CSci 402 - Operating Systems

Quiz 6

Fall 2023

Friay, Oct 6

Instructor: Bill Cheng

Teaching Assistant: Zhuojin Li

(This exam is open book and open notes.
Remember what you have promised when you signed your
Academic Integrity Honor Code Pledge.)

Time: (N/A) minutes

_____-
Name (please print)

Total: 10 points

_____-
Signature

Instructions

1. This is the first page of your exam. The previous page is a title page and does not have a page number. Since this is a take-home exam, no need to sign above since you won't submit this file.
2. Read problem descriptions carefully. You may not receive any credit if you answer the wrong question. Furthermore, if a problem says *"in N words or less"*, use that as a hint that N words or less are expected in the answer (your answer can be longer if you want). Please note that points may get *deducted* if you put in wrong stuff in your answer.
3. If a question doesn't say `weenix`, please do not give `weenix`-specific answers.
4. Write answers to all problems in the **answers text file**.
5. For non-multiple-choice and non-fill-in-the blank questions, please show all work (if applicable and appropriate). If you cannot finish a problem, your written work may help us to give you partial credit. We may not give full credit for answers only (i.e., for answers that do not show any work). Grading can only be based on what you wrote and cannot be based on what's on your mind when you wrote your answers.
6. Please do *not* just draw pictures to answer questions (unless you are specifically asked to draw pictures). Pictures will not be considered for grading unless they are clearly explained with words, equations, and/or formulas. It's very difficult to draw pictures in a text file and you are not permitted to submit additional files other than the answers text file.
7. For problems that have multiple parts, please clearly *label* which part you are providing answers for.
8. Please ignore minor spelling and grammatical errors. They do not make an answer invalid or incorrect.
9. During the exam, please only ask questions to *clarify* problems. Questions such as "would it be okay if I answer it this way" will not be answered (unless it can be answered to the whole class). Also, you are suppose to know the definitions and abbreviations/acronyms of *all technical terms*. We cannot "clarify" them for you. We also will **not** answer any clarification-type question for multiple choice problems since that would often give answers away.
10. Unless otherwise specified and stated explicitly, multiple choice questions have one or more correct answers. You will get points for selecting correct ones and you will lose points for selecting wrong ones.
11. When we grade your exam, we must assume that you wrote what you meant and you meant what you wrote. So, please write your answers accordingly.

(Q1) (2 points) Which of the following statements are correct about **Kernel 1**? Please note that unlike the last quiz, we are making a clear distinction between "process" and "thread" in this problem.

(1) your thread must not call do_wait_pid() with a negative number being the first argument
(2) the 3rd argument of do_wait_pid() is used to get the return/exit code of a dead child process
(3) the 3rd argument of kthread_create() specifies the first procedure of the child thread you are trying to create
(4) when thread X calls sched_make_runnable(Y), you are adding thread Y to the run queue
(5) when your thread calls do_exit(n), you are setting your thread's return/exit code to n

Answer (just give numbers): _____

(Q2) (2 points) Which of the following statements are correct about **the life cycle of a kernel thread**?

(1) if your kernel thread is running inside the CPU, it can add itself to the run queue
(2) if kernel thread X is sitting in a mutex queue when thread Y cancels it, thread X's state is changed to the terminated/zombie state immediately
(3) if kernel thread X is sitting in an I/O queue and got woken up inside an ISR (interrupt service routine), the ISR code will move thread X into the run queue
(4) if kernel thread X is sitting in the run queue when thread Y cancels it, thread X's state is changed to the terminated/zombie state immediately
(5) none of the above is a correct answer

Answer (just give numbers): _____

(Q3) (2 points) Which of the following statements are correct about **thread implementation strategies**?

(1) in the 1 $\times$ 1 model, threads are created in user space but thread switching is done in the kernel
(2) in the M $\times$ N model, threads are created in the kernel but thread switching is done in user space
(3) in the N $\times$ 1 model, threads are created in user space and thread switching is done in user space
(4) in the 1 $\times$ 1 model, threads are not allowed to make system calls
(5) in the M $\times$ N model, threads are not allowed to make system calls

Answer (just give numbers): _____

(Q4) (2 points) Which of the following statements are correct about **mutex implementations**?

(1) under straight-threads, if thread Y is holding mutex m when thread X calls mutex_lock(m), thread X will go to sleep in the run queue
(2) when there are multiple CPUs and no interrupts are allowed, if thread Y is holding mutex m when thread X calls spin_lock(m), thread X will do busy-waiting
(3) spin_unlock() is a blocking call
(4) mutex_unlock() is a blocking call
(5) none of the above is a correct answer

Answer (just give numbers): _____

(Q5) (2 points) Assuming that you have only one CPU, which of the following statements are correct about a **non-preemptive kernel**?

(1) in a non-preemptive kernel, a kernel thread can be forced to give up the CPU when a hardware interrupt is being delivered
(2) in a non-preemptive kernel, a kernel thread can be preempted by another kernel thread
(3) in a non-preemptive kernel, there is no need to implement kernel mutexes
(4) the weenix kernel is a non-preemptive kernel
(5) none of the above is a correct answer

Answer (just give numbers): _____