CS 376	**Operating Systems**	Test 3
Name:	Winter 1998	Max Pts: 54

Important: This is an open book test. You can use any books, notes, or paper. *Do not log into the computer during the test.* Any calculations and rough work can be done on the back side of the test pages.

1. [5 pt] Below is the listing of a short assembly language program for a computer with 512-byte pages. The program is located at address 1020, and its stack pointer is at 8192 (the stack grows toward 0). Give the page reference string generated by this program. Each instruction occupies 4 bytes (1 word), and both instruction and data references count in the reference string.

```
Load word 6144 into register 0
Push register 0 onto the stack
Call a procedure at 5120, stacking the return address
/* The procedure at 5120 consists of a single Return statement */
Subtract the immediate constant 16 from the stack pointer
Compare the actual parameter to the immediate constant 4
Jump if equal to 5152
```

2. [6 pt] If FIFO page replacement is used with four page frames and eight pages, how many page faults will occur with the reference string

$$0\ 1\ 7\ 2\ 3\ 2\ 7\ 1\ 0\ 3$$

if the four frames are initially empty. Now repeat this problem for LRU.

3. [6 pt] A computer provides each process with 65,536 bytes of address space divided into pages of 4096 bytes. A particular program has a text size of 32,768 bytes, a data size of 16,486 bytes, and a stack size of 15,870 bytes. Will this program fit in the address space? If the page size were 512 bytes, would it fit? Remember that a page may not contain parts of two different segments.

4. [5 pt] It has been observed that the number of instructions executed between page faults is directly proportional to the number of page frames allocated to a program. If the available memory is doubled, the mean interval between page faults is also doubled. Suppose that a normal instruction takes 1 μsec, but if a page fault occurs, it takes 2001 μsec. If a program takes 60 sec to run, during which time it gets 15,000 page faults, how long would it take to run if twice as much memory were available?

5. [5 pt] Explain the difference between internal fragmentation and external fragmentation. Which one occurs in paging systems. Which one occurs in systems using pure segmentation?

6. [4 pt] Systems that support sequential files always have an operation to rewind files. Do systems that support random access files need this too?

7. [5 pt] UNIX uses `rename(2)` to change the name of a file. Is there any difference at all between using this call to rename a file, and just copying the file to a new file with the new name, followed by deleting the old one?

8. [4 pt] Contiguous allocation of files leads to disk fragmentation. Is this internal fragmentation or external fragmentation? How does it differ from memory fragmentation?

9. [5 pt] After getting your degree, you apply for a job as director of a large university computer center that has just put its ancient operating system out to pasture and switched over to UNIX. You get the job. Fifteen minutes after starting work, your assistant bursts into office screaming: "Some students have discovered the algorithm we use for encrypting passwords and posted it on the bulletin board." What should you do?

10. [9 pt] Disk requests come in to the disk driver for cylinders 10, 22, 20, 2, 40, 6, and 38, in that order. A seek takes 6 msec per cylinder moved. The arm is currently at cylinder 20. The cylinders are numbered from 0 to 49. How much seek time is needed for

    (a) FCFS scheduling

    (b) SSTF scheduling

    (c) Elevator algorithm (initially moving away from cylinder 0)