# Syllabus - CSCE 451/851 Operating Systems Principles

Department of Computer Science and Engineering
University of Nebraska – Lincoln
Spring 2021

**Instructor:**
Justin Bradley 290 Schorr
Office hours: By appointment via Zoom
Email: justin.bradley@unl.edu
Phone: (402) 472-5072

**TAs:**
Minh Vu
Office hours: Mondays 9:30-11:30 via Zoom
Email: minh.vu@huskers.unl.edu

Brennan Rhoadarmer
Office hours: Wednesdays and Thursdays 17:30-19:30 via Zoom
Email: brhoadarmer@huskers.unl.edu

# 1 Course Overview

This course will introduce students to key concepts of modern operating systems. These concepts include system organization, processes, threads, process management, process and thread synchronization, memory management, and storage management.

**Topics covered:** Organization and structure of operating systems. Processes and concurrent programming. Inter-process communication. Process synchronization using reads/writes, semaphores, and monitors. CPU scheduling. Deadlocks including detection, avoidance, prevention and recovery. Intro to real-time systems. Memory organization and management including paging, segmentation, virtual memory, and page replacement algorithms. File system concepts and structure. Protection and security. Topics are conveyed through lectures and a series of hands on programming assignments.

# 2 Course Requirements

**Prerequisites:** A grade of “P” or “C” or better in CSCE 230, CSCE 230H or CSCE 231 and CSCE 310, CSCE 310H, CSCE 311, SOFT 260, SOFT 260H or RAIK 283H.

**Text:** Bic, Lubomir. *Operating Systems Principles.* zyBooks, 2020, zyBook ISBN: 978-1-394-06717-6

# 3 Class Details

The two sections of the course will be identical with the (minor) differences indicated below.

## 3.1 Section 001 (CSCE 451/851-001)

This is a “Web Conferencing” version of the course. Asynchronous video lectures and readings in the book will be used to introduce concepts to students. Weekly synchronous, live Zoom lectures will be used to supplement those lectures with a focus on successfully completing the programming assignments. This allows students in this section to directly interact with me, ask questions, and receive feedback.

Class: Tuesdays 09:30 - 10:45 (all other lectures asynchronous)
Final exam: 00:00-23:59 Tuesday, May 4, 2021

## 3.2 Section 700 (CSCE 451/851-700)

This is an “Online” version of the course. Like Section 001 above, asynchronous video lectures and readings in the book will be used to introduce concepts to students. The weekly live Zoom lectures from Section 001 will be recorded and disseminated in this section.

Class: Asynchronous video lectures
Final exam: 00:00-23:59 Thursday, May 6, 2021

# 4 Website, Communication, and Other Logistics

### 4.1 Canvas

We will use Canvas to disseminate materials, host the class schedule, maintain all links, etc. To simplify everyone’s life the front page of the course is a module containing two pages:

1. **Announcements**: this will be a regularly updated, running list of announcements. You need to check this regularly. Canvas, depending on your settings, may send you a notification that the page has been updated. You’ll then need to go check it. I do it this way to reduce the number of Canvas announcement emails you receive. But it’s on you to check the announcements so you’re informed!

2. **Main Schedule**: this page holds a weeky-by-week schedule of the topics, resources, links, assignments, etc. Look here to find out what you should do, and where to get materials and resources. This will be updated over the course of the semester as topics, resources, and assignments are updated.

## 4.2 Other Resources

Other external materials, links, etc. will be used as follows (all links to these will be in the “Main Schedule” page in Canvas:

- **Piazza** will be used for all online discussion of homework, programming assignments, and logistics.
- **YouTube** will be used to host lecture videos.
- **Microsoft OneDrive** will be used to host programming assignments and other handouts.
- **Zoom** will be used for synchronous lectures (section 001 only).
- **zyBooks** will be used as our book. Readings and homework assignments will be given from this book, and homework assignments, based on the readings, will be given. You will have to subscribe to the book (it is much cheaper than the previous book).
- **CSE Handin** will be used to turn in programming assignments.

