Reg. No.										

MANIPAL INSTITUTE OF TECHNOLOGY

MANIPAL

(A constituent unit of MAHE, Manipal)

IV SEMESTER B.TECH. (COMPUTER AND COMMUNICATION ENGINEERING)

END SEMESTER EXAMINATIONS, APRIL 2019

SUBJECT: OPERATING SYSTEMS [ICT 2251]

REVISED CREDIT SYSTEM
(24/04/2019)

Time: 3 Hours MAX. MARKS: 50

Instructions to Candidates:

- Answer **ALL** the questions.
- Missing data, if any, may be suitably assumed.

1A. Consider the set of processes, their arrival time and burst time in ms as given in Table Q.1A. Draw a Gantt chart showing the scheduling of the processes using priority and preemptive shortest job scheduling algorithms. Find the waiting time, turnaround time of all the processes. Calculate the average waiting time and average turnaround time. Assume all processes arrive at the same time in the order P1, P2, P3, P4, P5. Note that lower value indicates higher priority. **5**

Table Q.1A

Process	Burst Time	Priority
P1	8	2
P2	4	1
P3	9	3
P4	5	4
P5	2	2

1B. List and explain three commonly used approaches to real-time scheduling. **3**

1C. Differentiate between resource-allocation graph and wait-for graph with a suitable example. **2**

2A. Explain in detail the steps in handling page fault with a neat diagram. List out the delays that occur while reading the disk content to a free frame? **5**

2B. Suppose a disk is having 200 tracks numbered from 0 to 199. The disk is currently serving the request of reading data from track 120 and at the previous request the service was at track 90. The pending requests at the order of arrival are for the track numbers 30, 70, 115, 130, 110, 80, 20, 25. Draw the read/write head movements for the disk policies SSTF, C-SCAN and LOOK. Calculate the total distance the read/write head will traverse for each of these scheduling policies. **3**

2C. Differentiate between static and dynamic real time systems. **2**

3A. Write and explain the monitor solution to the dining philosopher's problem. **5**

3B. Write a short note on internal and external fragmentation with suitable example. Explain the solutions for both. **3**

3C. List and explain the benefits of multithreaded programming. **2**

4A. Consider a real time system having three periodic tasks T1=(10,1,6), T2=(5,1,4), T3=(3,1,5) **5**

i. Find the release time, absolute deadline of first four jobs of each task in the system.
ii. Find utilization of each task and total utilization.
iii. Find hyper-period.
iv. Draw pre-emptive EDF schedule of the jobs on single processor for timeline from 0ms-20ms.

4B. What is segmentation? Explain segmentation hardware with a neat diagram. **3**

4C. Consider a system with a 1-KB frame size. Assume that the system consists of a small student process of 10 KB and an interactive database of 138 KB are the only two processes running in a system with 62 free frames. Find the number of frames allocated to each process using equal allocation and proportional allocation schemes. **2**

5A. Consider a system with processes P0 through P4 and four resource types A, B, C, D. Assume resource type A has 20 instances, resource type B has 4 instances and resource type C has 6 instances, resource type D has 9 instances. The maximum and allocation matrices are provided in Table Q.5A.

Table Q.5A

Process	Allocation				Max			
	A	B	C	D	A	B	C	D
P0	5	0	1	3	15	1	2	4
P1	3	0	2	0	5	4	3	9
P2	2	0	1	1	2	2	2	9
P3	6	1	1	0	8	1	1	0
P4	0	3	1	2	5	3	3	6

5

i. Draw the resource allocation graph for the above scenario.
ii. Check whether the system remains in safe state or not. If system is in safe state then give the safe sequence otherwise write the processes involved in the deadlock.
iii. For the following independent requests from P4 and P1, check if the system remains in the safe state or not. If system is in safe state then give the safe sequence otherwise write the processes involved in the deadlock. Process P4 requests (2, 0, 0, 2) and the process P1 requests (3, 0, 0, 1).

5B. Explain the following with the neat diagrams. **3**

i. hierarchical page table structure
ii. hashed page table structure

5C. With the help of a neat diagram explain process state transitions. **2**