CMR
INSTITUTE OF
TECHNOLOGY

SCHEME AND SOLUTION

Internal Assessment Test I – NOV 2017

Sub:	OPERATING SYSTEMS					**Code:**15EC553		
Date:	8/ 11 /2017	Duration:	90 mins	Max Marks:	50	**Sem:**	V	**Branch:ECE(A)/TCE (A &B)**

Note: Answer any five questions:

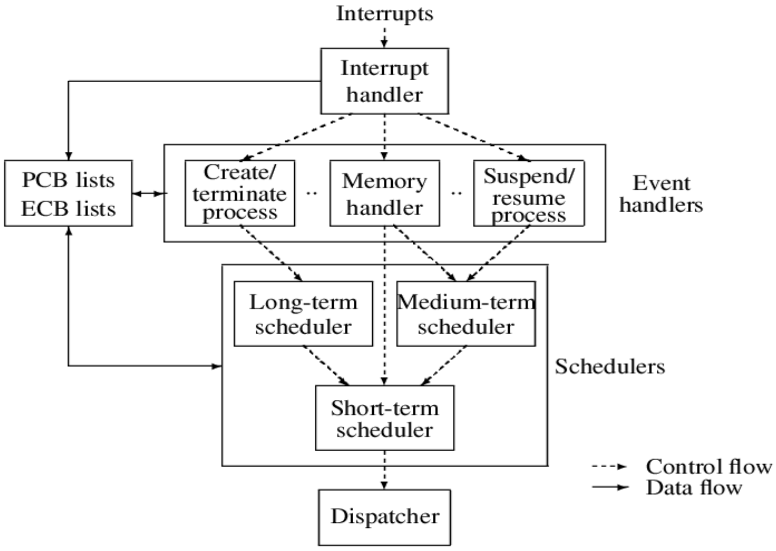
1a Explain the operation of long, medium and short term schedulers with neat diagrams . — 6M

Explanation -4M
Diagram-2M

A single scheduler cannot provide the desired combination of performance and user service, so an OS uses three schedulers

- Long-term scheduler
  - * Decides when to admit an arrived process
    - Uses nature of a process, availability of resources to decide
- Medium-term scheduler
  - * Performs swapping
    - Maintains a sufficient number of processes in memory
- Short-term scheduler
  Decides which ready process should operate on the CPU

- An event handler passes control to the long- or medium-term scheduler
- These schedulers pass control to the short-term scheduler

