CSCE 231: Computer Systems Engineering

Fall 2021

Schedule

Lecture

- Section 150: Monday, Wednesday, Friday 12:30pm-1:20pm in Brace Labs 206

Recitation

- Section 153: Monday 3:30pm-4:20pm in Avery 19
- Section 151: Tuesday 3:30pm-4:20pm in Avery 106
- Section 152: Wednesday 8:30pm-9:20pm in Avery 106

Laboratory

- Section 165: Monday 4:30pm-6:20pm in Avery 21
- Section 161: Tuesday 4:30pm-6:20pm in Avery 21
- Section 162: Tuesday 6:30pm-8:20pm in Avery 21
- Section 163: Wednesday 9:30am-11:20am in Avery 21
- Section 164: Wednesday 1:30pm-3:20pm in Avery 21

Instructor

Dr. Christopher Bohn
Avery 28C
bohn@unl.edu
Office Hours: Tu/Th 9:00-10:00, F 2:00-3:00

TAs

Name	Office Hours	email
Fahmida Afrin	Th 11:30-12:30, F 3:30-4:30 (SRC)	fafrin2@huskers.unl.edu
Sehee Sun	M 1:30-2:30, Th 3:30-4:30 (SRC)	ssun12@huskers.unl.edu
Dillon Howell	W/Th 5:00-6:00 (Zoom)	dillon.howell@huskers.unl.edu
Dalton Hulett	M 10:00-12:00 (SRC)	dhulett3@huskers.unl.edu
Nathan Kolabas	M 9:00-11:00 (Zoom)	nkolbas2@huskers.unl.edu
Fateh Sandhu	Tu/Th 1:00-2:00 (Zoom)	fatehkaran@huskers.unl.edu

TA office hours are in the Student Resource Center (SRC) in Avery 12 or over Zoom (https://go.unl.edu/csce231OfficeHours), as specified.

Catalog Description

Computer Systems Engineering (4 cr) Lec 3, rct 1, lab 2. Introduction to organization, structure, and applications of computer systems. Boolean Logic, Digital Arithmetic, Processor Organization, C Programming, Machine Language Programming, Input/Output, Memory Organization and Management, Building Embedded System Application.

Prerequisites

Grade of "P" or "C" or better in CSCE 235, CSCE 235H, or RAIK 184H.

Prerequisites by Topic

- *Mastery of* mathematical problem solving skills, mathematical maturity and competence to the level of introductory calculus
- *Familiarity with* programming, logarithms, integer and floating point numbers
- *Exposure to* recursion, operating systems services

Course Objectives

- *Mastery of* Boolean algebra, logic equations, binary numbers (including negatives), fixed-point binary arithmetic, hexadecimal notation, exponential numbers
- *Mastery of* C programming, accessing I/O devices, processor organization, stacks, subroutines, instruction set architecture (with hands- on experience in embedded system design and implementation)
- *Familiarity with* logic gates and diagrams, floating point arithmetic, memory devices and hierarchies, micro-operations & micro-architectures
- *Familiarity with* assembler language principles, flow of control, assemblers, linkers, loaders, the syntax & semantics of short assembler programs
- *Familiarity with* memory errors such as buffer overflows and segmentation faults, security vulnerabilities

Course Goals

How does a modern digital computer interpret programs written in a high-level programming language into the streams of electronic signals that actually do the thinking, the arithmetic, the keeping track of data, the sequencing of actions, and the input/output? What is the hardware-software interface and how can we use it to build modern computer systems and develop software? What system design considerations go into defining and designing it? In this course you will gain an understanding of the interface by first writing programs in a low-level assembly language that is close to the machine language,

developing the logic design of basic processor components, and then designing and implementing an embedded system. In the process you will gain practical experience with computer-aided design tools and C programming environments. The course comprises of lectures, recitation, and labs. The recitation sessions are used to prepare students to tackle the implementation and experimentation aspects of concepts discussed during lectures. The labs provide time for students to implement and experiment with these topics.

