Institute of Business Administration
MIS & CS Department
Operating Systems, Fall Semester 2003
BCS IV
Third Hourly Test
November 3, 2003

Time Allowed: One Hour **Total Marks: 100**

Instructions

a. Attempt all questions.
b. Maximum/Total Marks are 100.
c. Time allowed is 1 hour.
d. Do NOT write any thing on the Question Paper except your name. Provide your answers on the answer sheet provided for this purpose.

Question 1: State whether True (T) or False(F) (40 Marks)

1. Simple Paging results in no internal fragmentation.
2. In a uniprocessor machine concurrent processes cannot be overlapped, they can only be interleaved.
3. Semaphore is an operating system level mechanism to provide concurrency.
4. The circular wait condition can be prevented by defining a non-linear ordering of resource types.
5. Three conditions of policy that must be present for a deadlock to be possible are (1) Mutual Exclusion (2) No Preemption and (3) Circular Waiting
6. In a Deadlock Detection Strategy we do not grant an incremental resource request to a process if this allocation might lead to a deadlock.
7. A modified form of paging is used for UNIX kernel memory allocation.
8. A semaphore may be initialized to any negative value.
9. A relative address is a particular example of a physical address in which the address is expressed as a location relative to some known point, usually the beginning of a program.
10. Simple paging is similar to dynamic partitioning.
11. With the use of a special machine instruction to enforce mutual exclusion deadlock is not possible.
12. The smaller the page size, the less the amount of internal fragmentation.
13. If the page size is small then the page fault rate should be low.
14. The signal operation decrements the semaphore value.
15. With demand paging, pages other than the one demanded by the page fault are also brought in.
16. The use of a special machine instruction to enforce mutual exclusion has the disadvantage that it employs busy waiting.
17. The most popular virtual memory scheme is demand paging.
18. I/O Channels are an example of a consumable resource.
19. There is no single effective strategy that can deal with all types of deadlock.
20. Deadlock may be a consequence of the mutual exclusion enforcement.

Question 2: *(15 Marks)*

Suppose the page table for the process currently executing on the processor looks like the following. All numbers are decimal, everything is numbered starting from zero, and all addresses are memory byte addresses. The page size is 1024 bytes.

Virtual Page Number	*Valid Bit*	*Reference Bit*	*Modify bit*	*Page Frame Number*
0	1	1	0	4
1	1	1	1	7
2	0	0	0	-
3	1	0	1	2
4	0	0	0	-
5	1	0	0	0
6	1	1	0	1

What physical addresses, if any, would each of the following virtual addresses correspond to? Do not try to handle page faults, if any.

1. 1552
2. 3221
3. 6499

Question 3: *(15 Marks)*

A process has five page frames allocated to it. All the following numbers are decimal, and every thing is numbered starting from zero. The time of last loading of a page into each page frame, the time of last access to the page in each page frame, the virtual page number in each page frame, and the referenced (R) and modified (M) bits for each page frame are as shown in the following table. The times are in clock ticks from the process start time 0 to event.

Virtual Page Number	*Page Frame*	*Time Loaded*	*Time Referenced*	*R bit*	*M bit*
2	0	50	148	1	0
1	1	130	142	1	0
0	2	24	144	0	0
3	3	20	140	0	1
5	4	80	130	0	1

A page fault to virtual page 4 has occurred. Which page frame will have its contents replaced for each of the following memory management policies? Explain why in each case.

1. FIFO
2. LRU
3. Clock

Question 4: *(15 Marks)*

Consider the following snapshot of a system. There are no current outstanding queued unsatisfied request.

Available

R1	R2	R3	R4
2	1	0	0

	Current Allocation				***Maximum Demand***				***Still Needs***			
Process	R1	R2	R3	R4	R1	R2	R3	R4	R1	R2	R3	R4
P1	0	0	1	2	0	0	1	2				
P2	2	0	0	0	2	7	5	0				
P3	0	0	3	4	6	6	5	6				
P4	2	3	5	4	4	3	5	6				
P5	0	3	3	2	0	6	5	2				

1. Compute what each process still might request, i.e., Still Needs columns.
2. Is this system currently in a safe state or an unsafe state? Why?
3. If a request from P3 arrives for (0, 1, 0, 0), can that request be safely granted immediately? In what state (deadlocked, safe, unsafe) would immediately granting that whole request leave the system? Which processes, if any, are or may become deadlocked if this whole request is granted immediately?

Question 5: *(15 Marks)*

Very briefly state the Dining Philosophers' Problem and give one simple solution using semaphores to solve the problem.

-- THE END --