KIIT Deemed to be University
Online End Semester Examination(Spring Semester-2021)

Subject Name & Code: **Applicable to Courses:**

Full Marks=50 **Time:2 Hours**

SECTION-A(Answer All Questions. Each question carries 2 Marks)

Time:30 Minutes **(7×2=14 Marks)**

Question No	Question Type (MCQ/SAT)	Question	CO Mapping	Answer Key (For MCQ Questions only)
Q.No:1	**MCQ**	**Arrange the following Memory/Storage Device in increasing order of proximity to the CPU.** **a) Registers, Electronic disk, Main Memory(RAM), Cache.** **b) Registers, Cache, Electronic disk, Main Memory(RAM).** **c) Registers, Cache, Main Memory(RAM), Electronic disk.** **d) Cache, Registers Main Memory(RAM), Electronic disk.**	CO1	(c)
	MCQ	**Arrange the following Memory/Storage device in increasing order of speed of execution.** **a) Magnetic tapes,**	CO1	(c)

		Electronic disk, Optical disk, Magnetic disk. b) Magnetic tapes, Electronic disk, Magnetic disk Optical disk, c) Magnetic tapes, Optical disk, Magnetic disk, Electronic disk. d) Magnetic tapes, Optical disk, Electronic disk Magnetic disk.		
	MCQ	**A memory storage which is both volatile and non-volatile.** **a)Electronic disk** **b) Optical disk** **c) RAM** **d)Cache**	CO1	a)
	MCQ	**Arrange the following Memory/Storage device in increasing order of cost.** **a) Magnetic tapes, Registers, Optical disk, Magnetic disk.** **b) Magnetic tapes, Optical disk, Magnetic disk, Registers.** **c) Magnetic tapes, Cache, Magnetic disk, Registers.** **d) Magnetic tapes, Cache, RAM, Registers.**	CO1	(b)
Q.No:2	**MCQ**	**Which of the following is a valid process state transitions?** **a) Ready – Ready** **b) Ready – Waiting** **c) Waiting – Ready** **d) Waiting – Running**	CO2	(c)
	MCQ	**Which of the following is an invalid process state transitions?** **a) Ready – New** **b) Ready – Waiting** **c) Waiting – Running**	CO2	(d)

		d) All of the above		
	MCQ	**CPU scheduler involves the following queue** **a) Device queue** **b) Ready queue** **c) Both of above** **d) None of above**	CO2	(b)
	MCQ	**Mid-term scheduler involves the following transition** **a)Ready-Running** **b)Running-Waiting** **c)Both of above** **d)None of above**	CO2	(b)
Q.No:3	**MCQ**	**Which of the following transition relates to non-preemption** **a) Running-Ready** **b) Running-Waiting** **c)Ready-Running** **d) Waiting-Ready**	CO2	(b)
		If the fork() is called 'n' times by the parent process, then the number of total processes generated are: **a) 2n** **b) n^2** **c) n^2/2** **d) 2^n**	CO2	(d)
		Which of the following scheduling algorithms may produce starvation: **a)FCFS** **b)Priority** **c) Only a** **d) Both a and b**	CO2	(d)
		Increasing time quantum in Round Robin (RR) scheduling implies: **a)RR behaves like FCFS** **b)Higher average turn around time** **c)RR behaves like SJF pre-emptive** **d)Only b**	CO2	(a)

		e)None of the above		
Q.No:4		**Given a Resource allocation graph (RAG) with multiple instance multiple resources, choose the correct statement:** a) **A cycle in RAG guarantees deadlock.** b) **A cycle in RAG means no deadlock.** c) **A cycle in RAG may or may not guarantee deadlock.** d) **Absence of cycle may guarantee no deadlock.**	CO4	(c)
		Given a Resource allocation graph (RAG) with single instance multiple resources, choose the correct statement: **a) A cycle in RAG guarantees deadlock.** **b) A cycle in RAG means no deadlock.** **c) A cycle in RAG may or may not guarantee deadlock.** **d) Absence of cycle may guarantee no deadlock.**	CO4	(a)
		Given a Resource allocation graph (RAG) 'm' resource types and 'n' processes, choose the correct statement: **a) RAG algorithm is more efficient than Banker's algorithm by a factor of 'm'.** **b) RAG algorithm is more efficient than Banker's algorithm by a factor of '1/m'**	CO4	(a)

