CPS 210 – Introduction to Computer Systems
Spring 2023

(Last Modified: February 25, 2023)

1 General

Course

Time	Wed/Fri 3:30 - 4:45pm
Location	LSRC B101

Instructors

Name	Matthew Lentz
Email	mlentz@cs.duke.edu
Website	https://www.cs.duke.edu/~mlentz
Office Hours	TBD in LSRC D314 (or by appointment)
Name	Jeff Chase
Email	chase@cs.duke.edu
Website	https://www.cs.duke.edu/~chase
Office Hours	TBD in LSRC D306 (or by appointment)
Name	Yesenia Velasco
Email	yvelasco@cs.duke.edu
Website	https://yvelasco.github.io/
Office Hours	TBD (or by appointment)

Graduate TAs

Name	Luka Duranovic
Email	luka.duranovic@duke.edu
Office Hours	TBD
Name	Yongkang Li
Email	yongkang.li@duke.edu
Office Hours	TBD

Undergraduate TAs
(See Sakai for complete office hours)

Name	Aaric Han (Lead), Alyssa Zhao (Lead), David Bian, Khushmeet Chandi, Piper Epstein, Andrew Fate-Bolognone, Katherine Gallagher, Jiayi Hao, John Hession, Kathryn Kotler, Ajay Krishnamurthy, Ayush Jain, Owen Jennings, Nadeska Montalvan, Zaid Muqsit, Marisol Mata Nevarez, Michael Ruiz, Elizabeth Skinner, David Turner, Zijie Yang, Albert Yuan,

Resources

Website	https://courses.cs.duke.edu/spring23/compsci210d/
Sakai	https://sakai.duke.edu/portal/site/73852698-9364-4b86-ac5f-567d08373d92
Gradescope	https://www.gradescope.com/courses/475311
Ed	https://edstem.org/us/courses/32051/discussion/
GitLab	https://coursework.cs.duke.edu/compsci-210-spring-2023

2 Overview

CompSci 210 is an undergraduate introduction to computer systems software ”close to the metal” on real machines. It provides a programmer’s view of how computer systems execute programs and store information, with exercises using the C programming languge. It examines key computational abstractions behind high-level programming languages: number and data representations, instructions, memory hierarchy, programs and processes, and basics of multi-core/concurrency. The Big Ideas in this class are broadly applicable across the many subfields of computer systems and computer science, with specific instances demonstrated through labs and projects.

3 Expectations

3.1 Preconditions

The prerequisite for this course is that you have completed CompSci 201 (or equivalent). Therefore, we expect that you have experience programming in a high-level language (e.g., Java).

3.2 Postconditions

After completing this course, we expect you to be able to:

1. **(Overall) Understand how programs, written in high-level programming languages, run on real machines and operating systems**
2. Understand the key programming language concepts within the C language (e.g., types, operators, control flow) and the surrounding ecosystem (e.g., compiler, linker, debugger)
3. Understand the instruction set architecture (ISA) abstraction for CPU hardware and how C programs are mapped to machine-level instructions
4. Understand the role of the operating system and some of the key abstractions provided to user programs on top of underlying hardware (e.g., processes)
5. Implement non-trivial programs in C that interact with details of the underlying machine architecture (e.g., stack layouts) or interfaces provided by the operating system (e.g., threads, mutexes)
6. Understand various factors that impact performance and resource efficiency as a result of hardware and software running underneath your programs (e.g., caching)

4 Resources

Textbooks

1. **[Required] Computer Systems: A Programmer’s Perspective (CS:APP 3/E)**
   Bryant and O’Hallaron
   3rd Edition
2. **[Recommended] The C Programming Language**
   Kernighan and Ritchie
   2nd Edition

Sakai We will be using Sakai as a general course platform, to report grades, issue quizzes, and to (occasionally) make course announcements.

Gradescope We will be using Gradescope for submitting and automatically grading labs and projects. We will also use it for grading exams.

Ed We will be using Ed to serve as a discussion forum for the course and the primary place for making course announcements.

