E7: Introduction to Computer Programming for Scientists and Engineers

Fall 2024

Instructor:	Mohamad Hallal	**Lecture:**	Mon./Wed. 4pm–5pm, Dwinelle 155
Office Hours:	Tue. 4pm–5pm (Zoom)	and	Wed. 2pm–4pm, McLaughlin Hall 213
Email:	engin7@berkeley.edu		

Course Objectives

E7 is an introductory course on computer programming and numerical methods. The goal of the course is to introduce basic computer programming concepts and apply them to computer-based problem-solving methods. The first half of the class provides an accelerated introduction to programming the second half introduces several on numerical methods commonly used in engineering.

Any knowledge of programming can dramatically elevate the quality and efficiency of your work. Programming compels you to uncover your underlying logic and fosters transparency. Programming also prompts you to structure your work systematically, facilitating sharing with others and significantly magnifying its impact. Furthermore, programming makes your work reproducible. By employing your programs repeatedly, you can efficiently navigate mundane tasks and optimize your productivity. *All scientists and engineers should learn how to code – it will transform you.*

E7 this semester primarily uses the Python 3 programming language. Python is a popular language in both industry and academia. It is also particularly well-suited to the task of exploring the topics taught in this course.

Mastery of a particular programming language is a useful side effect of E7. However, our goal is not to dictate what language you use. Instead, our hope is that once you have learned the fundamentals, you will find that picking up a new programming language is but a few days' work. In fact, you will be asked to learn one other language, MATLAB, which is a popular scientific computing platform that continues to be widely used for engineering and scientific applications.

Why Python?

Python is an open-source, high-level, general-purpose programming language. Apart from its neat syntax and code readability features, Python's best aspect is that it comes equipped with a host of standard libraries for accomplishing different programming and computing tasks. Though it's been around for only 30 years, it has recently become one of the most popular programming languages, and its popularity continues to grow. Python has a very broad and supportive community. There are thousands of libraries developed and vetted by its community (all free, well-documented, and open-source) that are easy for anyone to install and use. This makes Python extremely versatile.

Not only does Python run Reddit, Dropbox, Pinterest, Instagram, and others, but the main dynamic language used at Google is Python. Also, Google's YouTube is largely written in Python. On the science and engineering side, the data to create the 2019 photo of a black hole was processed in Python, and major companies like Netflix and NASA use Python in their data analytics.

Why MATLAB?

But we will learn MATLAB too! MATLAB, short for Matrix Laboratory, is a computing platform that includes highly specialized libraries for engineering and scientific applications. However, when compared to Python, MATLAB exhibits relatively less versatility in domains such as data science, machine learning, and web scraping. That being said, MATLAB remains widely used in certain areas (it is the de facto standard for control systems, signal processing, numerical computations, etc.), so we will delve into MATLAB and see how it can be used to perform many of the same tasks. If you go into manufacturing and design, you may be working in MATLAB more exclusively.

There is one important difference between MATLAB and Python. MATLAB is proprietary, closed-source software, and a paid license is required to be able to use it. On the other hand, Python is a free and open-source software. Not only can you download Python at no cost, but you can also download, look at, and modify the source code as well. This is a big advantage for Python. If you are a researcher or scientist, then using open-source software brings greater integrity and accountability to your work and allows a much broader audience to use the code you develop.

Why Both?

Python and MATLAB are tools used to learn programming. However, programming is not confined to Python or MATLAB. Consider these tools as vehicles on the journey of programming, not as the ultimate destination. What you will learn in this course is not about a specific programming language, but can be applied to any other language. Learning multiple programming languages equips you with a broader skill set and will get you far more opportunities.

So, how will you learn all of that?!

1. Interactive lectures that will cover the building blocks
2. Lab assignments that build up from the building blocks to more practical applications to develop your programming and computational skills
3. Discussions to deepen your understanding of the material, work on more challenging exercises, and interact more closely with your peers

Course Catalog Description

Elements of procedural and object-oriented programming. Induction, iteration, and recursion. Real functions and floating-point computations for engineering analysis. Introduction to data structures. Representative examples are drawn from mathematics, science, and engineering.

Prerequisites

Math 1B (may be taken concurrently). Previous exposure to linear algebra and differential equations will be extremely helpful. Prior experience with computer programming is not required.

Course Format

The course includes many opportunities for learning: lecture, discussion, lab sessions, office hours, and more. Recognizing that each individual learns differently, it is recommended that you explore all these options to identify what works best for you.

