

# Syllabus

Fall 2025 | CS 128

University of Illinois Urbana-Champaign

---

## Instructor Information

**Prof. Michael R. Nowak** | Teaching Assistant Professor

Siebel School of Computing and Data Science, University of Illinois at Urbana-Champaign

- **Email:** [mnowak1@illinois.edu](mailto:mnowak1@illinois.edu)
- **Office:** 2338 Siebel Center for Comp Sci
- **Phone:** +1 (217) 244-8894
- **Website:** [michaelrnowak.com](http://michaelrnowak.com)

*See the Course Communication section of the syllabus for contact guidelines.*

---

## Learning Objectives

At the end of the course, a successful student should be able to:

- Manage, organize, and compile C++ projects of simple complexity using basic Unix-based tools
  - Use basic editing and debugging tools such as GDB and Valgrind
  - Understand the features of computer systems that make them useful for solving problems, including computation, memory, storage, data access and networking
  - Use object-oriented design to appropriately structure data and couple data and behavior
  - Decompose problems of modest complexity into simpler components with well defined interfaces
  - Formulate and implement useful algorithms that solve real problems and can be implemented and run on a computer
  - Construct and test code from a simple specification
-

# Course Structure

## Asynchronous Lectures

This class breaks from traditional collegiate lectures by delivering course content through a series of asynchronous daily lessons available on-demand through our course website. Each lesson introduces new material through a combination of text, video, and interactive walkthroughs. You will find interactive coding exercises interspersed throughout most lessons, which are designed to reinforce the material that you have been learning about.

## Synchronous Programs Recitation Sections

You have enrolled in the synchronous recitation section for CS 128. Your attendance to this section is mandatory, but your grade will not solely depend on attendance. More information about this section, including the grading policies, is found later in this document.

---

## Getting Help

### Office Hours

On days when office hours are scheduled, we will hold a block of in-person office hours located in the Tutoring Center in the Siebel Center for Computer Science, followed by virtual office hours online. For both formats, students will use the [course help site](https://queue.cs128.org) (queue.cs128.org) to join the queue.

The key difference is where students join the queue from:

- For in-person office hours, students must physically go to the Tutoring Center. Within the room, they should locate the CS 128 section, which may change from day to day since we do not have assigned spots. Once their name reaches the top of the queue, it is the student's responsibility to find the appropriate staff member for help.
- For virtual office hours, students may join the queue from wherever they are and will receive help online.

Office hour times are posted on the [course calendar](https://cs128.org/calendar) (cs128.org/calendar).

answer, post your question on the forum. While we do not debug solutions on the discussion forum, we're happy to provide clarification about smaller snippets of code. If you're posting code, please make sure your post is private so that you do not violate our academic integrity policies.

**Rule of thumb:** any general questions about the assignment -- e.g., I don't understand what I'm supposed to do, what does the prompt mean by x, etc. -- can be asked publicly on the course forum. However, any implementation specific questions or otherwise must be asked privately. If you are unsure whether your discussion forum post's contents could go against course policies, you should make your post private.

---

## Course Communication

We have set up a comprehensive and well-organized course website and discussion forum to help you find what you need to know. Our goal is to avoid email and other 1-to-1 forms of communication that don't scale well to large numbers of students.

## Primary Sources of Information for CS 128:

1. This website ([cs128.org](https://cs128.org))
2. The [course forum](https://discuss.cs128.org) ([discuss.cs128.org](https://discuss.cs128.org))

Most policy questions are answered in this syllabus. For almost anything else, search the forum—maybe another student has asked your question and we've already answered it. If you still can't find an answer, post your question on the forum.

## Addressing Course Staff

While I recognize that today's world is more casual than ever, I prefer to keep things formal and professional. Therefore:

- When addressing your professor, please address us with our professional title. For example, Prof. Nowak.
- When addressing a graduate TA, I suggest using the prefix Mr./Ms./Mx. unless your TA tells you otherwise.
- When addressing an undergraduate CA, feel free to use their first name.

## What You're Responsible For

## Contacting the Course Staff

The course forum is your primary point of contact with the staff. You may think the professor's five minutes of responding to your email is not a huge problem. Five-minute responses to 100s of students can consume many hours, though. Therefore, we kindly ask you to post to the course forum.

