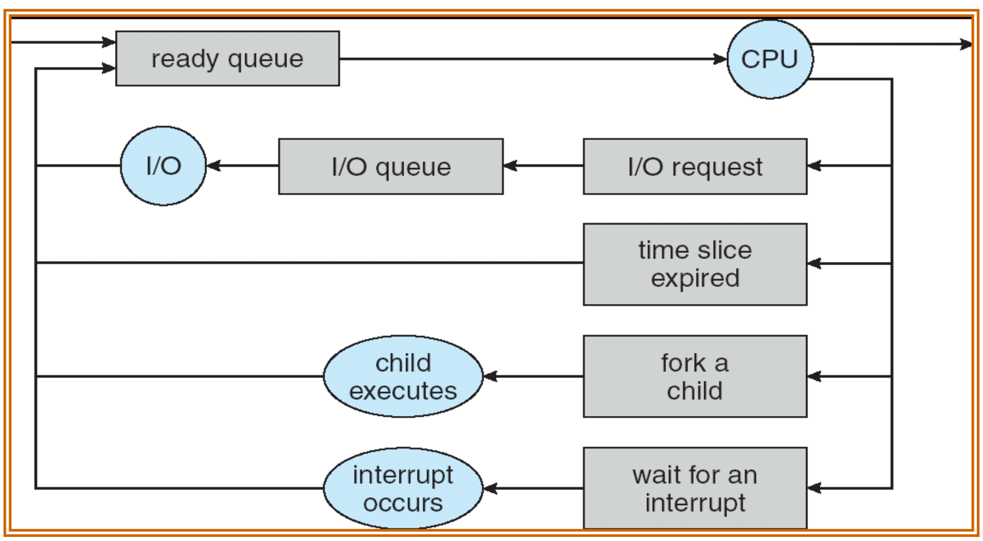
In above diagram of shared-memory model, a region of memory that is shared by cooperating processes is established. Processes can then exchange information by reading and writing data to the shared region.



In the message passing model, communication takes place by means of messages exchanged between the cooperating processes. **1 Mark**

The pros and cons among them are

1. Message passing is useful for exchanging smaller amounts of data.
2. Message passing is easier to implement than is shared memory.
3. Shared memory allows maximum speed and convenience of communication.
4. Shared memory is faster than message passing, as message passing systems are typically implemented using system calls and thus require the more time-consuming task of kernel intervention.
5. In shared memory systems, system calls are required only to establish shared-memory regions. Once shared memory is established, all accesses are treated as routine memory accesses, and no assistance from the kernel is required. **1 Mark**



A common representation of process scheduling is a queueing diagram. Each rectangular box repr a queue. Two types of queues: the ready queue and a set of device queues.
The circles represent the resources that serve the queues. Arrows indicate the flow of processes in the system. A new process is initially put in the ready queue. It waits there until it is selected for execution, or is dispatched. Once the process is allocated the CPU and is executing, one of several events may occur:

i) The process could issue an I/0 request and then be placed in an I/0 queue.

ii) Time slot allocated may lapse and process may be taken back from CPU and put in ready queue.

iii) The process could create a new subprocess and wait for the subprocess's termination.

iv) The process could be removed forcibly from the CPU, as a result of an interrupt, and be put back in the ready queue. **1 Mark**

3A. With the help of a Gantt Chart (Use the data given below) calculate Average Waiting Time and Turnaround time for a SRTF scheduler.

Process	**Arrival Time**	**Burst Time(msecs)**
P1	0	6
P2	2	4
P3	4	3
P4	5	1

Ans: Gantt Chart: 0.5M

P_1	P_4	P_3	P_2

0 6 7 10 14

0.25M for calculating TAT of each Process. Total (0.25 * 4)= 1M

0.25M for calculating WT for each process. Total (0.25 *4) =1M

0.25M for calculating average value for TAT and WT. Total (0.25 *2)= 0.5M

TAT = AT – FT

WT = TAT - BT

	Arrival Time	**Burst Time**	**Finish Time**	**Turn Around Time**	**Waiting Time**
$\mathbf{P_1}$	0	6	6	6	0
$\mathbf{P_2}$	2	4	14	12	8
$\mathbf{P_3}$	4	3	10	6	3
$\mathbf{P_4}$	5	1	7	2	1
	Average Value			**6.5 ms**	**3ms**

3B. Consider the exponential average formula used to predict the length of the next CPU burst i.e τ_1. What are the implications of assigning the following values to the parameters used by the algorithm.

a. $\alpha = 0$ and $\tau_0 = 100$ ms.
b. $\alpha = 1$ and $\tau_0 = 10$ ms.

Ans: Formula is $\tau_{n+1} = \alpha\, t_n + (1-\alpha)\tau_n$.

a. $\alpha = 0$

$\tau_{n+1} = \tau_n$. Therefore $\tau_1 = \tau_0$. So recent history does not count so τ_1

b. $\alpha = 1$

$\tau_{n+1} = \alpha\, t_{n.}$ Only the actual last CPU burst counts

Each step 1M.

4a) For the following set

	Allocation				Max				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P1	0	0	1	2	0	3	2	2	1	5	2	0
P2	1	0	0	0	1	7	5	0				
P3	1	3	5	4	2	3	5	6				
P4	0	6	3	2	0	6	5	2				
P5	0	0	1	4	0	6	5	6				

i. Write the content of the matrix Need?

Need				
	A	B	C	D
P1	0	3	1	0
P2	0	7	5	0
P3	1	0	0	2
P4	0	0	2	0
P5	0	6	4	2

//1 mark for the matrix

ii. Is the system safe? If so what is the sequence?
Yes the system is safe and we can say that since we have a safe sequence.
//Yes – 0.5 mark

The safe sequence is
P1 -> P3 -> P4 -> P5 -> P2 //0.5 mark – there can be multiple correct
p4-> p3 -> p1 -> p2 ->p5 // answers
p4->(followed by any order)

iii. If the request from a process P2 arrives for (0,4,2,0) can the request be granted immediately? Write the contents of the need matrix
The request cannot be granted immediately as there is no safe sequence by observing the need matrix. //0.5 mark
The content of need matrix is

Need				
	A	B	C	D
P1	0	3	1	0
P2	0	3	3	0
P3	1	0	0	2
P4	0	0	2	0
P5	0	6	4	2

4b) Explain 2 ways of recovering from deadlock in detail. 2 marks

1) **Process termination** – To eliminate deadlocks by aborting a process we use one of the methods
   a. Abort all deadlocked process – This will break the deadlocked cycle but at a great expense
   b. Abort one process at a time until the deadlocked cycle is terminated – This methods incurs considerable overhead, since after each process is aborted a deadlock detection algorithm must be invoked
2) **Resource Preemption** – We successively preempt resources and give these resources to other process until deadlock cycle is broken. If pre-emption is required to deal with deadlocks then three issues needs to be addressed
   a. Selecting a victim – We need to make a decision as to which resource and which processes are to be pre-empted so as to minimize the cost
   b. Rollback – Till what state should we rollback the process needs to be decided and we need to maintain the state of the system for this purpose. Since maintaining the state is difficult affair we can do a total rollback.
   c. Starvation – We need to make sure that the same process will not be pre-empted always. We need to take account of number of times a particular process is preempted into account.