CS 314 Operating Systems
Spring 2023
Quiz #5 on February 16, 2023 (SOLUTIONS)

Your Last Three Digits: ______________
(please do NOT write all of your student ID or your name)

Grade: ______

************************************************************

(1) What is “critical section”?

(a) A critical section is a part (section) in a program that can cause a race condition, (b) which requires “mutual exclusion” to prevent a race condition from happening.

Note: for full credit, either (a) or (b) should be presented.

(2) What does “atomic” in “atomic operations” mean?

Atomic operations are those (a) that will not be interrupted (of preempted) in such a way that (b) once an atomic operation starts (i.e., once a processor starts performing one) it will not be interrupted until it is completed (finished).

(c) An example of “atomic operations” is an assembly instruction, which is the minimum task unit (undividable to any smaller tasks) for processors.

Note 1: for full credit, either (a) or (b) is required.

Note 2: “assembly instructions” are just an example of “atomic operations” (i.e., assembly instructions are atomic operations, but atomic operations are not limited to assembly instructions). Thus, just saying “assembly instructions” for this question is not good enough.
(3) What “signal” system call to a semaphore exactly performs?

**Signal**
- If no one waiting on S, set $S = 1$
- If some one waiting on S, let the first proceed to CS and leave $S = 0$

(4) What are “counting semaphores” (how are “counting semaphores” different from “binary (mutex) semaphores”)?

(a) Counting semaphores are those semaphores that are initialized by a positive integer larger than ‘1’ (of those that take values of larger than ‘1’).

(b) Since counting semaphores can take a value larger than ‘1’, counting semaphores are not for protecting critical sections but for controlling the number of processes without using a spin-wait.

**Note:** the first response (a) is required for full credit.

(5) Operating systems use “queue (FIFO data structure)” for managing processes blocked on a semaphore. Why is FIFO-queue used (the best reason for using FIFO structure)?

The primary reason for operating systems to use a queue (FIFO) is primarily eliminating the chances for process starvations (for those processes that wait for a semaphore).

**Note:** as long as a solution presents a similar concept, such as “guaranteeing fairness”, it should earn full credit.