# 5 Policies

## 5.1 Grading Policies

This course will consist of 6 (graded) programming assignments, weekly reading check assignments, and a final exam. The grade breakdown will be:

- Programming Assignments: 60%
  - PA0 – 5%
  - PA1 – 11%
  - PA2 – 11%
  - PA3 – 11%
  - PA4 – 11%
  - PA5 – 11%
- Exams: 20%
  - Final Exam: 20%
- Homework - 20%
  - 7 reading/homework assignments – 20%/7 = ~2.86%
- Extra Credit:
  - Evaluation: 2% bump to final percentage (only if >80% of class fills out evaluation)

### 5.1.1 Late Work Policy

Late work will **NOT** be accepted. I *will* make exceptions for UNL-approved reasons. I *may* make exceptions for other reasons **if** you discuss it with me before the deadline.

**Explanation**: I have tried several variations on accepting late work. Accepting late work means more work for me and the TAs, and this class already has a high grading burden. My anecdotal evidence suggests the rate of students turning things in is about the same. Not accepting late work is simplest and encourages students to get started on assignments earlier.

### 5.1.2 Differences between 451 and 851

CSCE 851 students will take a different, more comprehensive final exam. Otherwise, assignments between 451 and 851 are identical.

### 5.1.3 A *very* important note on grading of programming assignments:

- Programming assignments will be graded on a 100 point scale.
- **Your program *must* compile and execute on the CSE servers (cse.unl.edu) - else you get 0/100. Make sure you check this prior to handing in.**
- All programming assignments must be handed into CSE web handin - else you get 0/100.
- Each programming assignment will indicate precisely how it will be scored. **Most** of the time points are gained by passing the test suites. Test suites are often broken into different parts so that you can get at least partial credit for partial implementations. You won't get any points for tests that you don't pass. This means there's no such thing as "being close" or "almost works."
- Despite the previous point, I also reserve the right to take off points for poor programming practices: poor commenting, function names, variable names, magic numbers, etc.

### 5.1.4 Explanations

Students have a variety of reactions to these policies. Here are some common ones, and associated explanations:

- "*That's stupid!*": When you develop code in industry you don't get "credit" if you break the nightly build (i.e., your code doesn't compile), or your program simply doesn't do what it is supposed to do (i.e., test suite fails). Even so, most PAs have a suite of tests that allow you to get partial credit for implementing as much as you can.
- "*That doesn't accurately reflect what I learned!*": I'm in the business of preparing our students to be strong employees. As 4th year undergraduates, or graduate students, good programming practice is simply an expectation of a computer scientist. Writing programs that meet specifications is the *bare minimum* any computer scientist can do. In most PAs you will have access to the test suites or an approximate set of test suites, so it should be straightforward to test all your code, and make sure it compiles and runs on the CSE servers.
- "*Why not partial credit?*" or "*I was so close…*" You will get the indicated points for passing various tests within the test suite. If your code doesn't pass a test you get 0 points for that particular test. There are several reasons for this:
    - I don't know how to give partial credit. If the output of your code doesn't match the required output, I (or the TA) would need to look at your code and make an assessment of how "close" you were to being correct. I've no idea how to do that fairly or efficiently.
    - I don't have the time to do this sort of partial credit grading and I won't ask the TAs to spend their time looking through and deciphering your code.
    - To my knowledge, there is no agreed upon metric for determining how close code is to being "correct."

**Bottom line**: write *good* code that fulfills the specifications (i.e., passes all the tests) and you'll get all the points.

### 5.1.5 Grading Appeals Policy

Sometimes you do your very best and something doesn't go right. You forgot a "\n" character in a printed statement, etc. In these minor cases, where you were *very* close, please write me an "appeals" email describing the scenario, submitting your new code, and I will, at my discretion, re-grade the assignment with a 10% penalty. All appeals must be filed within a week of grades being posted.

