

[Back to Course Info](#)

# Syllabus

## Office Hours

[Link to Office Hours Schedule](#) Office hours are held in the basement of Siebel. To add yourself to the office hours queue, go to [Queue](#). Make sure to include a good description of your location so the TA on duty can find you, and a meaningful question.

## Course Theme

This course teaches a variety of ways to store collections of data in a computer program and discusses the advantages and disadvantages associated with the different methods. You will learn how to build various data-storage structures, and you'll discover why you might prefer one over another in a particular situation. The combined arts of design, analysis and justification are the substance of the class. The class assumes basic skills programming in C++. The course will not review the language, but the first MP will mostly be a refresher.

## Prerequisites

- **CS 128:** Introduction to Computer Science
  - Programming basics
  - Programming structures (methods/functions)
  - Object-based programming (classes and encapsulation)
  - Recursion
  - Algorithm analysis
  - Linear search and binary search
  - Sorting algorithms
- **ECE 220:** Intro to Computing Systems
  - Programming basics (in C and C++)
  - Programming structures (methods/functions)
  - Pointers
  - Recursion
- **CS 173 or Math 213:** Discrete Mathematics
  - Order of growth ("Big-O Notation")
  - Recurrences
  - Proof by induction
  - Set definitions/terminology/facts
  - Tree definitions/terminology/facts
  - Graph definitions/terminology/facts

See also the [FAQ about prerequisites](#).

## C++ reference

A good C++ reference is a necessity. Here are some suggestions:

- A good reference of the language and standard library is available at <http://en.cppreference.com/>.
- A more tutorial-based approach is available at <http://www.learncpp.com/>. This spends more time teaching, rather than acting solely as a reference.
- If you like physical books, Ira Pohl's [C++ Distilled](#) is a good option. (Don't be put off by its age! In this case it just means it doesn't cost a fortune. yay!).

You are welcome to use any alternative references you prefer.

## Communication

### Email

For all administrative questions except for extensions (see below!) please use the [cs225admin@lists.cs.illinois.edu](mailto:cs225admin@lists.cs.illinois.edu) email to reach the professors. This will go to all the instructors and admins but none of the rest of the staff.

### Discord

We have a course Discord server to help students connect with staff and each other. To join the discord server follow this link [CS225 Discord](#).

We'd like you to use this virtual space to talk about class, work on the MPs/labs, and general bantering/socializing! We are also using it for lecture discussion so please get on it. You can keep using Piazza, office hours, or e-mailing us about anything you need.

Discord allows us to answer questions in real time. Please keep Discord questions brief so we may answer timely. This server isn't a way to get ahold of course staff for personal tutoring. If you have multiple questions or want 1-on-1 help in an assignment, please defer to office hours. **Please send any long form questions, that may include code, debugging, and error logs, to Piazza, so we can invest time in answering properly.** Piazza indexes questions and makes them searchable; this helps other students that may have the same issue!

Finally Do not DM staff with questions about the course on Discord.

## Piazza

While the Discord is great for socializing and quick responses, Piazza is a better tool to get asynchronous help on coding or conceptual problems. You are encouraged to use it to ask questions, request clarifications, express opinions, or give advice.

The Piazza site for this class is: [Piazza](#). You are welcome to sign up, and you can do so directly if you use your [@illinois.edu](#) email address. Doing so exposes you to the possibility of limited promotional material from Piazza, but you WILL be able to maintain your privacy within the system by posting private messages. If you are concerned about privacy, and you do not want to use your [@illinois.edu](#) account, send any [course instructor](#) an alternative account and we'll prompt Piazza to send you an invitation.

We expect that you will be courteous and post only material that is somehow related to cs225 (however slightly). The posts will be lightly moderated.

Piazza will be a great place to post most questions regarding assignments - both the course staff and other students are allowed to answer your questions. **Please try to avoid asking questions that have already been asked and answered. Also, do not post code!!!** Finally, try to avoid emailing the instructor or TAs if your question can be posted to Piazza.

## Grading

All grades for the course will be available via PrairieLearn throughout the semester (and no other platforms).

