E7 Intro to Computer Programming for Scientists and Engineers Spring 2016 Course Syllabus

Instructor
Prof. Tina Katopodes Chow
Email: tinakc@berkeley.edu
Website: www.ce.berkeley.edu/~chow

Office hours: Mon/Wed 9:30-11 in 621 Davis Hall, or by appointment

Head GSI: Brad Harken, harken@berkeley.edu

Class meetings: MWF 2pm-3pm, 155 Dwinelle Hall
Class website: http://bcourses.berkeley.edu

Overview:
Goals for this semester are to (1) introduce you to computer programming – including general concepts and theory as well as practical aspects, (2) learn about important numerical methods that are widely used in scientific computing, and (3) become comfortable as a programmer, not just this semester, but also for future classes and in the next stages of your career.

Engineers and scientists need to have strong computing skills so they can acquire and analyze data from laboratory experiments and field studies, simulate and gain insight into how natural and engineered systems behave, and design and test new products and processes. Effective use of computer graphics to help visualize and communicate results of these analyses is also important.

Computer programming skills are highly marketable. This means computing skills can help you to develop your own apps, get hired by top employers, and/or become an eminent researcher in your field of interest. You should think of programming knowledge as critical to your future.

Course objectives:
By the end of this course, you should be able to:

- Work comfortably with memory, data, and functions in the Matlab environment
- Implement data structures, iterative and recursive methods to solve problems efficiently
- Estimate computational cost of a computer algorithm
- Manipulate vectors and matrices within Matlab
- Derive and apply discrete methods for regression, interpolation, root finding, integration, differentiation
- Program solutions to ordinary differential equations that describe a physical process
- Collaborate on a team to design and code larger programs
- Apply algorithmic thinking to decompose problems into smaller components that can be programmed on computers
- Turn simple mathematical models into computer code to solve engineering problems
- Apply computer programming to your life (to make your life easier)

Active learning philosophy: You cannot learn to program just by watching someone else do it. This semester you will be actively engaged in the learning process to develop your own understanding of the material through group discussion and through lots of coding practice in lab sections and assignments. You are expected to come to lecture and lab prepared and ready to learn, participate, and ask questions. You are expected to attend classes regularly, arrive on time and stay until the end, and treat your peers and instructors with respect.

COURSE LOGISTICS

Prerequisites: Both Math 1A and Math 1B are required prerequisites if you are taking E7. Math 1B may be taken concurrently. The class does not assume any prior experience with computer programming; it is truly an introductory class.

Course website: http://bcourses.berkeley.edu The website will be used to post reading, lab assignments and solutions, handouts, etc. You will submit all assignments here and you can also check your grades online. Corrections/clarifications to lab assignments will be posted on bcourses as necessary. You are responsible for keeping up with E7-related e-mails and announcements. Online discussion facilities (Piazza) for E7 are available on bcourses. Everyone in the class will be able to ask questions, post comments, and review what is posted there. You may discuss course material, but do NOT post your code or copy another student's work. Please be courteous and respectful of others in all interactions, online and in person.

Reading: Reading will be assigned from *An Introduction to MATLAB Programming and Numerical Methods for Engineers* by Timmy Siauw and Alex Bayen (E7 Head GSI and Professor from previous years) and will be available electronically on bcourses.

Office hours policy: Please see GSIs in your lab/discussion sections for assistance, and/or try the online discussion forum (Piazza). Due to the large class size, the instructor and head GSI will not be able to provide assistance with lab assignments during office hours or by e-mail.

Exams: There will be one in-class midterm (Wed. **March 2 in class**) and a final exam (Tues. **May 10, 11:30am-2:30pm**) covering material discussed in class and in labs. You **must** bring your Cal Student ID to all exams. Please check the final exam schedule **now** so you can make changes to your course enrollments in case you have schedule conflicts or too many final exams on the same day.

Lab assignments: Lab assignments will be posted on Thursday afternoons, and must be completed during the following week. You will be expected to develop, test, and document MATLAB code, and to submit your code electronically via bcourses before **4 PM** on the Friday one week after the lab assignment is posted. To account for possible bcourses upload traffic jams, there will be a 2-hour grace period. There will be *no exceptions* in accepting late lab assignments after 6 PM, so push the limits of the grace period at your own risk. Solutions to assignments will be posted on bcourses the week after the due date.

Lab sessions: In labs you will spend time actually programming, working at it in a hands-on manner, and overcome challenges so you build confidence and improve your skills. You all signed up for a specific lab session that meets for a total of 4 hours per week: either M/W or Tu/Th. Your GSI should be your primary point of contact for ALL course-related questions as well as help with lab assignments. Your GSI will provide you with his/her e-mail address, but please post your question in the online forum (Piazza) first for a faster response. The best way to get help is to ask questions in person during your scheduled lab sessions. There will be a second GSI assisting in each lab session. We strongly encourage that you bring a removable storage device/flash drive to lab sections to save your work from labs. All data saved to the lab

computers will be erased upon logging out, so it is critical that you either back up your work on a flash drive or email your work to yourself.

