College of Computing

# Course Syllabus for CS 1371: Computing for Engineers

Summer 2026


| | | | |
|---|---|---|---|
| **Instructor:** | Idel Martinez Ramos (he/him) | **Time:** | Tu/We 5:00pm - 6:15pm |
| **E-mail:** | imartinez@gatech.edu | **Rooms:** | Love 183 and CCB 101 |


## Course Description

This course is designed to introduce you to problem-solving, designing, and analyzing algorithms with the MATLAB programming language. The course assumes no prior knowledge of programming skills. At the end of this course, students will develop a beginner's skill level for deriving algorithms, as well as become familiar with MATLAB. To achieve this, the course will use real-world data to guide students through understanding and applying it to achieve a goal, and then output or display the results in an appropriate format.

**Student Hours** If you'd like to discuss the class, programming, college in general, or have any questions, you can visit during student hours. These are scheduled on Tuesdays and Wednesdays 2:00pm to 4:30pm in the College of Computing (map), Room 203. Alternatively, you can schedule a meeting with me at your desired time.

**Teaching Assistants** We have a team of lovely TAs who will help you throughout this semester, and with whom you will likely interact online or in person. You'll find their information on Ed Discussion and Canvas.

**Important Dates:**

| | |
|---|---|
| Last day to add/drop classes | 05.22.2026 |
| Payment deadline | 4pm (EST) 05.26.2026 |
| Withdrawal and grade substitution deadline | 07.05.2026 |
| Last day of class | 07.27.2026 – 07.28.2026 |
| Final exams | 08.03.2026 – 08.06.2026 |
| Final grades available | 6pm (EST) 08.11.2026 |

**Learning Outcomes** Through active engagement and completion of the course, students will be able to:

1. Use the MATLAB programming language for solving numeric and text problems, plotting solutions, and applying numerical methods.
2. Develop problem-solving skills and apply a step-by-step process to design and test algorithmic solutions to problems.
3. Implement programming concepts (variables, comments, data types, functions, control flow) in custom MATLAB programs.

**Course Materials** There are no required materials to purchase for this course. Students are required to have access to a working computer with a functional webcam and microphone (see Computer Software and Hardware for requirements), along with a pencil for in-class work, quizzes, and exams. If you want to get started with setting up your computer, you can download and install the MATLAB Integrated Development Environment (IDE) via the MATLAB website. Otherwise, we'll be doing so on the first day of class.

## Course Organization

This course consists of in-person lectures and pre-recorded videos. Unlike other courses, our recitations will run the same way as the main course to maintain the fast pace of the summer semester. This means that classes will be a mixture of lectures, activities, and reviews to practice MATLAB and prepare for exams.

The course will be held on Tuesdays in the J. Erskine Love Manufacturing building, Room 183, from 5:00pm to 6:15pm and on Wednesdays in the College of Computing building (CCB), Room 101 from 5:00pm to 6:15pm. As a hybrid course, lectures will also be recorded and streamed on Zoom. This means that you can choose to attend class virtually. However, we strongly recommend attending in person so that we can assist you with any issues you encounter while programming in class. At times, pre-recorded lecture videos will supplement or replace the lecture, so that we can spend more time practicing in class.

This course will span 12 weeks. We will begin with an introduction to the course, covering computational problem-solving, self-regulated learning, and academic integrity. Afterward, we'll dive into coding using variables and functions, then introduce additional data types such as vectors, arrays, cell arrays, and structures, while working with concepts like masking, control flow, file editing, plotting, and numerical methods.

## Course Requirements

This course works on a 1000-point system. Assessment in this course is divided into participation (96 points), homework (154 points), quizzes (225 points), and exams (525 points). The following is the breakdown of these requirements.

