Computer Science 161

Midterm

80 minutes/80 points

Fill in your name and groupname below.

Name

Group

The points allocated to each problem correspond to how much time we think the problem should take. Please either place answers on a paper copy of the exam itself, or in a PLAIN TEXT file named lastname-161.txt. Mark the answers clearly so it is obvious what answers correspond to what questions. Please place your answers in order in the file. Send your completed exam to margo@eecs.harvard.edu. Please attach your answers to the email message.

This exam is open text book, open source code, and open notes. **You may not use computers**[1] **during the exam** (other than to record your answers or view OS/161 code). You must complete the exam on your own.

I have used only the appropriately designated materials while taking this exam.

Please sign your name here indicating that you used only designated materials while taking this exam.

Problem	Max	Actual
True/False	20	
Multiple Choice	15	
Short Answer	35	
Synchronization	10	
Total	**80**	

[1] Computers include cell phones, pdas, smart phones or any device with which you can talk to a network.

True/False (20 points)

For each statement, answer T if the answer is always true, else answer F. (2 points each)

1. The entire operating system must run in supervisor mode. T F F

2. It would be pointless to implement instructions in hardware that cannot be used some of the time. T F F

3. Every hardware device has a different interrupt handler. T F F

4. If the operating system is not multi-threaded, you cannot implement multi-threaded user processes. T F F

5. A context switch is the same as a domain crossing. T F F

6. Context switching is handled in hardware. T F F

7. Given a set of reasonable scheduling mechanisms including interrupts, it is possible to implement different scheduling policies. T F T

8. It is possible to implement multilevel feedback queues using lottery scheduling. T F T

9. There is no value in providing more physical memory than your virtual address space can map. T F F

10. The answer to the coming to class on time question is albatross. T F F

Multiple Choice (15 points) (3 points each)

1. The operating system on most of today's laptop computers is most similar in functionality and structure to: B

A. batch monitors

B. time-shared systems

C. 1980's era personal computers

D. 1980's era super computers

2. Consider a machine that requires all addresses by mapped through its 128 entry TLB. What is the maximum amount of memory that can be accessed, without incurring any TLB faults, if the page size is 4 KB? C

A. 128 KB

B. 128 MB

C. 512 KB

D. 512 MB

E. 4 GB

Consider an architecture that supports TLB superpages. A superpage is a region of memory, larger than the page size, that can be mapped with a single TLB entry. For example, let's consider a 64-bit architecture that has a base page size of 4 KB with superpages up to 1 MB. In this case, the TLB might look like:

	VPN (52 bits)	Mask (8 bits)	V (1)	PPN (52)	ATTR (4)	SIZE (8 bit)
A	0xFFFFFFFFFFFFF	0xFF	1	0x0000000000000	rwx	0xF0
B	0xFFFFFFFFFFFFF	0x00	1	0x0000000000000	rwx	0xF0
C	0xFFFFFFFFFFFFF	0xFF	0	0x0000000000000	rwx	0xF0
D	0xFFFFFFFFFFFFF	0x00	0	0x0000000000000	rwx	0xF0
E	0xDEADBEEFFADE0	0xFF	1	0x1111111111111	rwx	0x1

When we look up a virtual address, all entries with a 0 V (valid) bit are ignored. For the remaining entries, we compare the 52-bit page number in the address to the entries in the TLB, but apply up to an 8-bit mask, depending on the size of the superpage we wish to match (if all 8 bits are set, then we compare the entire 52-bit VPN, producing a 4 KB mapping; if none of the bits are set, then we compare only 44 of the bits in the VPN producing a 1 MB mapping. The ATTR field indicates the protections on the mapping and the SIZE indicates, in 4KB units, how much of the mapping is valid.

3. Which of the mappings in the table are non-sensical given the description of how this TLB works. (Select as many as you'd like.)A

A. A

B. B

C. C

D. D

E. E

4. Which of the following conditions requires that a contiguous 1 MB physical region of memory be mapped? (Select all that apply.) C, E

A. Yes, if the MASK field is 0x00.

B. Yes, if the SIZE field is 0x00.

C. Yes, if the SIZE field is 0xFF.

D. No, if the MASK field is something other than 0xFF.

E. No, if the SIZE field is something other than 0xFF.

5. Which of the following statements are true of the proposed architecture: A B D

A. Superpages increase the amount of memory that the TLB can map, but require more complex management of physical memory, because the different sized pages must be contiguous.

B. It is possible to map a region of size 24 KB with a single mapping.

C. It is possible to map a region of size 16 MB with a single mapping.

D. Two different TLB mappings can map into the same 1 MB of physical memory.

Short Answer (35)

6. (5 points) Propose a debugging feature that you wish you had while completing assignment 2. Explain how this feature would have been useful.

2 points for a clear description of a feature; 1 or 3 for the utility explanation.

7. (10 points) It turns out that scheduling an IT department is very much like scheduling processes on a heterogeneous multiprocessor. In this setup, the employees of the IT department are analagous to processors and support requests (SRs) are analagous to processes. This problem has the following constraints:

- Different employees have different skill sets.
- SRs require different types of expertise, but each SR requires only a particular kind of expertise (you may assume there are a small number of different kinds of expertise required).
- Requests from some customers are inherently more important than others (i.e., when the Dean wants something, that usually takes precedence).
- Some tasks have hard deadlines (i.e., if a course needs a website, then that probably needs to be satisfied before the semester begins).