Lecture

There are two 50-minute lectures per week. Prior to every lecture, slides and related materials will be made available. A course capture of the lecture will be posted soon after each lecture is given. This course moves fast, and the lectures are tightly coordinated with the discussion and lab sections. It is crucial that you attend or watch each lecture the day it is given and before you attend the discussion and lab sections. While the course capture is provided, you are expected to attend classes regularly, arrive on time, and treat your peers and the teaching staff with respect.

The lectures are dedicated to introducing and demonstrating the fundamental concepts of computer programming and numerical methods. Please be ready to take notes and work on programming exercises during the lectures. Bringing a laptop to the lectures is highly recommended, as it facilitates your involvement in programming drills and activities.

Discussion Section

There is one 50-min discussion per week, which will feature a worksheet with problems reviewing the material learned in the current week. In discussion sections, teaching staff will review the material and help students solve the problems on the worksheet. Moreover, the discussions will feature programming challenges, which are more demanding than the exercises presented during the lectures. These challenges are designed to further test your understanding and programming skills. If you wish to work on the coding challenges, please bring a laptop to the discussion section.

To receive credit for the discussion, you must attend in person. *You may only attend the discussion section in which you are officially enrolled.* We want to provide a high-quality experience for all students, which is why we limit the number of students in each discussion section. Discussion sections will not be webcast and are not recorded. Your active participation in these discussion sections will contribute significantly to your mastery of the course material.

Lab Section

In addition to the lecture and discussion, you will have two 120-min lab sections per week (total of 4 hours), where you will actively engage in programming and applying the concepts covered in the lectures. In these sections, teaching staff will introduce the assignment, review relevant material, and answer students' questions. The main purpose of the lab section is to provide you with dedicated time to work on the assignments while receiving immediate assistance from the teaching staff. The best way to get help for completing lab assignments is to be present from the very beginning of your lab section and to come prepared with questions.

You may only attend the lab section in which you are officially enrolled. We want to provide a high-quality experience for all students, which is why we limit the number of students in each lab section. You should bring your laptop to the lab section – the lab *is not* equipped with computers.

Textbooks

Python Programming and Numerical Methods: A Guide for Engineers and Scientists, by Qingkai Kong, Timmy Siauw, and Alexandre Bayen. Lectures will generally follow the outline of this textbook. This textbook is available freely online here.

An Introduction to MATLAB Programming and Numerical Methods for Engineers by Timmy Siauw and Alexandre Bayen. Parts of this book will be available electronically on bCourses.

You are responsible for material covered during the lecture, discussion, and lab section. Reading the assigned textbook sections is required, as they go into more detail than we have time to.

Support

You are not alone in this course; the teaching staff are here to support you as you learn the material. Some aspects of the course will take time to master, and the best way to master challenging material is to ask questions. For questions, use Ed, come to office hours, and ask your peers.

Discussion Forum

Our discussion forum will be Ed Discussion. For most questions, Ed is the right place to ask them. The course staff read it regularly, so you will get a quick answer. Furthermore, by posting on Ed as opposed to emailing us directly, other students benefit by seeing the question and the answer.

Any questions regarding course material (lectures, labs, discussions, exams, etc.) should be posted on Ed. You can access Ed from bCourses. When posting questions, click on the "New Thread" button, give your thread a title, then select its correct category (General, Lectures, Lab, etc.) and subcategory (e.g., Lab 01, Lab 02, etc.). If you have a question about something pertaining to your own code that shouldn't be shared with the class, you can make a private post on Ed, which will only be seen by the teaching staff. For more information, refer to the Quick Start Guide.

Office Hours

Office hours by the teaching staff, mostly in-person and some over Zoom, are spread out throughout the week, and are available to all E7 students. This is a very valuable resource in which you can ask questions about the material, receive guidance on assignments, and work with peers and course staff in a small group setting. The schedule of the office hours is available on bCourses.

Staff Email

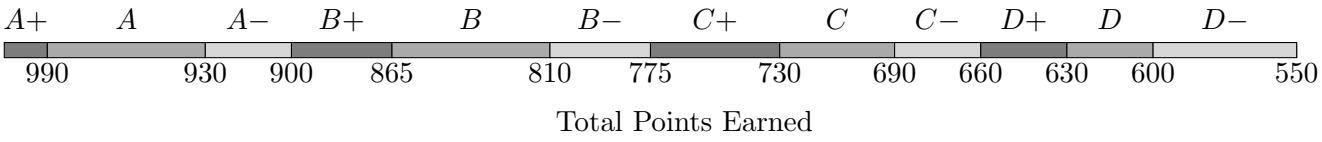
The email address engin7@berkeley.edu will send a message to the course staff. You can use it for correspondence related to personal matter. Both the lead teaching assistants and Mohamad monitor this inbox, so you can generally expect a response within a few days. If you send a question that is of general interest, we may post the response on Ed (with personal details omitted). If you have any concerns that require an exception to course policies (e.g., medical emergencies or sudden necessary travel that result in extended absences), please contact engin7@berkeley.edu. Please refrain from emailing Mohamad or course staff for exceptions – email engin7@berkeley.edu.