## Point breakdown

Category	Contribution	Notes
Programming Assignments	300 points	50 points each
Laboratory Assignments	120 points	10 points each
Exams	420 points	70 points each
Final Exam	160 points	

## Usual cutoffs

Points	Minimum Grade
[900, 1000]	A-
[800, 900)	B-
[700, 800)	C-
[600, 700)	D
[0, 600)	F

We might lower these cutoffs; for example, perhaps 670 points will turn out to be enough for a C-; however, we won't raise them. (In recent semesters these cutoffs have not moved significantly from these targets.)

We do not assign letter grades for individual scores. We also do not ever curve individual exam or assignment scores. If an exam or assignment should turn out to be significantly harder than we meant it to be, we would announce a lowering of the expected cutoffs above for the various letter grades, in effect lowering the percentage needed for a grade and curving the grades.

## Extra credit

There is an opportunity for significant extra credit in this course. Points for extra credit work will be assigned after grade cutoffs are determined, so they are a true bonus to your score. This semester the total amount of planned EC is 66 points. (This was raised to 70 points by the inclusion of an optional 13th lab, worth 4 points).

**Warning:** As per our [Academic Integrity](#) policy, a violation will result in loss of all extra credit points.

### MP extra credit via early submission

All MPs except the first `mp_sticker` are broken into two parts. The first part can be submitted early for up to +4 extra credit points. The result of consistent early submission is +20 points toward your final course score.

Partial extra credit is available; if you score an 80% on an early submission, you will get 80% of the extra credit weight extra credit. (eg:  $+8 * 80\% = +6.4$ )

### Problems of the Day (POTDs)

Starting early in the semester and continuing every weekday through most or all of the semester we will give you a small programming problem to solve. These exercises are designed to mimic the environment and scope of coding problems you will see on midterms. They will be distributed and collected via PrairieLearn. Each POTD is worth +1 extra credit point, to a maximum of +30 points.

### Course Feedback Surveys

Your thoughts and opinions about the course are pivotal towards shaping its future development. Throughout the semester there will be opportunities to provide constructive feedback about individual MPs and the class overall. However since these surveys are completely anonymous, points will be awarded collectively to the entire class if participation reaches above a certain threshold.

Each MP survey will be worth +2 points and the mid-semester Informal Early Feedback will be worth +4 for a total of +16 points.

## MPs and Labs

### Machine Problems

There will be 6 machine problems (MPs). They are of increasing difficulty and sophistication, and we consider them to be the meat and cheese of the course.

The MPs you will be doing in this class will serve many purposes:

- They are designed to give you substantial practice with the C++ syntax you learn in class.
- They are an exercise in the software development cycle.
- They are an exercise in attention to detail, as there is not much partial credit given on the MPs.
- They are an exercise learning to use the tools available to you to help you work more efficiently and to check the quality of your work.

You are given approximately two weeks for each MP. Exact MP due dates will be announced in lecture and on the MP specification page.

### Lab Meetings

There will be approximately 12 lab sections during the semester, each of which consists of a small intro and a lab exercise. See the top of this page for meeting times and locations. Lively discussion and collaboration with course staff and fellow students during the 2 hour labs should usually (but not always) result in significant progress toward finishing the exercise.

The purpose of labs is to help improve your programming abilities and reinforce concepts taught in lecture. Exact lab formats may change week-to-week, but a usual lab will consist of about 20 minutes of discussion over the material covered in lecture and about 90 minutes of collaborative coding.

Lab assignments will be released for all students before the first lab on Thursday, and will be due the following Sunday at 11:59pm.

## Machines and Environment

Your MP and lab solutions MUST compile and execute in the course docker container. All grading will be done in a container configured as the one we provide you. It will be run on the PrairieLearn platform.

## Extensions and Late Policies

To accommodate the complexity and stress of your college experience, CS 225 offers limited extensions on assignments. Specifically:

1. Each MP has a built-in extension window where you can submit an assignment up to 24 hours late for up to 93% credit. **Non-MP assignments do not have built-in extensions.**

2. Every student is allowed one free extension on any assignment for any reason. This extension is good for 48 hours after the original deadline for full credit. **There is no further extension window after a 48 hour extension.**

To request this extension, you must fill out the [Extension Request Form](#). Please be aware that you will never receive an email confirmation about your extension. Likewise please be aware that these extensions are added manually and are not going to be immediately resolved, especially if you are requesting them the night of the assignment due date. You are strongly encouraged to request an extension at least 48 hours before the assignment deadline.

## Exam Policies

See the [Exams page](#) for logistics and scheduling information for this semester's exams.

