CS 414	**Operating Systems**	Test 3
Name:	Fall 2002	Max Pts: 64

Important: This is an open book test. You can use any books, notes, or paper. *Do not log into the computer, or use any communications device during the test.* Any calculations and rough work can be done on the back side of the test pages. You will lose five points for not writing your name.

1. [8 pt] Assume a system with four resource types, $C = \langle 8, 7, 3, 5 \rangle$ (this is the total number of resources in the system, and not what is currently available), and the maximum claim table shown below.

Process	R_0	R_1	R_2	R_3
p_0	6	2	3	1
p_1	4	7	2	3
p_2	4	0	0	3
p_3	0	2	1	3

The resource allocator is considering allocating resources according to the following table:

Process	R_0	R_1	R_2	R_3
p_0	1	0	1	1
p_1	2	4	0	1
p_2	3	0	0	3
p_3	0	2	1	0

Run the safety algorithm on this system to determine if this state is safe. If it is safe, give the order of execution of processes. If it is not safe, specify the processes that may participate in a deadlock.

2. [4 pt] Why do we have memory fragmentation problem in C?

3. [8 pt] In a virtual memory system, 1 in 400 references (on average) causes a page fault. When the page fault is to be serviced, 1 in 80 pages have their dirty bit set. Let the average seek time for the disk be 10 milliseconds, the average latency be 1 millisecond, and the average wait time in device queue be 5 milliseconds. In addition, the transfer time per page is 1 millisecond. Let the memory access time be 50 nanoseconds when there is no page fault. Compute the effective memory access time in this system.

4. [10 pt] You have a memory of 64 frames, with each frame being 1K bytes. Current free-frame list (in order) is: 17, 1C, 16, 1B, 4, 9, 11, 2F, 2E, 10, 18, and 28 (hexadecimal numbers). You just scheduled a process that requires 8 frames for execution. Can you allocate the frames to this process? Show the resulting page table, free frame list, and how the pages are allocated into frames by drawing a picture. Show the translation of logical addresses 0x639E and 0x08CC into physical addresses using your page table.

5. [18 pt] If FIFO page replacement is used with 4 page frames, how many page faults will occur with the reference string

7 0 6 5 3 1 5 2 5 0 2 3 0 6 3 2 4 3 5

if the frames are initially empty. Now repeat this problem for OPT, LRU, LFU, and second chance algorithm. How will it perform with a window size of 5 under the working-set algorithm (assume unlimited number of frames available for working set algorithm but working set window size is 5)?

6. [4 pt] What is the advantage of using magic number compared to filename extension to recognize a file content type? What is the advantage of using filename extension compared to magic number for the same?

7. [4 pt] Why do you use `mmap` and `munmap` calls? Why is it better than relying on system's virtual memory?

8. [8 pt] I have a floppy disk with 1.44MB [unformatted] capacity. The data blocks are 512 bytes each. I am going to create a Unix filesystem on this floppy with an empty boot block of size 1 block. Consider 1 block to be allocated for super block. Let each inode require 1024 bytes. What can be the maximum formatted capacity of the floppy? What is the maximum file size that can be stored on this floppy if the system uses 12 direct blocks, 1 single indirect block, and 1 double indirect block?