General Information

http://iel.ucdavis.edu/course/EME5/fall17

Instructor	Harry H. Cheng, Professor Office: 2231 Academic Surge Building Phone: 752-5020 Email: hhcheng@ucdavis.edu Office hours: 10:00pm-11:20pm MW, or by appointment
Teaching Assistant	Binsen Qian Office: 1067 Bainer Phone: 752-1028 Email: bqian@ucdavis.edu Discussion Hour: 3:10-4:00pm Tuesday, 233 Wellman Lab hours: 3-4pm Wed, 3-5pm Thursday, 2121A Bainer Hall Office hour: by appointment only
Lecture Hours	9:00-9:50am MWF, 118 Olson
Discussion Hour	3:10-4:00pm Tuesday, 233 Wellman **for Session 001**, CRN number 62378
Computer Lab Hours	3-4pm Wednesday, 3-5pm Thursday, 2121A Bainer Hall TA is available during the lab hour in 2121A. Lab hours are optional.
Course Description	Structured programming in C for solving problems in engineering. Introduction to MATLAB and comparison study of C/C++ with MATLAB.
Prerequisites	MATH 16A or 21A, may be taken concurrently
Textbooks	(1) Harry H. Cheng, *C for Engineers and Scientists: An Interpretive Approach*, McGraw-Hill, 2009, ISBN: 978-0-07-337605-9. (2) Instructor's lecture notes.
Course Handouts	The course handouts are distributed at lecture time. Some of them are available on the Web of the home page for EME5 at http://iel.ucdavis.edu/course/EME5/fall17 For example, this handout is stored as `general.pdf` under *General Policy*.
Outcome	This course addresses the following Educational Outcomes for the Mechanical Engineering and Aeronautical Science and Engineering Programs (a) work comfortably and competently with mathematics, science, and basic engineering principles; (e) identify, formulate and solve engineering problems; (k) use the techniques, skills, and modern engineering tools necessary for engineering practice.

Homework

Homework is given out weekly through email or posted in UC Davis Smartsite http://smartsite.ucdavis.edu. Hardcopy of the homework, including programs, must be submitted in a homework collection box outside 2018 Bainer Hall before 10am on the due date, typically on Thursday. Programs must also be submitted through the UC Davis SmartSite before 10am on the due date.
Homework sent through email will not be accepted.
If you use computer programs to solve a problem, you should hand in computer programs in hardcopy also. Otherwise, you will not get the full credit for the homework. Remember that computer programs are always treated as an Appendix. If without text, appendix alone is not sufficient enough to receive a full credit.
Late homework will not be accepted. without exception unless there is a documented medical excuse.

Examinations

Midterm examination: this is an open book/open notes examination. The specific date of examination will be announced one week before the examination date. No early or late exam will be given. If you miss the exam for medical reasons (You **must** document this; no other excuses are acceptable), the other parts of the course will be counted proportionally more or you may be allowed or required to take a make-up exam (the choice is the instructor's).
Final examination: a comprehensive open book/open notes examination. Tuesday, December 11, 8:00am-10:00am in 118 Olson

Grading System

Written and computer homework 25%
Midterm examination 25%
Final examination 50%

Academic Integrity

(1) All work submitted for credit must be your own. You may discuss your assignment with classmates and instructor, in the course to get ideas or a critique of your ideas, but the ideas and words you submitted must be your own. Unless **explicitly** stated otherwise in the homework assignment, collaboration is considered cheating and will be dealt with accordingly.
(2) For written homework, you must write up your own solutions and may neither read nor copy another student's solutions.
(3) **For computer programs, you must create and type in your own code and document it yourself**.
But, you are free to copy programs from toolkit and modify them to solve the similar problems. Modifications made should be clearly documented.
You are free to seek help from instructor and fellow students while you are debugging a program once it is written.