(1)

(Please write your Exam Roll No.) Exam Roll No. .....

END TERM EXAMINATION

FOURTH SEMESTER [B.TECH.] MAY-2010

Paper Code: ETCS212 Paper Id: 32212	Subject: Operating Systems
Time : 3 Hours	Maximum Marks :75

Note: Attempt one question from each unit. Q.1 is compulsory.

Q1 Answer the following in brief:- **(2.5x10=25)**

(a) What are the main advantages of multiprogramming?
(b) Define the essential properties of Batch Operating System.
(c) What are the differences between a trap and an interrupt?
(d) Define the differences between preemptive and non-preemptive scheduling.
(e) Describe the actions taken by a thread library to context switch between user-level threads.
(f) What are the benefits and the detriments of each of the following? Consider both the systems and the programmer's level.
  (i) Symmetric and Asymmetric communication.
  (ii) Fixed-sized and variable-sized messages.
(g) Discuss three major complications that concurrent processing adds to an operating system.
(h) Can "busy waiting" be avoided altogether? Explain your answer.
(i) What is the cause of thrashing? How does the system detect thrashing?
(j) In what situations would using memory as a RAM disk be more useful than using it as a disk cache?

UNIT-I

Q2 (a) Describe the following allocation algorithms:- **(4)**
  (i) First Fit (ii) Best Fit (iii) Worst Fit
(b) Why are sometimes segmentation and paging combined into one scheme? **(4)**
(c) Under what circumstances do page faults occur? Describe the actions taken by the operating system when a page fault occurs. **(4.5)**

OR

Q2 (a) Why are page sizes always powers of '2'? Explain. **(3)**
(b) Consider the following segment table: **(5)**

Segment	Base	Length
0	219	600
1	2300	14
2	90	100
3	1327	580
4	1952	96

What are the physical addresses for the following logical addresses?
(i) 0430 (ii) 110 (iii) 2500 (iv) 3400 (v) 4112
(c) Under what circumstances would a user be better off using a time-sharing system, rather than a PC or single-user workstation? **(4.5)**

UNIT-II

Q3 (a) What are the two differences between user-level threads and Kernel-level threads? Under what circumstances is one type better than the other? **(4)**

P.T.O.

(b) Show that, if the 'wait' and 'signal' operations are not executed atomically, then mutual exclusion may be avoided (violated). **(3.5)**

(c) Suppose that a scheduling algorithm (at level of short term scheduling) favours those processes that have used the least processor time in the recent past. Why will this algorithm favour I/O bound programs and yet not permanently starve CPU-bound programs? **(5)**

OR

Q3 (a) Explain the differences in degrees to which the following scheduling algorithms discriminate in favour of short processes:- **(6.5)**
(i) FCFS (ii) RR (iii) Multilevel feedback Queues

(b) Write a bounded buffer monitor in which the buffers (portions) are embedded within the monitor itself. **(6)**

UNIT-III

Q4 (a) Consider a system consisting of six tape drives, with 'n' processes competing for them. Each process may need two drives. For which values of 'n', is the system deadlock free. **(4.5)**

(b) Why is rotational latency usually not considered in disk scheduling? How would you modify SSTF, SCAN and C-SCAN to include latency optimization? **(8)**

OR

Q4 (a) Describe three circumstances separately under which blocking I/O and non-blocking I/O should be used. Why not just implement non-blocking I/O and have processes busy-wait until their device is ready? Explain. **(4)**

(b) Is disk scheduling other than FCFS scheduling, useful in a single user environment? Explain your answer. **(4.5)**

(c) Is it possible to have a deadlock involving only one process? Explain your answer. **(4)**

UNIT-IV

Q5 (a) Systems that support sequential files always have an operation to rewind files. Do systems that support random access files need this too? Why or why not, explain. **(4.5)**

(b) Some systems provide file sharing by maintaining a single copy of a file, other systems maintain several copies, one for each of the users sharing the file. Discuss the relative merits of each approach. **(4)**

(c) Why must the bit map for file allocation be kept on mass storage, rather than in main memory? Explain. **(4)**

OR

Q5 (a) An operating system only supports a single directory but allows that directory to have arbitrarily many files with arbitrarily long file names. Can something approximating a hierarchical file system be simulated? How? **(5)**

(b) Why is it advantageous to the user for an operating system to dynamically allocate its internal tables? What are the penalties to the operating system for doing so? **(4)**

(c) Some systems support many types of structures for a file's data, while others simply support a stream of bytes. What are the advantages and disadvantages? **(3.5)**

****