Grading

Final course grades are determined according to your performance on lab assignments, discussions, surveys, midterm exams, final exam, and lecture participation. The points for each assessment are as follows:

Assessment	Count	Dropped	Total Points	Weight
Lab Assignments	11	1	250	25%
Discussions	13	3	50	5%
Study Quizzes	3		50	5%
Midterm Exams	2		300	30%
Final Exam	1		300	30%
iClicker	N/A		50	5%
			Total: 1000	*100%*

Your letter grade will be determined by the total points you earn out of the possible 1000. There is no curving in this course. Learning is not a competition. Your grade will reflect your individual performance, and it will not be influenced by the performance of your peers.



Lab Assignments: You will complete 11 lab assignments, due by Fridays at 11:59pm. Your 1 lowest lab grade will be dropped, and your 10 highest grades will count toward the final score. Each lab is worth 25 points, for a possible total of 250 points. Our policy on requesting an extension can be found in the Assignment Extensions section. Lab assignments will typically require 6–8 hours of work per week, so start lab assignments early and attend your lab section. You will be expected to develop, test, and document Python and MATLAB code. You will submit your work electronically on Gradescope or bCourses. It is your responsibility to confirm you have submitted your work correctly. Your lab assignments will be graded by an automatic grading system. Test cases will be provided in the lab assignments for you to check your work, and additional hidden test cases may be used in grading. While you are encouraged to work together on lab assignments, you must submit your own answers. *Lab assignments help you understand the material and serve as preparation for exams, so do not rely too heavily on other students or online resources for help.*

Discussions: There are 13 discussions throughout the semester. For each discussion you attend and submit the discussion worksheet, you will receive 5 points, for a possible total of 50 points. Attending any 10 (or more) discussions will earn a perfect discussion score. Since you can skip up to 3 discussions, this will give everyone some maneuvering room in case of illness, emergencies, travel plans, and so on. This policy is intended to give you flexibility to handle these situations without needing to email us regarding your absence. Plan to use your 3 drops for true emergencies only. No exceptions will be made to this policy. *Please do not email us to request special consideration for the discussions. Contact us only for exceptional situations (e.g., month-long illness).*

Study Quizzes: Prior to each midterm and the final exam, you will be required to complete an online quiz to help you prepare for the exams. They will be available on bCourses and include a combination of multiple choice, free response, and fill-in-the-blank questions. You can retake the quiz multiple times, and your highest score will be recorded. Each quiz will carry a different point value, for a possible total of 50 points. All quizzes are mandatory, and none will be dropped. Furthermore, late submissions for study quizzes will not be accepted.

Midterm Exams: You will complete 2 midterms for this course. Each midterm is designed to last around 1 hour and 15 minutes. All midterms are closed book, but you may bring one double-sided sheet of notes to each midterm. Your grades for the midterms will be distributed as follows: your highest score will be over 200 points (20%), and your lowest score will be over 100 points (10%), totaling a possible 300 points (30%). The midterms are not intentionally cumulative, but the course material builds upon earlier topics and cannot be fully separated.

All midterms must be taken in-person. There will be no alternate tests offered.

Tentative dates:

Midterm #1 ..... Monday, September 30, 7:30 – 9:00 PM
Midterm #2 ..... Monday, November 04, 7:30 – 9:00 PM

Final Exam: The final exam is cumulative and will be held at the University-scheduled time. The exam is closed book, but you may bring two double-sided sheets of notes to the final.

The final exam must be taken in-person. There will be no alternate final exam offered.

Date:

Final Exam ..... Thursday, December 17, 8:00 –11:00 AM

iClicker: During each lecture, several iClicker questions will be posed to emphasize key lessons. Follow these steps to create a student account by signing in through the UC Berkeley campus portal. To respond, you will need to download the iClicker Student mobile app or access the web app. The iClicker Student mobile app is free for Berkeley students.

iClicker questions will be asked in every lecture. It is your responsibility to bring a device to respond to these questions. If you miss a lecture or have issues accessing iClicker, you can complete a short online quiz on bCourses, due one day after the lecture. If you respond to both the iClicker questions and the bCourses quiz, your highest score for that lecture will be kept.