This is not because we don't like you. It's because there are many of you, a much smaller number of us, and many of the questions you have tend to be shared by other students. If you email us, we can answer your question with one person: you. If you post on the course forum, we can answer your question for the entire class. You may find that your question has already been answered by searching before posting or that another student can answer it for you before the course staff.

### General guide about how to contact the course staff:

- I need help installing... **post on the course forum.**
- I'm confused about the concept... **post on the course forum.**
- I need help with... **post on the course forum.**
- I can't find... **post the course forum.**

### Cases where you can and should contact the course instructors:

- I think that my friend is cheating in CS 128... **contact the course staff.**
- I'm really sick and getting behind in the class... **contact the course staff.**
- I'm feeling really overwhelmed and need someone to talk to... **contact the course staff, or an academic advisor, or a friend.**

## Grading

### Grade Scale

Only exceptional performance will constitute an A grade. If you're okay with just getting by or are a quiet quitter, this course will likely be too fast-paced for you.

Course grades will be assigned according to this scale:

Minimum % of Total Points Earned	Letter Grade
93.0%	A

83.0%	B
80.0%	B-
77.0%	C+
73.0%	C
70.0%	C-
60.0%	D

*You will fail this course if you earn below 60.0% of the total available points.*

**Note:** Final grade percentages are *truncated* (not rounded) to the nearest tenth to determine the letter grade. For example, a final score of 89.99% will be treated as 89.9%.

## Method of Evaluation

Course grades will be calculated using the following weights:

Category	% Contribution
Machine Problems	40%
Quizzes	40%
Graded Lessons	15%
Recitation	5%

*We reserve the right to modify these weightings during the semester. Note: No categories will overflow—you cannot earn more than the total number of available points for each respective category.*

## Last Day to Submit Coursework

We do not accept any submissions past the semester's last day of instruction. The date for the current semester's last day of instruction is found on the [University's academic calendar](https://registrar.illinois.edu/faculty-staff/calendars/academic-calendars/) (registrar.illinois.edu/faculty-staff/calendars/academic-calendars/). Search the academic calendar for "Last day of instruction".

## Assessments

quiz by completing all work individually and without collaboration.

- Machine problems are also called MPs throughout this document and the course.
- New MPs will be released according to the schedule below.
- To achieve the best outcomes, begin working on your solutions as soon as each MP is released and continue steadily until the due date—waiting until the last minute often leads to poor results.
- Each MP increases in difficulty and sophistication compared to the previous one.
- All MPs are automatically graded by the PrairieLearn autograder, and partial credit may be awarded for passing some of the test cases. However, any solution that does not compile will receive no points.

## MP Schedule

#	Release Date	Due Date	Points
0	2025-08-29 05:00:00	2025-09-11 23:59:59	10
1	2025-09-12 05:00:00	2025-09-18 23:59:59	100
2	2025-09-19 05:00:00	2025-09-25 23:59:59	100
3	2025-09-26 05:00:00	2025-10-09 23:59:59	100
4	2025-10-10 05:00:00	2025-10-23 23:59:59	100
5	2025-10-24 05:00:00	2025-11-06 23:59:59	100
6	2025-11-07 05:00:00	2025-11-13 23:59:59	100
7	2025-11-14 05:00:00	2025-11-20 23:59:59	100
8	2025-11-21 05:00:00	2025-12-10 23:59:59	100

*All date-times are in the Champaign-local time zone.*

*Each MP's Release Date and Formal Due Date are subject to change; any changes will be communicated via the course forum.*

## MP Late Policy

The late policy for each MP follows a graduated credit reduction system designed to encourage timely submission while still providing opportunities for partial credit.

### Credit Reduction Schedule:

- **1 day late:** 95% credit
- **2 days late:** 90% credit
- **3 days late:** 85% credit
- **4+ days late (up to the last day of instruction):** 70% credit

submission is possible.

- **Note:** A "one-day late" submission is not measured on a 24-hour rolling clock but instead by calendar days. For instance, if an MP is due on Monday at any time, then all of Tuesday is considered the one-day late period. This means that Tuesday, regardless of the hour of submission, is treated as a single late day. For emphasis, if an MP is due by Monday 23:59:59 (Champaign local time), the 1 day late penalty comes into effect on Tuesday 00:00:00.

