Indian Institute of Technology, Kharagpur
End-Spring Semester 2017-18

Date of Examination: 24-04-2018 **Session:** AN (2-5 pm) **Duration:** 3 hrs
Subject No.: CS31702
Subject: COMPUTER ARCHITECTURE AND OPERATING SYSTEMS
Department/Center/School: Computer Science and Engineering
Specific charts, graph paper, log book etc., required: NO **Total Marks :** 100
Special instructions (if any): ANSWER ALL QUESTIONS
Note: All parts of the question (a,b,c,....) should be answered at a stretch.

---

1. (a) What is meant by dual mode of operation in the context of execution of a process?
   (b) What are the services offered by operating system from system point of view?
   (c) What is direct memory access (DMA)?
   (d) With appropriate example discuss how user program, API, system call interface, system call and operating system are related?
   (e) What is core-dump and crash-dump? Where these are used?
   (f) With a neat sketch (process stete diagram), explain various states the process may undergo from its creation to its termination. Mark all possible transitions and states and clearly explain the details of states and state-transitions.
   (g) Briefly discuss the pros and cons of shared memory and message passing schemes in the context of inter-process communication.
   (h) Discuss the salient features of ordinary pipes in the context of inter-process communication.

   (1+1+1+2+2+4+2+2 = 15M)

2. (a) How process and thread creations differ? Why threads are considered as light-weight-processes?
   (b) Mention various multi-threading models, and discuss their features.
   (c) Briefly explain the following in the context of threads: (i) Signal Handling and (ii) Thread pools.
   (d) What is preemptive and non-preemptive scheduling mechanisms? Provide an example to each.
   (e) Mention various parameters (criteria) considered for CPU scheduling.
   (f) Briefly discuss the following in the context of multi-processor scheduling: (i) processor affinity, (ii) soft affinity, (iii) hard affinity, (iv) push migration and (v) pull migration.
   (g) Consider the following set of processes, with the length of the CPU burst given in milliseconds:

Process	Burst Time (ms)	Priority
P1	2	2
P2	1	1
P3	8	4
P4	4	2
P5	5	3

i. **Draw four Gnatt charts that illustrate the execution of these processes using the folloing scheduling algorithms: FCFS, SJF, non-preemptive priority (a larger priority number implies a highest priority), and RR (quantum = 2).**

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 these scheduling algorithms?**

iv. **Which of the algoritms results in the minimum average waiting time (over all processes)?**

(2+2+2+2+2+5+8 = 23M)

3. (a) **Define a critical section problem. Specify the requirements to be satisfied by the solution to the critical section problem.**

   (b) **Provide the codes correspond to entry and exit sections of a process Pi in view of Peterson solution to critical section problem.**

   (c) **Write the pseudo code with TestAndSet(&lock) to solve the critical section problem addressing the issues of bounded-waiting and mutual-exclusion.**

   (d) **What is a semaphore? What are the standard operations that modify semaphore value? How semaphore handles the critical section problem using entry and exit sections? Provide the detailed code segments for the standard semaphore operations in case of buzy-waiting and without buzy-waiting?**

   (e) **Illustrate the problem of deadlock using pair of processes want to access a pair of critical sections.**

   (f) **mention the necessary conditions to be satisfied for deadlock.**

   (g) **Consider the following snapshot of a system. P0, P1, P2, P3, P4 are the processes and A, B, C, D are the resourse types. The values in the table indicates the number of instances of a specific resource (for example: 3 3 2 1 under the last column indicates that there are 3 A-type, 3 B-type, 2 C-type and 1 D-type resources are available after allocating the resources to all five processes). The numbers under allocation-column indicate that those number of respources are allocated to various processes mentioned in the first column. The numbers under Max-column indicate the maximum number of resources required by the processes. For example: in 1st row under allocation-column 2 0 0 1 indicate there are 2 A-type, 0 B-type, 0 C-type and 1 D-type resources are allocated to process P0. Whereas 4 2**

1 2 under Max-column indicate that process P0's maximum requirement is 4 A-type, 2 B-type, 1 C-type and 2 D-type resources. Answer the fol-

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

lowing questions using banker's algorithm by providing all intermediate steps:

i. How many instances of resources are present in the system under each type of a resource?

ii. Compute the Need matrix for the given snapshot of a system.

iii. Verify whether the snapshot of the present system is in a safe state by demonstrating an order in which the processes may complete.

iv. If a request from process P1 arrives for (1,1,0,0), can the request be granted immediately?

v. If a request from process P4 arrives for (0,0,2,0), can the request be granted immediately?

(h) Briefly discuss the policies to recover from the deadlock.

(2+2+4+7+2+2+6+2 = 27M)

4. (a) What are logical and physical addresses of a process? How logical address is converted to physical address? With appropriate figure, discuss how memory management unit (hardware) protects the memory address space of other processes and operating system.

(b) Discuss 3 different stages of address binding.

(c) Briefly discuss about external and internal fragmentations in the context of contiguous memory space allocation.

(d) Show the process of paging (conversion of logical address to physical address) with TLBs using neat diagram.

(e) Consider a computer system with a 32-bit logical address and 4-KB page size. The system supports up to 512-MB of physical memory.

i. How many entries will be there in a conventional single-level page table and inverted page table?

ii. What will be the memory requirement for storing these tables?

iii. If a memory reference takes 50 nanoseconds, how long does a paged memory reference take in the context of conventional page table?

iv. If we add TLBs, and 75% of all page-table references are found in the TLBs, what is the effective memory reference time? (Assume that finding a page-table entry in the TLBs takes 2 nanoseconds, if the entry is present.)

(f) **What is hashed-page table? How address translation (logical to physical) is carried out using hashed-page table?**

(g) **What is page fault? With appropriate diagram clearly discuss the steps involved in handling the page fault by an operating system.**

(h) **With appropriate diagram explain the concept of copy-on-write in the context of process creation.**

(i) **Consider the following page reference string: 7,2,3,1,2,5,3,4,6,7,7,1,0,5, 4,6,2,3,0,1. Assume demand paging with 3 frames, how many page faults would occur for the following replacement algorithms?**

i. **LRU replacement**

ii. **FIFO replacement**

iii. **Optimal replacement**

(j) **What is thrashing? What is the working-set model (WSM) of a process? How do you track the working-set? Comment on the relation between WSM and size of the physical memory in view of thrashing.**

(3+3+2+2+7+2+4+2+6+4 = 35M)