

HAND IN

To make sure I can read your student number correctly, please write it as digits (987-6543) and as words (nine eight seven six five four three)

Student Number (digits) _____

Student Number (words) _____

Queen's University, Faculty of Arts and Science, School of Computing

Final Examination for CISC 324 and CMPE 324

April 21, 2014

D. Blostein

Instructions from Professor Blostein:

This examination is 3 hours in length. No aids are allowed. Answer all questions in the space provided on this exam question paper. There are 2 extra pages at the end of the exam if you need more space to answer questions. **Please write neatly.** Illegible work will not be marked.

There are a total of [100] points on this exam. Budget your time accordingly: the time limit of 180 minutes gives you 18 minutes to answer a 10 point question.

Page 2 [19] points

Page 3 [20] points

Page 4 [21] points

Page 5 [20] points

Page 6 [20] points

I cannot answer questions during the exam. If you find an exam question unclear, then make reasonable assumptions, write down those assumptions, and answer the question.

Instructions from Queen's University:

PLEASE NOTE: Proctors are unable to respond to queries about the interpretation of exam questions. Do your best to answer exam questions as written.

This material is copyrighted and is for the sole use of students registered in CISC/CMPE 324 and writing this exam. This material shall not be distributed or disseminated. Failure to abide by these conditions is a breach of copyright and may also constitute a breach of academic integrity under the University Senate's Academic Integrity Policy Statement.

[Circle an answer and **write a justification** that shows you understand this page replacement algorithm.]

- [6] 1. The optimal page replacement algorithm is optimal in the following sense:
- (a) it minimizes the amount of CPU time needed to calculate which page to replace
 - (b) it minimizes the number of page faults for one process (local page replacement algorithm)
 - (c) it minimizes the number of page faults for all processes (global page replacement algorithm)
 - (d) it minimizes the number of times dirty pages are replaced

[Circle an answer and **write a justification**.]

- [5] 2. Are there situations in which it is useful to have a recursive procedure inside a monitor? (In Java terminology: is it sometimes useful to make a “synchronized” method that is recursive?) YES NO

[Justify your answer, stating the purpose of the TLB and the meaning of *locality of addressing*.]

- [8] 3. Would a TLB (Translation Lookaside Buffer) be useful for a program that does not demonstrate locality of addressing? YES NO

[20] 4. Segmentation with paging, using notation from assignment 4: A virtual address SSPPEE (segment SS, page PP, offset EE) is translated to a 4 digit (hexadecimal) main memory address. The main memory contents below show the values in segment tables and page tables: a * means the page is on disk, and a two digit value is a frame number. Append 00_{16} to convert a frame number to a main memory address.

Address	Value	4200	12
1200	*		*
	*		C0
	*		
		C000	*
1800	*		*
	*		
	C1	C100	*
	12		20

STBR when process P1 is executing: 42
 STBR when process P2 is executing: 18

List of free page frames: D3, 9A, 80, 41, 06

P1 and P2 execute the following instructions. [Fill in the blanks; circle *Not Enough Info* if the answer cannot be determined from the given information. Show **changes to the segment and page tables** in the above figure, and **briefly justify** each answer.]

Process P1 executes the following four instructions

Instruction 1 MOV AX, 1717 ; Move the value 1717 into AX
 Instruction 2 MOV BX, 3939 ; Move the value 3939 into BX
 Instruction 3 MOV [010008], AAAA ; Move the value AAAA to virtual memory address 010008

VM address 010008 is translated to main memory address _____ *Not Enough Info*

Instruction 4 MOV [000102], FFFF ; Move the value FFFF to virtual memory address 000102

VM address 000102 is translated to main memory address _____ *Not Enough Info*

There is a context switch. Process P2 executes the following two instructions

Instruction 5 MOV AX, 8181 ; Move the value 8181 into AX
 Instruction 6 MOV BX, [030102] ; Move the value from VM address 030102 into BX

VM address 030102 is translated to main memory address _____ *Not Enough Info*

Now the value in the AX register is _____ *Not Enough Info*

Now the value in the BX register is _____ *Not Enough Info*

There is a context switch. Process P1 executes the following instruction

Instruction 7 MOV CX, 0 ; Move the value 0 into CX

Now the value in the AX register is _____ *Not Enough Info*

Now the value in the BX register is _____ *Not Enough Info*

- [5] 5. A process that uses a large virtual memory has a larger PCB than does a process that uses a small virtual memory. TRUE FALSE [Justify your answer]

[Circle an answer and **write a justification** that shows you understand this CPU scheduling algorithm.]

- [6] 6. The Shortest Remaining Time CPU scheduling algorithm is optimal in the following sense:
- (a) it minimizes the average turnaround time
 - (b) it maximizes throughput
 - (c) it minimizes response time
 - (d) it maximizes performance of real-time processes

[Your answer should briefly describe *capability-based protection* and *buffer overflow attacks*.]

- [10] 7. Explain how capability-based protection can be used to prevent buffer overflow attacks.

- [4] 8. The unix pipe, written as "|", is an implementation of which of the following? **[Justify your answer]**
 (a) readers and writers (b) bounded buffer (c) dining philosophers (d) elevator algorithm

- [16] 9. (a) A computer system uses segmentation. Segments are paged with three levels of paging. A virtual memory address is 32 bits long, with the top 6 bits used for the segment number:

6 bits Segment number	6 bits 1st level paging	6 bits 2nd level paging	6 bits 3rd level paging	8 bits Page offset
--------------------------	----------------------------	----------------------------	----------------------------	-----------------------

[Fill in the blanks; circle *Not Enough Info* if the answer cannot be determined from the given information.]

- (a1) The page size is _____ *Not Enough Info*
 (a2) The maximum number of segments is _____ *Not Enough Info*
 (a3) The maximum size of a segment is _____ *Not Enough Info*
 (a4) The size of main memory is _____ *Not Enough Info*
 (a5) The number of bytes in a page table entry is _____ *Not Enough Info*

[Draw a diagram similar to the above, to show the fields of the VM address. **Justify your answer.**]

- (b) Quadruple the page size: the new pages are four times as big as in (a1). No change is made to values (a2), (a3), (a4) or (a5). Now how is the virtual memory address divided into fields?

[20] 10. Write semaphore code to simulate customers buying donuts. The simulation uses two DonutMakerThreads and a large number of CustomerThreads. The donut shop has a display case that can hold up to 200 donuts, and DonutMaker threads make donuts whenever the display case isn't full. Each customer buys a dozen donuts. Write code for sections (1) to (4) in the following code outline. Use semaphores for synchronization, not busy waiting. Assume that semaphore queues are FIFO.

(1) **DECLARE SHARED VARIABLES. SHOW INITIAL VALUES, AND DOCUMENT THE PURPOSE OF EACH VARIABLE**

Outline of DonutMakerThread (two instances)

loop

(2) **SYNCHRONIZATION CODE BEFORE MAKING A DONUT** // Wait if the display case is full

sleep (10) // This "sleep" simulates the 10 seconds it takes to make a donut

(3) **SYNCHRONIZATION CODE AFTER MAKING A DONUT**

end loop

Outline of CustomerThread (many instances)

loop

sleep (randomly-chosen long time) // Simulate the time taken by other activities.

(4) **BUY DONUTS AT THE DONUT SHOP** // If other customers are already in the shop, wait

// until they have been served. Then buy a dozen donuts; this

// might require waiting for the DonutMakers to make more donuts.

end loop

Student Number _____

Page 7 of 8

Extra Page

Student Number _____

Extra Page