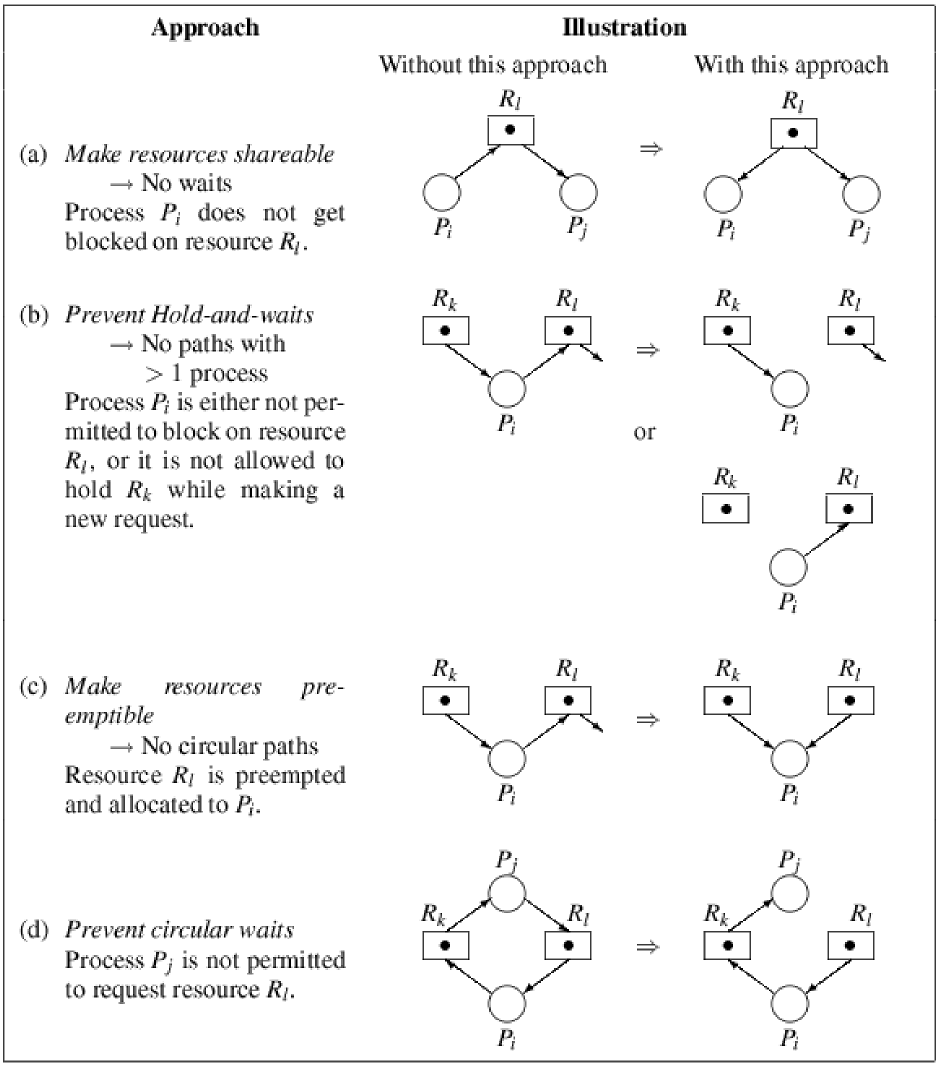


1b Write Short notes on a)Direct and Indirect Naming. b)Blocking and Non blocking Sends. — 4M

Explanation about Direct and Indirect Naming-2M
Explanation about Blocking and Non blocking Sends-2M

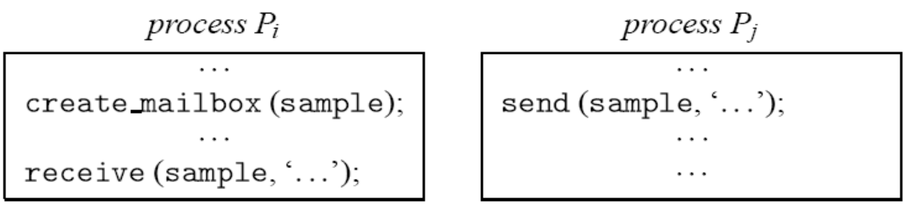
- * Direct and indirect naming
  - Direct: Process names are specified in send / receive commands
  - Indirect: Process names are inferred by the kernel
  - Synchronous message passing: sender is blocked
    - » Simplifies message passing, saves memory
  - Asynchronous message passing: sender is not blocked

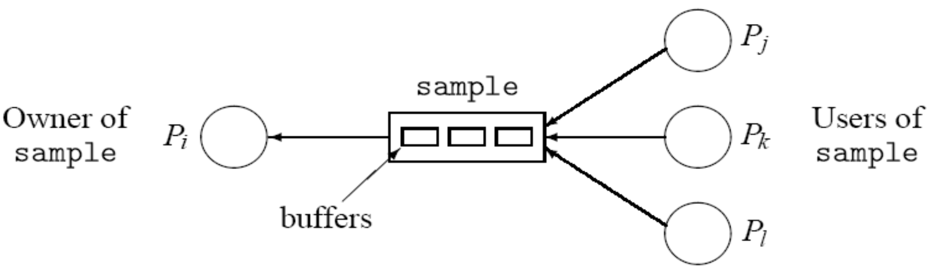
2a	Explain the various issues in message passing with exceptional conditions in detail. **Explanation about issues-2 M** **Explanation about exeptional conditions-4M** Issues 1. Naming of processes. 2. Method for transferring messages. 3 .Kernel responsibilities. Exceptional conditions 1.Destination process donot exist 2.Receiving process doesnot exist 3.buffer full condition 4.No message exist	6M
2b	Compare Contiguous and Non Contiguous memory allocation. **4 differences each carrying 4M** Contiguous — Noncontiguous 1.Overhead — No — address translation 2.Allocation — single area — several areas 3.Reuse of memory — Internal fragmentation — paging-No external, Segmentation-No internal 4.Swapping — relocation register — swapped in process can be placed anywhere.	4M
3	Mention the various deadlock handling methods.Explain the deadlock prevention approach by ensuring the deniel of 4 necessary conditions of deadlocks with appropriate diagrams. **Mentioning 3 methods-2M** **Explnation with diagram-8M** 1.Deadlock detection and resolution 2.Deadlock prevention 3.Deadlock avoidance • Use resource allocation policies that make deadlocks impossible – How to design a deadlock prevention approach? * Consider the conditions for deadlock * Ensure that they cannot hold simultaneously; i.e., make sure that one of them cannot arise – A simple policy * Allocate all resources required by a process together. Hence the hold-and-wait condition is never satisfied ▪ This policy is expensive in practice as resources may be requested much before they are actually needed by a process	10M



