Midterm Study Guide

CSCI 432: Fall 2023

Exam due date: Tuesday, Oct 24, at 5pm.

This is a closed book exam. You are allowed one sheet of notes (front side only); use of any other sources such as students, the Internet, or ChatGPT are prohibited. **Please submit your notes page with your exam.**

Read each question carefully and note all that is required of you. Keep your answers *clear and concise*. Often when you write more than you need, you end up saying things that are incorrect. I don't expect you to need more than a page for any sub-question. You may use the back side of the exam pages for scrap work. **If you use additional scrap paper, please submit it with your exam.**

If you have questions or need clarifications, you can try to find me in my office during normal working hours. If you cannot find me, state your assumptions clearly. You will not be penalized as long as they are reasonable.

You have 2.5 hours to complete this exam. Please indicate the start and end time below. If you have extra time accommodations, you may take a break. If you plan to use this option, however, please do not read ahead or stop in the middle of a question. Your total thinking time should not exceed your time limit.

You are responsible for anything we covered in class or in the projects, and everything in the assigned readings (i.e., the papers). This exam will focus on topics related to threads, concurrency, synchronization, and scheduling. You will **not** be tested on any newer material related to memory management.

The following list covers many of the topics we have touched on:

- Processes and threads: differences and similarities.
- Atomic actions: what are they and why do we need them.
- Mutexes/Locks: general use and purpose.
- Monitors and CVs: general use and purpose.
- Semaphores: general use and purpose.
- Compare and contrast use of monitors versus semaphores.
- Classic synchronization problems covered in class: Bounded buffer (soda machine), got milk, dining philosophers, sleeping barbers.
- Producer-consumer and reader-writer synchronization problems and solutions.
- Implementing threads and locks: project 1, context switching, shared and private state, implementing synchronization primitives, test and set, interrupts.
- Deadlock: what conditions are required for deadlock.
- Mechanisms for deadlock avoidance (Banker's algorithm).
- Scheduling algorithms: FCFS, RR, STCF, STCF-P, EDF.
- Tradeoffs between scheduling algorithms.
- Papers: Lampson, UNIX, and Eraser.