

MATLAB BOOTCAMP SYLLABUS – SEPTEMBER 2022

SIO 209, section 96772, 1 unit S/U

Instructor: Sarah Giddings, sgiddings@ucsd.edu

Class meetings: Eckart 225, computer lab, 19-20 Sept 08:30-13:00

Office hours: 19-20 Sept 14:00-15:00, or [email me](#) to set up an appointment

Course Summary:

This course will provide a hands-on introduction to MATLAB. No prior experience is necessary. The course will cover variables, plotting, scripts, matrices and other ways to store data including NetCDF, loops, and an introduction to more advanced techniques. While the focus is MATLAB, a brief comparison to other programming languages will be included and several of the coding approaches and best practices are applicable to other languages.

Requirements:

You do not need any background with MATLAB or computer programming. Programming is learned best by practice, so in order to follow along and do the homework, you will need to have access to a computer that runs MATLAB. You can bring your own laptop or use a computer in the Eckart lab. UCSD has a campus-wide MATLAB license, [instructions for students are here](#). Other options for MATLAB access are [included here](#). Worst case scenario, a student version is available at the UCSD bookstore (\$100), but that should not be necessary.

Before coming to class on the first day:

- [download MATLAB](#) and test that it works
- log into the course via Canvas
- download notes for the first day of class
- download files provided

What if I have more or less experience than my classmates?

It is important to note that because this course is open to all levels, the level of difficulty will vary by student. In order to make the course valuable for a variety of experience levels, two sets of homework aimed for different levels (no prior experience and some-moderate experience) will be available as well as extensive office hours. I strongly encourage students to ask questions during class, slow me down if need be, and repeat some of the coding at home. Finally, a variety of references and tutorials are listed below including some free on-line learning tutorials for a range of experience levels.

Note on the short format:

This class is scheduled for the very start of the fall quarter as some students may need to use MATLAB starting early in their fall quarter classes. Unfortunately, because of other competing time constraints (new student orientation, the Scripps Student Symposium, etc.), the course is packed into just 2 days this year. Prior offerings have been 2-3 days. While 4-5 days would be far preferable, most students find this course provides a good starting point. The goal is NOT to master the MATLAB programming language but rather to introduce students to the wide range of its capabilities and programming in general. For students with little to no experience, the main goal is to make them comfortable using the tool and knowing where to look for help in the future. For more experienced students, the goal is to learn some tricks and best practices.

Credit & homework:

To enroll in the class, search for the section ID #96772 [here](#). ****You can only take this class credit/no credit, so please make sure you are appropriately enrolled.**** To get course credit, you must attend both days of class and complete the homework. Homework will not be graded, it will be self-check. Again, you learn programming best by doing! As mentioned below, if any students are in need of remote-learning options, I should be able to accommodate that request and will explain to those students how to get the attendance credit.

COVID-19 circumstances for Fall 2022:

Instruction at UCSD has resumed to in person. However as of now, masking is still required in instructional settings and there are various vaccination and testing requirements. For most up-to-date COVID-19 requirements please see [UCSD's return to learn](#). I expect everyone to follow these requirements and to adapt should the requirements change. Most importantly, let us all pledge to remain respectful, supportive, and adaptable to ensure that educational goals are met. All participants in the course are bound by the [UCSD Code of Conduct](#).

Please reach out to me directly if there are issues prohibiting your full engagement in the course so that we can find a workable solution.

Recording Instruction Policies:

I will be posting all course material in Canvas, and also sending announcements via Canvas, so please make sure you are able to access Canvas and contact me if you are not.

This course will be taught *in person*, and credit will require attendance. However, all lectures will be recorded and posted to Canvas in a timely manner so that they are available asynchronously to re-watch if desired. Lecture notes will always be available in the event that lecture recording fails.

If there are student(s) in need of remote attendance, please notify me *in advance*, and I will also provide synchronous remote participation (live, online audio and visual instruction via zoom) during the regular class times. If this is put in place, the course lectures will be initiated using Zoom. If Zoom fails for a given lecture and/or there are other online disruptions (due to WiFi drops, service drops, etc.) recordings will be available later on-line to be viewed asynchronously (links available through Canvas or contacting instructor). Please be aware that you will be recorded if you choose to turn on your video and audio.

Netiquette:

Hopefully with our small class, this will not be a major issue, however, given that these lectures will be recorded during class, methods to minimize background noise and to promote clear communications are important. Please minimize background discussions, and raise your hand with questions (that I will try to repeat for the recording). If you need to participate remotely, please keep your microphone on MUTE when you are not talking and consider using headphones. If you have a question, you can either “raise your hand,” put a note in the chat, or best, please do unmute yourself and just chime in!

References:*Textbooks:*

There are many textbooks covering MATLAB, so you might check out the selection at the UCSD bookstore. Here are a couple of suggestions:

- MATLAB, Third Edition: A Practical Introduction to Programming and Problem Solving by Stormy Attaway ([bookstore link](#)) ([Amazon link](#)). This one is well reviewed and recent.
- Physical Oceanography: A Mathematical Introduction with MATLAB by Reza Malek-Madani ([Amazon link](#)). This book seems useful for those interested in using MATLAB to solve numerical problems in physical oceanography. For those of you in different fields, look on-line, there are great books for applications to engineering, numerical methods, biology, etc.
- Introduction to Programming with MATLAB for Scientists and Engineers, second edition, by William W. Broenkow

Online tutorials:

- Mathworks (the company who developed MATLAB) has [online tutorials](#) as well as a detailed [users manual](#) is available.
- Regular Google searches work pretty well too as there is a very large user community and on-line responses to questions as well as application specific code sharing. Some code you will find through individual websites (e.g., [TEOS-10](#)) others you can find on the MATLAB code [file exchange](#).

MATLAB help files (online and offline)

MATLAB help files are available within the program and [online help](#). For functions, (e.g. the “plot” function), you can use the help command: `>> help plot` OR `>> doc plot`.

APPROXIMATE SCHEDULE

Monday September 19th (08:30-13:00)

Workspace, startup, basic math, matrices & arrays, element-by-element vs. matrix math, symbolic algebra, characters & strings, scripts, best practices, basic plotting, saving figures, saving & loading data, loading and saving .txt or ascii files, basic statistics, basic fitting, MATLAB dates, figure properties, 2D plotting of 3D data, mapping, scatterplots

Tuesday September 20th (08:30-13:00)

Indexing & logical, functions, if statements, for & while loops, vectorizing code for efficiency, structures, cell arrays, Native MATLAB NetCDF, SNCtools NetCDF