These questions are graded for completeness, not correctness. You will earn a perfect iClicker score by answering all questions, even if incorrectly. You will receive 1 point for each question you answer and an additional 1 bonus point for each correct answer. To earn a perfect iClicker score, you need a total of 50 points. Any points beyond 50 will count as bonus points, up to +15 (the maximum you can earn is 65/50).

Given that there are bonus points for answering a question correctly, there will be no makeups for iClicker questions. If you miss a lecture and do not complete the bCourses quiz, your score for that lecture will be 0. *Please do not email us to request special consideration for iClicker questions. Contact us only for exceptional situations (e.g., month-long illness).*

Technology Requirements

Hardware

In today's world of technology, restricting the use of devices in class is not possible or effective. However, we need to work together to ensure appropriate incorporation of devices into the classroom. Phones, laptops, and tablets can only be used as an educational tool.

Technology is a main feature in this programming class, with iClicker questions being used as a part of the lecture experience. Some may also find it helpful to bring their laptops to class – lots of lecture time will be spent working through programming problems and being able to follow along on your own laptop will be extremely important. In addition, please bring a device to the lectures to participate in the iClicker questions.

With all these devices, it is easy to get sidetracked by other features of the technology we are using. When you do so, you are hampering your own understanding of the lecture topics and 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. We may revisit this subject later to establish guidelines that ensure an equitable learning environment for all.

Software

There are different ways to install Python and related packages. We recommend to use Anaconda, which provides easy ways to manage Python work environment. To install Anaconda, please go to this link. Depending on the operating systems (OS) you are using, i.e, Windows, macOS, or Linux, you need to download a specific installer for your machine. After downloading the installer, click on it and run it to complete the installation.

However, to keep things smooth, we provide dedicated servers that will allow you to run Python from the comfort of a web browser. Please go to DataHub and use your CalNet credentials (ID and password) to log in. For each lab, we will provide a link that directs to your DataHub account, and the Jupyter Notebook will automatically launch. The lab materials will be automatically pushed to your account and you can directly work on them.

We strongly recommend installing MATLAB on your laptop. As a UC Berkeley student, you are eligible for a free MATLAB license on your personal devices, which you can acquire by visiting this link. Alternatively, you can use MATLAB online, eliminating the need for installation. However, you will still need to create a MathWorks account, which is free for Berkeley students.

Academic Integrity

Code of Student Conduct

"The Chancellor may impose discipline for the commission or attempted commission (including aiding or abetting in the commission or attempted commission) of the following types of violations by students, as well as such other violations as may be specified in campus regulations:

102.01 Academic Misconduct: All forms of academic misconduct including but not limited to cheating, fabrication, plagiarism, or facilitating academic dishonesty."

Collaboration

Collaboration is important and a powerful way to learn. We encourage you to study together. No matter your academic background, you will learn more if you work alongside others than if you work alone. Ask questions, answer questions, and share ideas. However, it is important to keep in mind the limits to collaboration. You and your friends are encouraged to discuss course content and approaches to problem solving. But you are not allowed to share your code or answers. Doing so doesn't help students who are facing challenges with the course content, and is considered academic misconduct. It sets them up for trouble on upcoming assignments and on the exams.

Lab Misconduct

Deadlines can be stressful, and we know that under extreme pressure, it becomes tempting to start rationalizing actions that you would otherwise consider inappropriate. Please don't do this. We care that you learn the material, and that your grade reflects your learning. If you feel that you need to engage in academic misconduct to meet a deadline, please reach out to course staff.

Any work you turn in must be 100% your own. You must not email, post, or otherwise share your code with others, and you may not copy the computer code of another student or from any other source. More generally, you may not submit work that is not your own, or claim jointly authored or copied work to be your own. We will use special software to check all submitted computer code for plagiarism. Copying answers from *any source* is unacceptable and will be penalized.

The penalty for lab misconduct will be a score of -25 (negative score) on that lab assignment for all involved (copiers and original author, if any). You will also be required to write an anonymous reflection to be posted on the course website. A repeat offense will receive a failing grade and will also be reported to the Center of Student Conduct. A negative score means you will lose points earned from completing other lab assignments. Your lowest lab score can still be dropped, with the exception of the lab score of -25. Given our friendly lab extension policy (see below), there will be no exceptions. Make a serious attempt at every assignment yourself. When you need help, reach out to the course staff on Ed or in office hours. It's normal to struggle and you are not alone!

