

CS1301: Introduction to Computing (Online) (FA26)

Course syllabus · Georgia Institute of Technology · Canvas course 539830

Quick Links



Course Content

Course lecture content, exercises, and problem sets are all driven by Canvas.



Syllabus

Find the course calendar, assignment breakdown, and course policies in the course syllabus.



Tests

The six Programming Quizzes are launched from the Honorlock page in Canvas, which directs you to the assignment launch in Canvas and then opens the quiz in Vocareum.



Exam Sign-Up

Reserve your seat for the two in-person exams using the sign-up forms linked on the Exam Sign-Up page.



Forum

Join the forum to discuss with classmates, ask questions, and reply to important announcements.

Welcome!

Hi! I'm David Joyner, and along with Melinda McDaniel, welcome to the online version of CS1301—and by "the online version", I mean the version that we built to be online even before the pandemic.

This syllabus page will walk you through everything you need to know to succeed in this course. However, I know there's a lot of information on this page to process, and it can be hard to know what you're really expected to do. So, here's my recommendation: jump on over to [Modules](#) and get started! It'll welcome you to the course, present to you this syllabus, then walk you through the start-of-course survey, onboarding quiz, and launch you into the content.

While everything in this syllabus is important and binding, it also makes the course sound more complex than it is. In simple terms, for this course, you'll:

1. Complete some assignments each week, all launched from [Modules](#). These assignments *include* the instructional material you'll learn from; they're embedded together.
2. Complete a digitally-proctored Programming Quiz in weeks 4, 6, 8, 11, 13, and 15.
3. Come to campus for an in-person test in weeks 10 and 16.
4. Complete four extra credit options by the end of the semester.

That's all! From there, if you're ever lost or confused, just make sure you're reading the weekly start-of-week announcement each Monday, which tells you everything you need to do for the course. As long as you read that

weekly announcement and do everything it says to do, you won't overlook anything.

Immediate Attention Required

All that said, two things deserve your attention **right now** before anything else:

First, this class has **six digitally proctored programming quizzes**. The digital proctoring tool requires an **external webcam** because you'll be required to provide a side-angle view of your test-taking area. If your only webcam is the one built into your laptop, you need to obtain an external webcam before the first quiz, which is due **September 4th**. It doesn't need to be a nice camera—any \$20 camera from Amazon will likely work, although depending on your workspace we also recommend one with a longer cord or an extension cord (at least six feet) to get a good view.

Second, this course includes **two required in-person exams**, during the weeks of **October 26** and **December 7**. Everything else in this class is online and flexible, but those two **are not**: they are taken in person, on campus. You choose your own 90-minute block from a large range of options (including both daytime and evening slots), but you must sign up by the Monday two weeks before, or a block will be assigned to you. Put both weeks in your calendar today, and read the [In-Person Exams](#) section carefully.

Teaching Team

This course is taught by two instructors and supported by a team of teaching assistants led by four Head TAs. A note on how to reach us before you read any further: **for anything about the course, post on Ed Discussion rather than emailing us directly**. The forum is monitored by the whole team, so you will get a faster answer there than in any individual inbox. The email addresses below are listed for situations that genuinely require reaching one person privately—see [Communicating with Instructors and TAs](#) for the details.

Instructors



Dr. David Joyner
Instructor

david.joyner@gatech.edu



Dr. Melinda McDaniel
Instructor

mcdaniel@cc.gatech.edu

Head Teaching Assistants

The Head TAs coordinate the teaching assistant team, run the help desk schedule, and handle most day-to-day questions on the forum. In practice they are the people you will interact with most.



Pranav Bhagam
Head TA
bhagam@gatech.edu



Benito Karkada
Head TA
bkarkada3@gatech.edu



Aparna Jagannathan
Head TA
ajagannathan3@gatech.edu



Bickston Laenger
Head TA
bickston@gatech.edu

The full roster of teaching assistants, along with the help desk schedule, will be posted to the course forum within the first two weeks of the semester.

Course Platforms

This course uses two main platforms for delivery: Canvas and Vocareum.



