University of Minnesota
Department of Computer Science
CSci 5103 - Fall 2016 (Instructor: Tripathi)
Midterm Exam 2 — Date: November 21, 2016
CLOSED BOOK/NOTES
(Time: 75 minutes) Total Points – 100

STUDENT NAME:
STUDENT ID:

Problem 1	Problem 2	Problem 3	Problem 4	Problem 5	Problem 6	Total Score
40	10	15	10	15	10	100

You do not need to use a calculator to write an exact numerical answer. Clearly show the steps that you follow to derive your answer.

Problem 1: (40 points) Eight short questions of 5 points each.

(a) (5 points) A system has two processes and three identical resources. These resources are acquired and used in exclusive mode by the two processes. Each process needs a maximum of two resources. Is deadlock possible? Briefly explain your answer.

(b) (5 points) A system has three files, A, B, and C, which are accessed by two processes, P and Q, in exclusive mode, by acquiring exclusive-mode locks on them. Suppose that process P always locks them in the order ABC. For process Q, we have the following six sequence orders possible for acquiring locks on these three files: ABC, ACB, BAC, BCA, CAB, and CBA. Which one of these orderings are guaranteed to be deadlock-free?

ABC	Yes	No
ACB	Yes	No
BAC	Yes	No
BCA	Yes	No
CAB	Yes	No
CBA	Yes	No

(c) (5 points) List two advantages and two disadvantages of having a large page-size in a virtual memory system.

(d) (5 points) Briefly state the relative advantages/disadvantages of global vs. local page replacement approaches.

(e) (5 points) Briefly state the relative advantages/disadvantages of using a disk partition vs. files for the swap space.

(f) (5 points) Under what conditions the swapper process would swap a process from the RAM to the secondary storage? How such conditions are detected by the kernel?

(g) (5 points) List at least three important fields contained in a *segment descriptor*.

(h) (5 points) Identify the cases under which the contents of the TLB in a virtual memory system are updated.

Problem 2: (10 points) A system consists of four processes and only one type of resource with 10 identical instances. The maximum need and the current allocation for each process is shown below. In this state, 8 instances are allocated and 2 are currently available. The current state of the system is given below:
Available = 2

Process	Allocation	Maximum Need
A	1	6
B	1	5
C	2	4
D	4	7

Question (a) (5 points) Is this system in a safe state? If yes, then give a completion sequence for the processes.

Question (b) (3 points) If process D requests one more resource, should the resource be granted to this request when Banker's Algorithm is used?

Question (c) (2 points) What if request came from C instead of D?

Problem 3: (15 points) A computer has 38-bit logical address with 16-KB page-size. It uses a two-level page table with 12-bits for the top-level page-table field, and 12-bits for the second-level page-table field.

On this system we have a process with three logical pages, one contains *text*, second contains the initialized and uninitialized data, and the third page contains the stack and other data such as environment variables and arguments. Assume that all these three pages are in the physical memory.

a (1 point) How large is the virtual address space? Provide your answers in powers of 2.

b (1 points) What is the total number of pages in the virtual address-space of a process? Provide your answers in powers of 2.

c (4 points) Assume that a page-table entry requires 4 bytes. How many page-frames are required for the first-level page table? How many page-frames are required for a second-level page table?

d (3 points) For the process described above, show how *text*, *data*, and *stack* pages are placed in its virtual address-space.

e (6 points) Assuming that all pages of this process are in memory, how many page-frames of the physical memory are needed to maintain all of the page-table related information for this process to do address translation. Show the organization of the top-level page-table, the second level page-tables, and the pages in memory. Show, using pointers, how these are inter-related.

Problem 4 :(10 points)

Suppose that a computer system has primary memory with access time of 200 nanoseconds. It also supports demand paged virtual memory. Assume the following:

(1) It takes 20 nanoseconds to search the TLB.

(2) The average time to read or write a page on disk is 10 milliseconds.

Answer the following questions: (Just write the formula, no need to use a calculator.)

a (2 points) What is the memory access time when we have TLB hit?

b (2 points) What is the memory access time when we have a TLB miss and the page-table contains a valid mapping for the accessed page, i.e. no page-fault is encountered.

c (3 points) What is the memory access time when we have a TLB miss, there is a page-fault, and the page selected for replacement is <u>not</u> dirty.

d (3 points) What is the memory access time when we have a TLB miss and also a page-fault, and the page selected for replacement is <u>dirty</u>.

Problem 5 (15 points) : Consider the following program:

```
var A, B: array[1..400] of integer;
for i := 1 to 400 do
    B[i] = A[i] + B[i];
endfor;
for i := 400 to 1 do
    B[i] = A[i] + B[i];
endfor;
```

An integer occupies a word and page size is 100 words. Assume that the code and the variable i are placed in logical page 0, which is always kept in memory so that access to i and instruction fetch do not produce a page fault. Assume that array A is stored in logical pages 1 through 4, and B is stored in pages 5 through 8. The main memory is initially empty.

Question (a) (7 points) What is the reference string generated by this program? (*Ignore access to page 0 for instructions and access to* i.)

Questoin (b) (8 poiints) When 4 frames are given to this program (in addition to the one for logical page 0 containing code and i) and under the LRU replacement policy determine the following:

- (4 points) Total number of page faults.
- (4 points) State of the 4 page frames after handling each page fault.

continue answer for Problem 5...

Problem 6: (15 points) Consider a system using the WSclock scheme for *working set* based virtual memory management. Suppose that the current virtual time 2000. This system had 8 page frames. Figure 1 shows the position of the clock hand, and the values of the *last-time-of-use* and the R bit for each frame.

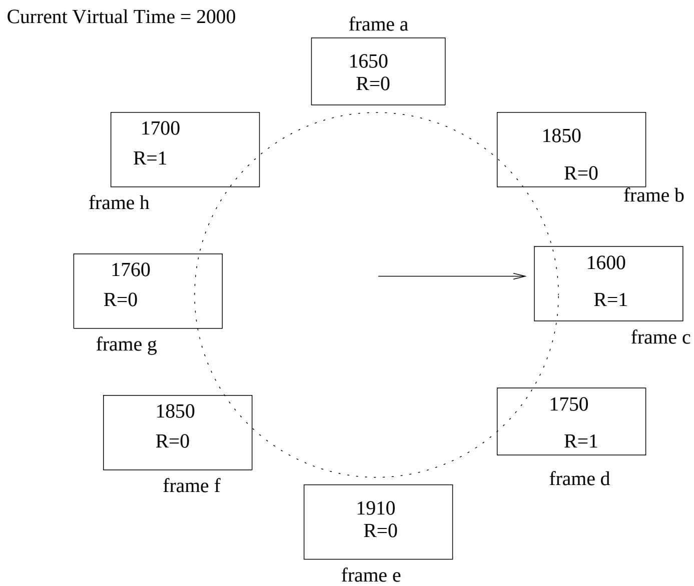

Figure 1: Frame status at the time of a page fault at virtual time = 2000

Question [a] (2 points) Suppose that the periodic clock interrupt comes virtual time 2000. Show the new contents of all the frame-list records.
Question [b] (6 points) Now suppose that instead of the clock interrupt a page-fault occurs at virtual time 2000. Answer the following questions for the working-set window size parameter τ equal to **200**:

1. (3 points) Indicate the frame whose page is replaced.
2. (3 points) Show the new contents of the frame-list.

Question [c] (2 points) Which page should be selected for replacement if τ is equal to **400** instead of 200.

continue answer for Problem 6...