### Participation

The participation grade in this class will be split up across four types:

- **MATLAB Setup** (10 points). This mini-assignment is here to ensure you were able to successfully set up the MATLAB IDE on your computer and access the homework website (MATLAB Grader, explained below).
- **Surveys** (32 points). There will be three main surveys on Canvas that you will need to complete as part of this course, which will also serve as a verification of participation for financial aid. The intro and syllabus acknowledgment open at the start of the semester, and an academic integrity survey opens three weeks into the semester. There will also be an optional exit survey for 10 bonus points.
- **Lecture Questions** (20 points). This course will have "Clicker" questions throughout the semester for in-person lectures. These can be completed in-person or online and will serve as check-ins to check if we need to review any material (so they are not graded for correctness). Each in-person lecture may contain up to three of these questions. Your final score will only count 80% of the total questions, meaning you can miss 20% of the questions to receive all 20 points.
- **Ed Discussion** (34 points). We will use Ed Discussion, a private online forum for courses, to support course organization and announcements, and to serve as a help desk. This is a place where you can ask questions, post tips or general comments, and answer your peers' questions. To obtain full points, you must meaningfully engage in a minimum of three questions, comments, or answers. More information on forum etiquette can be found in Student Conduct and Etiquette.

### Homework

We'll be using MATLAB Grader, an online website for automated code assessment, to submit homework. There will be 11 assignments, totaling 154 points. Each assignment's instructions will be provided in a PDF document (since MATLAB Grader is not the best place for formatting text). Homework is divided into small problems across levels, each worth a different number of points. This allows you to skip some problems if you get tired of one (although all must be completed to receive full points, so be sure to return to them eventually!). We break down the levels below.

- **First level problems** are designed to test your conceptual knowledge of the material, with some required coding. This is a great place to review your understanding and see if there's anything we can help clarify.
- **Second level problems** are intermediate problems that check your application of constructs and concepts. These will require a bit more planning than the first-level problems.
- **Third level problems** require you to combine multiple of the concepts we've discussed in class. This means that this level will most likely use previous topics to complete the problem. There may be an extra credit problem (worth 1 point) of the same difficulty as the other third-level problems.

To tackle the homework, we recommend reading the problem in the provided PDF, then coding, testing, and saving your solution in the MATLAB IDE. Once you have a working solution, copy & paste your code into MATLAB Grader for testing and submission. Although it may not be necessary for the first few homework assignments, as problems become more complex, the features of the IDE will become useful. You can save these in a "Homework" folder under a larger "CS 1371" folder. (We will go over this in class.)

Please note that although homework is graded automatically, it will be analyzed manually to make sure that the code you submitted is your own. If a discussion is needed, you will be requested to meet with the instructor. More details can be found in Course Policies.

**Assignment Deadlines** You will have 10 days to complete each homework. All but the first two assignments will open on Tuesdays and will be due the following Thursday at 11:59pm (EST). These times are subject to change; in such cases, I will notify you.

**Late Submissions** You will have 48-hour time credits that you can use to submit assignments late without penalty. These credits will be used up automatically whenever homework is submitted late. For example, if you submitted the first assignment 2 hours late, your time credit will decrease to 46 hours. If you submit the second assignment 18 hours late, it will further reduce the time credits to 28 hours, and so on. After the 48 hours are used up, each day that an assignment is submitted late will result in a penalty of 25% for that homework.

**Point Distribution** Each homework will be worth 14 points, where first-level problems (four total) are worth 1 point each, second-level problems (two total) are worth 2 points each, and third-level problems (two total) are worth 3 points each. Each problem consists of 3-5 automated tests, where partial credit can be awarded. The extra credit problem (from the third level) will be worth 1 point, where all tests must be passed to receive credit.

**Writing Solutions** Most homework problems can be solved using the material covered in class. In cases where a function is needed that was not covered in class, it will be noted in the assignment instructions. If you solve the problem using a different function (one not covered in class and not in the assignment instructions), you will be required to add a comment indicating (1) how you found that function (e.g., StackOverflow, the MATLAB IDE, a friend, an AI-agent, etc.) and (2) a brief description of what the function does in your own words (i.e., don't copy & paste the documentation). These will be checked manually, as it helps us identify other functions that can be covered in class and ensures that you understand the use of the function. Failing to do so will result in a 0 for that problem.