## MP Late Penalty Waivers

Students can elect to waive the late penalty on **TWO (2)** machine problems (past, present, or future) from the "MP Late Penalty Waivers" section of My.CS128 ([cs128.org/my/waivers](https://cs128.org/my/waivers)).

- This is a "No Questions Asked" waiver that is available for everyone.
- This is not automatic, and you are not required to use these waivers should you choose not to.
- Without exception, students must select their late penalty waivers by the semester's last day of instruction.
- For any MP you've elected a late penalty waiver, submit the assignment to PrairieLearn by 23:59 Champaign local time on the semester's last day of instruction.
- The late penalty waiver will not appear in PrairieLearn in real time. As a result, late submissions in PrairieLearn will display the score *with* the late penalty applied. The waived penalty will *only* be reflected on your official CS 128 Grade Report.

## MPs and PrairieLearn

In this course, you will submit your MPs to PrairieLearn using your CS 128 account instead of your NetID account used in your other classes. Therefore, you must access our CS 128 course in PrairieLearn through the links provided in our assessment prompts. Each assessment submitted to PrairieLearn in CS 128 will have an "Access Autograder" blue button. Upon clicking that button, you will be logged into your CS 128 PrairieLearn account. If your other courses use PrairieLearn, you must log out and back in using your NetID account to access your other classes.

## Quizzes

### Overview

#### No Collaboration Allowed.

No course staff are involved in grading CS 128 quizzes, so please do not appeal your grade to the course staff. If you have concerns about the questions themselves, please post privately on the discussion forum after the quiz ends, usually meaning at least one day later than the CBTF end date (as defined below).

## Important Quiz Information

- The policies of the CBTF are the policies of this course, and academic integrity infractions related to the CBTF are infractions in this course.
- If you have accommodations identified by the [Division of Rehabilitation-Education Services \(DRES\)](https://disability.illinois.edu) (disability.illinois.edu) for quizzes, please [submit your Letter of Accommodations \(LOA\) here](https://cbtf.illinois.edu/students/dres) (cbtf.illinois.edu/students/dres) before you make your first quiz reservation. This must be done each semester you use the CBTF.
- If you have any issue during an quiz, inform the proctor immediately. Work with the proctor to resolve the issue at the time before logging off. If you do not inform a proctor of a problem during the test then you forfeit all rights to redress.
- [Review all instructions on the CBTF website](https://cbtf.illinois.edu/students) (cbtf.illinois.edu/students) before your first quiz.

## Quiz Schedule

Name	Duration	CBTF Start Date	CBTF End Date	CBTF Self-Reserve
Quiz #1	50 min	2025-09-04 00:01:00	2025-09-07 23:59:00	2025-08-21 00:01:00
Quiz #2	50 min	2025-09-18 00:01:00	2025-09-21 23:59:00	2025-09-04 00:01:00
Quiz #3	50 min	2025-10-02 00:01:00	2025-10-05 23:59:00	2025-09-18 00:01:00
Quiz #4	50 min	2025-10-16 00:01:00	2025-10-19 23:59:00	2025-10-02 00:01:00
Quiz #5	50 min	2025-10-30 00:01:00	2025-11-02 23:59:00	2025-10-16 00:01:00
Quiz #6	50 min	2025-11-13 00:01:00	2025-11-16 23:59:00	2025-10-30 00:01:00
Quiz #7	50 min	2025-12-04 00:01:00	2025-12-07 23:59:00	2025-11-13 00:01:00

*All date-times are in the Champaign-local time zone.*

## Quiz Format

- CS 128 quizzes *may* include multiple-choice questions, programming exercises, debugging challenges, and other formats as appropriate. All questions are automatically graded.
- Quizzes typically feature conceptual questions drawn from lesson content, along with small programming problems. Programming problems may be inspired by prior lesson activities or adapted from previously assigned machine problems.
- You have unlimited attempts on programming and debugging questions without penalty. These are designed for practice, and we encourage you to refine your solutions. Keep in mind, however, that quizzes are timed, so you do not have unlimited time to complete them.
- Conceptual questions should be straightforward if you have been keeping up with daily lessons. Programming and debugging questions may be more challenging—especially under time constraints. While you won't have access to outside resources (such as course staff or the internet) during quizzes,



Do not contact the course staff regarding missed quizzes. You must take them at the time you scheduled in the CBTF and not at some other time. We will drop your lowest quiz score (only one quiz will be dropped) when computing the quiz component of your final grade.

