

CS 2110: Object-Oriented Programming and Data Structures

Syllabus

Course description

Intermediate programming in a high-level language and introduction to software engineering. Topics include object-oriented programming (classes, objects, subclasses, types), graphical user interfaces, algorithm analysis (asymptotic complexity, big “O” notation), recursion, testing, program correctness (loop invariants), searching/sorting, data structures (lists, trees, stacks, queues, heaps, search trees, hash tables, graphs), graph algorithms. Java is the principal programming language.

Prerequisites

CS 1110 or CS 1112 or equivalent course on programming in a procedural language.

Learning outcomes

- Employ recursion and object-oriented programming concepts (e.g. classes, objects, inheritance, and interfaces) to solve computational problems.
- Design and implement nontrivial Java programs (roughly 1000 lines of code), starting from an English language specification.
- Build and modify event-driven graphical user interfaces (GUIs) expressed in Java.
- Analyze the asymptotic complexity of algorithms and programs to determine their running times.
- Implement basic data structures taught in the course (linked lists, binary search trees, heaps, hash tables, adjacency lists) and be able to use them in programs.

Course staff

Instructor

Curran Muhlberger

Lecturer, Computer Science

cdm89@cornell.edu

Gates Hall 460

Office hours: Tuesdays, 11:45–13:00 (subject to change; see [office hours](#))

Individual appointments: To schedule an individual appointment with an instructor, see the [office hours](#) page.

Teaching assistants

Discussion sections are led by graduate and undergraduate teaching assistants (TAs). The TA roster can be found in [Canvas](#).

Consultants

Undergraduate consultants assist groups during discussion sections and hold consulting hours for one-on-one assistance with programming assignments.

Administration

Ms. Corey Torres (ct635@cornell.edu) serves as the Course Coordinator for CS 2110. You should coordinate with her in the event of last-minute exam conflicts and may see messages from her related to SDS accommodations and conflicts you report on our surveys. Please do not email the instructor about exam logistics; instead, email the Course Coordinator. Note that messages received after business hours will not be replied to until the next business day.

Meeting times

Lectures are held Tuesdays and Thursdays from 10:10am to 11:25am in Bailey Hall (BLY 101). Discussion sections meet once per week for 50 minutes on either Tuesday or Wednesday (see [class roster](#) for times and locations). Class begins on August 22 and ends on November 30, totaling 28 75-minute lectures and 13 50-minute discussions (see [schedule](#)). Attendance and participation are expected in both lectures and discussions.

Lectures will start promptly on time, so please do your best to be settled *before* 10:10am. The instructor will try hard to end lectures on time as well, so please be respectful and refrain from packing up until class has been dismissed. Show the same courtesy to your discussion TAs and classmates.

Please attend the discussion section in which you are enrolled—many classrooms are filled to capacity, and TAs occasionally send follow-up notes to the students enrolled in their section. If you have a one-time conflict with your regular section, you are allowed to attend another section so long as it is not full (look for the green dot on the [class roster](#)). But if you develop a habit of attending another section, you should change your enrollment on StudentCenter to match. Note that instructors cannot change your section enrollment; you must go through the registrar.

We will attempt to record audio and screen captures from lectures on a best-effort basis to help students who miss lecture for illness keep up with the material; these recordings may be viewed on Canvas.

Credits and grade basis

4 credits, letter or S/U grades, audits allowed.

Credits reflect scheduled contact hours and associated work, including roughly biweekly programming assignments.

Auditors are permitted to participate fully in the class, including submitting assignments, attending office hours, and receiving scores on assignments and exams.

Office hours and consulting

See our [office hours](#) page for times, locations, and appointment reservation instructions for office hours offered by instructors, TAs, and consultants. If a last-minute change must be made to the schedule, this will be announced on [Ed Discussion](#). Consulting hours and most TA office hours take place in Rhodes 405. Be advised that “camping out” in office hours is not a strategy for success in this course; rather, when you come to office hours, you should have focused questions that you have already written down.

During **instructors’ office hours**, we are happy to answer high-level questions about course content for any students who stop by. This is intended to be a small-group setting, and we may poll several students for questions and answer them out of order for the benefit of the group. We can also share related anecdotes from our experience applying course concepts in research and industry. Questions about assignments will not be prioritized unless you have been referred to us by a consultant or TA.