Design a scheduler for the IT department in question. Describe it in terms of algorithms and mechanisms that we've discussed in class or in the book. Justify the decisions you make. Your answer should take the form of a paragraph or two.

8. (6 points) Recently, Professor Seltzer was auditing some code and she encountered the following code section with comment:

```
/*
* On every buffer put we update the buffer generation number and
* check for wraparound (comparing with MPOOL_MAX_PRIORITY). The
* increment doesn't need to be atomic, because it's OK if we
* sometimes lose an increment; __memp_reset_lru handles race
* conditions.
*/

if (++c_mp->lru_count == MPOOL_MAX_PRIORITY &&
     (t_ret = __memp_reset_lru(env, infop)) != 0 && ret == 0)
          ret = t_ret;
```

Do you agree/disagree with the author's explanation in the comment about not needing to protect the increment? Why/Why not?

Bad bad bad! Because we are checking against MPOOL_MAX_PRIORITY, if you happen to have a couple of threads through here at the same time, you could completely miss triggering the MPOOL_MAX_PRIORITY setting and wrap around without ever getting into the code after the &&.

2 points for identifying a problem and 4 (0,2,4) for clearly explaining the problem.

9. (10 points) Phase-change memory is a new memory technology that is nearly as fast as main memory, but is persistent, that is, it does not disappear when you lose power. Imagine that you had persistent main memory -- how would you either modify your VM system to exploit it or change how you use your VM system in managing a process's data.. Think about ways that main memory and persistent storage are different on today's machines and how making them the same might change things.

The key thing here in my mind is that a process can scribble on its own address space and then lose those scribbles; if your persistent data were also stored in memory, it might be more vulnerable to corruption, so you might want to use page-protection to keep more things read-only and handle faults to write.

There may be other good answers.

Grade on a 0, 3, 6, 8, 10 scale.

10. (4 points) You are designing a new processor. How do you determine which parts of the processor state must be stored into special registers or designated memory by the hardware on a trap. Be as precise as possible and give examples.

There are two key characteristics: if the particular piece of processor state can change once software begins to execute AND if the particular piece of state must be restored in order for what was previously running to continue to run correctly.

For example, because we need to know where the faulting process was executing, we need to capture the PC. However, if the kernel never uses K0, K1 with interrupts on, then we can trash those at random.

0, 2, 4, scale

Synchronization (10 points)

For each of the problems described below, write GOOD if the proposed synchronization primitive is a good choice with which to solve the problem or BAD if the synchronization primitive is not a good choice. If you wrote GOOD, include a one-sentence explanation of why it is a good choice. If you wrote BAD, suggest a better primitive and explain why in one to two sentences. For the purposes of this problem, you should consider the following primitives.

Counting and binary semphores
Lock w/condition variable
Locks
R/W Locks
Monitors

11. You have a friend that sends you mail once a week with a list of ten really entertaining URLs, many of which are videos with sound. You typically try to open all the links in parallel and then read them in whatever order they load, but one problem you always encounter is that while you're reading one page, you'll start to hear the audio track of some other page that has just finished loading. You've decided to build a new browser that performs all this synchronization for you automatically and figure that a simple binary semaphore will work -- which ever URL gets the semaphore is the page that gets control of the speakers and is the one you can currently hear. If you pause that one, you relinquish the semaphore. This can be GOOD or BAD depending on your implementation. Basically you need a LOCK, but it doesn't have to be ordered, so you can use a semaphore as a mutex and not worry about locking and it's OK. The problem is that one tab COULD V the semaphore that another tab P'd and that would be wrong. But, you don't have to use semaphores that way, so I'd be inclined to call this GOOD.

12. 2 points: 1 for GOOD/BAD with a 1 for a proper explanation.**Professor Seltzer has one gerbil and four cats and they all like to play in the same room. However, unlike her hamster, the gerbil does not do well playing with cats. She proposes using a binary semaphore for the gerbil and a counting semaphore for the cats to control access to the room.**

13. BAD; you need to have the cats and hamsters share a synchronization primitive; this is a classic case of RW locks.3 points: 1 for BAD; 1 for RW; 1 for explanation.**The Town of Lincoln has a 5-way intersection and most drivers are not clueful enough to properly determine when they should pass through the intersection. The Lincoln police have decided that they will distribute small in-car GPS-enabled, computerized sensors that will perform proper synchronization at the intersection. The sensors will display either GO or DON'T GO to the drive. The police, who happen to all be operating systems experts, claim that they can implement this easily with locks and condition variables.**

BAD: while locks and CVs seem like the proper primitive, signaling does not guarantee ordered activation and that would seem to be essential here. I supposed you could do this with locks and CVs if you had separate CVs for “waiting in line” versus “waiting at head of line” but I think that’s going to be more complicated than a queue with a lock around it or actually a monitor “arrive at intersection”, “advance” and “proceed through intersection”3 points: 1 for understanding the ordering problem; 1 for proposing a solution;

14. When it snows a lot in Cambridge, many two-way streets get reducd to one lane. This creates enormous problems, because if two cars get in the middle of the street heading in opposite directions, deadlock ensues. The City of Cambridge asked a former 161 student to help them solve the problem and they have now allocated a lock to each such street that can be requested and released from either end of the street.

GOOD; locks can do ordered entry into the street, and as long as you can’t proceed until you get the lock, all is good.2 points: 1 for GOOD; 1 for explanation