Bharatiya Vidya Bhavan's
SARDAR PATEL INSTITUTE OF TECHNOLOGY

DEPARTMENT OF ELECTRONICS AND TELECOMMUNICATION

ESE

Class: - TE EXTC **Sem: - VI** **Total Marks: - 100**

Duration: - 3 Hour **Subject: - Operating System (ETC605)** **Year: - 2017-18**

1) (a) What is Kernel? Discuss different types of kernels. [CO1] [10]

Definition along with its functions................... [4]

Two types.. (1) Layered (2) Microkernel (diagrams needed)...........[6]

(b) List all the fields of Process Control Block (PCB). State clearly the necessity of each of these fields. [CO2] [10]

Total 8 different fields of PCB (may be nine). Context switching is the major reason for having PCB for each process apart from the individual field requirement.

2) (a) Consider a following set of processes, with length of CPU bursts given in milliseconds as follows:

Process	Burst Time	Arrival Time	Priority
P1	8	0	3
P2	1	1	1
P3	2	2	2
P4	3	3	3
P5	6	4	4

Draw the Gantt charts for FCFS, Preemptive SJF, Preemptive priority and RR (Quantum=2)
What is the waiting time of each process for each of the above algorithms?
What is the turnaround time of each process for each of the above algorithms?
Which algorithm results in the minimum average waiting time? [CO2] [10]

OR

Consider a following set of processes, with length of CPU bursts given in milliseconds as follows:

Process	Burst Time	Arrival Time	Priority
P1	8	0	3
P2	4	1	4

P3	9	2	2
P4	5	3	1

Draw the Gantt charts for FCFS, Preemptive SJF, Preemptive priority and RR (Quantum=2)
What is the waiting time of each process for each of the above algorithms?
What is the turnaround time of each process for each of the above algorithms?
Which algorithm results in the minimum average waiting time? [CO2] [10]

2) (b) In UNIX, each file is associated with inode. File allocation is an important operation required to use the space efficiently and to access it with high speed. Justify the use of inode in order to satisfy these two requirements. Support your answer with neat diagram of the inode structure. [CO3] [10]

UNIX inode structure has 15 block pointers. First 12 are direct data block pointers. 13th slotis known as single indirect pointer. 14th slot is known as double indirect pointer. 15th slot is known as triple indirect pointer. The complete detailed diagram of the inode structure is to be drawn.

OR

Discuss security features in UNIX OS. [CO3] [10]

User Accounts

File Permissions

Data Verification

Encrypted Storage

Secure Remote Access with OpenSSH

Software Management

Host Integrity Testing

System Recovery

Resource Allocation Controls

Monitoring and Audit Facilities

The System Firewall

Application Isolation

A Note on Viruses and Malware

P3	9	2	2
P4	5	3	1

Draw the Gantt charts for FCFS, Preemptive SJF, Preemptive priority and RR (Quantum=2)
What is the waiting time of each process for each of the above algorithms?
What is the turnaround time of each process for each of the above algorithms?
Which algorithm results in the minimum average waiting time? [CO2] [10]

2) (b) In UNIX, each file is associated with inode. File allocation is an important operation required to use the space efficiently and to access it with high speed. Justify the use of inode in order to satisfy these two requirements. Support your answer with neat diagram of the inode structure. [CO3] [10]

UNIX inode structure has 15 block pointers. First 12 are direct data block pointers. 13^{th} slotis known as single indirect pointer. 14^{th} slot is known as double indirect pointer. 15^{th} slot is known as triple indirect pointer. The complete detailed diagram of the inode structure is to be drawn.

OR

Discuss security features in UNIX OS. [CO3] [10]