Instructors may also be available for a limited number of **one-on-one appointments**, which must be reserved in advance. These are best suited for discussing accommodations, getting advice on how to study, reviewing your performance, or resolving partner conflicts (please specify your agenda in advance); they should not be used for assignment help. If you reserve an appointment, you are expected to attend.

All **TAs’ office hours** are open to all students (you are not restricted to the TA who teaches your discussion section). As with instructors’ open hours, TA office hours are able to accommodate small groups of students who may have related points of confusion; therefore, they are great for addressing high-level questions about course content (lectures & reading), past quizzes, lab exercises, and general approaches to assignments. If the session is quiet, the TA may be able to provide individualized assistance with debugging your assignment code, but priority will be given to assistance that can benefit everyone present.

Consulting hours are dedicated to one-on-one assistance (or two-on-one if working with an assignment partner). Consultations are first-come, first-served. Our consultants have all taken a version of CS 2110 or 2112, so they can answer questions about material as well as assignments. Because of the one-on-one setting, consulting is the best way to get individualized assistance with assignment debugging.

Students should not expect more than 10 consecutive minutes of individualized attention during a consultation, as we need to keep the queue moving. Therefore, please come prepared with specific questions, having already read the lecture notes or assignment + skeleton code, attempted a design or test plan, and identified a failing diagnostic. Remember that consultants are not there to implement assignments for you, so expect to be asked to work independently and rejoin the queue even if consulting hours are quiet.

Materials and services

Textbook

The primary textbook for this course is [*Data Structures and Abstractions with Java, Fifth Edition*](#), by Frank M. Carrano and Timothy M. Henry. A paper copy is recommended, as it will likely improve your retention of the material, but an ebook is also acceptable.

Readings may also be assigned from the following free online references:

- [JavaHyperText](#)—online textbook originally prepared for CS 2110 by Cornell professor David Gries
- [Object-Oriented Design and Data Structures](#)—online course notes originally prepared for CS 2112 by Cornell professors Andrew Myers and Dexter Kozen
- [The Java Tutorials](#)—official tutorials from the creators of the Java language

We recommend that students at least skim relevant readings before lecture, then re-read them carefully afterwards.

iClicker remotes

Lectures for this class will include problems for you to solve in your seats and answer using an iClicker remote. Therefore, iClicker remotes are *required* for this class. You will need a physical remote, not an app on your phone (neither the WiFi in our lecture hall nor the iClicker instructor software are stable enough to use it at the scale of 600+ students).

Your remote does *not* need to be new—a used “original,” “plus,” or “2” model would be fine, and it can be used in future courses as well (such as CS 3410). If you signed up for the Cornell Academic Materials Program (CAMP), you should be able to pick up a remote from the bookstore for free. You will need to register your remote on our [remote registration form](#).

Try to have your remote ready by the start of classes, but don't worry if it takes a few extra days to acquire one—responses will not count towards your grade until after Cornell's "add" period ends on September 5. When using your remote in class, look for a positive confirmation that your response was received (e.g. a green light). If you see a negative confirmation (e.g. an orange light), ensure that your remote is [set to the correct frequency](#) (we will use the default, "AA"). If you see no confirmation at all, check your batteries. A "score" counting your responses will be regularly updated on Canvas; check it to ensure that your remote is registered and successfully communicating.

Personal technology

Students will need regular access to a computer in order to complete this course. You will need the ability to interact with course websites, read PDF documents, extract and create ZIP archives, and run a Java development environment (preferably IntelliJ IDEA). A USB stick will come in handy should you need to use a loaner laptop or computer lab at some point during the semester (have a plan for what to do if your primary computer breaks down).

If you have a personal laptop, we recommend bringing it to discussion sections—some activities will require at least one group member to use a computer. Please try to keep your laptop well charged, as most classrooms lack accessible power outlets (and charging cables that present a trip hazard will not be allowed). A laptop is *not* required in lectures, however. Multitasking on phones or laptops is known to impede learning and lower grades for both you *and your neighbors*, so we ask that you refrain from using them and take notes on paper if possible. Students who may nevertheless be tempted to multitask on a notetaking device should sit in the back of the lecture hall (and classmates should not hesitate to ask distracting neighbors to relocate there). A section near the front will be reserved as a laptop-free zone.