Assistance Teaching staff are available to assist you via Ed and posted office hours. We want to help you and we encourage you to ask questions and visit office hours. Please be respectful of staff time: start early, seek help early, and do not expect extra time beyond the schedule. We expect you to make a good faith effort to solve problems yourself before seeking help. Keep up with the Ed board and check to see if your question is already answered. If demand is high, we may use a queuing system or other tools to allocate time. In that case, we may ask you to write down the steps you have already taken before receiving help. If a TA refuses to help you, please respect their decision and raise the issue with the instructors if you feel you are treated unfairly.

5 Grading

Your final grade in the course will be determined by the following percentage allocations:

Type	**%**	**Description**
Exams	50	Two midterm exams (15 each) and a final exam (20)
Projects	25	Larger programming assignments that build on labs and lectures
Labs	20	Small programming assignments associated with each lecture
Discussion	5	Participation in weekly discussion sections for labs
Quizzes	0	Assigned readings with self-assessment quizzes

Note that attending lectures is not mandatory; however, we strongly encourge it. You are responsible for all material covered and assignments given out during any class that you miss.

We generally use a 10 point scale for grading in this course (A 90-100; B 80-89; C 70-79; D 60-79; F < 60), with these ranges including + and -. We map the total scores to letter grades according to cutoffs drawn ”by thumb” at the end of the semester. Options for extra credit, such as those as part of the labs, will be considered in the assignment of final grades. Note that there is no ”curve” to assign anyone a weak grade by algorithm or to penalize anyone for someone else’s credit or success.

5.1 Exams

There will be two midterm exams and one final exam. The midterm exams will be held in class, with the dates specified via the online schedule here: https://courses.cs.duke.edu/spring23/compsci210d/schedule.html The final exam is scheduled for Monday May 1st from 9:00am-12:00pm in LSRC B101. All exams are considered to be cumulative.

5.2 Projects

There are six programming projects spread throughout the duration of this course. Each project builds on concepts and skills obtained from lectures, readings, and labs. You may work on projects individually or with at most one other person.

Be sure to carefully read the README file associated with each project and follow directions appropriately. Add a text file (statement.txt) listing the nature and sources of any assistance that you received, as well as anything else you want to tell us. If you choose to work in a team of two for the project, please add a reflection of the collaboration within your team to the file. You can find more information on the projects here: https://courses.cs.duke.edu/spring23/compsci210d/assignments.html.

5.3 Labs

There are short (1-3 hours), individual programming labs associated with each lecture that help: 1) reinforce concepts learned from readings and lectures, and 2) prepare you for the projects. Labs are released after each lecture (Wednesday and Friday) and are due the following Wednesday. The Monday discussion sections will focus on the labs due that week; you should start working on both labs prior to your discussion section. We will drop your three lowest lab scores.

Be sure to carefully read the README file associated with each lab and follow directions appropriately. Add a text file (statement.txt) listing the nature and sources of any assistance that you received, as well as anything else you want to tell us. You can find more information on the labs https://courses.cs.duke.edu/spring23/compsci210d/assignments.html.

5.4 Discussion

Our undergraduate TAs lead the discussion section meetings on Mondays. Attending a discussion section and working on the labs will get you credit; however, if you fully complete all of the required lab material prior to the start of your discussion section, you will automatically receive credit for that discussion section without needing to attend. The TAs may also provide assistance with projects at the meetings as time allows. We will drop up to two missed discussion sections.

5.5 Quizzes

There is one (optional) quiz per lecture. The quizzes are multiple choice and correspond to assigned material (e.g., readings) as listed on the course schedule. The goal for these quizzes is to provide you with a means for self-assessment at various points: 1) after the readings and prior to the lecture, and 2) after the lecture.

