Name	UBID	Seat	Side

Question:	1	2	3	4	5	6	7	8	Total
Points:	10	5	5	5	5	5	5	20	50
Score:									

CSE 421/521 Midterm Exam

31 Mar 2017

Please fill out your name and UB ID number above. Also write your UB ID number at the bottom of each page of the exam in case the pages become separated.

This midterm exam consists of three types of questions:

1. **10 multiple choice** questions worth 1 point each. These are drawn directly from lecture slides and intended to be easy.
2. **6 short answer** questions worth 5 points each. You can answer as many as you want, but we will give you credit for your best four answers for a total of up to 20 points. You should be able to answer the short answer questions in four or five sentences.
3. **2 long answer** questions worth 20 points each. **Please answer only one long answer question.** If you answer both, we will only grade one. Your answer to the long answer should span a page or two.

Please answer each question as **clearly** and **succinctly** as possible. Feel free to draw pictures or diagrams if they help you to do so. **No aids of any kind are permitted.**

The point value assigned to each question is intended to suggest how to allocate your time. So you should work on a 5 point question for roughly 5 minutes.

I have neither given nor received help on this exam.

Sign and Date: ____________________

Multiple Choice

1. (10 points) Answer all **ten** of the following questions. Each is worth **one** point.

   (a) What does Carl say sometimes?

   ◯ right ◯ on that ◯ ummm ◯ thing-in-a-Bob

   (b) Intra-process (within) communication is easier than interprocess (between) communication.

   ◯ True ◯ False

   (c) A virtual address might point to all of the following *except*

   ◯ physical memory. ◯ a disk block. ◯ a port on a hardware device. ◯ a register on the CPU.

   (d) Con Kolivas is

   ◯ the maintainer of the Linux scheduling subsystem. ◯ a Turing award winner. ◯ opposed to the use of profanity. ◯ an Australian anaesthetist and Linux hacker.

   (e) Normal users are not aware of laptop

   ◯ responsiveness. ◯ screensavers. ◯ resource allocation. ◯ weight.

   (f) Process terminate by calling

   ◯ `kill_me_now()`. ◯ `kthanxbai()`. ◯ `_exit()`. ◯ `terminate()`.

   (g) Which of the following is the simplest scheduling algorithm to implement?

   ◯ Rotating staircase. ◯ Multi-level feedback queue. ◯ Round-robin. ◯ Random.

   (h) Which of the following is a requirement for deadlock?

   ◯ A single resource request. ◯ A cycle in the dependency graph. ◯ Unrestricted access to shared resources. ◯ Preemptible resources.

   (i) Which of the following is *not* an example of an operating system mechanism?

   ◯ Virtual to physical address translation. ◯ Using priorities to choose the next thread to run. ◯ An interrupt handler. ◯ Identifying interactive threads by observing their wait patterns.

   (j) True or false: the following code ensures that variable `foo` is protected? (Assume `foo_lock` exists and was properly initialized.)

```
lock_acquire(foo_lock);
// modify foo
lock_release(foo_lock);
```

   ◯ True. ◯ False.

 UB ID: | | | | | | | |

Short Answer

Choose **4 of the following 6** questions to answer. You may choose to answer additional questions, in which case you will receive credit for your best four answers.

2. (5 points) Given a simple MLFQ approach that rewards any thread that sleeps or yields before its quantum expires, first describe a way that a computationally-intensive thread can take advantage of a weakness in this approach to remain in one of the top queues (2 points). Second, propose a modification to MLFQ that addresses this problem (3 points).

UB ID: | | | | | | | |

3. (5 points) Describe two changes to the OS virtual memory system that you would need or might want to make to accommodate 48-bit virtual and physical addresses (1 point each). For each, describe how it would affect the computation or memory overhead of virtual to physical translation (2 points each).

 UB ID:

4. (5 points) We have discussed several cases where operating systems provide a useful illusion to processes. Name one (1 point), describe why it is useful (2 points), and briefly explain how it is provided, identifying any hardware cooperation required (2 points).

 UB ID:

5. (5 points) Provide one example of a exception that would terminate a running process (1 point) and one example of an exception that would not (1 point). Describe what happens when an exception occurs (3 points).

UB ID:

6. (5 points) We've presented synchronization primitives that use both active (or busy) and inactive (or blocking) waiting. First, explain the difference (3 points). Second, for each describe a scenario in which that form of waiting is more efficient and why (1 point per scenario).

UB ID: | | | | | | | |

7. (5 points) Operating systems require special privileges to multiplex memory. Below, describe:

   - **what** special privileges are required (1 point),
   - **how** they are used (2 points),
   - and **why** they are needed (2 points).

 UB ID:

Long Answer

Choose 1 of the following questions to answer. **Do not answer both questions.** If you do, we will only read the shorter one. If you need additional space, continue and clearly label your answer on other exam sheets.

8. (20 points) Choose one of the following questions to answer:

   1. **Wait Time Prediction.** When discussing schedulers, one of the aspects of future that we wanted to be able to predict was the wait time when a thread blocks. To be more concrete, when a thread blocked and was moved to the waiting queue, we might want to know *how long* the wait will take.

      First, discuss how online prediction of wait times might be accomplished on a real system—no crystal balls allowed (10 points). Identify a few things that a thread might block waiting for, and for each discuss whether a prediction algorithm makes sense and how it might be implemented. You might think about applying some of the system design principles we've discussed in class.

      Second, discuss how to incorporate this information into a scheduling algorithm (10 points). Feel free to choose one of the scheduling algorithms we've discussed in class in order to make your solution more concrete. You should explain how to use the output of your wait predictor and argue that it can improve some aspect of scheduler performance.

   ---

   2. **Jumbo Pages.** While operating system pages have traditionally been 4K, some modern operating systems support "jumbo" pages as large as 64K. Based on your excellent and fast implementation of virtual memory for CSE 421 ASST3 you are hired as a kernel developer for the new Lindows © operating system company. Unfortunately, your boss didn't take CSE 421 and derives most of his understanding of operating systems from the movie "Her"[1]. At present, Lindows does not support jumbo pages, but once your boss hears about them he is desperate to include them into Lindows © version 0.0.0.2. He asks you for help.

      First, explain why how and in what cases 64K pages would improve or degrade OS performance (10 points). What information about virtual memory use could help the OS decide whether to locate content on a jumbo or regular-sized page? Second, explain how, in certain cases, you can implement jumbo-page-like functionality on top of an existing system that supports 4K pages *without* changing the underlying memory management hardware (10 points). What MMU features are required for this to work? Which benefits of jumbo pages are preserved or lost by your approach?

[1] Her is a 2013 American science fiction romantic comedy-drama film centering on a man who develops a relationship with an intelligent OS with a female voice and personality. (Wikipedia)

 UB ID: