

COMP/ENGG 150 Programming in MATLAB for Engineers

4 Units (3 Lecture + 1 Lab), typically 96 – 102 contact hours (16-17 weeks x 2 days x 3 hrs)

Prerequisite: Calculus 1

Course description: COMP/ENGG 150 is designed to meet computer programming requirements for engineering transfer students. The course utilizes the MATLAB environment to provide students with a working knowledge of computer-based problem-solving methods relevant to science and engineering, including programming and numerical analysis techniques. Students outline, write, test, and debug computer programs to solve problems and display results, with emphasis on proper documentation of computer code and reports. Common examples and applications of physics and engineering are used throughout the course.

The course resources are structured around a flipped approach where students watch video lectures at home and engage in programming activities during class time. Therefore, optimum scheduling is two weekly class periods of roughly equal duration.

Course objectives: There are essentially four major overlapping objectives in the course,

1. Reinforce a structured, top-down approach to formulate and solve problems.
2. Introduce common approaches, structures, and conventions for creating and evaluating computer programs, primarily in a procedural paradigm, but with a brief introduction to object-oriented concepts and terminology.
3. Apply a variety of common numeric techniques to solve and visualize engineering-related computational problems.
4. Introduce the MATLAB software environment.

Learning Outcomes: By the end of this course, students should be able to

1. Use MATLAB effectively to analyze and visualize data.
2. Apply numeric techniques and computer simulations to solve engineering-related problems.
3. Apply a top-down, modular, and systematic approach to design, write, test, and debug sequential MATLAB programs to achieve computational objectives.
4. Design and document computer programs and analyses in a careful and complete manner so as to effectively communicate results, to facilitate evaluation and debugging by another programmer, and to anticipate and resolve user errors.
5. Demonstrate understanding and use of fundamental data structures (classes).
6. Create and control simple plot and user-interface graphics objects in MATLAB.

Resource Links:

[Detailed course learning objectives for students.](#)

[Sample Course Syllabus:](#) textbook info, course requirements, course content, etc.

[Topic Sequence:](#) with links to all student resources (videos, slides, labs, HW)

[Course Documents:](#) Directory containing all documents (.doc and .pdf files)

Alternate [3-unit version of the course](#), aligned with [C-ID ENGR 220](#).