5.6 Late Policy

We expect you to turn in your work by the day and time it is due. Note that if a time is not listed, you may assume the deadline is 11:59pm ET on the day listed. However, we have the following late policy for labs and projects:

$[0, 24]$ hours late: 10% penalty
$(24, 48]$ hours late: 20% penalty
$(48, \infty)$ hours late: No Credit

Exceptions to this late policy are allowed only for Dean's excuses and short term illnesses as indicated by submitting the Short-term Illness Notification Form, which you can find here: https://trinity.duke.edu/undergraduate/academic-policies/class-attendance-and-missed-work. Note that you also must fill out the appropriate form on the Resources page.

5.7 Regrading Policy

All regrading requests (labs, projects, quizzes, exams) must be submitted within one week of the graded item being returned/available using the provided form on the Resources page. Requests after one week will be denied.

6 Academic Integrity

We expect everyone to uphold the Duke Community Standard, which you can find here: https://studentaffairs.duke.edu/conduct/about-us/duke-community-standard. In particular, this standard is comprised of:

- I will not lie, cheat, or steal in my academic endeavors
- I will conduct myself honorably in all my endeavors
- I will act if the Standard is compromised

Please ask us if you are unsure which actions may (or may not) violate the community standard as part of this course.

6.1 Collaboration Guidelines

Projects You may complete the projects individually or in groups of two. Within a group, there are no limits to your collaboration. Outside of your groups, you are more than welcome to discuss the projects with your fellow classmates. This includes discussing the project specification, proposed approaches to solving the project, working through high-level designs on a whiteboard, or asking questions on the forums. However, all of you are responsible for developing your own implementations. Do not share code, either snippets or solutions as a whole, with one another; this also extends to the forums, unless there is explicit approval from the instructor or one of the TAs.

You can switch partners for different projects, but you cannot switch during a project (i.e., don't drop someone). You may choose your partner or we can help to match you with someone on a first-come-first-served pairing. Fill out the project pairing form on the Resources page if you'd like us to try to match you to a partner. If you work as a team, you must submit a text file that contains a reflection about the collaboration as part of your solution.

Labs The labs are to be completed individually. As with projects, you can discuss the labs with your fellow classmates. This includes discussing the lab specification, proposed approaches to solving the lab, working through high-level designs on a whiteboard, or asking questions on the forums. However, all of you are responsible for developing your own implementations. Do not share code, either snippets or solutions as a whole, with one another; this also extends to the forums, unless there is explicit approval from the instructor or one of the TAs. Discussion sections are dedicated time for help on labs, so you should use that to your benefit.

7 Students with Disabilities

Duke University is committed to providing equal access to students with documented disabilities. Students with disabilities may contact the Student Disability Access Office (SDAO) to ensure your access to this course and to the program. There you can engage in a confidential conversation about the process for requesting reasonable accommodations both in the classroom and in clinical settings. Students are encouraged to register with the SDAO as soon as they begin the program. Please note that accommodations are not provided retroactively. More information can be found online at `access.duke.edu` or by contacting SDAO at 919-668-1267, `SDAO@duke.edu`.

8 Environment

Let's all work together to stay safe and healthy. Please treat teaching staff and other students with kindness and respect. Keep your remarks on the message board respectful. We will disable anonymous posting if we see any threatening or disruptive posts. Please monitor and comply with Duke mandates for precautions against infection. Let us know if you have concerns we can address regarding your safety or health. Please know that we are concerned for you. We understand that you may be facing negative reactions to stress and pressure, other personal challenges, or just the burdens of managing your life and future. Be mindful of your needs for sleep, exercise, proper food, recreation, social connection, and constructive engagement with your problems. We encourage you to take advantage of Duke resources for wellness and mental health. Ask for help when you need it.

9 Course Evaluations

Please take a moment of your time at the end of the semester to submit a course evaluation. These evaluations are incredibly useful to both us personally as well as to the department as a whole. You can provide your feedback at the following link: `http://duke.evaluationkit.com/`. Note that if you have suggestions for how we improve the course, feel free to reach out at any time.

10 Modifications

We tried to make this syllabus both correct and complete; however, we reserve the right to modify the contents of the syllabus while the course is underway. We will make sure that any modifications are clearly communicated to you with sufficient advance notice.