THE UNIVERSITY
OF ADELAIDE
AUSTRALIA

SUB CRUCE LUMEN

The University of Adelaide

Examination for
Bachelor of Computer Science
Graduate Diploma in Computer Science
Master of Information Technology
Master of Computer Science
Master of Software Engineering

Semester 2, November 2008

Operating Systems
COMPSCI 3004, 7064

Official Reading Time:	10 mins
Writing Time:	120 mins
Total Duration:	130 mins

Questions	Time	Marks
Answer all 6 questions	120 mins	120 marks
		120 Total

Instructions

- Begin each answer on a new page
- Examination material must not be removed from the examination room

Materials

- 1 Blue book
- Foreign Language Dictionaries are Permitted

DO NOT COMMENCE WRITING UNTIL INSTRUCTED TO DO SO

Processes & threads

Question 1

(a) List three types of inter-process communication.

[3 marks]

(b) Which of the following are shared between multiple threads running in the same process?

i. CPU registers

ii. File descriptor table

iii. Data segment

iv. Stack segment

[4 marks]

(c) Describe the differences between a monolithic kernel and a microkernel design, and the advantages of each.

[5 marks]

(d) Under Unix, if `fork` did not return a value, could the program itself distinguish between the child and parent process? Explain your answer.

[4 marks]

(e) Some programming languages provide light-weight threading mechanisms implemented directly in user space. What are the advantages of doing this?

[4 marks]

[Total for Question 1: 20 marks]

CPU scheduling

Question 2

(a) List two factors that contribute to the cost of context switches.

[3 marks]

(b) Consider a typical a single-user desktop system, with a combination of interactive and background jobs running. Which of the following scheduling algorithms would you choose? Why?

- First come first served
- Round robin
- Priority scheduling

[4 marks]

(c) There are two processes running on a system, each with the following combinations of CPU and I/O bursts (numbers are in multiples of scheduling quanta):

A: 4 CPU, 2 I/O, 5 CPU, 3 I/O, 1 CPU, 4 I/O
B: 3 CPU, 4 I/O, 2 CPU, 2 I/O, 2 CPU, 1 I/O

For each of the following scheduling algorithms, what percentage of the total running time will the CPU be kept busy? Show your working.

i. Non-pre-emptive
ii. Pre-emptive round robin

[8 marks]

[Total for Question 2: 15 marks]

Input/Output

Question 3

(a) Compare and contrast two different strategies which may be adopted by an operating system to deal with deadlock.

[8 marks]

(b) What makes disk scheduling different from processor scheduling?

[4 marks]

(c) List 4 potential errors the disk driver needs to handle.

[4 marks]

[Total for Question 3: 16 marks]

Process Synchronization

Question 4

(a) Consider an atomic broadcast in which a producer shares a 1-item buffer with multiple consumers. Every item deposited by the producer has to be fetched by all k consumers before the producer can deposit another item into that buffer. Below is a draft solution for this problem. Explain why it won't work.

```
// Shared variables
item buffer;
int count = 0;
semaphore next = 0;
```

```
// Producer process                    // Consumer Process j
item x;                                item y;
while(true) {                          while(true) {
        // produce data x ...                  wait(next);
        while (count > 0) {                    y = buffer;
        }                                      count = count -1;
        buffer = x;                            signal(next)
        count = k;                             // use the data y ...
        signal(next);
}                                      }
```

[6 marks]

(b) Write a correct solution for the atomic broadcast problem described above when the number of consumers is k=3.

[9 marks]

[Total for Question 4: 15 marks]

Memory Management

Question 5

(a) You are given the following sequence of page accesses:

12, 14, 9, 11, 12, 9, 10, 11, 3, 14, 15

Evaluate the following page replacement policies: FIFO, LRU and Optimal, for a machine with four physical memory frames available, which are initially empty. Trace through the execution of all three policies, showing when page faults occur, where pages are loaded and the total count of page faults in each case.

[6 marks]

(b) Assume a modified FIFO policy that loads the page adjacent to the required page, as well as the page itself. Pairs of adjacent pages are those for which integer division (which rounds down) by two gives the same result. So for example:

pages 2 and 3 are adjacent

pages 10 and 11 are adjacent

pages 9 and 10 are NOT adjacent

Trace through the execution of the modified FIFO policy, showing when page faults occur, where pages are loaded and the total count of page faults.

[3 marks]

(c) For each of the questions below, write the name of the memory allocation methods (segmentation, demand paging or both) that will answer that question.

Which method ....

i. ...allows easier sharing of code, data?

ii. ...allows easier protection (R/W/E) of code, data?

iii. ...allows internal fragmentation?

iv. ...offers virtual memory?

v. ...requires less storage for map tables?

vi. ...requires less storage per map table entry?

vii. ...requires a TLB?

viii. ...uses variable sized partitions of physical memory?

[8 marks]

(d) What is a 2-level page table? Explain the need for it.

[5 marks]

(e) A computer keeps the page tables for its currents processes in memory. The overhead required to read an entry from the page table is 500 ns. To reduce the overhead, the computer has a TLB which holds 32 pair entries and can do a loop up in 100 nsec. What hit rate is needed to reduce the mean overhead to 200 ns? Note: Hit rate is the percentage of virtual page accesses that found the physical page frame in the TLB.

[4 marks]

[Total for Question 5: 26 marks]

Filesystems

Question 6

(a) Describe the main performance benefit of indexed block allocation (I-nodes) over linked allocation (FAT).

[3 marks]

(b) Describe the differences between a hard -ink and soft-link.

[4 marks]

(c) Consider a file system which uses a linked list to keep track of free blocks. The following sequence of operations is performed on a newly initialised filesystem, in which the pointer to unallocated blocks is set to block 0, and the free list is empty:

- A = new file of 2 blocks
- B = new file of 4 blocks
- C = new file of 1 block
- Append 1 block to A
- Append 3 blocks to B
- Append 2 blocks to A
- Append 1 block to C
- Append 2 blocks to B
- Delete C
- Delete A

Show the contents of the free list (as a sequence of block numbers) once these operations have completed.

You may assume that blocks are initally allocated in sequential order from the beginning of the disk. You may also assume that a file deletion causes each of its blocks to be freed in increasing numeric order, and that each free operation adds that block to the beginning of the free list.

Show your working.

[8 marks]

(d) Why is it important for an operating system to provide high level APIs for reading and changing the contents of directories, instead of having applications access directories as normal files, and manipulate them in their native on-disk format?

[4 marks]

(e) Three different protection mechanisms are: capabilities, access control list and the UNIX rwx bits. For each of the following protection problems, tell which of these mechanisms can be used:

i. Ken wants his files readable by everyone except his office mate.

ii. Mitch and Steve want to share some secret files.

iii. Linda wants some of her files to be public

[9 marks]

[Total for Question 6: 28 marks]

End of exam