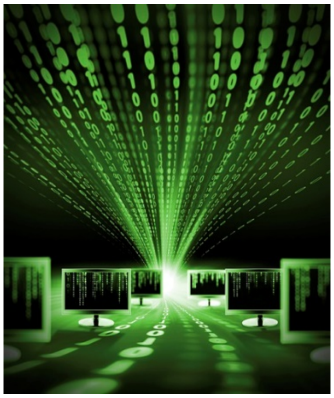
# ASEN 4057 Aerospace Software

*Material is preliminary and subject to change*



**Instructor:** Prof. Alexandra Le Moine (Alexandra.LeMoine@colorado.edu)
**Lecture:** Monday/Wednesday 2:45-3:35pm, Aero N100 (COPILOT)
**Lab:** Monday/Wednesday 3:45-4:35pm, Aero N100 (COPILOT)
**Office Hour:** Friday 3-4 pm, Aero N209
**TA:** Connor O'Reilly (connor.t.oreilly@colorado.edu)
**TA Office Hour:** TBD

**Canvas Webpage:** https://canvas.colorado.edu/courses/80590
**Github Private Repos:** https://github.com/Aerospace-Software

## Course Objectives

Aerospace engineers may go through their entire undergraduate education curriculum and have only a single formal course in *computing,* which often does not even cover formal programming, much less any details of the underlying processes by which the *computing* is accomplished. This is true despite an ever-increasing reliance on software by academia and industry for simulation and operational purposes. The purpose of this course is an attempt to fill that void.

## Course Learning Goals

The goal of this course is to (1) provide aerospace engineers with an overview of key software and hardware computing concepts utilized in academia and industry and (2) give the background necessary to tackle programming projects confidently on different computing platforms with various software tools and programming languages. Students will: (A) gain deeper and broad technical computing experience including debugging, code management and optimization, documentations, and collaborative software development; (B) actively apply these technical skills to solving relevant aerospace engineering problems; and (C) develop the key skills and traits to be a good programmer and software developer in academia and industry.

## Prerequisites

Students should have an extensive background in MATLAB programming and should understand programming fundamentals. It is assumed that students in the course have taken GEEN 1300, COEN 1300, ECEN 1030, ECEN 1310, CSCI 1300, CSCI 1310, or CSCI 1320. Previous experience with C programming is recommended.

## Course Textbook, References and Material

There is no required textbook for the course. Instead, reference material will be suggested throughout the course as applicable to the current lecture/assignment topics.

## Course Topics

**1. Fundamentals of Computer Programming and Software Design**

**2. MATLAB as a Tool for Software Design**

- a. Review of Basic Concepts
  - i. Classes, Arrays, Computations
  - ii. If Statements
  - iii. While and For Loops
  - iv. Plotting
  - v. Scripts and Functions
- b. Advanced MATALB Concepts
  - i. Debugging
  - ii. Profiling
  - iii. Numerical Integration and Optimization
  - iv. Handle Graphics
  - v. GUI Development

**3. Moving Beyond MATLAB**

- a. Compiled Languages versus Interpreted Languages
- b. Basic Sequential Computer Architectures and Operating Systems

**4. Introduction to Unix/Linux**

- a. Bash and Command Line Interface
- b. Bash Programming

**5. Version Control and Git**

**6. C as a Tool for Software Design**

- a. Review of Basic Concepts
- b. Pointers and Memory Management
- c. Compilation and Linking
- d. Building Programs with Make
- e. Defensive Programming and Debugging
- f. Performance and Profiling
- g. Code Optimization
- h. Scientific Libraries: BLAS and LAPACK
- i. Calling C within MATLAB with MEX Files

**7. Moving Beyond Sequential Computing**

- a. Introduction to Parallel Computing Architectures
- b. Parallel Computing with C and MPI
- c. Parallel Computing with C and OpenMP
- d. Parallel Computing with MATLAB’s Parallel Computing Toolbox

## Course Overview

The course will begin with a cursory overview of computer programming and software design. The course will then proceed with an overview of MATLAB as a tool for software design, reviewing basic concepts as well as exploring advanced programming techniques including debugging, profiling, handle graphics, graphical user interfaces, and numerical integration and optimization. CU-Boulder has a full MATLAB site license for students (https://oit.colorado.edu/software-hardware/software-catalog/matlab ).

The course will continue with a discussion of compiled languages (C, C++, FORTRAN) versus interpreted languages (MATLAB, Python) and basic sequential computer architectures and operating systems. This will set the stage for an introduction to Unix/Linux, including bash shell programming.

Understanding Unix/Linux and its environment is one of the primary goals of the course. The bash shell is the main interface with that environment, providing sophisticated configuration and programming capabilities. Another main goal of this course is to show students the similarities between programming languages and to demonstrate how it is fairly easy to work in any programming language with an understanding of basis programming constructs.