Announcements

Syllabus

Course Surveys

Quizzes

Tests

Grades



Lecture Content

Exercises

Problem Sets

Notebook Assignments

You will spend the overwhelming majority of your time in this course in two places: Canvas and Vocareum. **Canvas is your home base and your to-do list.** Navigate the course through the [Modules tab](#). **Vocareum is where you actually learn and write code.** The course videos, the exercises embedded throughout each chapter, and the coding problem sets all live here. Canvas will hand you off to Vocareum throughout the modules.

Beyond those two, we have three other platforms we use in this course. You'll visit these far less often, but you should know what each one is for:



[Ed Discussion](#) is our course forum, where you can ask questions to be answered by the teaching team.



Weekly recitations and the help desk meet via [Zoom](#), which you can access from within Canvas.



The six Programming Quizzes are proctored via [Honorlock](#). You'll launch these from the Honorlock page in Canvas.

Honorlock-proctored assessments require a webcam, microphone, and photo ID (driver's license, school ID, passport, etc.) and must be taken via Google Chrome (regardless of operating system). Note that graduate-level seminar students do **not** take the proctored assessments.

Readiness Questions

This class has no strong prerequisites in terms of prior knowledge. To succeed, no math beyond arithmetic is necessary, and all course materials are presented in English. However, the class will be rigorous and demand persistence and self-discipline.

To succeed in this course, you should be able to answer **yes** to the following questions:

- Are you prepared to manage your time in consuming this class material and completing its assignments, including watching lectures and starting assignments well in advance of their deadlines?
- Are you prepared to start approaching a field that requires significantly different ideas and skills than any you've approached before?
- Do you have a growth mindset, the belief that your abilities can be improved through dedication and hard work?
- Do you view failure as an opportunity to learn, rather than a referendum on your own ability?

We recommend a pace akin to a traditional lecture class: dedicate a couple hours a day every day to watching lecture material, completing exercises, and getting started on problems. In computing, it's common to encounter challenges that require you to step back and give your mind a day to think over them, but you can only do that if you start early!

Course Description & Structure

This course is an introductory course in computing. It does not presuppose any advanced knowledge of computer science, and requires no more background in math than basic arithmetic. The class covers the

foundations of computing, starting with the fundamental idea of computer programs, the basics of variables and operators, the control structures for directing program flow, and the data structures for advanced computing.

This course has three high-level learning objectives. At the conclusion of this course, you will be able to:

- Describe and apply the fundamental thought processes and theories of computing.
- Write command-line computer programs using fundamental principles of programming.
- Apply those abilities to a real target domain.

This course is broadly broken into six modules. Each module has its own Programming Quiz.

- **Module 1: Computing & Procedural Programming** (Chapters 1.1–1.3, 2.1–2.4). We'll cover the basics of computing—what it means to write computer code, how to read it, and what it means to run and debug it—and then the basic paradigm of programming, procedural programming, in which a series of commands are executed in order. Here we'll discuss variables, logical operators, and mathematical operators.
- **Module 2: Conditionals & Loops** (Chapters 3.1–3.3). We begin control structures: lines of code that control when other lines of code run. Conditionals let your program choose between alternatives, and loops let it repeat work.
- **Module 3: Functions & Error Handling** (Chapters 3.4–3.5). We finish control structures with functions, which let you package and reuse code, and error handling, which lets your program respond gracefully when something goes wrong.
- **Module 4: Strings & Lists** (Chapters 4.1–4.3). We begin data structures: how data is structured to be operated upon by a computer. Strings and lists are the two you'll use most often.
- **Module 5: File I/O & Dictionaries** (Chapters 4.4–4.5). We continue with file input and output, which lets your programs read and write data that outlives a single run, and dictionaries, which associate keys with values.
- **Module 6: Objects, Algorithms & AI** (Chapters 5.1–5.3). We preview the next topics in computing: object-oriented programming, computer algorithms, and artificial intelligence, covering what modern AI systems actually do, how they are built out of the programming constructs you've learned, and how to reason about their output as a programmer.

Course Assessments

Your grade in this course is derived from the following categories. From Canvas, anything listed under Modules with points attached counts towards your Module Content grade.