Topics Covered

1. *System Level Organization*: CPU, memory systems (main memory, cache, virtual memory), storage technologies, I/O devices & processes
2. *Arithmetic and Logic Level*
   (a) Boolean algebra, truth tables, logic equations, logic gates, combinatorial logic, sequential logic & memory devices, clocks, waveform diagrams
   (b) Signed & unsigned binary numbers, floating-point numbers, BCD, hexadecimal, ASCII
   (c) Signed & unsigned fixed-point arithmetic, floating-point arithmetic, ALU structures
3. *Micro-Architecture Level*: data paths & components, micro- operations, memory interfacing, the fetch/execute cycle
4. *Instruction Set Architecture Level*: instruction types & formats, opcodes, operands, immediate values, addressing modes, flow of control, branching & procedure calls
5. *Assembler Language level*: syntax, directives vs. instructions, assemblers, linkers, loaders, semantics of simple programs, stack management, procedure calls
6. *High-Level Programming Level*: C programming syntax, pointer arithmetic, I/O access, polling & interrupt handling, dynamic memory management, memory errors, security vulnerabilities

Lecture, Recitation, and Laboratory

While the lecture will occasionally address practical matters, the principal purpose of the lectures is to provide you with a theoretical understanding of computer systems engineering. Through the individual labs, the group lab, and the group project, you will learn to apply this theory to practice. The recitation will help you transition from the theory of the lectures to the practice of the labs.
It is a common misunderstanding that the purpose of this course is to program in C. That is not the case. A course such as this requires working with computer systems much closer to the hardware level than most software developers typically do. Few high-level languages afford access to the underlying computer system, and of those that do, C is by far the most commonly-used.

Textbooks

- *Programming at the Hardware-Software Interface*, 1st ed., Christopher A. Bohn
- Other useful references available in the "Pages" section of the course's Canvas site, including *The C Programming Language*, 2nd ed., Brian W. Kernigan & Dennis M. Ritche ("K&R")

Grading

Your grade will be calculated based on your performance in the following areas:

- *Homework* (20%) Homework, in the form of chapter quizzes integrated into the textbook, will help cement the concepts from the lecture and the textbook. Doing so will pay off in the other components of the course grade. Because group study is an effective learning technique, I encourage you to work with your peers to discuss the homework, but **be sure to write your own solutions and indicate whom you worked with**. Except as noted below, chapter quizzes will be due one week after we have finished covering the relevant material in lecture.
- *Labs* (35%) The labs will provide hands-on practice to provide better understanding of what you've learned in the lectures. Some labs will be programmed in C; others entail assembly programming. Unless explicitly stated otherwise, lab assignments are to be accomplished as individual effort. I encourage you to discuss the concepts in the assignment with your peers; however, **you must arrive at your own solution**. Except as noted below, you will have one or two weeks to complete each lab (this will be specified in the lab assignment), starting at the beginning of your lab section the week the lab starts and due *before the start of your lab section* in the week the lab ends.

  You will have a Hardware Pre-Lab to complete before the I/O labs. You may begin it at any time; it is due before the start of your lab section in Week 11.

  You will have two hours to complete the Memory Measurement Lab, starting at the beginning of your lab section in Week 15 and due at the end of your lab section.
- *Group Project* (10%) The group project will involve the design, implementation, and testing of an embedded system. You and one partner will be tasked to complete several functions before integrating them into your final design. The project provides an excellent opportunity to put into practice what you'll learn in this course. The group project will be done in teams of two; we'll allow one team of three in any lab sections with an odd number of students.
- *Midterm Exams* (20%) We will have two midterm exams, that you will take in the DLC. They are scheduled for **September 28-October 1** and **November 2-5**; if there are changes, I'll announce them at least a week in advance. *Unless I explicitly specify otherwise, the exams will be closed book/closed note/closed neighbor; and programmable calculators, laptops, tablets, and smartphones will be disallowed.* You may have one $8\frac{1}{2}$" $\times$ 11" sheet of notes, double-sided, printed or hand- written. The Respondus Lockdown browser has a built-in calculator that you may use to perform arithmetic.

- *Final Exam* (15%) This will be a comprehensive exam held in the DLC, scheduled for **Decenber 11-17**. As with the midterm exams, unless I explicitly specify otherwise, the exams will be closed book/closed note/closed neighbor; no programmable calculator/laptop/tablet/smartphone; one sheet of notes; you may use the calculator built into the Respondus Lockdown browser.

- Final grades will be assigned based on the traditional cutoff percentages.