From this point of the semester forward, the development environment will be the Unix/Linux operating system. For Spring 2022, students will use the CSCI OpenStack Cloud Platform to work on Unix/Linux assignments. Before moving forward to the C programming language, the concept of version control will be introduced. The open-source version control system Git will be introduced for this purpose. The course will then proceed with an overview of C as a tool for software design. The course will review basic concepts, syntax, and structure before proceeding forward and discussing advanced concepts such as building programs with Make, and defensive programming and debugging. Various software tools which can dramatically improve a programmer's efficiency as well as his or her understanding of the underlying code will be introduced. These tools include debuggers, profilers, and compiler components. A number of approaches will be introduced to improve code performance including compiler and memory access optimization, and various software libraries will also be introduced to emphasize that many tools have already been exhaustively developed and should not be re-implemented by the programmer.

Finally, the course will move on to parallel computation. There is currently a massive paradigm shift away from a single hardware-processing element to parallel computational units. In order to take advantage of state-of-the-art computer hardware architectures, much of the burden is placed on the programmer. This course will not spend any significant detail on embedded software/programming, as that is the focus on ASEN 4519/5519 – Microavionics. Nor does this course provide any instruction on the programming of applications for the popular tablet/smartphones genre - as these are not currently used for solving traditional aerospace computational problems. Although the concepts of this class could definitely be applicable and useful for such related applications.

## Course Format

The course will follow a blend of traditional lectures with lab/computing assignments. There will be two lectures and two lab periods per week. There will be 7 individual quizzes throughout the semester, based on material covered in the lectures in addition to 2 survey question quizzes. There will be 7 programming assignments throughout the semester. For some of these programming assignments, you will need to work in groups, and for other assignments, you will need to work individually. For the group assignments, groups will be *randomly* determined. A midterm exam will be given in approximately the 9-10th week. There will be both in-class and take-home portions associated with this midterm exam. The in-class portion will test concepts, while the take-home portion will require programming. A final project (14-15th week) will replace a final exam, and students may elect to work in pairs of their own choosing for the final project. Student assessment will be based on individual quizzes, assignments, the individual midterm exam, and the group final project.

## Course Grading

10% Participation (Weekly feedback and self-assessment)
40% Assignments (7 assignments in total)
15% Quizzes (9 quizzes – Quiz #0 and Quiz #8 are survey questions)
20% Midterm (In-class exam & take-home exam)
15% Final Project

Grades will be posted to the course website on Canvas. Each assignment/quiz/exam will be based on a 100-point scale and weighted accordingly.

## Assignments Policy

Assignments will initially be uploaded to the Canvas course website. After version control is introduced, students will employ Git to turn in assignments. Students should make an effort to turn in assignments that are organized with a professional appearance. Proper documentation and commenting should be used to explain programming and software concepts employed. ***Late assignments will have a 10% deduction immediately, and an additional 4% deduction will be applied for each hour the assignment is late. No assignments will be accepted 24 hours after the original due date.*** Students will work individually or in *randomly* assigned groups for each programming assignment. Collaboration is permitted, even between groups. Students may discuss basic concepts related to the programming assignments, but students are **NOT** free to copy another student's assignment (except, of course, if the other student is a group member). ***Students who are caught copying (or providing his or her assignment to another) will receive an "F" for the course and reported to the Dean's office for further punitive action.***

## Quizzes Policy

Quizzes will take place at the beginning of class, and no make-up quizzes will be allowed. The lowest quiz grade will be dropped in computing the average to account for any missed quizzes. Quizzes will be announced approximately one week in advance. Quizzes will be closed book, and collaboration on the quizzes will not be tolerated. ***Students who are caught in these activities will receive an "F" for the course and reported to the Dean's office for further punitive action***

## Examination Policy

The midterm examination will cover all material in the course including lecture, discussions, assignments, and quizzes. There will be two portions to the midterm exam, an in-class portion and a take-home portion. The in-class portion will cover concepts and be closed book. It will be very similar in style to the quizzes, and collaboration will not be tolerated. The take-home portion will require programming and be open book. Students may use class notes, homework solutions, and any additional material included on the course website for this portion of the exam. Students may not, however, consult with classmates or reference code from external sources. Submissions will be compared with submissions from other classmates as well as an online repository to ensure this policy is followed. ***Students who are caught consulting with classmates or referencing external sources will receive an "F" for the course and reported to the Dean's office for further punitive action***.

## Final Project

A final project will replace a final exam and will be assigned toward the end of the semester and due before the Final Week. The final project will consist of improving performance of an existing serial code and parallelization. Students may work in pairs of their own choosing for the final project.

# CU BOULDER POLICIES

## CLASSROOM BEHAVIOR

Both students and faculty are responsible for maintaining an appropriate learning environment in all instructional settings, whether in person, remote or online. Those who fail to adhere to such behavioral standards may be subject to discipline. Professional courtesy and sensitivity are especially important with respect to individuals and topics dealing with race, color, national origin, sex, pregnancy, age, disability, creed, religion, sexual orientation, gender identity, gender expression, veteran status, political affiliation or political philosophy. For more information, see the policies on classroom behavior and the Student Conduct & Conflict Resolution policies.

