

CS 4210 - Advanced Operating Systems - Summer 2002

Midterm Exam

- There are 100 points on this test. Point values are in () on each question.
 - Read the whole test before you begin - allocate your time wisely.
 - Longer answers are not necessarily better.
 - **Write legibly!**
1. (10) What does it mean for a function or system call to be *async safe*? What is the potential problem with functions that are not async safe? (Hint: `async safe == signal safe`)
 2. (10) You've graduated and gotten a programming job writing multithreaded programs. You walk past the office of a co-worker who is busy diagnosing performance problems in her code. You hear her muttering "A chain is only as strong as its weakest link" over and over. What multithreaded design pattern is she using? Why would she say what she said?
 3. (or, Be Smarter Than Me) (15) In class I managed to make a bog of writing the path expression for the One-Slot-Buffer problem. Recall that One-Slot-Buffer is a special case of readers-writers that involves a single "mailbox" whose contents must be protected against overwriting.
 - Give the correct path expression for One-Slot-Buffer.
 - The (closely related) One-Slot-Bulletin-Board problem uses a "bulletin-board" instead of a "mailbox". It is different in that all threads must see any message posted on the bulletin board before the message can be overwritten. Why can't we use path expressions to effectively address this problem?
 4. (10) When the kernel scheduler has no knowledge of user-level threads, it may pre-empt a thread that holds a lock for which all other threads will then have to wait. How did the designers of the Psyche system deal with this issue?
 5. (35) Consider the following synchronization problem. An application has three kinds of threads, red, green, and blue. Red threads have higher priority than green threads, which in turn have higher priority than blue threads. This means that no blue thread can acquire the resource if any red or green threads are waiting, and no green thread can acquire the resource if any red thread is waiting. Only one thread can be in the critical section at a time.

Show the synchronization code for all three kinds of threads (i.e. the synchronization variables, as well as what a thread must do before entering a critical section and what it must do after leaving a critical section. Use the Lock, Mutex, and Condition pseudocode from class.
 6. (10) You're designing a windowing system for Solaris 8 (which runs on the SunOS 5 multithreaded kernel). Which scheduling class would you use for the thread that monitors the mouse? Why?
 7. (10) Are monitors a good choice to implement the many-readers-one-writer problem? Why or why not?