4 Explain the operation of mailboxes with appropriate system calls and diagrams.Also mention its advantages considering an example of airline reservation server using three mailboxes. 10M

Explanation about system calls-2M
Mailbox diagram→-2M
Explanation-4M
Psedocode-2M





- A mailbox has a name and is a repository of messages

	• Processes connect to a mailbox and send / receive messages from it
	• Benefits of using a mailbox
	– Anonimity of receiver
	* A sender process need not know identity of receiver process
	– Classification of messages
	* A process can use numerous mailboxes, one for each kind of message
	*
	• An airline reservations process uses three mailboxes– *book, enquire, receive*
	• This way it can process cancellations followed by bookings followed by queries
	
	repeat
	while `receive` (*book, flags*$_1$, *msg_area*$_1$) returns a message
	while `receive` (*cancel, flags*$_2$, *msg_area*$_2$) returns a message
	process the cancellation;
	process the booking;
	if `receive` (*enquire, flags*$_3$, *msg_area*$_3$) returns a message **then**
	while `receive` (*cancel, flags*$_2$, *msg_area*$_2$) returns a message
	process the cancellation;
	process the enquiry;
	forever	

5	Consider the following snapshot of resource allocation at time t1 for a system which has 4 processes P1,P2,P3 and P4 with R1-5, R2-7 and R3-5 units of resource classes respectively. a) Show that the system is not deadlocked by generating a safe sequence. b) At an instance of time t2, process P2 requests for an additional resource of R1 class. Show that the system is in deadlock if the request is granted and show the deadlock processes.	10M

	R1	R2	R3
P1	2	1	0
P2	1	3	1
P3	1	1	1
P4	1	2	2

Allocated resources

	R1	R2	R3
P1	2	1	3
P2	1	4	0
P3			
P4	1	0	2

Requested resources

Solving the safe sequence-8M
Showing the deadlock condition-2M

(a) Initial state

	Allocated resources R1	R2	R3	Requested resources R1	R2	R3
P1	2	1	0	2	1	3
P2	1	3	1	1	4	0
P3	1	1	1			
P4	1	2	2	1	0	2

Free resources (R1 R2 R3): 0 0 1

(b) After simulating allocation of resources to P_4 when process P_3 completes

	Allocated resources R1	R2	R3	Requested resources R1	R2	R3
P1	2	1	0	2	1	3
P2	1	3	1	1	4	0
P3	0	0	0			
P4	2	2	4			

Free resources: 0 1 0

(c) After simulating allocation of resources to P_1 when process P_4 completes

	Allocated resources R1	R2	R3	Requested resources R1	R2	R3
P1	4	2	3			
P2	1	3	1	1	4	0
P3	0	0	0			
P4	0	0	0			

Free resources: 0 2 1

(d) After simulating allocation of resources to P_2 when process P_1 completes

	Allocated resources R1	R2	R3	Requested resources R1	R2	R3
P1	0	0	0			
P2	2	7	1			
P3	0	0	0			
P4	0	0	0			

Free resources: 3 0 4

Figure 8.4 Operation of Algorithm 8.1, the deadlock detection algorithm.

	Allocated resources R1	R2	R3	Requested resources R1	R2	R3
P1	2	1	0	2	1	3
P2	1	3	1	1	4	0
P3	0	1	1			
P4	1	2	2	1	0	2

Total resources (R1 R2 R3): 5 7 5

Free resources (R1 R2 R3): 1 0 1

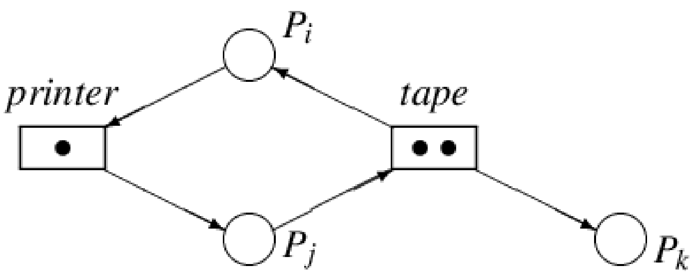
5 — 10M

An OS contains one printer and two tapes, and three processes Pi, Pj ,Pk .Explain the above scenario using Resource Request and Allocation Graph(RRAG) ,Wait –for–graph(WFG)and Matrix model when Pi is allocated with one unit of tape and requesting for a printer, Pj is allocated with printer but requesting for a unit of tape, one unit of tape is allocated to Pk also mention the events related to resource allocation.

Explanation -5M
Diagrams -2 graphs-5M

- A system contains a printer and a tape and two processes which require both the resources

Process P_i	Process P_j
Request a tape	Request a printer
Request a printer	Request a tape



- The printer is allocated to P_j
- The units of tape are allocated to P_i and P_k
- P_i has requested a printer and P_j has requested a tape
- Matrix model

Allocated resources

	Printer	Tape
P_i	0	1
P_j	1	0
P_k	0	1

Requested resources

	Printer	Tape
P_i	1	0
P_j	0	1
P_k	0	0

	Printer	Tape
Total resources	1	2
Free resources	0	0

Calculate the Mean turnaround time and Mean weighted turnaround time for the following processes using Least Completed Next(LCN) and Round Robin(RR) scheduling algorithm with suitable graph 4M

Process	P1	P2	P3	P4	P5
Arrival time	0	2	3	5	9
Service Time	3	3	2	5	3

LCN-5M
RR-5M
Solving the instance with table-2M
Timing diagram-4 M

4M

Time	Round-robin (RR) Processes	Round-robin (RR) Scheduled	Least completed next (LCN) Processes	Least completed next (LCN) Scheduled	Shortest time to go (STG) Processes	Shortest time to go (STG) Scheduled
0	P_1	P_1	P_1:0	P_1	P_1:3	P_1
1	P_1	P_1	P_1:1	P_1	P_1:2	P_1
2	P_2, P_1	P_2	P_1:2, P_2:0	P_2	P_1:1,P_2:3	P_1
3	P_1, P_3, P_2	P_1	P_1:2,P_2:1,P_3:0	P_3	P_2:3,P_3:2	P_3
4	P_3, P_2	P_3	P_1:2,P_2:1,P_3:1	P_2	P_2:3,P_3:1	P_3
5	P_2, P_4, P_3	P_2	P_1:2,P_2:2,P_3:1,P_4:0	P_4	P_2:3,P_4:5	P_2
6	P_4, P_3, P_2	P_4	P_1:2,P_2:2,P_3:1,P_4:1	P_3	P_2:2,P_4:5	P_2
7	P_3, P_2, P_4	P_3	P_1:2,P_2:2,P_4:1	P_4	P_2:1,P_4:5	P_2
8	P_2, P_4	P_2	P_1:2,P_2:2,P_4:2	P_1	P_4:5	P_4
9	P_4, P_5	P_4	P_2:2,P_4:2,P_5:0	P_5	P_4:4,P_5:3	P_5
10	P_5, P_4	P_5	P_2:2,P_4:2,P_5:1	P_5	P_4:4,P_5:2	P_5
11	P_4, P_5	P_4	P_2:2,P_4:2,P_5:2	P_2	P_4:4,P_5:1	P_5
12	P_5, P_4	P_5	P_4:2,P_5:2	P_4	P_4:4	P_4
13	P_4, P_5	P_4	P_4:3,P_5:2	P_5	P_4:3	P_4
14	P_5, P_4	P_5	P_4:3	P_4	P_4:2	P_4
15	P_4	P_4	P_4:4	P_4	P_4:1	P_4
16	–	–	–	–	–	–

Process	Round-robin (RR)			Least completed next (LCN)			Shortest time to go (STG)		
	C	*ta*	*w*	*C*	*ta*	*w*	*C*	*ta*	*w*
P_1	4	4	1.33	9	9	3.00	3	3	1.00
P_2	9	7	2.33	12	10	3.33	8	6	2.00
P_3	8	5	2.50	7	4	2.00	5	2	1.00
P_4	16	11	2.20	16	11	2.20	16	11	2.20
P_5	15	6	2.00	14	5	1.67	12	3	1.00

$\overline{ta}$ = 6.6 Seconds $\overline{ta}$ = 7.8 Seconds $\overline{ta}$ = 5.0 Seconds

$\overline{w}$ = 2.07 $\overline{w}$ = 2.40 $\overline{w}$ = 1.44

1.	(b) Define Turnaround Time and Throughput. Calculate the Mean turnaround time and Mean weighted turnaround time for the following processes using FCFS scheduling algorithm with suitable graph.	6M

Process	P1	P2	P3	P4	P5
Arrival time	0	2	3	5	9
Service Time	3	3	2	5	3

Definitions-2M
Final answer -4M

Throughput:The average number of jobs,programs,processes or subrequests completed by a system in unit time.
Turnaround time:Time interval between submission of a job and its completion.

TIME	ID	TA	W	PROCESS IN SYSTEM	SCHEDULED
0	-	-	-	{P1}	P1
3	P1	3	1.0	{P2,P3}	P2
6	P2	4	1.33	{P3,P4}	P3
8	P3	5	2.50	{P4}	P4
13	P4	8	1.60	{P5}	P5
16	P5	7	2.33	{}	-

Turn around Time=5.40 sec
Weighted turnaroung time=1.75