## Catch-up Quiz Grading

The material in CS 128 is cumulative, and in large part each daily lesson builds on the material covered previously. We realize that sometimes, you think you've understood a concept, but your performance on a quiz proves otherwise. If you perform poorly on a quiz, we want you to go back and review that material so that you catch up and are prepared to move forward!

That's why we provide a catch-up quiz grading policy. Here's how it works. If you do better on Quiz  $N + 1$  than on Quiz  $N$ , we'll increase your score on Quiz  $N$  to the average of the two scores. For example, let's say you struggle on Quiz 3 and earn a 70, but then rally on Quiz 4 for a 90. Since your performance on Quiz 4 indicates knowledge of the material tested on Quiz 3, we'll increase your Quiz 3 score to 80:  $(70 + 90) / 2$ . If you do worse on Quiz 4, your Quiz 3 score remains unchanged. This policy will never lower any quiz grade.

There are a few caveats. First, we don't apply catch-up quiz grading to quizzes that you miss. So you have to take Quiz  $N$  for us to apply it after you complete Quiz  $N + 1$ .

Finally, catch-up quiz grading does not ripple backward. So we increase your score on Quiz  $N$  because you did better on Quiz  $N + 1$ , we don't then go back and adjust Quiz  $N - 1$ , and so on.

## Graded Lessons

### Limited Collaboration Allowed

- We expect you to complete the activities and questions on the same day that the lesson is released.
- A lesson is considered a graded lesson if it has at least one graded programming activity or question.
- Lessons are designed to be completed in a study group setting. We strongly encourage you to form a study group and schedule a regular daily meeting time to go through each lesson together. Discuss the lesson content as it is presented, brainstorm high-level approaches to the activities, and help one another think through the concepts.
- You may review and discuss code with members of your study group—either in person or virtually—but you may not electronically share or receive source code (e.g., by copying, sending, or uploading files).
- All students are required to write and submit their own solutions. Even when working together on the lessons, we expect each student's work to reflect their own understanding, which naturally results in small differences between submissions.

Within a graded lesson:

- Each programming activity or question will display its point value, the number of allowed submissions (with unlimited attempts for programming activities), and its due date.
- All due dates follow the course's published due date and late policy. In the event of any discrepancy between the due date shown in an individual activity and the published due date/late policy stated below, the published policy will take precedence.
- Your score for each activity or question will be recorded as the maximum score across all of your submissions.
- Your score for a graded lesson will be the total number of points earned out of the maximum points available for that lesson. Each graded lesson contributes equally to the graded lesson component of your final grade.

## Due Dates and Late Policy

- **Due date:** Graded lessons are due by 23:59 (Champaign local time) on the day after they are assigned. For example, if a graded lesson is assigned on Monday, its activities must be completed by Tuesday at 23:59.
- **Late policy:** Graded lessons must be completed by their due date. Do not contact course staff about missed graded lessons. To account for illness, emergencies, or other excused absences, we will automatically drop the lowest five (5) graded lessons when calculating the graded lesson portion of your final grade.

## Recitation

### Collaboration Required

Recitation is a time to review and practice applying the material learned during the week. The meeting period will consist of a short presentation and a group activity. The activities will cover important extensions of the week's material that we think is vital for deeply understanding the course material. We also believe group work and an in-person environment can bring many benefits to students. For these reasons we made recitation a required part of the course with it contributing to your final course grade.

Your recitation grade is based on your attendance and the effort you put into the activity. There is no partial credit for a recitation session; you either attend and participate to earn full credit or receive a zero for that week. Additionally, two (2) of your recitation scores will be dropped when computing your final grade.

You are expected to be present, engaged, and respectful during recitation. You may not receive credit if you fail to meet these expectations. Potential reasons for losing recitation credit include:

- Not attending your scheduled recitation.