Online services

The course will utilize the following online services, most of which you can access with your Cornell account once you are logged into Canvas:

- [Course website](#) (lecture materials and assignments; accessible to public)
- [CMSX](#) (assignment submission, grades)
- [Canvas](#) (announcements, quizzes and surveys)
- [Ed Discussion](#) (class discussion board)
- Gradescope [access via Canvas] (exam feedback)

Assessment

Basis of grade determination

Student performance will be assessed using the following elements, weighted approximately as indicated to yield an overall performance score:

Participation	5%
Lecture (iClicker)	2.5%
Discussion (cooperative exercises)	2%
Course evaluations	0.5%
Quizzes	6%
Assignments	31%
Assignment 1	3%
Assignment 2	4%
Assignment 3	5%
Assignment 4	5%
Assignment 5	4%
Assignment 6	6%
Timeliness	4%
Prelim 1	16%
Prelim 2	16%
Final	26%

Grading scale

Letter grades are assigned in accordance with the [Cornell University grading system](#) based on the instructor's assessment of knowledge & understanding, perception, and originality reflected in a student's work (weighted as described above). Historically, about 35% of students receive a grade of A- or higher, and the median grade has been a B.

Grade boundaries are not known in advance, as assignments and rubrics vary between semesters. But to help you manage your expectations, we guarantee that:

- Performance scores above 90% will receive at least an A-
- Performance scores above 77% will receive at least a B-
- Performance scores above 60% will receive at least a C- as long as your exam scores exceed 40%

Attendance

Engaging in classroom activities both improves your learning and creates a more vibrant learning environment for your peers, so we want to reward you for it. Class participation will largely be

assessed via iClicker responses and submitting discussion activities with a group. *Students do not need perfect attendance or perfectly correct responses to maximize the participation component of their grade!* (Such a policy creates perverse incentives, such as rewarding coming to class when ill, which we do not want.) Responding to polling questions in at least 3/4 of lectures and submitting at least 3/4 of discussion activities is sufficient (responding to fewer will result in a proportional deduction). This provides flexibility to miss an occasional class for illness or a conflicting obligation without worrying about the impact on your grade. There are no make-up opportunities for participation credit and no “excused absences”; if you think you need to miss more than 25% of class meetings for legitimate reasons, discuss your situation with an instructor or advisor ASAP.

Lectures are offered exclusively in person and are not recorded. But be mindful of public health—if you do not feel well, please stay home. Posted materials, our discussion board, office hours, and your classmates are all available to help you get caught up after an absence.

Discussion activities

The majority of each discussion section is dedicated to completing cooperative activities defined on an online worksheet. Your responses to these activities (typically either a scan of the worksheet or some Java files) must be submitted to CMSX in order to earn participation credit for that week’s section. Credit is given for a good-faith effort to complete as much of the worksheet as allowed by class time with a group. Responses should ideally be submitted as soon as section ends, though we allow submissions through Friday. Only one representative from each group should submit; other group members are responsible for ensuring that they are grouped in CMSX and that someone in the group submits on their behalf. If for any reason you submit a solo response, it will receive half credit. You might choose to do that if you are unable to attend section, or if you failed to group properly but are still able to submit by yourself before the Friday deadline.

Quizzes

Take-home quizzes are assigned weekly on Canvas. You will have two attempts to respond to each quiz (note that the questions may be slightly different), and we will take your highest score. The quizzes are open-note, but they are timed and must be completed individually. Late submissions are not accepted, as questions will be discussed in discussion sections the following day. We will drop your lowest two quiz scores automatically to accommodate unexpected circumstances.

We recommend thoroughly reviewing the week’s material before taking your first quiz attempt. Quiz questions will require that you apply these concepts in new settings, focusing on the *implications* of rules more than the rules themselves, so experiment with corner cases as you study. After your first attempt, review the results and try to determine why any incorrect answers are incorrect (if you can code up an example, that usually helps). This should prepare you well for your second attempt.

There will also be a quiz on this syllabus that you must complete during the first week of classes. The syllabus quiz cannot be dropped (an extension will be granted for students who join late).

Programming assignments

The first two programming assignments must be completed individually to ensure that everyone in the class can individually compile, run, and test Java code. Later assignments may allow you to work with a partner of your choice. The code and supporting documents you produce for an assignment must be submitted to CMSX; we may also ask you to complete a survey about your experience with the assignment. New assignments are generally released a few days after the previous one was due, and we aim to cover all material required for an assignment before it is released; this gives you approximately 2 weeks to complete most assignments.

