OPERATING SYSTEMS

FINAL EXAMINATION APRIL 30, 2006

Name ___________________________

ANSWER ALL OF THE QUESTIONS BELOW.

1. Indicate which of the statements below are true or false and EXPLAIN your answer in ONE SENTENCE.

   a) A multi-level indexed file descriptor permits faster random access than a file descriptor with a single level of index.
   b) The dispatcher is the part of the Operating System that determines the priority of each process.
   c) A particular semaphore is used by only two processes at a time.

2. A number of jobs enter a system with characteristics shown in the table.

Job	Entry Time	Burst Time
1	0	3
2	2	2
3	4	6
4	5	5
6	7	1

Determine the average turnaround (or completion) time for these jobs using each of these scheduling mechanisms:

a) FCFS
b) Shortest job first with preemption.
c) Round robin with time quantum of 3 seconds (no preemption).

3. Code has been written (see below) which claims to provide support for a critical section. Evaluate the code and answer the following:

a) Does the code meet the three criteria for a critical section?
b) If yes, justify your answer. If no, explain why not.

[There are two equivalent processes contending for control of a critical section, only one of which is shown below.]

```
program    DoesThisWork

     var        p1_wants_to_enter, p2_wants_to_enter:  boolean;

     procedure  p1;
        begin
           while true do
              begin
                 p1_wants_to_enter := true;
                 while p2_wants_to_enter do
                    begin
                       p1_wants_to_enter := false;
                       delay( short_random_time );
                       p1_wants_to_enter := true;
                    end;           /* while p2.....        */
                 CRITICAL SECTION

                 p1_wants_to_enter := false;
                 OTHER-NON-CRITICAL-STUFF
              end;           /* while true                  */
       end;                   /* procedure p1                */

      begin
         p1_wants_to_enter := false;
         p1;
      end.                       /* program  DoesThisWork       */
```

4. A particular river crossing is shared by both managers and engineers. Managers are assumed to be dangerous for engineers and so certain rules are required. You have been hired to write synchronization code to make sure that the engineers are safe. The boats used to cross the river seat three people, and must always carry a full load. In order to guarantee the safety of the engineers, no boat must ever carry more than one manager.

   You are to write code containing TWO procedures:

   `"Check_Engineer" and "Check_Manager".`

   The procedures group the arriving engineers and managers into safe boatloads. The appropriate procedure will be called by each engineer or manager when it arrives at the river crossing. The procedures shouldn't complete until enough passengers have arrived for a safe boatload; in other words, the procedures provide a "gate" to prevent entrance onto the boat until all three participants are present. The procedures for each of the three lucky passengers should return at about the same time.

a) List the constraints that must be satisfied before these procedures can return?
b)
c) Use this page and the next to write the code for Check_Engineer and Check_Manager in C using semaphores. Be sure to describe any shared variables and semaphores and give their initial values. There must NOT be any busy waiting in your solution.

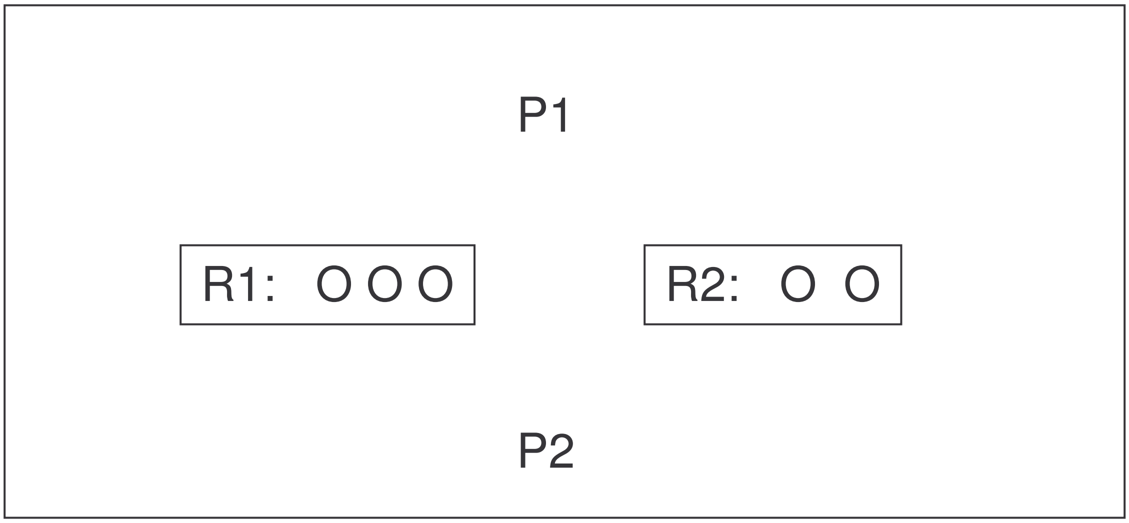
5. See the following snapshot of a system containing four resources A, B, C, & D.

