

CS 6210 Fall 2009 Midterm

Name: kishore GT Number:

Tuesday October 13, 2008 (9:30 to 11:00 AM)

Note:

1. Write your name and GT number on each page.
2. The test is **CLOSED BOOK** and **NOTES**.
3. Please provide the answers in the space provided. You can use scratch paper (provided by us) to figure things out (if needed) but you get credit **only** for what you put down in the space provided for each answer.
4. For conceptual questions, concise bullets (**not wordy sentences**) are preferred.
5. Where appropriate use figures to convey your points (a figure is worth a thousand words!)
6. Illegible answers are wrong answers.
7. Please look through the whole test before starting so that you can manage your time better.

Good luck!

Question number	Points earned	Running total
1 (1 min) (Max: 1 pts)		
2 (10 min) (Max: 10 pts)		
3 (15 min) (Max: 15 pts)		
4 (10 min) (Max: 10 pts)		
5 (15 min) (Max: 19 pts)		
6 (5 min) (Max: 10 pts)		
7 (5 min) (Max: 10 pts)		
8 (10 min) (Max: 10 pts)		
9 (5 min) (Max: 5 pts)		
10 (10 min) (Max: 10 pts)		
Total (86 min) (Max: 100 pts)		

1. (1 min, **1 point**)

Michael Vick

(Don't worry you get a point irrespective of your answer!)

- a. Plays for Chennai Superkings
- b. Plays for Atlanta Falcons
- c. Plays for Manchester United
- d. Plays for Philadelphia Eagles**
- e. Works for the Humane Society

CS 6210 Fall 2009 Midterm

Name: _____ GT Number: _____

2. (10 min, 10 points) (OS Structures - conceptual)

- (a) (5 points) What are the **highest two components** of potential overhead involved in border crossing from one protection domain to another? (choose the two that apply; -1 for each incorrect choice)
- A. Saving and restoring the process context blocks pertaining to the protection domains
 - B. Linked list maintenance for the process context blocks
 - C. Switching the page tables and associated processor registers
 - D. TLB misses
 - E. Cache misses
- (b) (5 points) A processor architecture provides segmented-paged virtual memory (similar to x86). The TLB contains virtual to physical address translations and is divided into two parts: USER and KERNEL.

(Answer True/False with justification) It is possible to avoid having to flush the TLB entries corresponding to the USER space upon a context switch.

3. (15 min, 15 points) (Exokernel - conceptual)

This question pertains to supporting multiple library OS images on top of Exokernel.

- (a) (5 points) What is the role performed by the software TLB?

- guaranteed $vm \rightarrow pm$ mappings for each library OS kept in S-TLB
- on TLB miss, Exokernel checks if mapping exists in S-TLB; if yes, it restores mapping to hardware TLB
- reduces startup memory penalty for a library OS upon a context switch

→ Yes. All domains colocated in same address space. Each domain given distinct segment register.

CS 6210 Fall 2009 Midterm

Name: _____ GT Number: _____

- (b) (5 points) What is the role performed by the Processor Environment?

PE is a data structure that contains 4 entry points corresponding to each library OS executing on top of Exokernel: (1) addressing context identifies the software TLB for that library OS, (2) interrupt context, exception context and protected entry context, respectively identify the handler entry points registered with exokernel for dealing with those three kinds of program discontinuities during the execution of a library OS.

- (c) (5 points) Enumerate the steps involved in context switching from one library OS to another.

- 1) The state of the currently executing library OS is saved using its PE. This includes saving hardware TLB in the specific software TLB for that library OS, and processor registers.
- 2) The state of the library OS chosen to be scheduled is loaded into the hardware TLB using the PE of the chosen one.
- 3) The volatile processor state (registers, PC, etc.) is also loaded using the PE of the chosen library OS.