		types. **c) RAG algorithm is more efficient than Banker’s algorithm by a factor of ‘m/n^2’.** **d) RAG algorithm is more efficient than Banker’s algorithm by a factor of ‘n^2/m’.**		
		Given a Wait for graph (WFG) ‘m’ resource types and ‘n’ processes, choose the correct statement: **a) WFG algorithm is more efficient than Deadlock detection algorithm by a factor of ‘1/m’.** **b) WFG algorithm is more efficient than Deadlock detection algorithm by a factor of ‘m’.** **c) WFG algorithm is more efficient than Deadlock detection algorithm by a factor of ‘n^2/m’.** **d) WFG algorithm is more efficient than Deadlock detection algorithm by a factor of ‘m/n^2’.**	CO4	(b)
Q.No:5		**Given two atomic operations on semaphore p(),v() choose the correct option:** **a) p(): s++; v(): s++** **b) p(): s--; v(): s++** **c) p(): s++; v():s--** **d) p(): s--; v(): s--**	CO3	(b)

		The problem of busy waiting in semaphore is solved by following functions: **a) wait(); sleep()** **b) wait(); signal()** **c) block(); wakeup()** **d) block(); wait()**	CO3	(c)
		Given that P and Q are two processes on semaphores S, Q such that: **Wait(S) Wait(Q);** **Wait(Q) Wait(S);** **Choose the correct option:** **a) The above sequence will lead to a deadlock.** **b) The above sequence may lead to a deadlock.** **c) No deadlock will happen.** **d) None of the above.**	CO3	(a)
		A solution for priority inversion may be: (a) **Priority acquiring** (b) **Priority reduction** (c) **Both of the above** (d) **None of the above**	CO3	(d)
		Let processes P and Q access a shared variable S with critical section. Pick the correct statement: a) **P, Q modify S simultaneously** b) **Q then P may modify S.** c) **P then Q may modify S** d) **Both (b) and (c)** e) **None of the above**	CO3	(d)
Q.No:6		**Given a logical**		

		memory of size 16KB and page size of 4B. If the physical memory has a total of 8 bits, (assume byte addressable memory), choose the most appropriate option: a) **Data insufficient** b) **64 frames, 256B RAM, 12 page table entries** c) **64 frames, 256B RAM** d) **64 frames, 256B RAM, 256 KB page table size** e) **Address mapping not possible**	CO5	(d)
		Given a logical memory of size 16KB and page size of 4B. If the physical memory has a total of x bits, (assume byte addressable memory), choose the most appropriate option: a) **Data insufficient** b) **2^(x-2) frames, 256B RAM, 12 page table entries** c) **2^(x-2) frames, 2^x B RAM, 2^x KB page table size** d) **64 frames, 2^x B RAM, 2^x KB page table size** e) **Address mapping not possible**	CO5	(c)
		Given a logical memory of size x KB and page size of 4B. If the size of physical memory and that of page table is 256B (assume byte addressable memory),	CO5	

		choose the most appropriate option: **a) No. Of pages is 4K** **b) Size of logical memory is 16KB** **c) Data insufficient** **d) Both (a), (b)**		(d)
		Given that the total number of TLB accesses is 100 and the total number of page table accesses is 20. If the TLB access time is 20 ns, page table and RAM access time is 100 ns each then there is: a) **45% slowdown in memory-access time** b) **40% slowdown in memory-access time** c) **50% slowdown in memory-access time** d) **Data insufficient**	CO5	(b)
Q.No:7		**External fragmentation may be dealt with schemes:** a) **Only Compaction** b) **Paging** c) **Only Paging, Segmentation** d) **Compaction, paging, segmentation**	CO5	(d)
		Perform FIFO with the following page sequence: 1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5 having a frame size of 4. The number of page faults: **a)9** **b)10** **c)11** **d) None of the above**	CO5	(b)