Final letter grades are assigned on a standard scale: a final average of **90 or above receives an A, 80 or above a B, 70 or above a C, 60 or above a D, and below 60 an F.**

Please note for dual enrollment students: we are not permitted to contact schools with any grade besides the officially-reported grade recorded by Georgia Tech. Your school will be told if you received a final letter grade of A, B, C, D, or F, but will not be given any extra information about how high or low your grade within that range is.

Category	Count	Each	Total
Module Content	—	—	50%
Programming Quizzes (online, Honorlock)	6, lowest dropped	6%	30%

In-Person Exams	2	10%	20%
<i>Extra Credit</i>	<i>4 activities</i>	<i>1–2 points</i>	<i>up to +5%</i>
Total			100% (+5% extra credit)

Content: 50%

Essentially everything present in the Modules tab counts toward your overall Module Content grade, which is worth 50% of your total average. Our expectation is that you'll earn 100% of these points: everything here allows unlimited attempts until its deadline, so as long as you keep working at it before Friday, full credit is entirely within your control.

Generally speaking, you'll find the following types of assignments under Module Content. All Module Content can be attempted unlimited times until the deadline until you earn full credit.

- **Exercises:** Most course content is presented via course chapters. Each chapter has multiple embedded exercises interspersed among the lecture videos. Some of these are multiple choice and fill-in-the-blank exercises, while others are programming exercises.
- **Notebooks:** There are a handful of Notebook assignments, a different kind of content.
- **Problem Sets:** Coding problem sets are sets of programming problems provided through Vocareum.
- **Surveys:** There are four course surveys: the Start-of-Course Survey, Quarter-Course Survey, Mid-Course Survey, and End-of-Course Survey. Completing these automatically earns their points.
- **Onboarding Quiz:** The Programming Quizzes in this class are proctored via Honorlock, and so we've provided an Onboarding Quiz so you can practice with these tools before the first graded quiz. Completing this counts towards your Content grade.
- **Syllabus Comprehension Quiz:** At the start of the course, you'll be asked to complete the Syllabus Comprehension Quiz/Survey to indicate you understand the contents of this syllabus.
- **Additional Other Assignments As Assigned:** Other assignments may be announced during the semester that will also go into your Content grade. These will always be announced at least two weeks before they are due.

For every type of content in Module Content, you'll see how many points it's worth listed below the assignment title:

For example, here you'll see that the 2.1 Exercises are worth 10 points, the Working with Vocareum Notebooks assignment is worth 3 points, the Chapter 2.2 Exercises are worth 41 points, the Chapter 2.2 Problem Set is worth 66.5 points. These points all count toward your Module Content grade.

The different number of points for each assignment are generally connected to the amount of work required, not the importance. For example, the 2.2 Problem Set is expected to take a couple of hours to complete, while the Working with Vocareum Notebooks assignment is probably only a few minutes of work. Again, our expectation is that you'll earn 100% of the Module Content points.

Programming Quizzes: 30%

There will be six Programming Quizzes, one for each of the six modules: Programming Quiz 1 covers Module 1, Programming Quiz 2 covers Module 2, and so on. Each quiz focuses on the material for its module, but may also require knowledge gained in earlier modules, because CS builds on itself going forward.

Each quiz is worth 6% of your average, and **your lowest Programming Quiz score is dropped**. Six quizzes at 6% each with the lowest dropped means five scores count, for 30% of your final average. The drop is automatic; you do not need to request it, and it is applied at the end of the semester once all six scores are recorded.

The Programming Quizzes feature the same kinds of programming exercises and conceptual questions you have seen in the lessons and problem sets, but they are timed and digitally proctored. You will still have access to the programming interface from the exercises, so you can run your code live and receive immediate feedback on whether it was successful. To start a quiz, navigate to Honorlock in Canvas and select the desired quiz; Honorlock will then guide you through setting up side-angle proctoring, then direct you to the assignment launch in Canvas, which opens the quiz in Vocareum just like any other Vocareum assignment.

