University of Waterloo
Midterm Examination #2 Model Solution
Spring, 2007

1. (8 marks)

In a virtual memory system that uses simple paging and no TLB, each page table entry includes a use (reference) bit, a dirty (modified) bit, a valid bit, and some protection bits. Complete the grid below to indicate how these bits are used by the operating system and by the MMU hardware. In each grid cell, write "-" if the bit is neither read nor written, write "R" if the bit is read but not changed, and write "W" if the bit may be changed. For example, if you believe that the MMU does not use or change the protection bits, you should write "-" in the fourth row of the MMU column.

	Operating System	Memory Management Unit (MMU)
use bit	W	W
dirty bit	W	W
valid bit	W	R
protection bits	W	R

2. (5 marks)

Normally, an application program is not able to access the page tables for the process in which it is running. These page tables are accessible only to the operating system. However, if the operating system itself runs in a virtual address space then it must set up and maintain the page tables for its own address space. That is, the page tables for the operating system's virtual address space must be visible to the operating system.

The diagram below shows the (single level) page table for an operating system's address space. Only the virtual page number and corresponding frame number are shown for each page table entry. Assume that each of the entries shown is valid. Suppose that the page size in this system is 4096 (2^{12}) bytes, and that the *physical address* of the beginning of the operating system's page table is $0x3420$ (hexadecimal). The size of a page table entry (PTE) is 8 bytes. At which *virtual address* (in the operating system's address space) will the operating system find the PTE for its virtual page number 3.

Virtual Page Number	Physical Frame Number
0	0
1	12
2	2
3	7
4	9
5	8
6	20
7	4
8	3
9	1

Physical address $0x3420$ is offset $0x420$ into frame 3. Frame 3 holds virtual page 8, so the corresponding kernel virtual address of the start of the page table is $0x8420$. The PTE for page 3 will be found starting at virtual address $0x8438$, since each PTE occupies 8 bytes.

3. (10 total marks)

Consider an operating system that does process scheduling using multi-level feedback queues. There are two levels, and the operating system's scheduler uses the same quantum length, q, at each level.

a. (5 marks)

Suppose there are two processes, P_S and P_L, in the system. P_S is an interactive process that has short CPU bursts of length $t_c < q$, and blocks for time t_b after each CPU burst. Process P_L never blocks. Both processes have been in the system for a long time. When process P_S unblocks, how long will it have to wait before it is chosen by the scheduler to run?

$$\lceil \frac{t_b}{q} \rceil q - t_b$$

b. (5 marks)

Suppose instead that the system contains one process P_S as described in part (a), and N processes like P_L from part(a). Also suppose that the CPU burst for P_S is longer: $q < t_c < 2q$. How much time will elapse from the time that P_S starts executing one of its CPU bursts until the time that it finishes that CPU burst?

$t_c + Nq$

4. (9 total marks)

Consider a virtual memory system that uses 2-level paging. The page size in this system is 256 (2^8) bytes. Each individual page table fits exactly into one memory frame, and the size of each page table entry (PTE) is 8 bytes.

a. (3 marks)

What is the maximum size (in bytes) of a virtual address space in this system?

There are $2^8/2^3 = 2^5$ PTEs per page table. There can be up to 2^5 page tables in the second level, each with 2^5 PTEs, for at total of 2^{10} PTEs. Each refers to a page of size 2^8 bytes, so the maximum total address space size is $2^{10}2^8 = 2^{18}$ bytes (256 Kbytes).

b. (3 marks)

Suppose that there is a process with a virtual address space of the maximum size. How many bytes of memory are occupied by the page tables for this process?

There are 2^5 page tables in the second level plus one at the first level, each of size 2^8 bytes. Total size is $2^8(2^5 + 1) = 2^{13} + 2^8 = 8448$ bytes.

c. (3 marks)

Suppose that there is a process with a virtual address space of size 10240 ($10 \cdot 2^{10}$) bytes. Also suppose that the entire address space is in memory. How many *valid* PTEs will exist in the page tables for this process?

This process will occupy $10(2^{10})/2^8 = 40$ pages. This will require 40 PTEs in two page tables at the second level, and 2 PTEs at the first level, for a total of 42 PTEs.

5. (8 total marks)

a. (3 marks)

Consider the following basic CPU scheduling policies that were discussed in class: FIFO, round robin (RR), shortest job first (SJF), shortest remaining time first (SRTF), highest response ratio next (HRRN). Under which of these five policies is starvation possible? List *all and only* those policies under which starvation may occur.

SJF and SRTF may result in starvation.

b. (3 marks)

Under which of the five policies listed in part (a) is it necessary for the scheduler to predict process burst lengths? List *all and only* those policies under which burst lengths must be predicted.

SJF, SRTF, and HRRN require burst length predictions.

c. (2 marks)

Briefly and clearly explain the difference between demand paging and prefetching.

Demand paging means that pages are brought into memory only when they are needed, i.e., when a page fault occurs for a page. Prefetching means that pages are brought into memory before they are needed (before they cause a page fault).