If working with a partner on later assignments, all code must be written *collaboratively*, ideally following “pair programming” principles—one partner proposes ideas while the other partner types, switching roles regularly. Partners should not attempt to divide the assignment and integrate independent work—both partners are responsible for *all* of the contents of their submission.

Partnerships must be declared by forming a group on CMSX before starting work (one partner sends an invitation, the other partner accepts). Forming a group gives you permission to collaborate at the source code level, and it is also a commitment to work together on a mutually-agreed schedule. Be sure to discuss work habits with your proposed partner before forming a group and discussing code.

Late work

Assignments are due at 11:59pm Eastern Time on their due date, and it is your responsibility to ensure that your submission is complete by that time, even in the presence of potential technical glitches (though we do have a short grace period to accommodate a last-minute server reboot). Assignment submissions will still be accepted up to 3 days after the due date, but each day late consumes a “slip day”. The first 4 slip days used incur no penalty; beyond that, each slip day deducts proportionally from your course timeliness score (so if every assignment is submitted 3 days late, your timeliness score will be 0). Slip days are intended to accommodate illnesses, emergencies, and unforeseeable conflicting commitments, but you do not need to justify their use to instructors. Slip days are consumed automatically when submitting late.

Assignment extensions are reserved for exceptional circumstances (e.g. extended hospitalization), and requests for extensions must be accompanied by a letter from SDS or your college advisor. Slip days and the late penalty, not extensions, are the appropriate accommodation in most circumstances.

Assignments will not be accepted more than 3 days late in order to facilitate timely grading. Resubmissions of incomplete work (along with critical bug fixes) will be accepted with penalty during the regrade window, but at that point most of your attention should be on the next assignment to avoid falling behind.

We have tried to avoid due dates that overlap with major religious observances, but in general you are responsible for budgeting your time with awareness of all of your obligations so that you can submit work by its due date. Starting early is the best way to ensure access to office hours and consulting hours compatible with your schedule.

Smoketests

Most assignments will be accompanied by a “smoketest” that checks whether your code successfully compiles and runs on graders’ computers. If your code does not pass the smoketests, you are responsible for fixing it and resubmitting before we will be able to grade it. If this is not done before submitting, you will be asked to fix your code during the regrade window for a cost of 10 points.

Exams

There will be two evening prelims during the semester (September 19 and November 14) and one final exam during finals week. Exams are cumulative (as the course content builds on previous topics), but there will usually be an emphasis on more recent topics. In the weeks approaching each exam, a study guide will be made available on the [exams page](#) detailing which topics are in scope. One or more practice exams will also be made available, but remember that topics and question types may vary. Creating your own questions based on our study guide skills is recommended—students who did so in the past scored higher than those who did not.

Evening prelims are 90 minutes in duration and are held at 5:30pm and 7:30pm to accommodate conflicts; your assigned time will be announced at least three weeks prior to the exam. If you have a conflict with your assigned time, you must respond to the exam conflict survey on Canvas at least two weeks before the exam date. If you have SDS accommodations related to exams, the SDS office will coordinate separate exam details with you.

The [university registrar](#) coordinates all evening prelims; any conflicting exam must appear on their schedule. If you have a severe illness or unavoidable approved conflict during the exam, an alternative assessment will be offered. A single makeup exam will be scheduled that will count for half of the weight of the original exam, with the remaining weight allocated to the final. If you cannot attend the scheduled makeup for an approved reason, then all of the weight will be shifted to the final. For last-minute emergencies, students must contact our course coordinator ASAP (in the case of illness, you must determine your viability of taking the exam by 4pm on the exam day). If a makeup is offered for the final exam, it will take place *after* the scheduled exam. If you are

unable to take the final exam for reasons *beyond your control*, your grade will be ~~INC~~ and you will need to take the exam in a future semester.

Please do not make plans to leave Ithaca before the end of the finals period until the finals schedule has been announced by the registrar—buying plane tickets too early is not considered “beyond your control.”

Regrades

Graded homework will be available on CMSX, and graded exams will be available on Gradescope. If you identify an error in how the rubric was applied to your assignment, you may submit a regrade request during the allowed window (typically within 1 week of an assignment or exam being returned). Errors not identified during the request window will not be corrected.

Regrade requests are submitted electronically in both CMSX and Gradescope. When submitting a request, explain precisely why the deduction in question does not apply to your solution. When evaluating a request, staff will review the entire program/question for grading accuracy and consistency, so it is possible for your score to decrease if errors in your favor are identified. If major systematic grading problems are identified during the regrade window, we will attempt to address them for all students.