Programming Quizzes are closed: **no notes, no books, no scratch paper, no other web sites, and no AI assistants of any kind**. During the quiz, you may not access any resources except for the quiz itself.

The regular quiz window runs from **Monday at 9:00 AM ET to Thursday at 4:00 PM ET** of the week in which the quiz is listed on the calendar. Thursday at 4:00 PM is the **latest you may start**, not the latest you may finish: you get your full **45 minutes** once you begin, even if that runs past 4:00 PM.

Additionally, every Programming Quiz has a **grace period** that extends from the Thursday 4:00 PM cutoff to **Friday at 4:00 PM ET**, which is the last moment you may *start* a quiz. The grace period is for emergencies or students who encountered technical issues during their original attempt. Technical support will **not** be given during the grace period except in the case of students who attempted the quiz and encountered problems **before** the grace period. Quizzes started during the grace period earn full credit, with the full 45 minutes, and there is no late penalty; you do not need to request permission to submit during the grace period, but you use it **at your own risk** if you did not have an emergency or technical issue during the original window.

In-Person Exams: 20%

This course includes **two required in-person exams**, each worth 10% of your final average, for 20% total. Unlike the Programming Quizzes, **neither in-person exam can be dropped**.

- **In-Person Exam 1** is given during the week of **October 26–30, 2026**. It covers Modules 1 through 3.
- **In-Person Exam 2** is given during the week of **December 7–11, 2026**. It emphasizes Modules 4 through 6 but is cumulative over the entire course, and serves as the final exam for this class.

Each exam is **90 minutes** long, is administered on **paper**, and consists entirely of **multiple choice questions**. There are no computers involved: you will not be writing or running programs, so there is nothing to compile and no partial credit to chase. Expect questions that ask you to read a snippet and predict its output, identify the bug in it, choose the correct construct for a situation, or reason about how the concepts from the modules fit together. The exam booklet and answer sheet are provided; **bring a #2 pencil**. Do not bring a laptop—you will not be able to use one.

Because these exams are on paper rather than in Vocareum, practicing only by running code is not enough preparation. You should be able to trace a program by hand and predict what it does before executing it, since on

exam day that is the only tool you will have. **The non-coding problems in each problem set—the multiple choice and fill-in-the-blank questions you answer without writing a program—are the closest match to the exam format.** When you work through a problem set, treat those questions as exam practice rather than as something to click through on the way to the coding problems.

For more information, review the [About the In-Person Exams](#) page.

Extra Credit: 5%

You can earn up to 5 extra credit points on your final average in this class. While these points are extra credit in the sense that they raise the maximum number of total points to 105, we hope everyone completes them. All extra credit is due **December 11, 2026**.

To summarize, they are:

- **Extra Credit 1 (1 point): Complete a MOOC.** Complete a MOOC available via Coursera, edX, or another platform. We offer one for free, or you can choose to earn a verified certificate in a different one. You may find more information or submit your proof of certificate [on the Assignment page](#).
- **Extra Credit 2 (1 point): Read a Book.** There are a handful of books we highly recommend. You can earn one extra credit point by reading one and earning at least a 90% on a quiz available via Canvas. You may find available book quizzes and their instructions [on the Assignments page](#).
- **Extra Credit 3 (1 point): Complete a Coding Activity.** There are a number of different additional coding-based challenges we offer for one additional extra credit point. More details will be made available during the semester.
- **Extra Credit 4 (2 points): Final Problem Set.** At the end of the course in Vocareum, there is a Final Problem Set you may complete for up to 2 extra credit points. It consists of a number of problems significantly more challenging than those in the rest of the course.

You may earn up to 5 extra credit points if you complete all activities. Note that Canvas is set to not factor in extra credit until the **end of the semester**; that's because Canvas's math is such that it appears as if *not* doing the extra credit hurts you until you complete all the other assessments. We'll set extra credit to factor into your average at the end of the term, but in the meantime, you may just mentally add the extra points you earn to your average.

Course Calendar

This course runs 17 weeks, from **August 24, 2026** through **December 18, 2026**. The calendar below provides the content we expect you to work on each week.