You should get help on the assignments during your lab sessions and/or using the online discussion capabilities included as part of the class bcourses site. If you are having major issues that you cannot resolve by talking with your GSI, you may contact Brad Harken (harken@berkeley.edu) who is head GSI for E7 this spring. The length/difficulty of lab assignments will ramp up gradually during the first half of the semester, so try not to fall behind in the early weeks or it may be difficult to catch up.

Grading: Your final course grade will be evaluated using both grading options listed below, and we will use the option that results in the *higher score* for each student. A minimum overall score of 60% is needed to pass the class.

Option 1:	Lab sessions participation	5%
	Lecture/discussion participation	10%
	Lab assignments	35%
	Programming project	10%
	Midterm exam	10%
	Final exam	30%
Option 2:	Lab assignments	35%
	Programming project	10%
	Midterm exam	15%
	Final exam	40%

Under option 1, you will score one participation point if you attend your first scheduled E7 lab section AND ask a question about your E7 lab assignment each week (i.e., on Monday or Tuesday). Lab participation points are capped at 10, but there will be 12 labs, so you do not need to ask a question every week in order to get full credit. Lab and lecture participation points will only be counted if they help to increase your final course grade. These points are easy to collect and are being offered to encourage you to (a) actively participate in lecture and labs, and (b) to get an early start on lab assignments each week. You can earn participation points only by attending the lab session for which you are actually registered. For the lecture points, full points will be awarded for answering at least 80% of the clicker questions posed in MWF 2-3pm classes (regardless of whether your answers are correct or not).

For both grading options, your **two lowest** lab assignment scores will not be counted – this will give everyone some maneuvering room in case of illness, personal/family issues, travel plans, crunch times in other classes, etc. This drop policy is intended to give you flexibility to handle these situations without needing to e-mail us to request deadline extensions or other special consideration. Your final letter grade will be computed on a straight scale unless upward adjustments are necessary. Your grade will not be curved down.

Class participation: To respond to questions asked in lecture, you will need to buy an iclicker or use the iclicker app on your smart phone. Bring your device or app to *every* lecture (MWF).

You can purchase an iclicker device OR you can use the REEF Polling app ($10 fee/semester) for your smart phone. You only need one or the other. Handheld iclicker devices are available at the student bookstore or online for ~$45. If you already have one, you can use it in this class too. Please follow the instructions here to activate and register your clicker or your smart phone app for E7: https://www.ets.berkeley.edu/discover-services/clickers/students-getting-started
Here is the link to the app version (2 week free trial):
https://www1.iclicker.com/products/reef-polling/

Lab grading procedure: Your lab assignments will be graded by an automatic grading system which has been designed for E7. Manual grading will be done to verify presence of appropriate explanatory comments and documentation in your code. You will be provided with some test cases (i.e., sample input data) and associated answers (i.e., expected outputs) that you can use to help check your code. The provided test cases are not exhaustive, and it is your responsibility to ensure that your code works in general, not just for a few supplied test cases. We will use additional undisclosed test cases in grading your lab assignments. You will receive a report for each lab assignment showing your score on each question and where you lost points. You should compare your output with the output in the solutions to understand what is going on in cases where you lost points.

Regrade requests: Any lab assignment grading disputes must be registered no later than one week after your lab assignment score is posted on bcourses. After that, your grade for that assignment will be final. To dispute grades, you should send an email to E7spring16@gmail.com with your name, SID, the lab assignment number, and file name that you are disputing as well as a memo of why you think the file was graded incorrectly. You will receive an email confirming the disputed problem, and we will resolve the problem as soon as we can. Please be patient and be assured that your dispute will be considered carefully.

Lab collaboration policy: We encourage you to get to know the GSIs and other students in your lab section, and to discuss lab assignments with them. It is fine to ask for and offer advice, and to answer questions from other students during lab sessions. **You must not e-mail, post, or otherwise share your code with others, and you may not copy the computer code of another student**. More generally, you may not submit the work of someone else, or work together in teams and claim jointly authored or copied work to be your own. We will use special software to check all submitted computer code for plagiarism. In situations such as header lines at the beginning of a file where everyone's code is similar, similarities do not matter and will be ignored. In areas where most codes are dissimilar, any remaining ones that show strong similarities will be flagged. The penalty for copying will be a score of **–100** (negative one hundred) on that lab assignment for all involved (original author and any copiers). A repeat offense earns a final grade of F and referral to the Center for Student Conduct. A negative score on a lab assignment means you will lose points earned from completing other lab assignments. Your lowest 2 lab scores can still be dropped, with the exception of the lab score of -100.