Grade	A+	A	A-	B+	B	B-	C+	C	C-	D	F
Score	97	93	90	87	83	80	77	73	67	60	<60

  - *Rounding of final grades* I am not in the habit of *giving* grades, but I am sympathetic to students who put in effort. In lieu of rounding, after calculating each student's score and adding extra credit points, I will make one final adjustment to scores near a letter-grade cutoff:

$$adjustment = \begin{cases} 0.50 & \text{if all assignments completed} \\ 0.015 \times number\_of\_assignments\_completed & \text{otherwise} \end{cases}$$

Notional Sequence

Topics		Chapter
Week 1: August 23-27 (M-F)		
lecture	Intro, Prerequisite review, Digital data	1, 2
recitation	Getting started with C / C refresher	
lab	Poker Lab	
Week 2: August 30-September 3 (M-F)		
lecture	Digital data, Integer Arithmetic	2, 3
recitation	ASCII table, number-base conversion	
lab	Broken Keyboard Lab	
Week 3: September 6-10 (M-F)		
break	No lecture, recitation, or lab September 6	
lecture	Integer, Floating Point Arithmetic	3, 4
recitation	Integer arithmetic	
lab	Broken Keyboard Lab, continued	
Week 4: September 13-17 (M-F)		
lecture	Floating Point Arithmetic	4
recitation	Integer arithmetic; Fractional number representation	
lab	Integer Lab	
Week 5: September 20-24 (M-F)		
lecture	Simple Assembly Language	5
recitation	Fixed- & floating-point conversion	
lab	TBD	
Week 6: September 27-October 1 (M-F)		
	Exam Review September 27	
lecture	Simple & Structured Assembly Language	5, 6
recitation	Addressing modes	
lab	Addressing Lab	

Topics		Chapter
	Exam 1 (weeks 1-5): Sep 28-Oct 1 at the DLC	
Week 7: October 4-8 (M-F)		
lecture	Structured Assembly Language	6
recitation	Applying "Compiler cookbook" recipes	
lab	Bomb Lab	
Week 8: October 11-15 (M-F)		
lecture	Assembly Language, Buffer Overflow	6, first half of ch11
recitation	Applying "Compiler cookbook" recipes	
lab	Bomb Lab, continued	
Fall Break: No lecture, recitation, or lab October 18-19		
Week 9: October 20-26 (W-Tu)		
lecture	Buffer Overflow, Concurrency	7, first half of ch11
recitation	Buffer overflow vulnerabilities/prevention	
lab	Attack Lab	
Week 10: October 27-November 2 (W-Tu)		
	Exam Review November 1	
lecture	Simple Processor Design	first half of ch8
recitation	Interleavings and Synchronizing	
lab	TBD	
	Exam 2 (weeks 6-10): Nov 2-5 at the DLC	
Week 11: November 3-9 (W-Tu)		
lecture	System I/O, Exceptional Control Flow	9
recitation	TBD	
lab	Polling-Based I/O Lab	
Week 12: November 10-16 (W-Tu)		
lecture	Advanced Processor Design	second half of ch8
recitation	Polling vs interrupts discussion	
lab	Interrupt-Driven I/O lab	
Week 13: November 17-23 (W-Tu)		
lecture	Advanced Processor Design, Memory	10, second half of ch8
recitation	Practice with Pipelining	
lab	Group Project	
Thanksgiving Break: No lecture, recitation, or lab November 24-26		
Week 14: November 29-December 3 (M-F)		
lecture	Memory, Tying it all together	10, 12, second half of ch11
recitation	Working with Cache	
lab	Group Project, continued	
Week 15: December 6-10 (M-F)		
	Exam Review December 10	
lecture	TBD	
recitation	TBD	
lab	Memory Measurement (due at end of lab)	
Final Exam: December 11-17 in DLC		

Policies and Resources

- *Attendance*: We will cover some material in class that is not covered in the text; your assignments and test questions may be based on that material. In particular, many of the details necessary to complete assignments will be presented in class. It is to your benefit to attend class. Because late arrivals are distracting, I also ask that you arrive to class on time.

- *Class Preparation and Participation*: I expect you to be prepared for class, participate in discussions, and ask and answer questions.

- *Assignments*: All work must be completed when due, and all quizzes and exams must be taken when given. **No make-up quizzes or exams will be given, and with rare exception no late homework or labs will be accepted.** If you have a valid excuse (e.g., illness or injury of yourself or your dependent, an academic conference, a job interview, military obligation, jury duty, bereavement, religious observances, participation in university-sponsored intercollegiate athletic events, etc.), the score for a missed quiz or exam score will be replaced by the average of the remaining quizzes or exams, respectively. Similarly, if you have a valid excuse then at my option you will either be given an extension on a homework or lab assignment or will have the missed score replaced by the average of the remaining assignments in the same category.