### Retake Exam

At the end of the semester, you will be given the opportunity to retake a single prior exam from earlier in the semester. **Your final grade for that exam will be the direct average of these two exam scores.** This does mean that there is a risk associated with a retake exam – the purpose behind this is to give students who have struggled to display mastery over a single topic an opportunity to learn and demonstrate their growth.

## Academic Integrity

For a more detailed description of our Academic Integrity Policy, please see [Academic Integrity](#). You are responsible for the content there, but as an overview:

- You may **NOT** reference any code outside of that provided in lecture and the textbook, or receive help from ANYONE outside of course staff.
- **In labs and labs alone**, you may freely discuss code implementations with your lab partners.
- **In all other assignments do not share code with other students for any reason!**
- Your turned in work must be your own product, representing your own knowledge. Any form of cheating is unacceptable.
- Code produced by generative AI tools is considered cheating / plagiarism. Using these tools is against class policy.
- Do not use a friend's netid to get around homework timeout restrictions. This is considered cheating (and code sharing).

**Be aware of the accessibility of your code (and keep it private)!** Companies are not interested in 200-level course code but students looking to cheat are! FAIR violations are submitted for all parties with shared code – the stress of proving you did not knowingly share code is not worth the risk.

## Statement on anti-racism and inclusivity

The intent of this section is to raise student and instructor awareness of the ongoing threat of bias and racism and of the need to take personal responsibility in creating an inclusive learning environment.

The Grainger College of Engineering is committed to the creation of an anti-racist, inclusive community that welcomes diversity along a number of dimensions, including, but not limited to, race, ethnicity and national origins, gender and gender identity, sexuality, disability status, class, age, or religious beliefs. The College recognizes that we are learning together in the midst of the Black Lives Matter movement, that Black, Hispanic, and Indigenous voices and contributions have largely either been excluded from, or not recognized in, science and engineering, and that both overt racism and micro-aggressions threaten the well-being of our students and our university community.

The effectiveness of this course is dependent upon each of us to create a safe and encouraging learning environment that allows for the open exchange of ideas while also ensuring equitable opportunities and respect for all of us. Everyone is expected to help establish and maintain an environment where students, staff, and faculty can contribute without fear of personal ridicule, or intolerant or offensive language. If you witness or experience racism, discrimination, micro-aggressions, or other offensive behavior, you are encouraged to bring this to the attention to any course staff you are comfortable sharing with. You can also report these behaviors to the [Campus Belonging Office](#), the [Office of Student Conflict Resolution](#), or [CS CARES](#).

## Statement on CS CARES and CS Values and Code of Conduct

All members of the Illinois Computer Science department - faculty, staff, and students - are expected to adhere to the CS Values and Code of Conduct. The CS CARES Committee is available to serve as a resource to help people who are concerned about or experience a potential violation of the Code. If you experience such issues, please contact the CS CARES Committee. The instructors of this course are also available for issues related to this class.

## Statement on Mental Health

Diminished mental health, including significant stress, mood changes, excessive worry, substance/alcohol abuse, or problems with eating and/or sleeping can interfere with optimal academic performance, social development, and emotional well-being. The University of Illinois offers a variety of confidential services including individual and group counseling, crisis intervention, psychiatric services, and

specialized screenings at no additional cost. If you or someone you know experiences any of the above mental health concerns, it is strongly encouraged to contact or visit any of the University's resources provided below. Getting help is a smart and courageous thing to do – for yourself and for those who care about you.

Counseling Center: 217-333-3704, 610 East John Street Champaign, IL 61820

McKinley Health Center: 217-333-2700, 1109 South Lincoln Avenue, Urbana, Illinois 61801

University wellness center: <https://wellness.illinois.edu/>

## Time Conflict Override

If you need a time conflict override for the lecture of this course you can use the following email

From: "Evans, Carl" <gcevans@illinois.edu>  
Subject: Time Conflict Override  
Date: January 30, 2025 at 1:48:08 PM CST  
To: "Evans, Carl" <gcevans@illinois.edu>

Hello,

I am approving a time conflict override for students with the CS 225 lecture. This is section AL1 and AL2. The lectures are recorded and can be watched later. Since our exams are scheduled by the students in the CBTF there is no need worry about exam conflicts. I am not approving a time conflict override for the lab sections since being able go in person is important to success in the course.

G Carl Evans