CS 6210 Fall 2009 Midterm

Name: _____ GT Number: _____

4. (10 mins, 10 points) (Xen - conceptual)

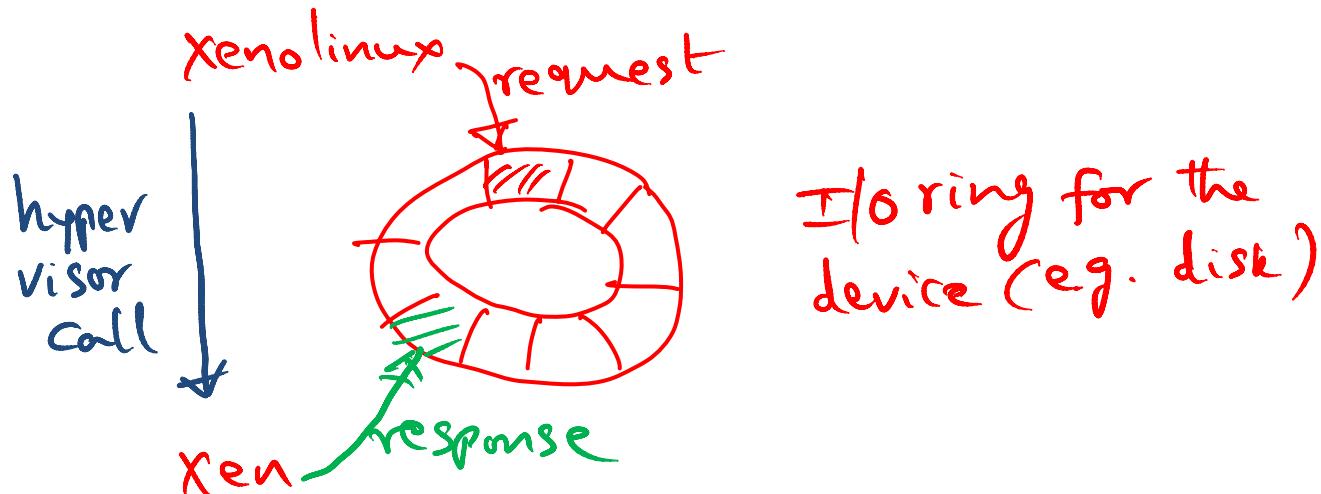
(a) (4 points) What is the role performed by I/O rings in the Xen architecture?

- Serves as a communication data structure between any given library OS and Xen
- Library OS places the necessary info corresponding to a hypervisor call in the I/O ring for Xen to pick up
- Similarly, Xen places its responses to hypervisor calls from a particular library OS in the associated I/O ring

(b) (6 points) A process "foo" executing in XenoLinux on top of Xen makes a blocking system call:

```
fd = fopen(<file-name>);
```

Show all the interactions between XenoLinux and Xen for this call. Use pictures to make your description easier to follow.



CS 6210 Fall 2009 Midterm

Name: _____ GT Number: _____

(Additional work space)

- xenolinux fields the `open` call (fastpath)
- in the `I/O ring` corresponding to the device that this pertains to, xenolinux places all the information for performing the `open` call (e.g. disk block address corresponding to `<filename>`).
- xenolinux makes a hypervisor call to convey the request to Xen
- Xen performs the request places response in `I/O ring`
- xenolinux picks up response and makes the blocked process runnable again

CS 6210 Fall 2009 Midterm

Name: _____ GT Number: _____

5. (15 mins, **19 points**) (Parallel systems - conceptual)

(a) (**4 points**) Consider the following simple code for lock acquisition.

Lock:

```
While (test-and-set(L) == locked) /* do nothing */;
```

Unlock:

```
L = unlocked
```

In an invalidation-based cache-coherent multiprocessor, the performance problems caused by the above algorithm are (choose all that apply; -1 for each incorrect choice):

- A. Increased network traffic upon lock release
- B. Increased latency for lock acquisition in the absence of any lock contention
- C. All processors spin on the same memory location in their respective caches
- D. All processors have to go to main memory every time through the loop

(b) (**5 points**) Qualitatively argue why Anderson's queuing lock algorithm that uses a unique memory location for each processor waiting for a lock mitigates the performance problem posed by the above algorithm.

- Each processor spins on a private variable awaiting its turn for the lock. Thus the spinning is local and does not cause bus traffic
- upon lock release, exactly one process's private variable is modified. So only this process goes to memory to perform the t+s operation while others continue to spin on their respective private variables in their caches

CS 6210 Fall 2009 Midterm

Name: _____ GT Number: _____

- (c) (5 points) In the absence of any optimization, for making an LRPC call from a client domain to a server domain on the same processor, there are four copies involved. Enumerate them.

- 1) client side stub creates a message of the actual arguments of the call
- 2) Kernel copies this message into a buffer in Kernel space
- 3) The Kernel copies this buffer from Kernel space into server domain
- 4) server side stub places the arguments on the stack before calling the actual procedure.