Programming project: In the second part of the semester, you will work in teams of two or three on a programming project; all team members should be enrolled in the same lab group. The project is designed to be a fun and comprehensive synthesis of the programming concepts you will learn throughout the semester. Further details about the project, including guidelines and

important dates, will be given later. This is the *only* time you are allowed to work together on writing code.

Honor code: The UC Berkeley honor code that was adopted by the ASUC reads: "As a member of the UC Berkeley community, I act with honesty, integrity, and respect for others." Collaboration or copying from others during an exam will not be tolerated and will result in zero credit and referral to Student Judicial Affairs. Bringing a fellow student's iclicker to class is also cheating and a violation of the University Honor Code. If you are caught with a remote other than your own or have votes in a class that you did not attend, you will forfeit all clicker points and may face additional disciplinary action.

Accommodations: Students with disabilities should contact the Disabled Students Program (DSP) office first, and then me, as soon as possible, but in any event at least a month in advance of any exam for which you are requesting special accommodations.

Concluding thoughts: Computing is a powerful tool that all engineers and scientists should know about. We will work hard to present E7 course material to you in a way that is clear and compelling, and you will work hard learning valuable new skills in this class. We hope you have a good semester, and GO BEARS!

FAQs

Why MATLAB? Why not some other language such as C, C++, Fortran, Java, Python, or …
MATLAB is a good programming language for engineers and scientists to use because it helps you to develop computational methods efficiently. This means you can implement your ideas and have a working prototype of your computer code up and running quickly. MATLAB can save you time and help you avoid "reinventing the wheel" because there is a large and growing library of special-purpose toolboxes that you can build on. Examples of available toolboxes include data acquisition, statistics, optimization, and bioinformatics. There are many others. MATLAB is short for MATrix LABoratory, and it will easily handle matrix arithmetic for you. Matrices arise often in engineering, so it is helpful to have a tool that allows you to work with them efficiently.

Another answer to the "*Why MATLAB?*" question is that E7 aims to teach general computer programming concepts and numerical methods that are important to know, even though you may be using other programming languages in the future. At this stage of your career, gaining more programming experience is helpful in general, regardless of the specific language that you use.

Can I get MATLAB on my own computer?
We encourage you to download the MATLAB software to install on your laptop or home computer. This is recommended but not required, since MATLAB is available in computer labs that are reserved for E7 this semester. MATLAB is also available in various other campus labs (check with another student, adviser, or computer support staff in your Department to find out more). To obtain a free license from Berkeley go to: https://software.berkeley.edu/MATLAB
Scroll down and fill out the form for student users and you will be instructed on how to download the software.

What if I need more time and/or help to complete lab assignments?

1109 Etcheverry is open to all on Fridays between 8 AM and 5 PM; there are no scheduled E7 lab sessions on that day. GSIs will be available in the lab on Fridays from 10 AM-noon to answer your questions. The lab is also open on Friday afternoons, but no GSIs will be available then to assist you. The online forum (Piazza) is available 24/7. 1109 Etcheverry is also available for drop-in use 3-4 PM on M/W. 1535 Tolman is available for drop-in use as long as there are not 2 classes scheduled in there (see calendar here: https://www.ets.berkeley.edu/discover-services/instructional-computer-facilities/staff-training/1535-tolman). Please **do not** crash other E7 lab sessions for which you are not enrolled. We want to provide a high-quality laboratory experience for all students; that means limiting the number of students present during each lab session.

COURSE TOPICS

A detailed schedule of lecture, exam, lab assignment, and project dates will be posted on bcourses.

Tentative lecture schedule:

Week	Week of	Topics	Reading
1	Jan 18	MATLAB as a calculator, Scratch programming	Ch. 1, Ch. 2
2	Jan 25	Data structures, functions	Ch. 3
3	Feb 1	Branching statements	Ch. 4
4	Feb 8	Iteration, recursion	Ch. 5, Ch. 6
5	Feb 15	(Holiday), complexity of algorithms, representation of numbers	Ch. 7, Ch. 8
6	Feb 22	Graphics, Data I/O	Ch. 10, Ch. 11
7	Feb 29	Review, midterm (Mar 2), debugging	Ch. 9
8	Mar 7	Root finding, linear equations	Ch. 16, Ch. 12
9	Mar 14	Linear algebra, regression	Ch. 13
	Mar 21	Spring break	
10	Mar 28	Interpolation, series	Ch. 14, Ch. 15
11	Apr 4	Numerical differentiation and integration	Ch. 17, Ch. 18
12	Apr 11	Ordinary differential equations	Ch. 19
13	Apr 18	Special topics	
14	Apr 25	Special topics, final projects, review	