Course Objectives
This course offers an introduction to numerical computing and modeling in engineering. Its objective is to build on a background in calculus, linear algebra, and differential equations as well as a working knowledge in basic computing and programming to solve relevant problems in civil and environmental engineering. The course will rely heavily on the use of Matlab as a software tool.

Textbooks
Pratap R. *Getting Started with Matlab*. Oxford University Press, 1999 *(recommended)*.

Prerequisites
Math 33A, CEE 15, Math 33B *(may be taken concurrently)*

Course Schedule
**cee103OL is an online course**
Video Lectures will be released at the beginning of every week. Each weekly package will contain two to four 45-min lecture videos.
Online Discussion A *(instructor-led ZOOM Session)*: Tuesday, 10:00-11:30 AM
Online Discussion B *(instructor-led ZOOM Session)*: Thursday, 10:00-11:30 AM
Online Discussion C *(TA-led ZOOM Session)*: Friday, 10 AM-12 PM

Instructors
Ertugrul Taciroglu, Professor *(etacir@ucla.edu)*  *(first half of course)*
UCLA, Civil & Env. Engineering Department, 5731E Boelter

Steven Margulis, Professor *(margulis@seas.ucla.edu)*  *(second half of course)*
UCLA, Civil & Env. Engineering Department, 5732D Boelter

Teaching Assistants  TBD

Course Website  http://ccle.ucla.edu/

Grading Basis  Homworks 75%, Final Project 25%
Class Logistics

The class is designed to cover basic theory and demonstration of concepts during video lectures. Live online discussion sessions—which will be carried out using the ZOOM teleconferencing app—will focus on hands-on examples, practice problems, and Q&As. The tentative list of topics to be covered is listed below and closely follows Chapters 1-9 in the textbook.

Before each online ZOOM session, it is your responsibility to read the appropriate sections of the book and watch the corresponding video lectures. The live ZOOM sessions are not intended to cover basic theory but rather to allow students to interactively solve problems with the Instructor and the TAs. Basic theory and examples are covered in video lectures. Students should sign up and download ZOOM teleconferencing software at the following link and test it prior to the beginning of class:

http://zoom.us

The theme of the class necessarily involves the use of computers to solve problems in engineering. The programming tool for this class will be Matlab. It is essential that you are willing and able to use Matlab in this course. Matlab access is provided through the UCLA School of Engineering SEASNet Computing Facility. You should be able to access Matlab remotely from your computer by using the Remote Desktop Connection app. To set up this remote server, follow instructions provided at the link below:

http://www.seasnet.ucla.edu/terminal-server/

Prior to the beginning of the class students are advised to review Matlab programming basics. A recommended resource for this is MATLAB OnRamp, which is provided free-of-charge by MathWorks and can be accessed through the following link:

https://matlabacademy.mathworks.com

All course materials (lecture videos, recorded discussion and recitation sessions, handouts, assignments, homework solutions, etc.) will be provided through the CCLE:

http://www.ccle.ucla.edu

Final Project

The general topic of the final project will be announced around the end of the 7th week, and will be due by the Friday midnight (Los Angeles local time), August 25, 2017. This project will require you to solve a well-defined problem in engineering using various tools you have learned during the class. You will generate computer codes, and a Technical Report that clearly explains your work. You will be graded on accuracy, organization, and efficiency of your code, as well as on the technical clarity, and quality of the presentation of your technical report.
Homework Assignments

There will be one or two homework assignments per week. For example, Homework 1a will be released after online Discussion A, and Homework 1b will be released after online Discussion B. Each homework assignment will be due one week after the release day. These assignments are designed to cement your basic understanding of the principles covered in class, and provide practical experience in solving problems.

For any homework assignment, you will be expected to submit online (through CCLE):

- A written report (in Word or PDF format) that provides your solutions, and interpretations of results
- Your unique source code (written in Matlab) that you’ve used to solve the posted problems.
- In the report, you are expected to carefully and neatly describe your problem-solving logic and the significance of your findings. If you hand in a sloppy (either in thinking or writing) homework, you will lose points.
- The score for source code is based on its ability to produce correct results when operated by us (i.e., the course instructors and graders), neatness of organization, use of commenting, and the explicit use of the programming tools learned in the class. There might be other criteria specified in each homework assignment.
- Policy on late homework (late means any time after 11:59pm on the due date): Late homework will still be accepted, but you will lose 10 points for each hour that it is past the deadline.