- (d) (5 points) Fixed Processor (FP) scheduling for a given thread has the following effect (choose the most appropriate selection):

- A. Good load balance
- B. Reduces the ill-effects of cache reload**
- C. Increases the ill-effects of cache reload

6. (5 mins, 10 points) (Tornado - conceptual)

- (a) (5 points) In building a subsystem (e.g., memory manager), you notice that an object is mostly shared in a read-only manner. You would choose to implement this object as a clustered object with (choose one of the following):
- A. Single copy shared on all the processors (a true shared object)
 - B. One copy per processor (fully replicated representation)**
 - C. One copy per group of processors (partially replicated representation)

- (b) (5 points) False sharing refers to (choose the most appropriate selection):

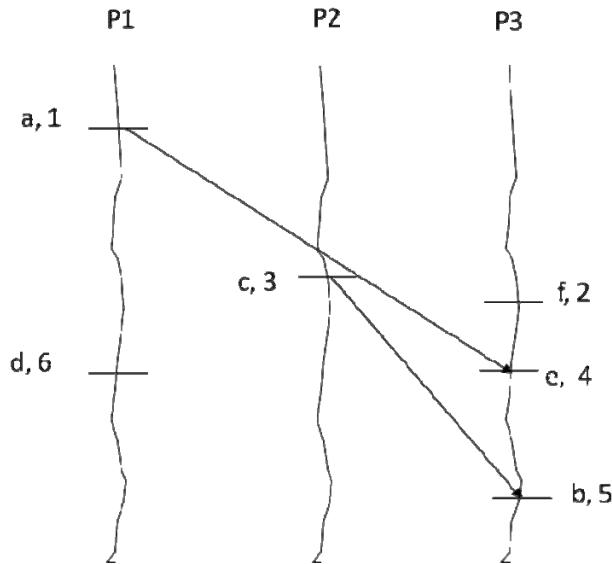
- A. A memory location being currently in the cache of a processor
- B. A memory location being present simultaneously in the caches of multiple processors in a shared memory parallel machine

- C. A memory location, which is private to a thread, appearing to be shared by multiple processors in a shared memory parallel machine**

CS 6210 Fall 2009 Midterm

Name: _____ GT Number: _____

7. (5 mins, **10 points**) (Lamport's clock - problem)



Choose \rightarrow or $||$ to show the relationship between the following events:

- (a, 1) \rightarrow (e, 4)
- (a, 1) $||$ (c, 3)
- (a, 1) \rightarrow (d, 6)
- (f, 2) $||$ (d, 6)
- (a, 1) \rightarrow (b, 5)

8. (10 mins, **10 points**) (Thekketh and Levy - conceptual)

- (a) (5 points) Identify the components of an RPC call that **ARE NOT** in the critical path of the latency (Choose all that apply; -1 for each incorrect choice):
- A. Time on the wire
 - B. Controller latency
 - C. Switching client out at the point of call
 - D. Interrupt service
 - E. Switching server in on call arrival
 - F. Switching server out on call completion
 - G. Switching client in upon call completion

CS 6210 Fall 2009 Midterm

Name: _____ GT Number: _____

- (b) (5 points) Assume DMA without scatter/gather. To send a message out at least one copy is needed in the kernel or user space before starting the DMA. Explain why.

Without scatter/gather, the DMA engine needs the entire message to be in one contiguous memory buffer. Thus either the RPC Stub or the kernel has to take parameters of an RPC call (or return values) and write into a contiguous memory buffer before starting the DMA.

9. (5 mins, 5 points) (Ensemble - conceptual)

Choose all that apply with respect to the advantages of component based design (-1 for each incorrect choice):

- A. Decouples specification, verification, and implementation
- B. Ease of development**
- C. Avoids side effects present in non-functional programming languages
- D. Ease of adaptation to meet the requirements of specific environments**
- E. Ease of extensibility**
- F. Optimal performance

CS 6210 Fall 2009 Midterm

Name: _____ GT Number: _____

10. (10 mins, **10 points**) (ANTS)

Consider an application wanting to send the same message to N of its peers. Let H_c be the number of common hops that the message has to traverse. Let H_u be the average number of unique hops to get to each of the intended destinations. Show an optimal active network for achieving this transmission (a picture is worth a thousand words).

