

Instructor

Prof. Tina Katopodes Chow

Email: tinakc@berkeley.eduWebsite: www.ce.berkeley.edu/~chowOffice hours: Mon/Wed 9:30-11 in 621 Davis
Hall, or by appointment**Head GSIs:** James Neher - jmtn@berkeley.edu, Jake Duncan - jduncan@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 programs or 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 all 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 Siau and Alex Bayen (E7 Head GSI and Professor from previous years) and is available electronically on bcourses. Reading quizzes will be assigned to unlock access to the lab assignments.

Exams: There will be one in-class midterm (Wed. **February 28 in class**) and a final exam (Tues. **May 8, 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 assigned weekly and due on Fridays. To gain access to the lab assignment on bcourses you will first have to complete a reading quiz. For each lab assignment, you will be expected to develop, test, and document MATLAB code, and to submit your code electronically via bcourses before **12 PM (NOON)** on the Friday due date. Late assignments can be submitted before 5 PM on Friday with a 25% deduction. There will be *no exceptions* in accepting lab assignments after 5 PM and zero credit will be given. Solutions to assignments will be posted on bcourses the week after the due date.

Your **two lowest** lab assignment scores will be **dropped** – 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 on your own – please **do not** e-mail us to request deadline extensions or other special consideration as they cannot be granted. Please e-mail us only about exceptional situations (e.g. month-long illness).

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. There will be a second GSI assisting in each lab session. We encourage you to 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 save on bDrive 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 one of the head GSIs. 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.

Lab session pair programming: Pair programming is the practice of writing computer programs in pairs, with one partner (the "driver") actively controlling the computer and the other (the "navigator") observing. As the driver writes the program, the observer helps plan and catch errors. An important feature of pair programming is that the partners switch roles frequently. This ensures that each partner has a good understanding of all parts of the program, and that the person in each role remains focused.

For attending lab sessions and completing pair programming assignments during the lab, you can earn **bonus points** toward your final grade. It is your responsibility to check your work in with your GSI to receive credit for participating. You will earn participation credit only if you attend your *first* scheduled E7 lab section AND complete the E7 pair programming lab assignment each week (i.e., on the Monday or Tuesday). If you complete at least 10 of the 12 lab pair programming assignments during your assigned lab section this the semester, you will receive 3 extra points on your final grade. If you complete less than 10 pair programming assignments, then you will earn that percentage of the 3 bonus points.

Office hours: The GSI office hours [in person and online] are spread out throughout the week, and are available to all E7 students. You may attend as often as you like. This is a very valuable resource and you are highly encouraged to bring questions here on a regular basis. See schedule on bcourses. Due to the large class size, the instructor and head GSIs will not be able to provide direct assistance with lab assignments during office hours or by e-mail, but we are happy to discuss lecture material.

iClickers: 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 (\$15 fee/semester) for your smart phone. Use one or the other (not both). Handheld iclicker devices are available at the student bookstore or online for ~\$40. 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://www.iclicker.com/pricing>

iClicker questions will be asked during each lecture starting the second week of class. **It is your responsibility to bring a functioning (keep extra batteries with you) properly registered iClicker to lecture every day** (register through the tab in the E7 bcourses page). Put your name on your phone and/or iclicker! iClicker questions will be based on the assigned reading and the material presented during the lecture. For each question, you will receive one point for answering the question and one point for answering the question correctly. If you earn 75% of the possible clicker points for the semester, you will receive all three **optional bonus points** on your final grade. If you earn less than 75% of the possible clicker points, then you will earn that percentage of the three clicker bonus points. There are no makeups for iClicker questions.

Technology in the classroom: Technology is a main feature in this programming class, with iClickers being used as a part of the lecture experience. Some may also find it helpful to bring their laptops to class, since they may be used to take notes* and test Matlab code. With all these devices in the classroom, it is easy get sidetracked by other features of the technology we are using. When you do so, you not only hamper your own understanding of the lecture topics, but you detract from the learning experience of those around you. With this in mind, we would like to ask that you please refrain from using your devices for purposes that are not related to the class material during the lectures, and putting your device away when you are not actively using it. We may revisit this subject later with the aim of establishing guidelines that ensure an equitable learning environment for all. (*Research shows that taking handwritten notes improves learning and understanding.)

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: You can submit a regrade request only if you miss *more than 2 points* on a problem. Any lab assignment grading disputes must be received 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 e7spring@gmail.com with your name, SID, the lab assignment number, and file name that you are disputing as well as a description 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 work in groups and ask for and offer advice, and to answer questions from other students during lab sessions as long as you are writing your *own* code. **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 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 or letting someone copy 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 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 directly 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.

Grading: Your final course grade will be evaluated using the point distribution below. The total possible number of possible points is 106, including 6 bonus points. Your final letter grade will be computed on an approximately straight scale (out of 100). A minimum overall score of 60 is needed to pass the class.

Lab assignments [lowest 2 scores dropped]	35
Programming project	15
Midterm exam	15
Final exam	35
<i>Lab session bonus points</i>	3
<i>iClicker bonus points</i>	3

Lab session participation and iClicker 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 lab participation points only by attending the lab session for which you are actually registered.

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 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?

Check out the in-person and online office hours schedule. The online forum (Piazza) is available 24/7. 1109 Etcheverry is 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 at the same time (see calendar here: <https://www.ets.berkeley.edu/discover-services/instructional-computer-facilities/staff-training/1535-tolman>). 1109 Etcheverry is open to all on Fridays between 8 AM and 5 PM; there are no scheduled E7 lab sessions on that day and there will be no GSIs to assist you. 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 schedule of exam, lab assignment, and project dates will be posted on bcourses.

Tentative lecture schedule:

Week	Week of	Topics	Reading
1	Jan 15	MATLAB as a calculator, Snap!	Ch. 1, Ch. 2
2	Jan 22	Data structures, functions	Ch. 3
3	Jan 29	Branching statements	Ch. 4
4	Feb 5	Iteration, recursion	Ch. 5, Ch. 6
5	Feb 12	Complexity of algorithms, representation of numbers	Ch. 7, Ch. 8
6	Feb 19	(Holiday), Graphics, Data I/O	Ch. 10, Ch. 11
7	Feb 26	Review, midterm (Feb 28), debugging	Ch. 9
8	Mar 5	Root finding, linear equations	Ch. 16, Ch. 12
9	Mar 12	Linear algebra, regression	Ch. 13
10	Mar 19	Interpolation, series, (no class Mar 23)	Ch. 14, Ch. 15
	Mar 26	Spring break	
11	Apr 2	Numerical differentiation and integration	Ch. 17, Ch. 18
12	Apr 9	Ordinary differential equations	Ch. 19
13	Apr 16	Special topics	
14	Apr 23	Special topics, final projects, review	