## REQUIREMENTS FOR COVID-19

As a matter of public health and safety, all members of the CU Boulder community and all visitors to campus must follow university, department and building requirements and all public health orders in place to reduce the risk of spreading infectious disease. Students who fail to adhere to these requirements will be asked to leave class, and students who do not leave class when asked or who refuse to comply with these requirements will be referred to Student Conduct and Conflict Resolution. For more information, see the policy on classroom behavior and the Student Code of Conduct. If you require accommodation because a disability prevents you from fulfilling these safety measures, please follow the steps in the “Accommodation for Disabilities” statement on this syllabus.

CU Boulder currently requires masks in classrooms and laboratories regardless of vaccination status. This requirement is a precaution to supplement CU Boulder’s COVID-19 vaccine requirement. Exemptions include individuals who cannot medically tolerate a face covering, as well as those who are hearing-impaired or otherwise disabled or who are communicating with someone who is hearing-impaired or otherwise disabled and where the ability to see the mouth is essential to communication. If you qualify for a mask-related accommodation, please follow the steps in the “Accommodation for Disabilities” statement on this syllabus. In addition, vaccinated instructional faculty who are engaged in an indoor instructional activity and are separated by at least 6 feet from the nearest person are exempt from wearing masks if they so choose.

If you feel ill and think you might have COVID-19, if you have tested positive for COVID-19, or if you are unvaccinated or partially vaccinated and have been in close contact with someone who has COVID-19, you should stay home and follow the further guidance of the Public Health Office (contacttracing@colorado.edu). If you are fully vaccinated and have been in close contact with someone who has COVID-19, you do not need to stay home; rather, you should self-monitor for symptoms and follow the further guidance of the Public Health Office (contacttracing@colorado.edu). Please email your instructor if you will be missing class due to illness or quarantine.

## ACCOMMODATION FOR DISABILITIES

If you qualify for accommodations because of a disability, please submit your accommodation letter from Disability Services to your faculty member in a timely manner so that your needs can be addressed. Disability Services determines accommodations based on documented disabilities in the academic environment. Information on requesting accommodations is located on the Disability Services website. Contact Disability Services at 303-492-8671 or dsinfo@colorado.edu for further assistance. If you have a temporary medical condition, see Temporary Medical Conditions on the Disability Services website.

## PREFERRED STUDENT NAMES AND PRONOUNS

CU Boulder recognizes that students' legal information doesn't always align with how they identify. Students may update their preferred names and pronouns via the student portal; those preferred names and pronouns are listed on instructors' class rosters. In the absence of such updates, the name that appears on the class roster is the student's legal name.

## HONOR CODE

All students enrolled in a University of Colorado Boulder course are responsible for knowing and adhering to the Honor Code academic integrity policy. Violations of the Honor Code may include, but are not limited to: plagiarism, cheating, fabrication, lying, bribery, threat, unauthorized access to academic materials, clicker fraud, submitting the same or similar work in more than one course without permission from all course instructors involved, and aiding academic dishonesty. All incidents of academic misconduct will be reported to the Honor Code (honor@colorado.edu; 303-492-5550). Students found responsible for violating the academic integrity policy will be subject to nonacademic sanctions from the Honor Code as well as

academic sanctions from the faculty member. Additional information regarding the Honor Code academic integrity policy can be found on the Honor Code website.

## SEXUAL MISCONDUCT, DISCRIMINATION, HARASSMENT AND/OR RELATED RETALIATION

CU Boulder is committed to fostering an inclusive and welcoming learning, working, and living environment. The university will not tolerate acts of sexual misconduct (harassment, exploitation, and assault), intimate partner violence (dating or domestic violence), stalking, or protected-class discrimination or harassment by or against members of our community. Individuals who believe they have been subject to misconduct or retaliatory actions for reporting a concern should contact the Office of Institutional Equity and Compliance (OIEC) at 303-492-2127 or email cureport@colorado.edu. Information about university policies, reporting options, and the support resources can be found on the OIEC website.

Please know that faculty and graduate instructors have a responsibility to inform OIEC when they are made aware of incidents of sexual misconduct, dating and domestic violence, stalking, discrimination, harassment and/or related retaliation, to ensure that individuals impacted receive information about their rights, support resources, and reporting options. To learn more about reporting and support options for a variety of concerns, visit Don’t Ignore It.

## RELIGIOUS HOLIDAYS

Campus policy regarding religious observances requires that faculty make every effort to deal reasonably and fairly with all students who, because of religious obligations, have conflicts with scheduled exams, assignments or required attendance. In this class, email your instructor as soon as possible regarding missing class or assignments due to religious holidays. See the campus policy regarding religious observances for full details.