For programming assignments, if you received a very low score from the autograder due to compilation/execution issues or a bug in a central piece of code, you may resubmit during the regrade window to address these problems for a cost of 10 points. An additional 15 point deduction applies if your original submission was incomplete and manual re-evaluation is required.

Community of learning and professionalism

We aim to create an inclusive learning environment where diversity and individual differences are respected and appreciated, and we expect students in this class to demonstrate diligence in understanding how others’ perspectives may be different from their own. Behaviors that contribute positively to our community of learning include:

- Recognize that everyone is starting from different bases of knowledge. Be respectful and constructive when pointing out mistakes.
- Listen to one another and, especially during group work, actively encourage everyone to contribute.
- Help build a lively and active online learning environment. Ask *and answer* questions on our discussion board, always remembering to be respectful and constructive.

Academic integrity

Integrity is a cornerstone of both our learning community and professional life; it is about respecting yourself and respecting others. You respect yourself by submitting work completed through your own effort; you respect others by acknowledging contributions from them when

collaboration is allowed (e.g., group projects). When your individual effort is required (such as on an exam), you may neither seek nor accept help from others. Always abide by Cornell's [Code of Academic Integrity](#) and any course-specific rules regarding its interpretation. We adopt the [CS department academic integrity policy](#) and the [CS 1110 academic integrity policy](#) in this course.

During lecture, you may only respond to polling questions using your own iClicker remote; it is a violation of academic integrity to respond on behalf of other students not in attendance.

On any programming assignment, it is a violation of academic integrity to:

1. Look at or be in possession of the code of another group in this semester or a previous one with a similar assignment.
2. Show or give your code to another student not in your group.
3. Post code on any communication platform (including Q&A sites and public posts on Ed Discussion) that other students can see.

You may discuss assignments with others at a high level, but the discussion should not extend to writing actual code, picking variable names, agreeing on specifications or comments, etc. If someone else contributes a key idea affecting your program design, you must credit them in a code comment, clearly specifying the scope of their contribution.

If you do an assignment with a partner, you must work together as much as possible. It is a violation of academic integrity to submit as a group if both members cannot claim joint authorship of all portions of the submission.

As large language models like ChatGPT become more prevalent, pay careful attention to the section on Automated Help. By that policy, generating submitted code with such tools is forbidden in this course. These tools may be used as a study aid during your own explorations of material (being aware that their responses are not verifiable and may be misleading or false), but they may not be used in the process of completing coursework.

If we suspect that the Code of Academic Integrity is not being upheld, we may upload student submissions to 3rd-party services that detect plagiarism; enrollment in this course implies consent for your submissions to be used in this manner.

Online discussion board

We will be using [Ed Discussion](#) as an online discussion forum. You are encouraged to post any questions you might have about the course material (after searching to see if they have already been asked). Course staff will monitor the forum daily, but If you know the answer to a question, you are encouraged to post it (though please avoid giving away hints on the homework or posting any part of a solution).

By default, your posts are visible to the course staff and other students, and you should prefer this mode so that others can benefit from your question and the answer. However, you can post privately so that only the course staff can see your question, and you should do so if your post might reveal information about a solution to a homework problem. If you post privately, we reserve the right to make your post public if we think the class will benefit. You can also post anonymously if you wish not to reveal your identity.

If you can phrase a precise question about where you are confused, or if you are curious about how a concept is generalized or applied, a public Ed post is a great way to get an answer, and reading and answering others' posts is a great way to form connections among the material. Ed is also a good place for soliciting general computer troubleshooting advice or recommendations. We would prefer that students use Ed for all course-related chat rather than external services, as it engages the whole class and is moderated by trained staff.

Ed is not ideal for individualized debugging of assignments, however, as asynchronous iteration is tedious, there is no guarantee of a timely response, and your code cannot be shared publicly. Prefer consulting hours for individualized assistance (discussing general debugging *strategies*, on the other hand, is an excellent discussion board topic). Ed is also not the place to request personal accommodations; in general, if a consultant could not possibly address your situation, it is probably best directed to an instructor via e-mail (after reviewing the course policies).

Accommodations

It is important that everyone enrolled in CS 2110 has access to and can participate in the course to the best of their abilities. [Student Disability Services](#) (SDS) manages accommodations for students with special needs in this regard. If you think you may have a disability that affects your ability to access this course, we encourage you to contact SDS for a confidential consultation.

