No of Pages : 4 **Course Code : 15XT44 / 15XW44 /15XD44**

Roll No:
(To be filled in by the candidate)

PSG COLLEGE OF TECHNOLOGY, COIMBATORE - 641 004

SEMESTER EXAMINATIONS, DECEMEBR 2018

MSc – TCS / SOFTWARE / DATA SCIENCE Semester: 4

15XT44 / 15XW44 / 15XD44 OPERATING SYSTEMS

Time : 3 Hours **Maximum Marks :100**

INSTRUCTIONS:
1. Answer **ALL** questions. Each question carries 20 Marks.
2. Subdivision (a) carries 3 marks each, subdivision (b) carries 7 (3+4) marks each and subdivision (c) carries 10 marks each.

1. a) Do you feel that an OS should include many common system functions or that it should contain only a minimum level of functions, leaving as much as possible to be in additional layers and libraries? Justify your answer.

   b) i) "The evolution of OSs has resulted in the present state in which most modern OSs are virtual machine OSs." – Comment on this statement.

   ii) List the states that a process may be in. Draw a diagram showing the possible transitions between theses states and identifying what causes state transitions.

   c) Explain how each of the following scheduling algorithms works. Be able to discuss how each one affects CPU utilization, job throughput, turnaround time, waiting time, and response time. Discuss how preemption might apply to each of the algorithms and what effects it might produce on the performance of the system.

   - first come, first served
   - shortest job first
   - shortest remaining time first
   - round robin
   - priority
   - multilevel queue

2. a) We said that for a deadlock to happen there had to be a sequence of processes, each holding a resource and waiting on another resource that was held by another process, with the last process waiting on a resource held by the first process. How many processes does it take to create a deadlock?

   b) i) With regard to process synchronization what is meant by race condition? In order to avoid race conditions, what four conditions must be satisfied?

   ii) Two processes, P_0 and P_1,are to be run and they update a shared variable. This update is protected by Petersons solution to the mutual exclusion problem.

   - Show Petersons algorithm and show the truth table for the part of the algorithm which dictates if a process is allowed to enter its critical region.
   - P_0 attempts to enter its critical region. Show the state of the variables that are created/updated. Will P_0 be allowed to enter its critical region? Why?

- P_1 attempts to enter its critical region. Show the state of the variables that are created/updated. Will P_1 be allowed to enter its critical region? Why?
- P_0 leaves its critical region. What effect does this have on the variables?
- Assume no processes are running and P0 and P1 try to enter their critical region at exactly the same time. What will happen?
- Critically evaluate the Petersons algorithm with reference to the conditions you listed in part (i)

c) Consider the following snapshot of a system (P=Process, R=Resource) :

Available			
RA	RB	RC	RD
8	5	9	7

Maximum Demand				
	RA	RB	RC	RD
P0	3	2	1	4
P1	0	2	5	2
P2	5	1	0	5
P3	1	5	3	0
P4	3	0	3	3

Current Allocation				
	RA	RB	RC	RD
P0	1	0	1	1
P1	0	1	2	1
P2	4	0	0	3
P3	1	2	1	0
P4	1	0	3	0

Answer the following questions using banker's algorithm:

- Calculate the *Needs* matrix:
- Is the system in a safe state? If so, show a safe order in which the processes can run.
- Can a request of one instance of RA by Process P0 be granted safely according to Banker's algorithm?

3. a) Eventually, page tables started to grow very big and sparse. What technique was employed to solve this problem?

b) i) Suppose a computer has 10,000 bytes of physical memory with a page size of 1000 bytes. The operating system has just begun a program which uses 20,000 bytes of virtual memory. None of the program is currently in memory, and all frames are empty. Assuming the following page reference sequence, which pages are in which frame at the end of the sequence? Show your working.

0,1,2,3,4,1,0,5,9,11,15,9,10,15,16,2,3,11,10,6,8,17,19,6,3

ii) What is the difference between external fragmentation and internal fragmentation?

c) Describe the operation of Translation Look aside Buffers (TLBs) in a paged operating system.

What is the sequence of events when

- a TLB is 'hit';
- a TLB is 'missed' but the required page is in main memory;

- a TLB is 'missed' but the required page is not in main memory.

Show clearly how the various components of the system fit together.

In a simple paged system with only one level of page tables the access times are as follows.

TLB HIT:

To read associative memory 1ns

To read main memory 6ns

TLB MISS:

To read associative memory 1ns

To add the page number to the page table origin register 2ns

To read page table 6ns

To read main memory 6ns

- Calculate the effective access time for a hit rate of 93 per cent.
- If the page fault service time is 1.2 ms, what is the maximum page fault rate that the system can tolerate without incurring more than a 15 per cent degradation?

4. a) Why do we worry about page replacement algorithms so much?

b) i) What is page fault? How is page fault frequency related to degree of multiprogramming of a computer?

ii) How does a buddy allocator allocate memory? Assume a computer with a memory size of 256K, initially empty. Requests are received for blocks of memory of 5K, 25K, 35K and 20K. Show how the buddy system would deal with each request, showing the memory layout at each stage and the status of the lists at the end. After allocating all the processes, what would be the effect of the 25K process terminating and returning its memory?

c) Consider a system where the virtual memory page size is 2K (2048 bytes), and main memory consists of 4 page frames. Now consider a process which requires 8 pages of storage. At some point during its execution, the page table is as shown below:

Virtual page	Valid	Physical page
0	No	
1	No	
2	Yes	1
3	No	
4	Yes	3
5	No	
6	Yes	0
7	Yes	2

- List the virtual address ranges for each virtual page.
- List the virtual address ranges that will result in a page fault.

- Give the main memory (physical) addresses for each of the following virtual addresses (all numbers decimal): (i) 8500, (ii) 14000, (iii) 5000, (iv) 2100.

5. a) How are directories organized internally to optimize searching time?

   b) i) If the FIFO disk scheduling algorithm is the fairest, why don't we just use that? How is the C-LOOK scheduling algorithm an improvement over LOOK?

   ii) Why was the concept of partitioning drives introduced?

   c) Disk requests come into the disk driver for cylinders 10, 22, 20, 2, 40, 6, and 38, in that order. Assume that the disk has 100 cylinders. A seek takes 6msec per cylinder moved. Compute the average seek time for the request sequence given above for

   - First-come, First-served
   - Shortest Seek Time First (SSTF)
   - LOOK (with the disk-arm initially moving towards higher number cylinders from lower number cylinders)
   - C-SCAN

   In all the cases, the arm is initially at cylinder 20.

/END/

FD/RL