		Perform FIFO with the following page sequence: 7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1 having a frame size of 3. The number of page faults: **a)15** **b)11** **c)13** **d)10**	CO5	(a)
		Perform LRU with the following page sequence: 7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1 having a frame size of 3. The number of page faults: **a)15** **b)12** **c)13** **d)10**	CO5	(b)

SECTION-B(Answer Any Three Questions. Each Question carries 12 Marks)

Time: 1 Hour and 30 Minutes **(3×12=36 Marks)**

Question No	**Question**	**CO Mapping (Each question should be from the same CO(s))**
Q. No :8	a) Explain the difference between batch, multiprogramming, time-sharing and distributed operating system. [2]	CO2

b) Consider process arrival as given below where N = right most significant digit of your Roll No.(ex:- for Roll No. 18085**4**, N=**4**):

Process	CPU Burst Time(ms)	Arrival Time	Priority
A	4	0	2
B	3	3	3
C	6	N	6
D	5	5	N
E	1	15	4

Calculate the following for *priority (non preemptive)* and *round robin* (time quantum = 2 ms) CPU scheduling algorithm: [10]

i. Average waiting time
ii. Turnaround time for each process
iii. Order of completion
(*hints:-higher digits indicate higher priority*)

a) What is the drawback of preemptive priority scheduling algorithm? Explain method to solve it. [2]

b) Consider process arrival as given below where N = right most significant digit of your Roll No.(ex:- for Roll No. 18085**4**, N=**4**):

Process	CPU Burst Time(ms)	Arrival Time	Priority
A	4	0	2
B	3	1	3
C	6	3	6
D	5	N	N
E	1	4	4

Calculate the following for *priority (non preemptive)* and *round robin* (time quantum = 2 ms) CPU scheduling algorithm: [10]

iv. Average waiting time
v. Turnaround time for each process
vi. Order of completion
(*hints:-higher digits indicate higher priority*)

A) A.Define and differentiate between preemptive & non-preemptive scheduling. [2]

B) Consider process arrival as given below where N = right most significant digit of your Roll No.(ex:- for Roll No. 18085**4**, N=**4**):

Process	CPU Burst Time(ms)	Arrival Time	Priority
A	4	0	2
B	3	3	3
C	6	4	1
D	5	6	N
E	1	N	4

Calculate the following for *priority (non preemptive)* and *round robin* (time quantum = 2 ms) CPU scheduling algorithm: [10]

(a)Average waiting time

(b)Turnaround time for each process

(c)Order of completion

(*hints:-higher digits indicate higher priority*)

Q. No :9

CO2 & CO6

(a)What is "starvation" of process and how to prevent it? What do you mean by cascading termination? (6+6)

(b)Find the total head movement for FCFS scheduling with following disk queue requests on I/O blocks: 98, 183, 37, 122, 14, 124, 65, 67. Head starts at N (where N= your Roll No. MODULUS 100) **.**

(a)In terms of the variable pid, differentiate between a child process and a parent process? Draw the Queuing diagram representation of process scheduling for all types of schedulers. (6+6)

(b)Find the total head movement for SSTF scheduling with following disk queue requests on I/O blocks: 98, 183, 37, 122, 14, 124, 65, 67. Head starts at Head starts at N (where N= your Roll No. MODULUS 100) **.**

(a) What are the criteria for a good CPU scheduling algorithm? State the differences between Multi level queue scheduling(MQS) and Multi level feed back queue scheduling(MFQS)? Which of MQS or MFQS can lead towards starvation of processes? (6+6)

(b)