  - If you wish to challenge the grade you received on an assignment, you must demonstrate that the answer you provided is correct; *i.e.*, that your grade is erroneous. We will not adjust your grade if your answer is too illegible for us to read. You will have two class-days in which to do so. We will not adjust grades for an assignment after two days past the day on which it was returned to you.
  - I will, at a minimum, honor the guidance for excused absences in the UNL Faculty Senate's Class Attendance Policy. See https://www.unl.edu/facultysenate/classattendancepolicy%20%20April%203%2C%202018.pdf
    * Notify me of planned absences in advance, and preferably early. I will not excuse missed assignments due to planned absences if you notify me after-the-fact.
    * Notify me of unplanned absences when you are able. In general, I will not require proof of the excuse; however, I will require proof of the excuse for missed exams, and I may require proof of the excuse if absences are extended or common.
    * If missed assignments due to excused absences become frequent enough that I cannot gauge how well a student has met the course objectives, then at my discretion I may award an "Incomplete" for the course and require the student to complete the assignments before receiving a letter grade.
  - The DLC will provide you with a link to sign up for each exam two weeks before the exam window opens. Before the exam window opens, if you attempt to sign up for an exam timeslot and no timeslots are available, I will make sure you have the opportunity to take the exam. If you wait until after the exam window has opened to sign up for an exam, I make no guarantees about whether you'll be able to take the exam.

- *Services for Students with Disabilities*: 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.

- *Counseling and Psychological Services* UNL offers a variety of options to students to aid them in dealing with stress and adversity. Counseling and Psychological Services (CAPS, https://caps.unl.edu/) 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 (https://resilience.unl.edu/big-red-resilience-well-being) provides fun events, innovative education, and dynamic services to help students understand emotions, manage stress, build strength, connect with others, develop grit and navigate transitions.

- *Diversity and Inclusion* The University of Nebraska-Lincoln does not discriminate on the basis of race, ethnicity, color, national origin, sex (including pregnancy), religion, age, disability, sexual orientation, gender identity, genetic information, veteran status, marital status, and/ or political affiliation.

- *Student Resource Center*: The CSE Department maintains a Student Resource Center in Avery Hall 12. See http://cse.unl.edu/src.

- *Check Your Email* The CSE Department's policy is that all students in CSE courses should regularly check their email so they don't miss important announcements.

  - We will make announcements through Piazza; make sure that Piazza is configured to forward these announcements to your preferred the email address.
  - When email is appropriate, we will contact you through your huskers.unl.edu email address and/or the email address associated with your Canvas account.
  - I strongly encourage you to check your huskers.unl.edu email daily. Beyond any emails that we may send, the university will send any announcements it makes to your huskers.unl.edu email.

- *Departmental Contact Form*: The CSE Department has an Anonymous Online 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. See http://cse.unl.edu/contact-form.

- *Academic Integrity Policy*: 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. See https://stuafs.unl.edu/DeanofStudents/Student%20Code%20of%20Conduct%20May%20Rev%202014%20a.pdf.

All homework assignments, exams, etc. must be your own work. The Computer Science and Engineering department has an Academic Integrity Policy. All students enrolled in any CSE 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. See http://cse.unl.edu/ugrad/resources/academic_integrity.php.

The Department requires me to report every offense to the Chair for further consideration. The key to avoiding cheating is to be totally open and transparent about any and all collaborations, noting that appropriate teamwork and collaboration will be highly encouraged. Here is some elaboration on the examples listed on the department's academic integrity webpage:

- I encourage discussions of *what* and *why*, but discussions of specific solutions or implementations are prohibited.
- Being in possession of a worked or partially-worked solution to an assignment (whether from a fellow student, from the internet, or from another source) before you have completed the assignment is unauthorized collaboration on the assignment.
- If another student has a copy of your worked solution to an assignment before s/he has completed the assignment, I will assume that you facilitated their cheating unless it can be demonstrated that they obtained the copy despite your reasonable precautions to prevent them from doing so. Students who share a computer should protect their files either by using separate accounts or by placing their coursework in password-protected folders.
- If we detect academic misconduct on group assignments, I will hold the full group responsible unless there is compelling evidence that only a proper subset of the group committed the misconduct and that the remainder of the group was unaware of the misconduct.

We will use software, including but not limited to MOSS, to help us detect academic integrity violations; however, we will also apply human judgement. We will retain your assignment submissions to be compared with future students' submissions.

- *Student Concerns and Feedback*: Your experience with remote learning in this course is important to me. If you have questions, concerns, or positive feedback, please contact me at bohn@unl.edu or 472-1803. If I am unable to respond, or you feel I've not adequately addressed your concerns, you can contact Dr. Marilyn Wolfe at mwolf@unl.edu or 472-2401. If your concern is still not resolved, please contact Dean Lance Pérez at lcperez@unl.edu or 472-5259.

Last updated: 19:41-2021/09/12