- Behaving rudely or disrespectfully toward classmates or course staff.
- 

## General Policies

### Late Penalty Waivers and Drops

When life happens—whether expected or unexpected—we recognize that there may be times when you're unable to attend class, complete an assignment, or meet a deadline. The course includes built-in flexibility in the form of drops and late penalty waivers to help you manage such situations.

We expect you to use these tools first, as they were built into the course to support you in navigating the kinds of situations where completing work on time isn't possible. **These accommodations are intended to cover a range of life events, from illnesses and personal emergencies to scheduling conflicts or other unexpected obligations.**

Please make sure to save documentation for any instance in which you use a drop or late penalty waiver (e.g., doctor's notes, official communications, etc.). This documentation may be requested, especially if you seek further accommodations beyond what is already built into the course.

#### In summary:

- **Use the built-in accommodations first:** The course design anticipates life's challenges, so we require you to use the provided drops and late penalty waivers as your first recourse.
- **Document every case:** Always save the necessary documentation for any drop or late penalty waiver you use.
- **Additional accommodations:** Once these options have been exhausted, we can discuss further accommodations if needed.

### Regrades and Audits

Regrade requests, challenges, and/or otherwise of the grade on any assessment must be submitted within one week (7 days) from the date that the respective student work is handed back.

For all works submitted to PrairieLearn or to the website (cs128.org), this is within one-week from the assignment due date (not the late due date).

We reserve the right to audit the grades for any assessment submitted to this course; during the audit process, we can decrease or increase your score. Examples when this might occur include (but are not limited to):

Whenever there is uncertainty, it is your responsibility to seek clarity. Never assume; instead confirm.

## Reporting Problems

If you believe that you have identified a problem with a quiz question, machine problem, lesson, or otherwise, please report the problem to the course staff through a private post on the discussion forum. Do not post your questions publicly.

Once we have received your report, we will do one of the following:

- If the question has a bug, we will correct it and ensure that all affected students receive full credit or have an opportunity to address the issue at our discretion—including those who completed the quiz before the bug was identified.
  - If the question has a minor typo that we don't think affects its ability to be correctly answered, we will fix it and distribute that change.
  - If the question is fine, we will not do anything.
- 

## Academic Integrity

The University of Illinois at Urbana-Champaign [Student Code](https://studentcode.illinois.edu) (studentcode.illinois.edu) should also be considered as a part of this syllabus. Students should pay particular attention to [Article 1, Part 4: Academic Integrity](https://studentcode.illinois.edu/article1) (studentcode.illinois.edu/article1).

Academic dishonesty will result in a sanction proportionate to the severity of the infraction, with possible sanctions described in [1-404](https://studentcode.illinois.edu/article1/part4/1-404) (studentcode.illinois.edu/article1/part4/1-404) of the [Student Code](https://studentcode.illinois.edu) (studentcode.illinois.edu). Every student is expected to review and abide by the [Academic Integrity Policy](https://studentcode.illinois.edu/article1) (studentcode.illinois.edu/article1) as defined in the Student Code. As a student it is your responsibility to refrain from infractions of academic integrity and from conduct that aids others in such infractions.

Review the [Students' Quick Reference Guide to Academic Integrity](https://provost.illinois.edu/policies/policies/academic-integrity/students-quick-reference-guide-to-academic-integrity/)

(provost.illinois.edu/policies/policies/academic-integrity/students-quick-reference-guide-to-academic-integrity/). Ignorance of these policies is not an excuse for any academic dishonesty. It is your responsibility to read this policy to avoid any misunderstanding. Do not hesitate to ask the instructor(s) if you are ever in doubt about what constitutes plagiarism, cheating, or any other breach of academic integrity. It is imperative that each student clearly understand those rules and the severe consequences that can result from the adjudication of an Honor Code Violation.

**By submitting anything to this course, electronically or otherwise, you are asserting the following:** "I have neither given nor received unauthorized aid on this academic work. In particular, I certify that I have

---

consequences as cheating.

**In other words, if you provide your solution to another student—even if that student never submits it—you have committed an act of academic dishonesty. Both students will be subject to the same penalties.**

Writing, inserting, or otherwise modifying code or comments to reveal hidden test cases, or to circumvent or bypass our auto-grader in any way, is considered an academic integrity violation. You may never hard-code against the test cases in our suite.

