Operating System

1. 20% [Segmentation] Consider the following segment table :

Segment	Base	Length
0	219	700
1	2315	15
2	100	90
3	1327	580
4	1920	100

What are the physical addresses for the following logical addresses?

(a). Segment 4, byte 50. (4%)

(b). Segment 1, byte 20. (4%)

(c). Segment 3, byte 500. (4%)

(d). Segment 2, byte 400. (4%)

(e). Segment 0, byte 112. (4%)

2. 15% [Virtual Memory] Assume we have a demand-paged memory. The page table is held in registers. It takes 8 milliseconds to service a page fault if an empty page is available or the replaced page is not modified and 20 milliseconds if the replaced page is modified. Memory access time is 100 nanoseconds. Assume that the page to be replaced is modified 70 percent of the time. What is the maximum acceptable page-fault rate for an effective access time of no more than 200 nanoseconds?

3. 10% [Scheduling] Consider a preemptive priority scheduling algorithm based on dynamically changing priorities. Larger priority numbers imply higher priority. When a process is waiting for the CPU (in the ready queue, but not running), its priority changes at a rate A; when it is running, its priority changes at a rate B . All processes are given a priority of 0 when they enter the ready queue. The parameters A and B can be set to give many different scheduling algorithms. What is the algorithm that results from $B > A > 0$?

4. 15% [Deadlocks] Consider the dining-philosophers problem where the chopsticks are placed at the center of the table and any two of them could be used by a

philosopher. Assume that requests for chopsticks are made one at a time. Describe a simple rule for determining whether a particular request could be satisfied without causing deadlock given the current allocation of chopsticks to philosophers .

5. 10% [Scheduling] Which of the following scheduling algorithms could result in starvation?

(a) . First-come, first-served

(b) . Shortest job first

(c) . Round robin

d) . Priority

6. 15% [Virtual Memory] Consider a paging system with the page table stored in memory.

(a) . If a memory reference takes 200 nanoseconds, how long does a paged memory reference take? (7%)

(b) . If we add associative registers, and 75 percent of all page-table references are found in the associative registers , what is the effective memory reference time ? (Assume that finding a page-table entry in the associative registers takes zero time , if the entry is there.) (8%)

7. 15% [Operating System Structure] What is the main advantage of using a virtual-machine architecture for a user?