**Disallowed Functions** There will also be some homework problems that will disallow the use of certain functions or commands (as otherwise they would defeat the purpose of the exercise). These will be noted in the assignment instructions and checked by the autograder.

### Quizzes

There will be four paper quizzes in this class, which will take place in the classroom two weeks before the exams. Each quiz will be worth 75 points, totaling 225 points, since the lowest quiz grade will be dropped. The goal for these is to help you study for the exams and allow more room for improvement. As such, they will cover part of the material for the next exam, but with shorter questions and answers. Quizzes will take place in person, but will not occupy the entire lecture time. You can read additional rules for quizzes in Course Policies.

### Exams

There will be three paper midterm exams and a final exam. Each exam will be worth 175 points, totaling 525 points. The midterm exams will build upon the quizzes, involving longer answers, such as annotating code, tracing, writing code, or explaining specific programming constructs. Exams will be held in the classroom, occupying the entire lecture time. In this programming course, each midterm exam will be cumulative, with a focus on specific topics. They will be closed notes, but a reference MATLAB sheet will be provided, since the goal is not to memorize functions but rather to apply them.

The final exam will be optional and can be taken to replace a midterm grade (if the final exam grade is higher than a midterm exam grade). It will be entirely cumulative and involve solving larger problems. The final will also be taken in person, but it runs slightly differently from traditional midterms. You can read additional rules for exams in Course Policies.

**Extra Credit** After every exam, you will have the opportunity to earn 7 points of extra credit by completing a survey (totaling 21 points). This survey will consist of brief questions that ask you to reflect on your learning habits. We'll go over how this looks in class. Although this will be for your reflection, we will check for any academic integrity violations, which may result in a score of 0 for the extra credit.

### Grading Scale

Your final grade will be assigned as a letter grade according to the following scale:

**A:** $[900, 1000]$, **B:** $[800, 900)$, **C:** $[700, 800)$, **D:** $[600, 700)$, **F:** $[0, 600)$

## Course Policies

**Buzzcard Requirement** You must bring your GT ID to every quiz and exam so that it can be scanned. If your Buzzcard is not scanned, and you have not arranged another identification method with your instructor, then you will receive an automatic zero on the quiz/exam.

**Quiz and Exam Instructions** The following are rules used in this course for in-person assessments (quizzes and exams):

1. Signing and/or taking an exam indicates you are aware of and in accordance with the Academic Honor Code of Georgia Tech and the Georgia Tech Student Code of Conduct.
2. If you take the quiz/exam, we will assume you are well enough for your performance to accurately reflect your knowledge, and you will not be allowed to retake it.
3. Once you start the assessment, you are not allowed to leave the exam room and return. If you leave the room for any reason, then you must submit your exam as complete.
4. Quizzes and exams are not allowed to leave the exam room. However, you may take the reference sheet we provide (different versions will be given in every exam).
5. No other media is to be used during your quiz/exam. This includes books, notes, phones, watches, earbuds, headphones, "smart" glasses, etc. The use of any of these items will be noted as a violation and result in a zero on the quiz/exam.
6. No sunglasses are allowed. Hats, hoodies, caps, and/or scarves of any kind must not cover your ears; they must be visible at all times during the exam. If you plan on using earplugs (foam or silicone, *not* earbuds) during the exam, you must show them to the instructor for approval.
7. Quiz and exam answers must be written in English.

**Makeups** CS 1371 will not offer any makeup quizzes or exams. This is a policy of the School of Computing Instruction. If you miss a quiz, the lowest quiz grade will be dropped. If you miss an exam, you can take the final exam to replace its grade. If you miss more than one exam, then the instructor will discuss alternatives on a case-by-case basis. For homework, you can use the pool of 48-hour time credits. Under extraneous circumstances or institute-approved absences, you may request an extension for homework. The final decision regarding any exception is made at the discretion of your instructor. Events such as vacations, travel plans, weddings, graduations, work conflicts, tardiness, forgetting the exam date and time, or being unaware of an assignment are not valid excuses.