### 5.1.6 Grade Assignment

Grades will then be assigned using the following table:

| Grade | Percent |
|---|---|
| A+ | [98-10] |
| A | [93-98) |
| A- | [90-93) |
| B+ | [88-90) |
| B | [83-88) |
| B- | [80-83) |
| C+ | [78-80) |
| C | [73-78) |
| C- | [70-73) |
| D+ | [68-70) |
| D | [63-68) |
| D- | [60-63) |
| F | [0-60) |

## 5.2 Class Policies

1. Attendance is not mandatory, even at live sessions, but you are responsible for anything that transpires during class or a live

conferencing session. **Please** be on time to class!! If you're consistently late and/or disruptive in class I reserve the right to dock you points on an assignment.

2. Put your name, course, and section on everything you turn in. Sometimes we print things out for grading. Also, this is generally just good practice.

3. Exchange of ideas and techniques is **highly** encouraged but **your work must be your own**. If someone helps you, please give them credit in your code (even if it's the TA). Myself and the TAs will follow up with suspicions of academic dishonesty in accordance with department and university policy (see below).

4. Students have one week from time of grade posting to challenge a grade.

5. If you have *technical content* questions please follow this order of operations for getting help:

    a. Look on Piazza and check for similar questions.
    b. Post your question on Piazza – then everyone benefits from the answer!
    c. Check the internet for similar questions or concepts.
    d. Attend the TAs' office hours.
    e. Email the TA asking your question or requesting a Zoom session.
    f. Email me asking your question or requesting a Zoom session.

6. If you have *personal questions* about logistics or similar please skip straight to contacting me by email explaining the situation and your question.

7. Please use professional email communication. A saluation, description, request, closing, and concise subject line are appropriate for professional communication.

8. I'm particularly concerned with the mental and emotional distress our current circumstances could have on each of us. Don't hesitate to reach

out and let me know how things are going. Large doses of patience are in order during the global pandemic.

## 5.3 CSE Policies

1. The CSE Department has an anonymous contact form that you may use to voice your concerns about any problems in the course or department if you do not wish to be identified.

2. CSE Department policy dictates that students in CSE courses are expected to regularly check their email so they do not miss important announcements.

3. Consider the Student Resource Center in Avery 12 if myself or the TAs are not available, or you otherwise need help.

4. All homework assignments, quizzes, exams, etc. must be your own work. No direct collaboration with fellow students, past or current, is allowed unless otherwise stated. The Computer Science & Engineering department has an Academic Integrity Policy. All students enrolled in any computer science course are bound by this policy. You are expected to read, understand, and follow this policy. Violations will be dealt with on a case by case basis and may result in a failing assignment or a failing grade for the course itself. The UNL College of Engineering also has an academic integrity policy you should read and understand.

## 5.4 UNL Policies

1. The University strives to make all learning experiences as accessible as possible. If you anticipate or experience barriers based on your disability (including mental health, chronic or temporary medical conditions), please let me know immediately so that we can discuss options privately. To establish reasonable accommodations, I may request that you register with Services for Students with Disabilities (SSD). If you are eligible for services and register with their office,

make arrangements with me as soon as possible to discuss your accommodations so they can be implemented in a timely manner. SSD contact information: 232 Canfield Admin. Bldg.; 402-472-3787.

2. Academic honesty is essential to the existence and integrity of an academic institution. The responsibility for maintaining that integrity is shared by all members of the academic community. The University's Student Code of Conduct addresses academic dishonesty. Students who commit acts of academic dishonesty are subject to disciplinary action and are granted due process and the right to appeal any decision.

3. UNL offers a variety of options to students to aid them in dealing with stress and adversity. Counseling and Psychological Services (CAPS) is a multidisciplinary team of psychologists and counselors that works collaboratively with Nebraska students to help them explore their feelings and thoughts and learn helpful ways to improve their mental, psychological and emotional well-being when issues arise. CAPS can be reached by calling 402-472-7450. Big Red Resilience & Well-Being provides one-on-one well-being coaching to any student who wants to enhance their well-being. Trained well-being coaches help students create and be grateful for positive experiences, practice resilience and self-compassion, and find support as they need it. BRRWB can be reached by calling 402-472-8770.