Release Schedule

Although all the course material (except the tests) is available on Day 1, we make periodic changes and improvements to the course. The week in which a chapter is listed in the calendar is the date by which you may assume the material for that chapter is finalized, and any additions after the start of the semester will be announced. For example, Chapter 2.2 is listed under Week 2: this means that on Monday of Week 2, you may assume that Chapter 2.2 has been finalized.

You are encouraged to work ahead, however we may make changes to chapters until their official release date. Generally, these changes will be adding new content or exercises, so you wouldn't risk completing something

that would later be removed. If you do work ahead, you should just make sure to check back and see if your score remains where you expect it to be after the release date has passed.

Deadlines

Each chapter is due on Friday of the week on which it appears on the calendar. Every chapter is finalized on the Monday of the week in which it appears and is due that same Friday at 11:59 PM ET. For example, Chapter 2.2 appears in Week 2 and is due on the Friday of Week 2. This includes the chapter exercises *and* the problem set: the exercises embedded in Chapter 2.3, and the Chapter 2.3 Problem Set, are both due the Friday of the week Chapter 2.3 appears.

Programming Quizzes

Programming Quizzes open on Monday at 9:00 AM ET in the week in which they are listed, and **Thursday at 4:00 PM ET** is the last time you may start one, with a grace period allowing you to start as late as **Friday at 4:00 PM ET**. You get the full 45 minutes from whenever you begin, even if that carries past the cutoff.

In-Person Exams

The two in-person exams occupy Weeks 10 and 16 and are not tied to the online quiz window described above; you attend, on campus, the 90-minute block you reserved on that exam's sign-up form. Sign-up opens on Monday, August 31, 2026 (the Monday of Week 2) and closes at 11:59 PM ET on the Monday two weeks before each exam week. Both exam weeks are marked in bold on the calendar below. There is no grace period of any kind on an in-person exam.

Don't Procrastinate

This course is intentionally flexible *within* each week to give you room to work around other classes' schedules, work, extracurricular activities, or personal issues. What it does not give you is extra time after a Friday deadline has passed. If you wait until Thursday night to start a week's chapters and then hit a bug you cannot solve, there is no cushion behind you—so start each week's work early enough that you can still get help before Friday. We recommend working ahead, especially early in the semester, to give yourself leeway to get stuck and struggle later.

School Breaks

You'll notice our calendar largely doesn't observe school breaks (fall break, spring break, etc.). This is because this class comprises both Georgia Tech students and Dual Enrollment students, many of whom have different school breaks. For this reason, we don't bother incorporating breaks into the recommended schedule. Because every deadline is firm, the way to handle a break that falls in the middle of the term is to **work ahead before it**. If you have an official conflict with the entirety of the time a Programming Quiz is open, let us know and we may be able to open it early.

Two exceptions. First, **the two in-person exam weeks are fixed**. Working ahead does not let you skip them, and unlike the online quizzes they cannot be opened early for you. Check October 26–30 and December 7–11 against your travel plans now.

Second, **Thanksgiving**. Week 14 is the week of November 23, and the Georgia Tech Student Recess runs November 25–27. The **Chapter 5.2 deadline is moved up to Wednesday, November 25**, so no course content

is due over the holiday itself. That is a shorter week than usual—three days instead of five—so start Chapter 5.2 on the Monday. No Programming Quiz falls in this week.

Course Calendar

All deadlines are Fridays at 4:00PM ET **except** during Week 1, when the deadline is extended to Sunday to accommodate students who register late in the week.

