UNIVERSITY OF CALIFORNIA, DAVIS

BERKELEY • DAVIS • IRVINE • LOS ANGELES • MERCED • RIVERSIDE • SAN DIEGO • SAN FRANCISCO SANTA BARBARA • SANTA CRUZ

Dr. Helen Dahlke
LAND, AIR AND WATER RESOURCES
235 VEIHMEYER HALL
TELEPHONE: (530) 302-5358
FAX: (530) 752-5262
WEB: http://dahlke.ucdavis.edu

ONE SHIELDS AVENUE
DAVIS, CALIFORNIA 95616-8628

Dear Student,

Welcome to the HYD298 course "How to use Matlab to solve problems in environmental, climatological, and earth science research?" offered for the fall quarter 2013.

This course will begin on **September 26, 2013 at 9:00 AM in room HH124** on the UC Davis campus. The full course schedule and structure accompanies this letter.

The course provides an overview of the programming language and numerical computation software Matlab using applications in environmental, climatologic and earth sciences. We focuses on teaching and applying 'basics' of programming using eight modules and a final student project covering data import, loops and logical statements, data analysis, function files, general plotting, equation solving and interpolation.

Note, computer lab space is limited to 12 stations and students will be admitted on a first-come, first-served basis and will secure a spot by contacting Angie Nguyen (htgnguyen@ucdavis.edu). Additional space is made available for those able to supply Matlab on their own personal laptop. A student license of Matlab can be purchased via UC Davis (http://software.ucdavis.edu) but is not mandatory for this course.

On behalf of all involved, I look forward to an exciting course this upcoming semester. Please feel free to contact me directly if you have any questions before the course begins. I can best be reached by email at hdahlke@ucdavis.edu.

Thanks!

Helen Dahlke
Land, Air and Water Resources
University of California, Davis

How to use Matlab to solve problems in environmental, climatological, and earth science research?
2+1 optional credits

Course booklet, Fall Term 2013

Helen Dahlke
Dept. of Land, Air and Water Resources
University of California, Davis

Introduction

Modeling and simulation of environmental processes are fundamental analytical steps towards the development of an understanding of environmental systems. Models, numerical methods and analytical software such as Matlab are important tools to gain this knowledge. This graduate level course focuses on teaching "basics" of programming using the computation and simulation software Matlab. The course will consist of eight modules introducing fundamental programming concepts such as data import, loops and logical statements, data analysis, function files, general plotting, equation solving and interpolation and their realization in Matlab. The achieved programming skills will be taken into practice in a small, self-chosen student project at the end of the course in order to apply and test the students' programming skills on "real" problem sets. The intended learning outcomes and grading system for this course are detailed below.

Contact information

Helen Dahlke (*course responsible*), hdahlke@ucdavis.edu, tel. (530) 302-5358

Course structure

The course and corresponding credits are divided into the following blocks:

- Programming Modules, 2 credits
- Student Project Work, 1 credits (optional)

Literature

There is no required textbook. For those interested, the following is recommended literature for this course:

Introduction to Programming And Numerical Methods In Matlab. Otto, S. and James D.P. (2005). Springer-Verlag London Ltd.

Intended Learning Outcomes

By the end of this course the student is expected to:

- Recognize and explain the basic terminology used in Matlab programming.
- Demonstrate understanding of the basic principles of programming by solving different problems and processing diverse data sets.
- Generate, compile and debug original Matlab scripts and function files to conduct data analysis (e.g., basic calculus and statistical analysis) and modeling simulations.
- Design coherent programming approaches (i.e. goal oriented, computationally efficient) and apply these in environmental, climatological and earth sciences research applications.

Teaching/Learning Activities

The intended learning outcomes for this course will be achieved using self-contained lab modules to teach programming skills and a student designed project to demonstrate ability to apply the achieved programming skills to real problems. The following section summarizes these activities and the relation to the overall structure of the course. The course is graded based on the pass/fail system (S/U). However, in order to pass, active participation, class attendance, and completion of in-course exercises and the student project are expected.

Course Schedule and Structure

Programming Modules

The following lists to date for each module and the theory covered in more detail:

1. Software introduction and Vectors — 23 September 2013
    a. Defining a vector
    b. Accessing elements within a vector
    c. Basic operations on vectors
2. Matrices — 3 October 2013
    a. Defining Matrices
    b. Matrix Functions
    c. Matrix Operations
3. Loops — 10 October 2013
    a. *for* Loops
    b. *while* Loops
4. Logical Expressions — 17 October 2013
    a. *if* statements
    b. *if-then-else* statements
    c. Boolean logic
5. Data Files — 24 October 2013
    a. Saving and Recalling Data
    b. Saving a Session as Text
    c. Stylized Read/Write
6. Plots — 31 October 2013
    a. Plots of different types
    b. 3D Plotting
7. Executable Files — 7 November 2013
    a. Curve fitting
    b. Basic statistics
8. Subroutines — 14 November 2013
    a. Function files
    b. Modular Programming

These modules are designed to be self-contained while still scaffolding the learning process. Participation and attention during the modules is fundamental to receive the course credits. If you miss a module, you are responsible catching-up on the missed material before attending the next module.

During each module the learning material will be presented in a short 30 min introductory lecture followed by a 2-3 hour session during which you will practice the new material in various exercises. The goal is to complete all exercises in class. Completed exercises have to be handed in latest one week after the assignment date on the course webpage on Smartsite.

Student Project Work

The project work aims at applying your new or improved programming skills for ‘real’ problems. Students that have completed all modules will propose a short project to the course responsible(s). This project will use the student’s own data and/or be connected to the students own research. As such, what comprises the ‘completion’ of a project will vary from project to project. Most often, the production of a key result (i.e., a key figure or graphic) will be enough to gauge project completion.

<u>Projects should be completed within about 2-3 weeks from the end of the last module.</u>