Students have previously defended FAIR violations by claiming that a friend or peer accessed their solutions while their device was left unattended or loaned out. **This is not a valid defense.** It is your responsibility to secure your coursework and protect your device. If another individual accesses your work due to negligence on your part, you will still be held accountable.

## Cheating

All work submitted in this course must be entirely your own. The following activities are considered cheating and will result in sanctions as outlined below. You may not:

- Submit work completed by anyone other than yourself.
- Copy or paste code from any source that you did not personally write.
- Misrepresent another student's work as your own or allow your work to be misrepresented as theirs.
- Examine, reproduce, or submit another classmate's machine problem or quiz solution.
- Share or discuss the content of quizzes or other private course assessments.
- Publish your MPs or coursework in any location where other students could access them.

*Note:* This includes posting assignments publicly on GitHub. If you want to showcase your skills to employers, use your GitHub portfolio for independent projects, not course work.

- Email, message, or otherwise share your code with anyone in this class now or in future semesters.

All submitted work will be analyzed using advanced cheating-detection software. These tools are highly accurate, and any evidence of plagiarism or academic dishonesty will immediately initiate formal academic integrity proceedings. We take this very seriously, and you should as well.

**Important:** Course staff (CAs and TAs) are not responsible for warning you about potential policy violations. It is solely your responsibility to know and follow the course's academic integrity rules. If you are ever uncertain about what is allowed, you must seek clarification directly from the instructor via the course forum *before* collaborating with another student, consulting an AI tool (which is never permitted), or seeking outside assistance on any assignment.

accessible repositories found on the internet. If there is a high similarity between your code and another student's code (or code from online solutions), and we believe it is more probable than not that an academic integrity violation has occurred, we issue a FAIR violation. It is assumed that college students know what is honest and what is not.

## Self-plagiarism

Self-plagiarism is reusing the work you have already submitted to a class. In CS 128, you cannot reuse/submit work you've completed during a previous semester. Instead, you need to think through the problems and compose new solutions. Past performance is typically indicative of future performance. Sitting out and disengaging while resubmitting old work is unsuitable for your learning. Accordingly, I will only allow submissions of a past semester's work if one has an extraordinary circumstance. For example, if one had to withdraw due to medical reasons. This policy is not meant to be harsh or otherwise; I sincerely want you to do well in my class and future courses in our program. Therefore, to encourage learning, I can only allow you to submit codes from a previous semester this semester if you have an extraordinary circumstance that I consider and subsequently decide to approve in writing.

Accordingly, if you are retaking the course this semester, you will need to compose solutions without any reference to your work a last semester. Failure to follow our self-plagiarism policies will result in a FAIR referral. Simply do not review your solutions from a previous semester, and you should be good.

## Policies on Collaboration

### Machine Problems

**No collaboration of any kind**—with other students, outside individuals, AI tools, or any other source—is permitted on Machine Problems (MPs). If you have questions about your solution, approach, or anything else related to an MP, you must seek help directly from course staff during office hours. Assistance from classmates, others, or AIs is not allowed.

Searching for or using solutions from the Internet is an act of cheating. If you are found to have viewed or submitted code obtained online, you will be referred to FAIR for an academic integrity violation. We take this very seriously. Do not cheat yourself—the purpose of these assignments is to develop your skills through independent effort.

Similarly, AI tools often generate solutions that use constructs not taught in this course and may produce nearly identical code for different users with similar prompts. **Using AI-generated code is strictly prohibited and readily detectable.** If you are found to have used AI on an MP, you will be referred to FAIR for an academic integrity violation.

Exchanging or soliciting ideas about how to solve a graded lesson's activity is not cheating, but submitting another's code is cheating. Feel free to discuss your solutions with other students as long as you do not provide them your source code.

## In General

You might look up how to do something in C++. For example, "How do I iterate through a `std::map`'s keys?", "How do I clear the contents of an `std::string`?", "How do I preallocate memory for an `std::vector`?", etc. During such lookup, if you find inspiration from a source not directly associated with an assignment's prompt and you adapt that code, be sure to make it your own and cite the source. After this exercise, your code should look very different from the inspired source. Looking at C++ documentation is not considered cheating, provided you do not plagiarize it.