In order to have adequate time to arrange your approved accommodation(s), you must [request your SDS accommodation letter](#) **no later than the “add” deadline for the semester** (September 5).

- **Students currently registered with SDS:** Once you request your accommodation letter and it is approved by SDS, it will be emailed to you, the instructor, and the course coordinator. Processing time can be up to 48-hours.
- **Students not registered with SDS:** The registration process for new accommodations can take up to three weeks. Once you are approved by SDS for accommodations, you will be able to request your accommodation letter for this course.
- **If you are approved for accommodations later in the semester:** you must request your accommodation letter as soon as possible.

When your SDS letter is sent to us, we will reply with a summary of how we intend to implement your accommodations. For the most common accommodations approved by SDS, here is what you

can expect:

Testing accommodations

This course will be participating in the **Alternative Testing Program (ATP)**. All exams will be centrally managed and supported by the ATP Testing Coordinator in the Office of Student Disability Services and accessible in your SDS student portal. ATP support includes:

Scheduling of accommodated exams

Evening prelim exams will begin promptly at 6:30 p.m.

Notification of accommodated exam logistics

All exam logistics are managed by the ATP and will ONLY be communicated to you via email from sds-testing@cornell.edu and accessible in your SDS student portal. **Please do not contact course staff with questions about SDS exam logistics, as we will not be able to answer them.**

- *10-days prior to the exam date:* ATP will automatically send an email with the exam date, time, location.
- *48-hours prior to the exam date:* ATP will send a reminder email about the exam.

Coordination of make-up exams (i.e., for students who have been granted prior permission to take the exam on a day other than the scheduled date of the main exam) will be handled by the course coordinator. The ATP will not be involved in the logistics for any make-up exams. If you miss your scheduled accommodated exam, you should notify the course coordinator, not the ATP.

Extended test-taking time

In addition to the above policies regarding exams, your extension will also be applied to time limits for quizzes (as enforced by Canvas).

Computer use for exams, if needed

If you need to use a computer in order to respond to exams, you must reserve a laptop from SDS at least one week before the exam (you may not use a personal laptop). Other exam logistics will be managed by the SDS Alternative Testing Program as described above. Note that our exams do not entail long essay responses and may require drawing diagrams, so handwritten responses may be viable and preferable for some students.

Temporary accommodations (e.g. COVID)

Please forward your temporary accommodation letter to our [course coordinator](#) for our records, but there is no need to correspond directly with the instructors unless your situation is not covered by our default policies. For example, you do not need to ask us about missing class or using a slip day, but you should inform us if you are unable to take an exam (so we will know to expect you at the makeup exam). You are responsible for catching up on missed material (e.g. by watching lecture recordings or reviewing a classmate's lecture notes).

Joining the class late

Students who join the class between August 22 and September 5 (including those who transfer from CS 2112) must contact their TA ASAP for procedures on how to make up missed work. In general, the following policies apply:

- Assignment A1 must be completed. For students joining *after* the CMS late deadline for A1, an extension will be provided. Students joining *before* the CMS late deadline should instead use slip days. Or, if you completed A1 for CS 2112, then notify the CS 2110 professors and your score on the CS 2112 A1 will be transferred instead.
- The syllabus quiz must be completed on Canvas. If you joined during the second week of classes, please email our course coordinator to ask for an extension.
- iClicker responses do not count during the “add” period, so no lecture participation credit has been lost. You are responsible for the material covered in missed lectures, however, so be sure to complete assigned readings and review notes with a classmate.
- Discussion activities cannot be made up, but because of our attendance policy, you may still earn full participation credit by submitting all remaining activities. You should still review the previous worksheets and complete them for practice.
- Quizzes (other than the syllabus quiz) cannot be made up, but because two quizzes are dropped, it is still possible to earn full credit for quizzes by completing all remaining ones.

Copyright

All materials distributed in this course are copyrighted and may not be distributed further (unless otherwise indicated). They are intended for your sole use and may not be reposted on any public or private website or collected in an offline archive. Public availability does not imply permission to redistribute, and materials on CMSX or Canvas are not public to begin with.

You hold the copyright on original work you create for this course. But note that assignments include significant “skeleton” code authored by the course staff and licensed for use only in the context of this class. Therefore, it is generally not permissible to share code for completed assignments. If you want to showcase your new skills in a portfolio, be creative and apply them in a novel setting.
