Code: CT31 **Subject: OPERATING SYSTEM**
Time: 3 Hours **Max. Marks: 100**

MARCH 2011

NOTE:

- **Question 1 is compulsory and carries 28 marks. Answer any FOUR questions from the rest. Marks are indicated against each question.**
- **Parts of a question should be answered at the same place.**

Q.1 **(7 × 4)**

a. List the four steps that are necessary to run a program on a completely dedicated machine.

b. What are the differences between a trap and an interrupt? What is the use of each function?

c. What is the main advantage of the layered approach to system design?

d. What two advantages do threads have over multiple processes? What major disadvantage do they have? Suggest one application that would benefit from the use of threads, and one that would not.

e. What are the benefits of a DFS when compared to a file system in a centralized system?

f. What is the need-to-know principle? Why is it important for a protection system to adhere to this principle?

g. List out the main requirements, which should be satisfied by a page replacement policy.

Q.2 a. List five services provided by an operating system. Explain how each provides convenience to the users. Explain also in which cases it would be impossible for user-level programs to provide these services. **(10)**

b. What are interrupts? How are they handled by the operating system? Explain. **(8)**

Q.3 a. Describe the differences among short-term, medium-term, and long-term scheduling. **(6)**

b. Consider the following set of processes, with the length of the CPU-burst time given in milliseconds:

Process	Burst Time	Priority
A	10	3
B	1	1
C	2	3
D	1	4
E	5	2

The processes are assumed to have arrived in the order A, B, C, D, E, all at time 0.
(i) Draw four Gantt charts illustrating the execution of these processes using FCFS, SJF,a nonpreemptive priority (a smaller priority number implies a higher priority), and RR (quantum = 1) scheduling.
(ii) What is the turnaround time of each process for each of the scheduling algorithms in part (i)?
(iii) What is the waiting time of each process for each of the scheduling algorithms in part (i)? **(12)**

Q.4 a. What is a critical-section problem? What are the three requirements that a solution to the critical-section problem must satisfy? **(8)**

b. Consider the following snapshot of a system:

Processes	**Allocated resources**				**Maximum requirements**			
	A	**B**	**C**	**D**	**A**	**B**	**C**	**D**
P0	0	0	1	2	0	0	1	2
P1	1	0	0	0	1	7	5	0
P2	1	3	5	4	2	3	5	6
P3	0	6	3	2	0	6	5	2
P4	0	0	1	4	0	6	5	6

Available: 1 5 2 0

Answer the following questions using the banker's algorithm:
(i) What is the content of the matrix Need?
(ii) Is the system in a safe state?
(iii) If a request from process P1 arrives for (0,4,2,0), can the request be granted immediately? **(10)**

Q.5 a. When do page faults occur? Describe the actions taken by the operating system when a page fault occurs. **(8)**

b. Why are segmentation and paging sometimes combined into one scheme? **(4)**

c. Explain the following allocation algorithms giving a suitable illustration:
(i) First-fit
(ii) Best-fit
(iii) Worst-fit **(6)**

Q.6 a. Discuss some most common schemes for defining the logical structure of a directory. **(6)**

b. Suppose that a disk drive has 5,000 cylinders, numbered 0 to 4999. The drive is currently serving a request at cylinder 143, and the previous request was at cylinder 125. The queue of pending requests, in FIFO order, is

86, 1470, 913, 1774, 948, 1509, 1022, 1750, 130
Starting from the current head position, what is the total seek distance (in cylinders) that the disk arm moves to satisfy all the pending requests for each of the following disk-scheduling algorithms?
(i) FCFS (ii) SSTF **(6)**

c. Describe the deadlock-detection algorithm in a fully distributed system. **(6)**

Q.7 a. Explain various types of programming threats and system threats. **(5)**

b. Compare Caching and Remote services used in distributed file systems. **(8)**

c. Linux runs on a variety of hardware platforms. What steps must the Linux developers take to ensure that the system is portable to different processors and memory-management architectures, and to minimize the amount of architecture-specific kernel code? **(5)**