

## Course Syllabus

 Edit

### Professor's contact information

email	doug.blough@ece.gatech.edu
Office	Klaus Advanced Computing Building (KACB) 3356
Office hours	Mon 12:45 - 1:45 PM Wed 3:30 - 4:30 PM

### Course description and objectives

Object-oriented software methods for engineering applications using C++. Numerical analysis methods; simulations and graphical presentation of simulation results; analysis of numerical precision. Programming projects.

At the end of the course, you should be able to do the following:

1. Write a syntactically and semantically correct program using an object-oriented programming language such as C++.
2. Define and develop a computer program to implement specific tasks associated with a well-defined engineering application.
3. Choose the appropriate data structures to maintain the system state, and explain why the selected one is appropriate.
4. Choose appropriate algorithms to maintain the progression of the system state, and explain why the selected algorithm is appropriate.
5. Implement those data structures and algorithms using an object-oriented programming language such as C++.
6. Use the developed program to analyze performance and behavior of the engineering application being studied.
7. Create a class hierarchy of software objects with public inheritance and virtual functions to enable the use of polymorphism.
8. Create class templates and function templates to be used in a generic programming style.
9. Analyze the numerical error in floating-point calculations of linear equations of at least one multiplication and one addition.

10. Use commonly available tools for software development, source code maintenance, and debugging.

## Textbook

Deitel & Deitel, C++ How to Program (10th edition), Prentice Hall. (required)

We will cover approximately one chapter each week, so make sure to keep up with the readings! You have free access as long as you are a student at Georgia Tech. Through O'Reilly, students can enter their Georgia tech email address as the username and leave the password blank. The URL to access the book is:

<https://learning.oreilly.com/library/view/c-how-to/9780134448930/xhtml/cover.xhtml>   
<https://learning.oreilly.com/library/view/c-how-to/9780134448930/xhtml/cover.xhtml>

## Grading

Programming Assignments/Labs	25%
Exam 1	25%
Exam 2	25%
Final Exam	25%

**Assignments/Labs:** There will be 5 assignments/labs during the semester. The last lab *may* be due on the last instructional day, which is April 27.

You will complete general C++ programming assignments using the Georgia Tech ICE cluster.

There is Georgia Tech documentation on using ICE at [Information about ICE \(Instructional Cluster Environment\)](https://gatech.service-now.com/home?id=kb_article_view&sysparm_article=KB0042102). [https://gatech.service-now.com/home?id=kb\\_article\\_view&sysparm\\_article=KB0042102](https://gatech.service-now.com/home?id=kb_article_view&sysparm_article=KB0042102)

 will also demonstrate how to get started with ICE in class.

Later in the semester, you will complete labs using the ESP32 microcontroller kit and the Arduino C++ software development environment (IDE). ESP32 kits with peripherals can be checked out from the library. This contains most of what you need for the microcontroller labs. **You will need to get a breadboard and wire kit on your own** - you can purchase those through any outside vendor or through Eta Kappa Nu.

**Responsibility for material:** Students are responsible for **all** material in assigned sections of text, even if not specifically covered in lecture. Students are also responsible for all material covered in lecture.

**Makeup Exams and Incompletes:** As a rule, makeup exams will be offered **at the discretion of the professor** and only for scheduled Georgia Tech approved absences that are **requested in writing at least one week in advance**. Medical emergencies are the only exception to this rule and in case of such an emergency, the student must contact the professor with documentation of the emergency as soon as possible to discuss the makeup. Incomplete grades will be given only in extraordinary circumstances.

**Late Turn-in and Regrading:** It is expected that some leeway for late turn-in of assignments/labs will be allowed with penalty. A detailed late policy will be announced when the first assignment is posted. Exams will not be considered for regrading later than the **next class period after they are returned**. Regrading requests should be submitted **in writing** with a specific explanation of the possible grading error. Photocopies of completed exams will be made by the instructor prior to returning them.

**Academic Honesty:** Although students are encouraged strongly to communicate with each other to assist in learning the course material, all students are expected to complete course work individually, following all instructions stated in conjunction with exams and assignments. All conduct in this course will be governed by the Georgia Tech honor code, which can be found at:

<http://www.policylibrary.gatech.edu/student-affairs/academic-honor-code> ↗ (<http://www.policylibrary.gatech.edu/student-affairs/academic-honor-code>)

Additionally, it is expected that students will respect their peers and the instructor such that no one takes unfair advantage of any other person associated with the course. Any suspected cases of academic dishonesty will be reported to the Dean of Students for further action.

**Academic Honesty for Lab Assignments:** You MAY NOT copy code from others in any way. You MAY NOT use solutions that others have developed as the basis for your solutions. However, you ARE allowed to discuss the problems with others, including fellow students, teaching

assistants, and the instructor. Periodically, automatic plagiarism detection algorithms will be used to compare source code against all students in the course. You ARE allowed to solicit and obtain help in design and debugging your solutions. You CAN show others your BROKEN code and ask for advice about why it is not working or how to make it work better. But to be totally clear, you MUST implement your own solution. If someone helps you, you still MUST enter every line of code of your solution personally, and you MUST fully understand every part of your submission. Students should be prepared to explain each lab assignment and their work when demoing selected labs to the TA.

A similar policy relates to the use of AI tools, such as chatGPT and Claude, in the course.

---