Academic integrity

While we encourage you to discuss the course material and homework with others, the code and reports you submit must be your own.

Unauthorized collaboration, and copying or viewing another person’s work, including the transfer and/or use of another person’s computer files, are considered acts of academic dishonesty by the University Academic Integrity Policy and the UCLA Student Code of Conduct (www.deanofstudents.ucla.edu), to which we will hold you accountable. Under no circumstances should you share (by electronic, printed, visual, or any other means) any of your work with another student.

We will be actively monitoring the source code and reports and comparing your submitted work with that of others. If we find that you’ve copied each other or passed another’s work off as your own, we will have to take steps to report the breach of academic integrity. The bottom line here: A university education is about so much more than gaining knowledge and a good GPA. Most importantly, it is about challenging and developing your mind and your character. Honesty in your academic work will develop into professional integrity.
Course Topics

1. Error and computer arithmetic (floating-point numbers, accuracy, propagation of error, etc.)
2. Root-finding (bracketing methods, open methods, etc.)
3. Interpolation and approximation (polynomial interpolation, splines, least squares approximation)
4. Numerical integration and differentiation (trapezoidal and Simpson’s rules, Gaussian quadrature, numerical differentiation)
5. Solution of systems of linear equations (matrix arithmetic, Gaussian elimination, LU factorization)
6. Advanced numerical linear algebra (least squares data fitting, nonlinear systems)
7. Numerical solution of ordinary differential equations (one-step methods, boundary-value problems, initial condition problems, explicit and implicit methods)
8. Numerical solution of partial differential equations (finite difference methods for partial differential equations: Poisson equation, one-dimensional heat equation, one-dimensional wave equation)
Tentative Schedule of Topics

Prof. Taciroglu

06/26/17:
Week 1, Video Set 1: Syllabus, introduction. Taylor Series [Sections 1.1-1.2]
Week 1, Video Set 2: Computer arithmetic and computing errors: floating point numbers; sources of error; error propagation [Sections 2.1-2.3]

07/03/17:
Week 2, Video Set 1: Solution of nonlinear equations: Bisection method, Newton’s method, Secant method, Matlab’s fzero [Section 3.1-3.3]
Week 2, Video Set 2: Interpolation & approximation: Polynomial interpolation, Lagrange basis functions [Sections 4.1]

07/10/17:
Week 3, Video Set 1: Interpolation & approximation (cont’ed): divided differences. Errors in polynomial interpolation, Natural cubic splines [Sections 4.2, 4.3]
Week 3, Video Set 2: Numerical integration: Trapezoid and Simpson’s methods, error analysis [Section 5.1]

07/17/17:
Week 4, Video Set 1: Numerical integration (cont’ed): Gaussian quadrature [Sections 5.2]
Week 4, Video Set 2: Numerical differentiation: Differentiation using Interpolation [Sections 5.3, 5.4]

Prof. Margulis

07/24/17:
Week 5, Video Set 1: Solution of systems of linear equations: Systems of linear equations, Matrix Arithmetic, Gaussian elimination, LU factorization, errors in solving linear systems [Sections 6.1-6.5]
Week 5, Video Set 2: LU factorization, errors in solving linear systems, Iterative methods [Sections 6.4-6.6]

07/31/17:
Week 6, Video Set 1: Advanced numerical linear algebra: Least squares, Nonlinear systems [Section 7.1, 7.3]
Week 6, Video Set 2: Numerical solution of ordinary differential equations: Introduction, Euler’s method [Sections 8.1-8.2]

08/07/17:
Week 7, Video Set 1: Numerical solution of initial value problems: Convergence analysis of Euler’s method, numerical stability, implicit methods [Sections 8.3-8.4]
Week 7, Video Set 2: Numerical solution of initial value problems: Runge-Kutta and multi-step methods, Numerical solution of systems of equations and boundary value problems: Runge-Kutta [Sections 8.5-8.8]

08/14/17:
Week 8, Video Set 1: Finite-difference methods for partial differential equations (PDEs): Poisson’s Equation, One-dimensional heat equation [Sections 9.1-9.2]
Week 8, Video Set 2: Finite-difference methods for partial differential equations (PDEs): One-dimensional wave equation [Section 9.3]