User Accounts

File Permissions

Data Verification

Encrypted Storage

Secure Remote Access with OpenSSH

Software Management

Host Integrity Testing

System Recovery

Resource Allocation Controls

Monitoring and Audit Facilities

The System Firewall

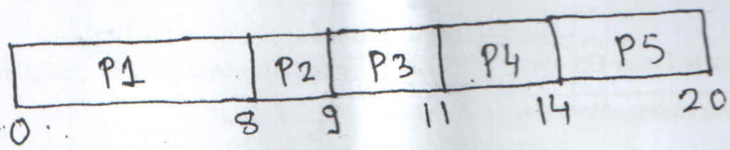
Application Isolation

A Note on Viruses and Malware

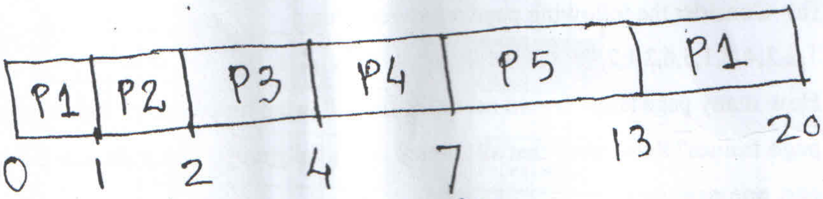
Q.2 ⓐ (First Part)

Grantt charts

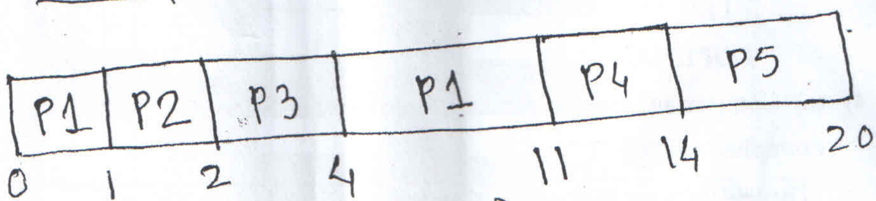
① FCFS:-



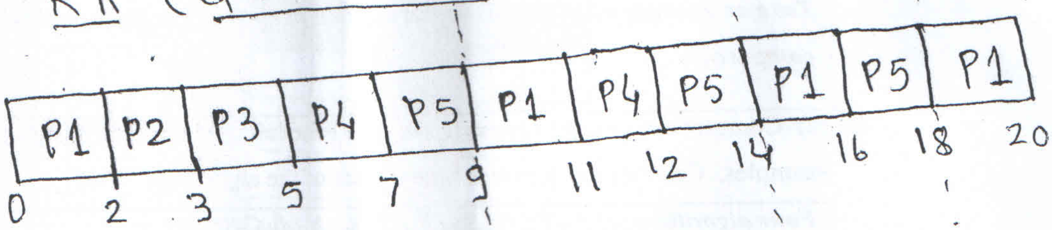
② Preemptive SJF:-



③ Preemptive Priority:-



④ RR (Quantum = 2)



Process	Burst Time	Arrival Time	Priority	FCFS		Pre-SJF		Pre-Priority		RR (Q=2)	
				TW	TA	TW	TA	TW	TA	TW	TA
P1	8	0	3	0	8	12	20	3	11	12	20
P2	1	1	1	7	8	0	1	0	1	1	2
P3	2	2	2	7	9	0	2	0	2	1	3
P4	3	3	3	8	11	1	4	8	11	6	9
P5	6	4	4	10	16	3	9	10	16	8	14
Average value =				6.4		3.2		4.2		5.6	

Q. 2 (a) (Second Part)

Gantt charts

① FCFS:-

P1	P2	P3	P4

0 8 12 21 26

② Preemptive SJF:-

P1	P2	P4	P1	P3

0 1 5 10 17 26

③ Preemptive Priority:-

P1	P3	P4	P3	P1	P2

0 2 3 8 16 22 26

④ RR (Quantum) = 2

P1	P2	P3	P4	P1	P2	P3	P4	P1	P3	P4	P1	P3	P3

0 2 4 6 8 10 12 14 16 18 20 21 23 25 26

Process	Burst Time	Arrival Time	Priority	FCFS		Pre-SJF		Pre-Priority		RR (Q=2)	
				TW	TA	TW	TA	TW	TA	TW	TA
P1	8	0	3	0	8	9	17	14	22	15	23
P2	4	1	4	7	11	0	4	21	25	7	11
P3	9	2	2	10	19	15	24	5	14	15	24
P4	5	3	1	18	23	2	7	0	5	13	18
Average =				8.75		6.5		10		12.5	

3) (a) Discuss, with respect to Linux OS, the virtual memory management scheme (Paging scheme). [CO3]

[10]

Memory is divided into segments. Then a segment can be divided into pages of fixed size.   ***Hence, first describe the segment translation process with diagram. [4]***   ***Second, describe the page translation process with diagram. [6]***

OR

With respect to Linux OS, Draw and discuss the process state transition diagram. [CO3] [10]

Five process states: New, Ready, Running, Blocked, Halted   ***Draw the process state transition diagram with justification.***

3) (b) Consider the following page reference string:

1,2,3,4,2,1,5,6,2,1,2,3,7,6,3,2,1,2,3,6

How many page faults would occur for the following replacement algorithms, assuming four page frames? Remember that all frames are initially empty, so your first unique pages will cost one page each.