Week #	Week Of	Module	Chapters	Other Assignments	Deadline
1	08/24/2026	1	1.1, 1.2, 1.3	Start-of-Course Survey Syllabus Comprehension Quiz/Survey	08/30/2026*
2	08/31/2026	1	2.1, 2.2	Onboarding Quiz	09/04/2026
3	09/07/2026	1	2.3, 2.4		09/11/2026
4	09/14/2026	2	3.1, 3.2	Programming Quiz 1	09/18/2026
5	09/21/2026	2	3.3		09/25/2026
6	09/28/2026	3	3.4	Programming Quiz 2 Quarter-Course Survey	10/02/2026
7	10/05/2026	3	3.5	In-Person Exam 1 Sign-Up	10/09/2026
8	10/12/2026	4	4.1	Programming Quiz 3	10/16/2026
9	10/19/2026	4	4.2		10/23/2026
10	10/26/2026	4	4.3	In-Person Exam 1	10/30/2026
11	11/02/2026	5	4.4	Programming Quiz 4 Mid-Course Survey	11/06/2026
12	11/09/2026	5	4.5		11/13/2026
13	11/16/2026	6	5.1	Programming Quiz 5 In-Person Exam 2 Sign-Up	11/20/2026
14	11/23/2026	6	5.2		11/25/2026
15	11/30/2026	6	5.3	Programming Quiz 6	12/04/2026
16	12/07/2026	—		In-Person Exam 2	12/11/2026
17	12/14/2026	—		End-of-Course Survey	12/17/2026

Course Policies

The following policies are binding for this course.

Official Course Communication

In terms of course administration, you are responsible for knowing the following information:

1. This entire syllabus.

2. Anything emailed directly to you by the teaching team (including announcements via the course forum or Canvas), 24 hours after receiving such an email.

Generally speaking, we will post announcements via Canvas and cross-post their content to the course forum; you should thus ensure that your Canvas settings are such that you receive these announcements promptly, ideally via email (in addition to other mechanisms if you'd like). If we need to contact you individually, we will do so via email. Georgia Tech generally recommends students to check their Georgia Tech email once every 24 hours. So, if an announcement or message is time sensitive, you will not be responsible for the contents of the announcement until 24 hours after it has been sent.

Note that this means you won't be responsible for knowing information communicated in several other methods we'll be using. You aren't responsible for knowing everything posted to the course forum that isn't linked from an official announcement. You aren't responsible for anything said in recitation or the help desk or any other third-party sites we may sometimes use to communicate with students. You don't need to worry about missing critical information so long as you keep up with your email and understand the documents on this syllabus.

Communicating with Instructors and TAs

Communication with the course teaching team should be handled via the discussion forum. If your question is relevant to the entire class, you should ask it publicly; if your question is specific to you, such as a question about your specific grade or submission, you should ask it privately. Please refrain from emailing the instructor or teaching team unless you are sharing information that should only be seen by the instructor; you will generally get a more rapid response by posting privately to the forum and having the TAs either answer your question or quickly alert the instructor that a question needs to be answered.

Our workflow is to regularly filter the forum for Unresolved posts, which includes top-level threads with no answer accepted by the original poster, as well as mega-threads with unresolved follow-ups. If your question requires an official answer or follow-up from an instructor or teaching assistant, make sure that it is posted as either a Question or as a follow-up to a mega-thread, and that it is marked Unresolved. Once an instructor or TA has answered your question, it will automatically be marked as Resolved; if you require further assistance, you are welcome to add a follow-up, but make sure to unmark the question as Resolved in order to make sure that it is seen by a member of the teaching team.

One category of post is worth calling out before you write it: **please do not post or email asking for a deadline extension.** We cannot grant one without an approved letter from the Dean of Students, so these posts cannot be answered with anything except a pointer to the Late Work policy below. Go to the Dean of Students first; see [Extensions](#).

Similarly, in order to keep the forum organized, please post as a Post or Note instead of a Question if your question does not require an official response from the teaching team. For example, if you are interested in getting multiple perspective from classmates, getting feedback on your ideas, or having a discussion that does not have a single answer, please use Post or Note instead of Question. Please reserve Question threads for questions that will likely have a single official response. TAs and instructors will regularly convert Questions to Posts or Notes that do not need a single official answer, but it will save time and allow them to focus their attention on other students if you correctly categorize your post in the first place.

Late Work

Although this class is online and asynchronous, its deadlines and schedule should be treated just as an in-person course. If an excuse or emergency would not be significant enough to miss an in-person class session or test, then it is not sufficient to request an extension on this class's work.

Therefore, late work **is not accepted** without documented approval from the Dean of Students.

