CS213: Introduction to Computer Systems

Syllabus - Winter 2025

Course Staff

Instructor

Branden Ghena	branden@northwestern.edu

TAs

Connor Selna	connorselna2021@u.northwestern.edu
Jess Mingyoung	mingyoungjeng@u.northwestern.edu

PMs

Christian Lee

Daniel Lee

Ethan Foong

Ethan Hamilton

Haresh Wedanayake

Jay Park

Jerry Han

Leo Zhang

Max Smith

Milind Maiti

Natalie Hill

Nathan Wang

Ryan Wong

Sean Rhee

Overview

This course introduces the lower levels of computer systems. We will peel back the abstractions you have grown accustomed to from CS111 and CS211, and look “under the hood”, so to speak, to see how they work. In doing so, we will explore the hierarchy of abstractions and implementations that comprise contemporary computer systems.

The goals for doing so are two-fold. First, this will provide you with an overview of “what’s out there” in the fascinating world of computer systems, which you can flesh out in our upper-level computer systems courses. Second, and more important in the long term, your familiarity with the lower levels of systems will allow you to investigate and resolve issues that arise when the upper levels break down due to bugs or performance issues.

This is a learn-by-doing kind of class. You will write pieces of code, compile them, debug them, disassemble them, measure their performance, optimize them, etc.

The specific computer architecture we will focus on in this class is the 64-bit Intel/AMD x86 architecture, which is used in virtually all supercomputers, clouds, clusters, servers, and desktops computers today[1]. The specific operating system we will use is Linux, which is used in most supercomputer, cloud, cluster, and server environments, and is the operating system of Android smartphones and ChromeBooks. The specific programming toolchain we will use is GCC (and GDB), which is an extremely widely used core toolchain on pretty much all platforms, except Windows. The ideas and concepts embodied in this architecture, operating system, and programming toolchain are commonly found in others.

Textbook

Computer Systems: A Programmer’s Perspective, ***Third Edition,***
Randal E. Bryant and David R. O’Hallaron*,*
Prentice Hall, 2015, (ISBN-13: 978-0134092669, ISBN-10**:** 013409266X)

- Details at: http://csapp.cs.cmu.edu/3e/students.html

- **Make sure you have the third edition of the book**. This edition is the first to focus on the 64-bit operation of the machine, which we will make extensive use of in this course.

- If you buy a non-U.S. version, acquire a pdf through some means, etc., please be aware that these can have differences from the U.S. version.

- This class may be eligible for the Books for Cats program; the textbook is available via library reserves for eligible students.

[1] The 64-bit x86 architecture is also called “x86-64” and just “x64”. We may also touch on the ARM architecture used heavily in smartphones and the relatively new RISC-V architecture. If this doesn’t make sense to you yet, don’t worry about it, this course will teach you.

Location and Time

Lecture time: 2:00 - 3:20 PM Central, Tuesdays and Thursdays
Location: Tech Institute, Room LR3

We **strongly encourage** keeping up with the course lectures. Much of the difficulty in CS213 comes from the new concepts, which we take care to introduce in lecture. We will attempt to record all lecture sessions so that you can later review them if you want, but the expectation is that students will attend class.

Pre-requisites

CS211 or equivalent: experience with C or C++ and Unix environments.

CS213 is a required core course in the Computer Science curriculum in both McCormick and Weinberg. It is also a required course for CS minors in both schools. CS213 can also be taken for credit within the Computer Engineering curriculum. 300-level computer systems courses generally have CS213 as a prerequisite.

Communication

All course materials will be posted to Canvas including grades, lecture materials, and class recordings. Piazza will be used for course discussions and questions. **All questions should go to Piazza rather than to email.** We will enroll you in Piazza.

Office hours will also be available, with the regular schedule available on Canvas. Office hour appointments can also be made with the instructors or TA by Piazza post to “Individual Student(s) / Instructor(s) → Instructors”.

Class Structure

Schedule

The course schedule is available on the Canvas homepage for the course. Be aware that it is subject to change, although warnings will be given to students for any major changes.

Homework

There will be four graded homework assignments. Homework assignments are practice problems worked out on paper to help you understand course topics. They must be completed **individually** and are important for preparing for exams.

Labs

There will be four lab assignments. Lab assignments are longer hands-on activities performed on a computer. Some involve programming, while others involve interacting with programs you are given. Their goal is to make you apply the concepts you have learned and understand them more deeply. The labs are the largest portion of work in the class. Labs can be done individually or in groups of two.

Midterm Exams

There will be two midterm exams. The exams will be given in person. One will take place near the middle of the class, with results returned to students before the drop deadline. The other will take place during the class exam slot of exam week. Midterm exams are not cumulative and focus on their own part of the course. However, the nature of the course is that the material in the second half builds on some knowledge from the first half.

Exams are taken individually. Students will be allowed the use of some amount of notes, which the instructor will provide guidance on in advance.

Minimum Midterm Average Rule: CS213 requires a minimum average across the two midterms of 60%. The rationale here is that midterm exams are one of our few chances to gauge individual understanding of a student. Having a minimum bar ensures that you have the background to succeed individually in future Systems courses and that we aren't throwing you into future classes that you're not ready for. Further, this rule allows for a minimum bar without unduly increasing the weight of exams and their impact on your final letter grade.

Students not achieving a 60% average or better across both midterms will fail the course. However, to ensure that student improvement is rewarded, there is an exception to this rule: a student who achieves 85% or better on the second midterm will have this rule waived. Also, in the case of overly challenging exams, the instructor may apply a grade boost to all students across-the-board, which will apply before considering average score. Students will never be penalized for overly easy exams.