	Current Allocation				**Maximum Need**				**Available**			
	A	**B**	**C**	**D**	**A**	**B**	**C**	**D**	**A**	**B**	**C**	**D**
p0	0	0	1	2	0	0	1	2	1	5	2	0
p1	1	0	0	0	1	7	5	0				
p2	1	3	5	4	2	3	5	6				
p3	0	6	3	2	0	6	5	2				
p4	0	0	1	4	0	6	5	6				

Answer the following questions using the Banker's/safety algorithm:

a) What is the content of the matrix "need"?
b) Is the system in a safe state? Prove it!
c) If a request from process p1 arrives for (0,4,2,0), can the request be immediately granted?

6. Consider the following resource graph in a state S, with maximum claims shown by dashed lines. P1 and P2 are processes, R1 and R2 are resources holding the number of resources shown:



In state S, none of the resources have been allocated, they just MIGHT be allocated.

a) Show a sequence of operations leading from state S to a deadlock state.
b) Show how deadlock could have been prevented, in your example in part a), by using an appropriate deadlock prevention technique.

7. A pure paging system has a page size of 512 words, a virtual memory of 512 pages numbered 0 through 511, and a physical memory of 10 page frames numbered 0 through 9. The current content of physical memory is shown to the right:

```
Physical                Frame
 Address    PHYSICAL    Number
       \     MEMORY     /
       0+------------+0
        ~            ~
        ~            ~
    1536+------------+3
        |  Page 34   |
    2048+------------+4
        |  Page 9    |
        +------------+5
        ~            ~
    3072+------------+6
        | Page Table |
    3584+------------+7
        |  Page 65   |
        +------------+8
        ~            ~
    4608+------------+9
        |  Page 10   |
        +------------+
```

a) Assuming that page tables contain frame numbers (rather than physical memory addresses), show the current content of the page table.
b) Show the content of the page table after page 49 is loaded at frame number 0 and page 34 is replaced by page 12.
c) What physical address is referenced by each of these virtual addresses:

4608,
5119,
5120,
33300.

d) What happens when virtual address 33000 is referenced?

8. Assuming a physical memory of FOUR page frames, give BOTH the sequence of faults AND the GRAND TOTAL number of page faults for the reference string

   a b g a d e a b a d e g d e

   for each of the following policies. (Initially all frames are empty.) It is recommended for a) and b) below that you draw the sequence of events by showing the pages which are in the FRAMES. For c) however, the simplest picture is one based on PAGES; showing which pages are physically in memory at each event.

   a) FIFO
   b) LRU
   c) Working Set (with T = 3). Do this one in terms of pages rather than frames.

How many page faults occur for each of these scheduling types?

9. We wish to evaluate various seek algorithms for a moveable head disk. This disk has 100 cylinders. Suppose the head is currently at cylinder 40, having just completed a transfer. During that transfer, the following requests (in chronological order) were received:

60, 40, 30, 70, 20, 10.

For EACH of the algorithms, FCFS, SSTF, SCAN, and C-SCAN

a) List the order in which the cylinders are accessed.
b) Determine the total seek distance travelled.