The Dean of Students is there to be an advocate and partner for you when you're in a crisis; we wholeheartedly recommend taking advantage of this resource if you are in need. Justifiable excuses here would involve any major unforeseen disruption to your classwork, such as illnesses, injuries, deaths, and births, all for either you or your family. Additionally, the Dean of Students office can coordinate with you and alert all your classes together instead of requiring you to contact each professor individually. You may find information on contacting the Dean of Students with regard to personal emergencies here: <https://studentlife.gatech.edu/request-assistance>

You are responsible for ensuring that your assignments are submitted and that credit has been granted. All exercises, problem sets, surveys, and Programming Quizzes are graded automatically, so you may confirm your grade has appeared before the deadline.

We understand this policy appears restrictive for an online asynchronous course, but it serves two primary purposes. First, we have observed over several years that it is far harder to catch up than to stay on pace in the first place, so the policy is written to only permit falling behind when it is absolutely necessary. Second, research has shown there are wide disparities in individuals willing to ask for extensions and under what circumstances; restricting extensions only to documented emergencies prevents students with a higher willingness to request special accommodation from having a different course experience from those who do not recognize that as an option.

Note that this policy does not apply to pre-approved extensions, such as extensions for Office of Disability Services accommodations and Institute Approved Absences.

Academic Honesty & AI Collaboration Policy

Every Student is expected to read, understand and abide by the [Georgia Tech Academic Honor Code](#). Academic misconduct is taken very seriously in this class.

Exercises and problem sets are collaborative. You are encouraged to work with your classmates to talk through problems, debug code, explain a concept, etc. However, the work submitted for a grade must be your own. You should not copy answers directly from other students, nor should you copy any code directly from other students.

Similarly, you should not send any code you write to your classmates. You can show it off either side-by-side or via teleconference, but it should not be shared in its copyable text format. When asking questions about your code on the course forum, we ask that you post screenshots of your code along with its output rather than copying the code itself. This makes it slightly more difficult to directly borrow from someone else's work, as well as often helps us answer your question by seeing its output and referencing specific line numbers.

Additionally, you should not copy/paste questions from this course to any third-party web sites, such as Stack Overflow. You are welcome to use these resources for additional support, but when doing so, you should have already attempted the problem and be able to explain what your goal and issue are in your *own* words. You should not copy the entire problem description and ask, "Can anyone help with this?" Instead, you should attempt the problem and be prepared to ask a question like, "I'm trying to reverse this string, but right now the

string never seems to get changed. What's wrong? Here's my current code:" Not only is this restriction necessary to maintain academic honesty, but it is also good programming practice: you will often find that in the process of explaining where you're stuck, you solve the issue yourself.

These policies apply straightforwardly to collaboration with AI agents such as ChatGPT and Github Copilot as well. We encourage use of these tools as assistants when working on exercises and problem sets, but you should make sure you are using them as *assistants*. You should learn from these tools, then let that improved understanding show on the problems you complete. You should never send the entirety of a problem description to an AI assistant, nor should you copy the response you receive back into your own submission. Instead, you should attempt problems yourself, ask AI assistants for help if you are stuck, learn from their responses, and recreate those responses in your own work. You can ensure you are using AI assistance effectively by following these two heuristics: (a) never Copy (literally, Ctrl/Cmd+C copy) anything you receive from an AI assistant, and (b) do not have your own code open while interacting with the assistants. If you follow these two heuristics, you guarantee that any assistance you receive will contribute to your own understanding, which you can then demonstrate on the assignments.

Note that Module 6 includes a chapter on artificial intelligence. Studying how these systems work does not change the policy above or below: the AI chapter is content to be learned, not a license to use AI assistants differently on graded work.

No collaboration is permitted during Programming Quizzes or in-person exams. You may not interact with anyone during an assessment: not in person, not via phone, not via the computer. During an assessment, you should only access the quiz or exam itself; no other resources, including AI assistants. Honorlock will automatically flag any violations on the Programming Quizzes, and the in-person exams are proctored in the room.

