56

SU EXAM

COMPUTING SCIENCE DEPARTMENT

CMPUT 379: Operating System Concepts

Friday, 20 April 2001

C379 Section B2

Closed book examination:

Two hours (1400-1600)

Note:

There are 7 questions, worth a total of 90 marks.

Closed book examination

Use of a basic calculator is permitted.

Concise, clear answers are expected.

Student ID:

Prepared by: Professor T.A. Marsland (April 2001)

Marks:

Q1	Q2	Q3	Q4	Q5	Q6	Q7	Total
15	10	15	10	15	15	10	90

1 [2 + 8 + 5 = 15 marks].

A feature of many real-time systems is that they run a set of cyclic tasks. In the table below there are three tasks. Task Ti is released Ri seconds after task T1, and restarts every Fi seconds thereafter. During its execution-window, task Ti must receive Ai seconds of CPU activity before its deadline (i.e., the starting time of the next occurrence of the same task).

(a). In the respective columns of the table below, fill in the priorities of these 3 tasks according to the two strategies: RMS (most frequent first) and U (largest utilization, Ai/Fi, first).

Task i	Release Ri	Frequency Fi	Activity Ai	RMS Priority	U Priority
T1	0	10	4		
T2	4	7	1		
T3	*6*	*12*	*3*		

(b). Complete the table below showing the release time, deadline time, start time and completion time for the first three initiations of each task, according to a priority pre-emptive scheduler employing your U priority strategy of part (a). Are all the deadlines met?

Task i	Ri1	Di1	Si1	Ci1	Ri2	Di2	Si2	Ci2	Ri3	Di3	Si3	Ci3
T1												
T2												
T3												

(c) Is the system RMS schedulable on theoretical grounds (hint use the RMS schedulability condition, below, and the data 2^(1/2) = 1.4142, 2^(1/3) = 1.2599, 2^(1/4) = 1.1892 etc.)? Is there a valid RMS schedule?

$$\sum_{i=1}^{k} (A_i/F_i) \quad < \quad U(k) = k \times (2^{1/k} - 1)$$

2 [10 marks].

a. Assume we have a demand-paged memory system with a TLB (Translation Lookaside Buffer) to hold the most active page-table entries. Assume also that the TLB has an access time of B seconds.

If the page table is held in primary memory, and memory access time is T seconds, give an expression for the effective access time to a resident page when P percent (or P fraction) of all memory references are resolved by the TLB. What does your expression reduce to as P tends to 100 (or 1).

Suppose we have a demand-paged memory, where the page table is held in registers (assume zero access time) instead of primary memory. The primary memory access time is given as T seconds. Assume it takes V seconds to service a page fault if an empty frame is available (or if the victim page/frame has not been modified), but D seconds if the victim page/frame is dirty (has been modified). Furthermore assume that the page to be replaced is dirty M% of the time.

b. Define the term page fault rate.

c. What is the biggest acceptable page-fault rate, F, for an effective (average) primary memory access time of no more than E seconds?

3 [15 marks].

Suppose we have a file system with a block size of 128 bytes and pointers of size 4 bytes. If a linked indexed (the index blocks are linked together) file allocation scheme is used:

a. What is the maximum file size? Explain your answer.
b. In order to read block number 68 of a file, how many blocks must be accessed? Explain your answer. Make (and state) reasonable assumptions about what is in main memory.
c. If a file is of size 1 Mbyte (10^6 bytes) how much external, internal and table (i.e., index block) fragmentation is there? Explain your answer.
d. Re-answer part (a) for the case of a 2-level indexed allocation scheme (one root block).
e. Re-answer part (a) for the case of a combined scheme (like Unix) where the top-level index block has 4 indirect pointers, 2 double indirect pointers and the rest of the space is used for direct (to blocks) pointers.

4 [10 marks].

a. What are the advantages and disadvantages of a memory management system using multi-level page tables?

Suppose we have the following form for a virtual address in a two-level page table:

P1	P2	offset
3 bits	5 bits	10 bits

b. What is the page size?

c. What is the maximum size of the logical address space for a program in this system?

d. How much table space will be required for this scheme, given that a page table entry requires 3 bytes of storage?

e. State briefly the advantages and disadvantages of having a large page size versus having a small page size in a demand paging system.

5 [7 + 7+ 1 = 15 marks].

A binary semaphore has only two integer values. 0 and 1.

(a) Show how a general (counting) semaphore can be implemented using binary semaphores. That is, present pseudo code for your binary P and V operators and for the general-purpose Wait and Free functions.

(b) Clearly identify the variables (including semaphores) you use by

* providing an appropriate declaration for each
* specifying their initial value
* stating what they are used for

(c) What is the legal range of values for your counting semaphore?

6 [10 marks].

Consider the following page reference string:

1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6, 1, 2, 3, 4, 2, 1, 5, 6, 2, 5

Fill in the following table with the number of page faults that would occur for a given number of page frames (1, 2, 3, and 5) and for a given page replacement algorithm. Briefly define the LRU and FIFO replacement methods in the context of this question Also state the optimal replacement stragegy.

Assume that all frames are initially empty, so your first unique pages will all cost one fault each. By some clear means show details of your work for at least the 3 and 5-frame LRU systems.

Number of Frames	Number of page faults		
	LRU	FIFO	OPT
2	xxxxxxxx		xxxxxxxx
3			
5			

7 [10 marks].

What is the impact of the size of the I/O buffers on the frequency with which the CPU scheduler is invoked? Choose (and circle) the best answer from the following 4 possible answers. Explain your choice.

1.large I/O buffers tend to reduce the frequency with which the CPU scheduler is invoked
2.large I/O buffers tend to increase the frequency with which the CPU scheduler is invoked
3.I/O buffers have nothing to do with the CPU scheduler
4.the relationship between I/O buffer size and the CPU scheduler is complicated and difficult to predict