Ref. No. : Ex/IT/T/316/2019

B.E. INFORMATION TECHNOLOGY
THIRD YEAR, FIRST SEMESTER EXAM 2019
Subject: Operating Systems

Time - Three Hours | Full Marks – 100

Different parts of the same question should be answered together.

CO1 [10]

Q.1
i) What are the differences between trap and interrupt? What are the use of these functions?
ii) Mention two important facilities provided by a conventional operating system.
iii) What is the purpose of system calls and system programs? **(2+2)+2+4=10**

CO2 [25]

Q.2
i) Provide two programming examples in which multithreading does not provide better performance than a single-threaded solution
ii) Suggest a suitable algorithm for solving critical-section problem for n processes.
iii) What resources are used when a thread is created? How do they differ from those used when a process is created?
iv) IPC is best provided by message-passing system – give the mechanism of message passing for process communication.
v) Write a multi-threaded program that outputs prime numbers. On getting a number from user the program will create separate thread that outputs all the prime numbers less than the number that the user entered

4+7+4+4+6=25

CO3 [25]

Answer any one from (a) and (b) in this block

Q.3a

i) Describe the notion of deadlock avoidance. Write a solution to the Dinning-Philosopher problem which avoids deadlock.

ii) Two concurrent processes P1 and P2 use four shared resource R1, R2, R3 and R4 as shown below

P1	P2
Use R1	Use R1
Use R2	Use R2
Use R3	Use R3
Use R4	Use R4

Both processes are started at the same time, and each resource can be accessed by only one process at a time. The following scheduling constraints exist.

P2 must complete use of R1 before P1 gets access to R1
P1 must complete use of R2 before P2 gets access to R2
P2 must complete use of R3 before P1 gets access to R3
P1 must complete use of R4 before P2 gets access to R4

What is the minimum number of binary semaphores required for enforcing above scheduling constraints? Provide a solution.

iii) Show how to implement the wait()and signal()semaphore operations in multiprocessor environments using the TestAndSet() instruction. The solution should exhibit minimal busy waiting.

(2+8) +7+8 = 25

Q.3b

i) A certain processor provides a 'test and set' instruction that is used as follows TSET register flag. This instruction atomically copies flag to register and sets flag to 1. Give pseudo code for implementing the entry and exit code to a critical section using this instruction.

ii) Two shared resources R1 and R2 are used by processes P1 and P2. Each process has a certain priority for accessing each resource. Let Tij denotes the priority of Pi for accessing Rj. A process Pi can snatch a resource Rk from process Pj if Tik > Tjk. Find condition(s) that ensure(s) that P1 and P2 can never deadlock.

iii) A uni-processor computer system has only two processes, both of which alternate 10 ms CPU burst with 90 ms I/O burst. Both the processes were created at nearly the same time. The I/O of both processes can proceed in parallel. Calculate the CPU utilization (over a long period of time) for SJF and RR scheduling.

iv) What is the meaning of the term busy waiting ? What other kinds of waiting are there in an operating system? Can busy waiting be avoided altogether? Explain your answer.

7+5+5+8 =25

CO4 [25]

Answer any one from (a) and (b) in this block

Q.4a)

i)What is page thrashing and how might it be avoided?

ii) Consider a small two-way set-associative cache memory consisting of four blocks. For a block to be replaced, LRU scheme is used. Calculate the number of cache misses for the following sequence of block addresses: 8, 12, 0, 12, 8

iii) Describe the actions that follow a page fault. You should include those performed in hardware prior to a handler being started, and those that are taken within the handler.

iv) Why it is easier to share a reentrant module using segmentation than it is to do so when pure paging is used. In a segmentation scheme, if there are 64 segments and the maximum segment size is 512 words, what is the length of the logical address in terms of bits?

v) Let the page reference and the working set window b e c c d b c e c e a d and 4 respectively. The initial working set at time t = 0 contains the pages {a, d, e}, where 'a' was referenced at time t = 0, d was referenced at time t = -1, and e was referenced at time t = -2. Determine the total number of page faults and the average number of page frames used.

5 X 5 =25

Q.4b)

i) Describe the ideal algorithm for handling the case of victim page replacement, and explain why it cannot be implemented.

ii) Now give a practical algorithm for handling the above case. Can the algorithm suffer from Belady's Anomaly? Justify your answer.

iii) Consider the following memory heap in which blank regions are not in use and hatched regions are in use. Which of the policies first-fit, best-fit, and worst-fit satisfies the sequence of requests for blocks of size 300, 25, 125 and 50?

50	150	300	350	600

iv) What is the cause of thrashing? How does the system detect thrashing?

v) Consider the following page reference string: 1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6. How many page faults would occur for each of LRU, FIFO and Optimal page replacement algorithm, assuming three frames (initially empty)?

5 X 5 =25

CO5 [15]

Answer any one from (a) and (b) in this block

Q.5a)

i) List four items of metadata that you might find in a File Control Block (FCB).

ii) Consider a file system on a disk that has both logical and physical block sizes of 512 bytes. Assume that the information about each file is already in memory. For each of the three allocation strategies (contiguous, linked, and indexed) how is the logical-to-physical address mapping accomplished in this system?

iii) Give the structure of an inode with a diagram.

4+6+5=15

Q.5b)

i) What is free- space list. How does Bit Vector manage free space? Mention the limitation of this technique if any.

ii) Describe the structure of a FAT with a diagram.

iii) What are the advantages of the variation of linked allocation that uses a FAT to chain together the blocks of a file?

(2+4+2)+4+3=15