**Plagiarism and Academic Integrity** Students are expected to maintain the highest standards of academic integrity. All work submitted must be original and properly cited. Plagiarism, cheating, or any form of academic dishonesty will result in immediate consequences as outlined in the university's academic integrity policy. Every Student is expected to read, understand, and abide by the Georgia Tech Academic Honor Code.

As a programming course, discussions about course sessions, programming concepts, and algorithms are encouraged. Assignments and practice problems are collaborative, allowing students to work together by discussing problems, exploring MATLAB functions and documentation, etc. However, the work submitted by a student must be their own original work. Students should not copy or send code to peers, but high-level discussions on their solutions are permitted. Additionally, students should not copy & paste from/to other sources, such as Stack Overflow, AI agents, or other platforms, that would compromise the integrity of their work or violate the course's guidelines on original submission. Students may use these resources for support, such as further inquiring about algorithms, programming language syntax, etc., but they should be used after first attempting the problem independently. The goal of these resources should be to supplement students' learning process and promote good programming practices, not replace it.

We analyze assignment submissions with Moss, "a system for detecting software similarity," and manually analyze flagged cases. Additionally, we cross-check solutions to those generated by AI agents and search for defined "tells." Any student suspected of cheating or plagiarizing on a quiz, exam, or assignment will be reported to the Office of Student Integrity (OSI), which will investigate the incident and determine the appropriate penalty for the violation. However, in this course, you will be given a 72-hour period to come forward and admit to cheating before you are reported to OSI.

**Attendance and participation** Participating in hands-on work is essential to success in CS 1371. Because of this, you are expected (but not required to) to attend class and come prepared to each class session. Although the course lectures are recorded, having live help will be useful, as it will also help the instructor assess whether students are understanding.

**Accommodations for Students with Disabilities** If you are a student with learning needs that require special accommodation, contact the Office of Disability Services at (404) 894-2563 or https://disabilityservices.gatech.edu to make an appointment to discuss your special needs and to obtain an accommodations letter. Please also email the instructor as soon as possible to set up a time to discuss your learning needs.

## Student Conduct and Etiquette

As a hybrid course, we will meet both in person and virtually. As with any course, you must follow the Georgia Tech Student Code of Conduct when attending both the in-person and Zoom lectures. For the asynchronous portion of the class, we will mainly communicate through EdDiscussion. It's a private forum, which makes it the ideal place to share messages, ask questions, and post comments for your instructor and TAs, as well as for other students. You will be able to create private or public posts and categorize them for easier lookup, which will make it easier for the instructional team to reply as soon as possible. To make this space as efficient and welcoming as possible,

- do not publicly post your code,
- use professional etiquette when communicating with your instructor, TAs, and peers,
- show respect to others and avoid language that may come across as strong or offensive, given that language can be easily misinterpreted (especially humor and sarcasm) in written electronic communication,
- follow the language rules of the Internet: do not write using all capital letters, do not spam, etc.
- consider the privacy of others (i.e., ask permission before giving out a classmate's information),
- do not post or send any inappropriate material; do not forward virus warnings, chain letters, jokes, etc. to classmates or the instructional staff; the sharing of pornographic material is forbidden.

Please note that the instructor reserves the right to remove posts that are not collegial in nature and/or do not meet the Online Student Conduct and Etiquette guidelines listed above.

## Computer Software and Hardware

- High-speed Internet connection
- Laptop or desktop computer with a minimum of 8GB of RAM and any Intel or AMD x86-64 processor with two or more cores (Windows/Linux/Mac), or any M-series chip (Mac)
- Windows 10 or higher, macOS 14 or higher, Ubuntu 22 LTS or higher
- Complete Microsoft Office Suite or comparable and ability to view Adobe PDF documents
- A modern web browser running on Windows, Mac, or Linux (e.g., Firefox, Chrome, Safari)

## 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, and Time Management.