## Sanctions

If you are caught cheating in CS 128 you will (without question) receive a FAIR violation. The penalty for the first infraction will be:

- A letter grade reduction in the class *and* a zero score on each assessment involved in the violation.

For a second offense or extreme act affecting other students:

- An F in the course.

Please note that these sanctions deviate from the departmental recommendations.

---

## Additional Policies

### Style Guide

In this course, we will use the Google C++ Style Guide (<https://google.github.io/styleguide/cppguide.html>).

### Copyright Statement

#### Course Materials

The course materials used in this course are copyrighted. All material prepared for this class is copyrighted; this includes the syllabus, lecture slides and notes, exams, machine problems, lab work

Students may not record audio or video of any course activity unless the student has an approved accommodation from Disability Resources & Educational Services permitting recording. This accommodation letter must be presented to the instructor in advance of any recording being done. Students who are allowed to record classes are not permitted to redistribute audio or video recordings of statements or comments from the course to other individuals without the express permission of the faculty member and of any students/staff members who are recorded.

## Classroom Climate

Our course goals can only be accomplished in a setting of mutual respect. Your instructors are committed to creating a classroom environment that welcomes all students, regardless of their identities—race, class, gender, sexual orientation, religious beliefs. We value diversity in all of its definitions, including who we are, how we think, and what we do. We cultivate an accessible, inclusive, and equitable culture where everyone can pursue their passions and reach their potential in an intellectually stimulating and respectful environment. 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 behavior that goes against these expectations, you are encouraged to bring this to the attention of the course instructor. Please let us know if there is anything we can do to make sure everyone is encouraged to succeed in this class.

## Our 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.

## Religious Observances

Illinois law requires the University to reasonably accommodate its students' religious beliefs, observances, and practices with admissions, class attendance, and the scheduling of examinations and work requirements. Therefore, we recommend that students examine each syllabus at the beginning of the semester for potential conflicts, notify the instructor/professor, and follow their directions on how to request accommodations.

This class requires students to notify the instructor via email of any class activities or assessments affected by a religious observance. To best facilitate planning and communication, you must notify your



If you have DRES Accommodations, it is required that you formally submit your letter of accommodations for this course. Instructions on how to fulfill this requirement will be found on the the Start Here page. We must to receive your accommodations letter before you start using them. It's important to note that we won't be able to apply accommodations retroactively in any situation.

## ADA Statement

The Americans with Disabilities Act (ADA) is a federal anti-discrimination statute that provides comprehensive civil rights protection for persons with disabilities. Among other things, this legislation requires that all students with disabilities be guaranteed a learning environment that provides for reasonable accommodation of their disabilities. To obtain disability-related academic adjustments and/or auxiliary aids, students with disabilities must contact the course instructor and the Disability Resources and Educational Services (DRES) as soon as possible. To contact DRES, you may visit 1207 S. Oak St., Champaign, call 333-4603, e-mail [disability@illinois.edu](mailto:disability@illinois.edu) or go to <https://www.disability.illinois.edu>. If you are concerned you have a disability-related condition that is impacting your academic progress, there are academic screening appointments available that can help diagnose a previously undiagnosed disability. You may access these by visiting the DRES website and selecting "Request an Academic Screening" at the bottom of the page.

## Computer Requirements

Please ensure that you have a laptop device that meets the College of Engineering's recommendations ([https://techzone.illinois.edu/site\\_engineering\\_recommendation.asp](https://techzone.illinois.edu/site_engineering_recommendation.asp)), with either Window, Mac, or Ubuntu operating system, and the Google Chrome web browser (used to access our website) installed.

---

## Tentative Schedule

Week Topics	
1	Introduction to C++
2	Introduction to C++ cont.
3	Navigating the command line; compilation and execution
4	Build systems, version control, and software and errors
5	Testing and debugging
6	Compound types (pointers, references, arrays); user-defined types
7	Streams and input validation; data representation
8	Functions and the stack; the free store

- |    |  |
|----|--|
| 12 | Trees                                  |
| 13 | Graphs                                 |
| 14 | Design patterns; smart pointers        |
| 15 | Concurrency, networking, and databases |