Find the total head movement for SCAN and C-SCAN scheduling with following disk queue requests on I/O blocks: 0, 14, 37, 53, 65, 67, 98, 122, 124, 183. Head starts at N (where N= your Roll No. MODULUS 100) **. Which algorithm is more**

	preferable and why?	
Q. No :10	Consider a paging memory management system, where each entry of the page table consists of frame address, valid bit(1 bit), present bit(1 bit), protection bit(2 bits), modified bit(1 bit). The virtual address is 18bits, page size is 512 bytes, and page table is exactly fit into one page. The program is allocated with 3 page frames. The program generates the following 16 page numbers (which are part of the virtual address produced by): 0, 17, 18, M, 20, 2, 20, M, 17, 32, 0, 2, 2*M, 0, 16, M (Note: The page numbers are in decimal and M=sum of digits of your Roll No. MODULUS 10). **(6+6)** I. Calculate the number of page faults generated for the above page request, assuming a Optimal page replacement algorithm. II. Find the maximum size pf the program for single level paging.	CO6
	Consider a paging memory management system, where each entry of the page table consists of frame address, valid bit(1 bit), present bit(1 bit), protection bit(3 bits), modified bit(1 bit). The virtual address is 18bits, page size is 512 bytes, and page table is exactly fit into one page. The program is allocated with 3 page frames. The program generates the following 16 page numbers (which are part of the virtual address produced by): 0, 17, 18, M+2, 20, 2, 20, M, 31, 32, 0, 18, N, 0, 17, M (Note: The page numbers are in decimal and M=sum of digits of your Roll No. MODULUS 10). **(6+6)** I. Calculate the number of page faults generated for the above page request, assuming a Optimal page replacement algorithm. II. Find the maximum size pf the program for single level paging. .	
	Consider a paging memory management system, where each entry of the page table consists of frame address, valid bit(1 bit), present bit(1 bit), protection bit(2 bits), modified bit(1 bit). The virtual address is 18bits, page size is 512 bytes, and page table is exactly fit into one page. The program is allocated with 3 page frames. The program generates the following 16 page numbers (which are part of the virtual address produced by): 0, 17, 18, M, 20, 2, 20, M+4, 17, 32, 0, 2, M+2, 0, 16, M (Note: The page numbers are in decimal and M=sum of digits of your Roll No. MODULUS 10). **(6+6)** I. Calculate the number of page faults generated for the above page request, assuming a Optimal page replacement algorithm. II. Find the maximum size pf the program for single level paging.	
Q. No :11	**Find a safe sequence(if any) for the following resource allocation table using deadlock detection algorithm.** **Process Alloc Req Avail** P0 0 1 0 2 2 2 4 3 3 P1 1 1 1 1 2 2	CO4

P2	1 0 1	3 2 2
P3	2 1 2	4 2 3
P4	0 0 0	3 2 3

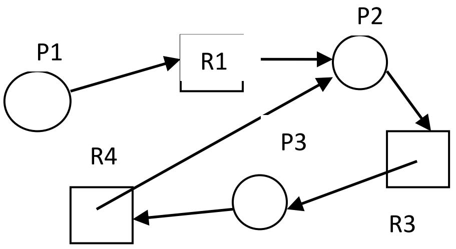
Does the system has deadlock, if so then which are the deadlocked processes? Also find any unsafe sequence and provide the value of 'Work' vector for it. (6+2+4)

Find a safe sequence(if any) for the following resource allocation table using Banker's algorithm.

Process	**Alloc**	**Max**	**Avail**
P0	0 1 0	2 2 2	4 3 3
P1	1 1 1	1 2 2	
P2	1 0 1	3 2 2	
P3	2 1 2	4 2 3	
P4	0 0 0	3 2 3	

Also find any unsafe sequence and provide the value of 'Work' vector for it. processes? (6+4+2)

Convert the following Resource Allocation Graph (RAG) to Wait for Graph (WFG)



Identify the cycle in the corresponding WFG. List the processes which are a part of deadlock.