FIFO

LRU

OPTIMAL [CO2]

[10]

4) (a) Compare and contrast various allocation methods for files. Justify the answer with examples. [CO2]

[10]

Three Allocation Methods (a) Contiguous (2) Linked (3) Indexed   ***Take an example with diagram representation in each of the three methods for the comparison.***

4) (b) Compare and contrast various Disks Arm Scheduling algorithms. Justify the answer with examples. Calculate average seek time in each of the algorithms. [CO2] [10]

Four algorithms (a) FCFS (b) SSTF (c) SCAN (d) C-SCAN   ***Take an example with diagram and table for calculating the seek time in each of the four algorithms for the comparison.***

5) (a) Give an example of Priority inversion, and how to overcome for a three task system. [CO3]

[10]

(b) Consider a system with two tasks, which we'll call Task 1 and Task 2. Assume these are both periodic tasks with periods T1 and T2, and each has a deadline that is the beginning of its next cycle.
Task 1 has P1 = 50 ms, and an execution time of C1 = 25 ms. Task 2 has P2 = 100 ms and C2 = 40 ms. Consider static priority scheduling (RMA), draw the time space diagram and calculate processor utilization, schedulability. [CO3]

[10]

Q.3 (b)

1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
FIFO (Page Faults = 14)																			
1*	1*	1*	1*	1*	1*	5	5	5	5*	5*	3	3	3	3	3*	1	1	1	1
	2	2	2	2	2	2*	6	6	6	6	6*	7	7	7	7	7*	7*	3	3
		3	3	3	3	3	3*	2	2	2	2	2*	6	6	6	6	6	6*	6*
			4	4	4	4	4	4*	1	1	1	1	1*	1*	2	2	2	2	2
LRU (Page Faults = 10)																			
1*	1*	1*	1*	1*	1	1	1	1*	1	1	1	1*	6	6	6	6*	6*	6*	6
	2	2	2	2	2	2	2*	2	2	2	2	2	2*	2*	2	2	2	2	2
		3	3	3	3*	5	5	5	5*	5*	3	3	3	3	3	3	3	3	3
			4	4	4	4*	6	6	6	6	6*	7	7	7	7*	1	1	1	1*
OPTIMAL (Page Faults = 8)																			
1*	1*	1	1	1	1	1	1	1	1*	1*	1*	7*	7*	7*	7*	1*	1*	1*	1*
	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
		3*	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
			4*	4*	4*	5*	6*	6*	6	6	6	6	6	6	6	6	6	6	6

6

5 (a) Ans:

Priority Inheritance provides a "solution" to the problem of priority inversion.

Basic rule: When a high-priority (HP) thread attempts to lock a mutex already locked by a lower-priority (LP) thread, the Priority Inheritance Protocol (PIP) temporarily raises the priority of the low-priority thread to match that of the blocked thread until the low-priority thread unlocks the mutex.

The LP task will complete very fast and release mutex which can be acquired by the HP task.

5 (b) Ans:

Utilization:

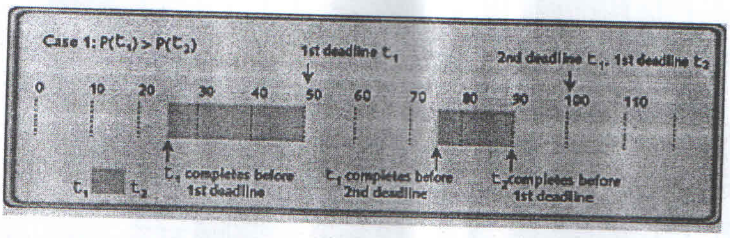
$Ui = \Sigma\, c_i / p_i$; Thus U1 = .5 and U2 = .4

This means total requested utilization Ui = U1 + U2 = 0.9

Schedule bound:

$$U = n * (2^{(1/n)} - 1) = 0.828$$

RMS schedule always exists if Ui < U



$\boxed{\text{I}}$