Because the in-person exams are administered in multiple blocks across a week, students will take them at different times. **Discussing the contents of an in-person exam with anyone who has not yet taken it is a violation of the Honor Code**, and that includes describing the topics, the number of questions, or how hard it was. This applies to the course forum, group chats, conversations in the hallway outside the exam room, and messages to classmates at other schools. If you are asked, the correct answer is "I can't talk about it." Exam versions vary across blocks, and we do compare submissions across the week.

To summarize:

- You are permitted and encouraged to seek help from others on course exercises and problem sets, but you may not copy anyone else's code into your own submission.
- You are permitted and encouraged to help your friends and classmates, but you may not send them your actual code in an easily copyable format.
- You are permitted to seek additional support from third-party services like Stack Overflow and other sites, but you may not copy the actual assignment instructions into your question; instead, you must present your request in your own words.
- During Programming Quizzes and in-person exams, you may not consult any other resources or interact with any other people via any mechanism.
- You may not discuss the contents of an in-person exam with anyone until the entire exam week has concluded.

Core IMPACTS

This is a Core IMPACTS course that is part of the Institutional Priority area.

Core IMPACTS refers to the core curriculum, which provides students with essential knowledge in foundational academic areas. This course will help master course content, and support students' broad academic and career goals.

This course should direct students toward a broad Orienting Question:

- How does my institution help me to navigate the world?

Completion of this course should enable students to meet the following Learning Outcome:

- Students will demonstrate the ability to think critically and solve problems related to academic priorities at their institution.

Course content, activities and exercises in this course should help students develop the following Career-Ready Competencies:

- Critical Thinking
- Teamwork
- Time Management

Disability Services & Testing Accommodations

Georgia Tech supports students with documented disabilities through the [Office of Disability Services](#). If you have a disability and need accommodations in this course, please register with ODS and provide us with your accommodation letter as early in the semester as possible.

Accommodations matter especially for the two in-person exams. If your accommodations include extended time, a reduced-distraction environment, assistive technology, or anything else that a standard 90-minute block in a shared room cannot provide, **do not submit the exam sign-up form**. Instead, **you are responsible for scheduling your exam directly with the ODS Testing Center** for a time during the exam week. Also send us a private forum post letting us know you have done so, so that we can get your exam materials to the Testing Center in time and know not to expect you in the general room.

Schedule early. Testing Center appointments are booked on a first-come, first-served basis across all of Georgia Tech, and they fill up fast—particularly in October and December, when every course is running exams at once. Do not wait until the week before. As soon as each exam week is on your calendar, book your appointment. If you find that the Testing Center has no availability during the exam week, tell us immediately by private forum post rather than waiting; the earlier we know, the more options we have.

You will need to do this separately for each of the two exams. Booking Exam 1 does not reserve anything for Exam 2.

Accommodations for the Programming Quizzes are configured in Canvas and Honorlock once your letter is on file, and apply automatically from that point forward.

Help Desk

Aside from the two in-person exams described above, this course has no mandatory in-person meeting times or locations. There are no lectures. In lieu of lectures, you are expected to complete the online material for this class.

Students may also seek additional support at the TA help desk, administered [via Zoom](#). At the TA Help Desk, TAs will be on hand to answer questions live via videochat and screenshare. A schedule for the TA help desk, along with directions for accessing it, will be posted to the course forum within the first two weeks of the semester starting. Check the Announcements in Canvas or the Help Desk folder on the forum for the schedule and details

Feedback

Although this course has now been running for several semesters, we're always experimenting with, expanding, and fixing things. As a result, there are bound to be things that go wrong. We ask your patience and support as we figure things out, and in return, we promise that we, too, will be fair and understanding, especially with anything that might impact your grade or performance in the class.

Second, we want to consistently get feedback on how we can improve and expand the course for future iterations. You can take advantage of the feedback box on the forum (especially if you want to gather input from others in the class), give us feedback on the surveys, or contact us directly via private forum messages. You can also complete [this form](#) especially to give feedback on the content, but note that if you want a response quickly, you should use the other methods as this survey is shared with public MOOC students. It's also off-site and purely anonymous. Feel free to complete it as many times as you want!

Please don't hesitate to share any thoughts, no matter how small. We want this to be a fantastic experience for everyone.