In summary:

- >= 60% average across both midterms: account for grades normally
- >= 85% on the second midterm: account for grades normally
- < 60% average across both midterms AND < 85% on the second midterm: fails CS213

Grades

Percentage grades will be converted to letter grades using the standard letter grade system (93% A, 90% A-, 87% B+, etc.). However, these grade bins may be moved at the instructors' discretion for the advantage of students. Note that the percent grade displayed by Canvas is not always accurate and may not take late penalties or slip days into account, as described below.

Each category of assignment has a total value, which is divided evenly between assignments.

Category	Count	Total Value
Homework	4	12%
Labs	4	48%
Midterm Exam 1	1	20%
Midterm Exam 2	1	20%

Late Policy

Midterm exams may not be taken late without prior coordination by the instructor.

Homework and labs may be submitted late with a penalty of a 20% reduction in maximum points per day late with a minimum of zero points. For example, a homework assignment submitted two days late has a maximum score of 60%. Lateness is rounded up to the whole day, so an assignment that is five minutes late has the same penalty as an assignment 23 hours late.

Extra credit, when available, is applied after late penalties. So a student scoring 110% on an assignment that is one day late would receive 90% instead (maximum of 80% plus 10% extra credit), five days late would receive 10%, and six or more days late would receive 0%.

Slip Days

To help you more flexibly manage deadlines, we will give you **three slip days**, which allow you to submit a homework or lab assignment late without penalty. Slip days are used in units of whole days, meaning a homework or lab submitted five minutes late consumes an entire slip day. Slip days may only be applied to homework or lab assignments, not exams.

You do not need to notify staff that you are using a slip day. We will track the total number of late days for your submissions and automatically apply slip days to optimize their usage. Slip days will not be assessed against homework or lab assignments you did not submit. No extra credit is awarded for avoiding the use of slip days. However, it is in your best interest to avoid turning in homework or lab assignments late, as the next assignment is often released slightly afterwards.

Slip days are applied individually, so for partner assignments be careful to communicate about plans to use slip days. It is possible for an assignment submitted one day late to have no penalty for one student (due to spending a slip day) and a one day late penalty for their partner with no slip days remaining.

Example slip day usage:

- Use two slip days to receive no penalty on a homework submitted two days late.
- Use two slip days to receive no penalty on two separate lab assignments each submitted one day late.
- Use three slip days to receive just a one-day late penalty on a homework submitted four days late.

Slip days are meant to automatically handle minor issues. If you are having a major issue, please contact the instructor as soon as possible, and we will work together on a solution. Particularly for issues outside of the student's control, such as major injury or sickness, deadlines can be shifted without penalty if you contact the instructor.

Academic Integrity

Collaboration is a really good thing, and we encourage it. On the other hand, cheating is a very serious offense, which carries serious consequences. It is OK to meet with colleagues, form study groups, discuss assignments with them, compare alternative approaches, go over examples from textbooks or other sources. **But it is never ok to share code or homework solutions, or even to see each other's code or solutions.**

What you turn in must be your own work. Copying (or even studying) code, solution sets, etc., from anywhere (e.g., other people, web, GitHub, AI) is strictly prohibited. Be aware that we use a number of tools to detect and discover integrity violations. If you discuss your work with other people, please acknowledge them by listing their names in your submission. It is also forbidden to share, post, or otherwise publicize course materials. This includes (but is not limited to) homeworks, exams, solutions, or your own submissions. This extends even after the quarter ends; course material remains private information which you may not share or reproduce.

It is the responsibility of every student in this class to be familiar with and to adhere to the Academic Integrity Policies of Northwestern University and the McCormick School of Engineering. Any suspicion of violation of these policies will be reported immediately to the Associate Dean for Undergraduate Studies. If you are in doubt whether your actions constitute a violation of the above policies, ask the instructor (before doing what you are unsure about).

Sickness and Common Sense

Generally, if you are sick do not attend class. Instead contact your instructor as soon as possible and we'll figure out a way to handle the situation. We expect all students to use their discretion and make good choices for the community.

Accessibility

We believe in providing reasonable accommodations that allow for full access to learning for all. Please contact me for anything that might have an impact on your participation in this course (documented disability, language challenges, absences for religious observations, etc.).

Class Recordings

This class or portions of this class will be recorded by the instructor for educational purpose and available to the class during the quarter. Your instructor will communicate how you can access the recordings. Portions of the course that contain images, questions or commentary/discussion by students will be edited out of any recordings that are saved beyond the current term.

Diversity and Inclusion

We consider this classroom to be a place where you will be treated with respect, and we welcome individuals of all ages, backgrounds, beliefs, ethnicities, genders, gender identities, gender expressions, national origins, religious affiliations, sexual orientations, ability—and other visible and nonvisible differences. All members of this class are expected to contribute to a respectful, welcoming, and inclusive environment for every other member of the class.

Northwestern is committed to fostering an academic community respectful and welcoming of persons from all backgrounds. To that end, the policy on academic accommodations for religious holidays stipulates that students will not be penalized for class absences to observe religious holidays. If you will observe a religious holiday during a class meeting, scheduled exam, or assignment deadline, please let me know as soon as possible, preferably within the first two week of class. If exams or assignment deadlines on the syllabus fall on religious holidays you observe, please reach out so that we can discuss that coursework.

This course will also include a mix of undergraduates and graduate students with differing backgrounds in programming and computer systems. Do not feel discouraged by this. Each student will bring a different aspect of their knowledge to discussions, and we'll all be contributing towards increasing each other's understanding.

Northwestern University Syllabus Standards

This course follows the Northwestern University Syllabus Standards. Students are responsible for familiarizing themselves with this information.