CS 471 – Operating Systems
Fall 2003
Examination I
Points: 125
October 4, 2003
Time: 8:30-11:30 AM
CLOSED BOOK

Turning in this exam under your name confirms your continued support for the honor code of Old Dominion University and further indicates that you have neither received nor given assistance in completing it.

Name:__________________________ SSN: __________________

Question#	Points	
	Maximum	Obtained
1a	6	
1b	6	
1c	7	
1d	6	
2a	8	
2b	8	
2c	9	
3a	8	
3b	8	
3c	9	
4a	6	
4b	6	
4c	6	
4d	7	
5a	8	
5b	9	
5c	8	
Total:	125	
Letter Grade:		

Question 1.

a. For exach of the following entries in a PCB, explain what specific information is saved and why it is saved by an OS when a process is switched from running to waiting state.

   i Program counter

   ii. CPU Registers

   iii. Memory management information

b. What is the state of a process in each of the following scenarios:

   i Executing "LOAD X" instruction that loads variable X into the accumulator.

   ii. Process just issued a "READ diskaddress, nbytes" instruction to read nbytes of data starting from diskaddress from a disk.

   iii. Process is in the ready queue.

c. If a system has 10 processes in the ready queue, 1 process running, 4 waiting for read/write on disk-1, 1 waiting for read on disk-2, 3 waiting for user input, 2 waiting to enter the system in the job queue (due to the multiprogramming limitation), and 5 that have terminated in the last 1 minute, determine the degree of multiprogramming of the system.

d. Explain the difference between the elements in each of the following pairs.

   i. Direct versus indirect communication (between processes)

   ii. CPU-bound versus I/O-bound process

   iii. Short-term versus long-term scheduler

Question 2.

a. Given the following four processes with required CPU bursts and arrival times, determine the **average wait time** with RR (Round-Robin) as CPU scheduling policy with time quantum of 5 units. Also show the **Gantt chart** illustrating the execution sequence of the jobs.

Process#	Arrival Time	CPU Burst Time	Priority
P1	0	11	4 (Lowest)
P2	4	8	1 (Highest)
P3	12	12	3
P4	19	5	2

b. For the same four processes shown in (a), determine the **average wait time** with Shortest-Job-First **with** preemption as CPU scheduling policy. Also show the **Gantt chart** illustrating the execution sequence of the jobs.

c. For the same four processes shown in (a), determine the **average turnaround time** with priority-scheduling with preemption (Note: Priority 1 is highest priority; 4 is the lowest priority). Also show the **Gantt chart** illustrating the execution sequence of the jobs.

Question 3.

a. Using the monitor solution to a single resource allocation problem (refer to Figure 7.23 provided in the appendix), answer the following questions. (Treat each question independently.)

   i. What happens if a process A has released a resource R (through R.release()) that it never acquired?

   ii. What happens if a process B executes two R.acquire(100) operations consecutively (i.e., without a R.release() operation in between them).

b. Two threads running concurrently are updating a shared integer variable x. Initially, $x = 100$. While thread T_1 is incrementing x by 50, thread T_2 is decrementing it by 75. Explain the the consequences (if any) of not placing the updating code in critical sections using this example.

c. Using examples (one example each for the two cases), show how the following solution for 2-process CSP does not satisfy the

Case 1: Mutual exclusion requirement

Case 2: Bounded waiting requirement

(Note: The code corresponds to process P_i. The other process is P_j.)

```
do{
    flag[j] = false;
    flag[i] = true;
    while (flag[j]);

    Critical section

    flag[i] = false;

    remainder section

    } while(1);
```

Question 4

a. What is the "circular wait" condition for a deadlock? Show through an example how deadlocks may be prevented by relaxing this assumption.

b. Suppose a system is in unsafe state. Does it mean that it is in a deadlock state? Justify your answer with an example.

c. Given the state of the resources (A, B, C, D) and processes $(P_0 - P_4)$ in a system, determine if the system is in a **safe** state. *Including* the ones already allocated, the system has 7 instances of A, 11 instances of B, 13 instances of C, and 14 instances of D. **Show your work**. (Hint: Use banker's algorithm.)

	Current Holding				Maximum Needs			
	A	B	C	D	A	B	C	D
P_0	1	1	1	2	1	2	1	2
P_1	1	0	0	1	2	7	5	2
P_2	2	3	5	4	2	3	5	6
P_3	1	6	4	2	1	6	5	2
P_4	2	1	1	5	3	6	5	6

d. Given the following allocation/request/available information of resources and processes, determine whether or not the system is in a deadlock. **Show your work.**

	Allocation				Request				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P_0	2	4	3	2	1	0	0	2	2	0	1	3
P_1	1	3	0	0	2	4	4	2				
P_2	0	5	3	2	2	6	5	2				
P_3	2	3	2	5	2	4	3	5				

Question 5.

a. Given memory partitions of 400 KB, 100 KB, 300 KB, and 50 KB (in order), show how the best-fit algorithm would place processes of 165 KB, 250 KB, 200 KB, 20 KB, and 80 KB (in order).

b. Consider a paging system with page table in main memory as well as TLBs. Each memory access takes 100 nanoseconds, and accessing page-table entries in a TLB (when the entry is there) takes 10 nanoseconds. Assuming that 70% of all page-table references are found in TLB, determine the effective memory access time? (To help you recollect the TLB, Figure 9.10 from the book is reproduced in the appendix.)

c. Given the following segment table, map the following logical addresses to physical addresses?

Segment	Base	Length
0	400	200
1	800	555
2	720	75
3	1550	245
4	3000	2000

Logical address	Physical address
(2,48)	
(3,145)	
(0,150)	
(4,2020)	
(1,400)	