Exam Misconduct

No collaboration is allowed on any of the exams. Anyone involved in academic misconduct in any exam will receive a failing grade and will also be reported to the Center of Student Conduct. Please keep your eyes on your own materials and do not converse with others during exams.

Lecture and Discussion Misconduct

Answering iClicker questions for a class that you did not attend is also a violation of the University Honor Code. In this case, you will forfeit all iClicker points and may face additional disciplinary action. This also applies to discussions that you did not attend. In this case, all parties involved will forfeit all discussion points and may face additional disciplinary action. In addition, posting course content on any third party websites is considered academic misconduct.

Penalties for academic misconduct in E7 are severe and these policies will be strictly enforced.

Don't engage in academic misconduct – it's not worth the consequences.

Resources and Accommodations (DSP and Otherwise)

We are committed to creating a learning environment that meets the needs of the diverse student body. If you anticipate or experience any barriers to learning in this course, please feel welcome to discuss your concerns directly with the instructor.

Disability Accommodations

We will provide appropriate accommodations to all students enrolled in the Disabled Students Program (DSP). To ensure that you receive the appropriate accommodations, have your DSP specialist submit a letter confirming your status and accommodations. We also aim to provide fair and appropriate accommodations to any students who, because of extenuating circumstances, may need them. For information on how to request accommodations, please see the sections below.

Lab Assignment Extensions

Our assumption is that you will complete each assignment by its due date. Especially in the first few weeks, the course content builds on itself, and not completing an assignment will make it much more difficult to do well in the course. However, we understand that life happens, and we want to ensure that all students are provided with the support and flexibility they need to succeed.

If you need to request an extension, regardless of your DSP status, please fill out the form on bCourses. *This only applies to lab assignments – other aspects of the course cannot be extended.* There are two types of extensions:

1. Short: extensions submitted for up to 24 hours (no reason required)
2. Long: extensions submitted for 24–72 hours (valid reason required)

After Midterm 1, the course content builds on itself in a less time-critical manner so, the maximums for "short" and "long" extensions will be increased to 48 and 120 hours, respectively. There will *never* be a penalty for requesting an extension. However, continuing to work on an assignment after the original deadline may lead to slower response times on Ed/longer wait times in office hours, as staff will actively prioritize assignments with upcoming due dates.

Mental Wellness and Basic Needs

You may experience a range of issues that can cause barriers to learning. UC offers services to assist you with addressing stressful events you may be experiencing. Counseling and Psychological Services (CAPS) is available as part of University Health Services. Services are offered at many locations, including on-site in the College of Engineering. CAPS services are available to all students, regardless of insurance, and initial visits are free. You can receive help immediately with same-day counseling at (510) 642-9494 and a 24/7 counseling line at (855) 817-5667.

The Basic Needs Center provides housing, food, among other support services. Specifically, the Food Pantry aims to reduce food insecurity among students. Students can visit the pantry as many times as they need and take as much as they need while being mindful that it is a shared resource. The pantry operates on a self-assessed need basis; there are no eligibility requirements. The pantry is not intended for supplemental snacking food, but rather, core food support.

Tentative Schedule

Week	Date	Lecture Topics	Lab	Discussion
1	08/28–08/30	Introduction		1
2	09/02–09/06	*Monday no class* Python Basics	1	2
3	09/09–09/13	Data Structures NumPy Arrays	2	3
4	09/16–09/20	Functions Conditionals	3	4
5	09/23–09/27	Iteration Recursion	4	5
6	09/30–10/04	*Midterm 1, Monday 09/30* Number Representations		6
7	10/07–10/11	Errors, Exceptions Reading and Writing Files	5	7
8	10/14–10/18	Plotting	6	8
9	10/21–10/25	MATLAB	7	9
10	10/28–11/01	Linear Algebra Regression	8	10
11	11/04–11/08	*Midterm 2, Monday 11/04* Series, Root Finding		
12	11/11–11/15	*Monday no class* Root Finding	9	11
13	11/18–11/22	Numerical Differentiation Numerical Integration	10	12
14	11/25–11/29	Ordinary Differential Equations *Wednesday no class*		
15	12/02–12/06	Ordinary Differential Equations	11	13
16	12/09–12/13	*RRR week*		

Final Exam Date: Thursday, 12/19, 8:00 – 11:00 AM