

## **MSL S494: Computer Programming for Scientific Applications Class Syllabus**

Spring 2015

Instructor: Dr. Seth Danielson

Office: 112 O'Neill

Tel: (474) 7834

email: sldanielson@alaska.edu

Class location: 214 O'Neill

Class times: MWF 11:45-12:45

Office hours: MWF 1-2 pm

3 credits

### **Course Description:**

Catalog Description: MSL S494; Computer programming for scientific applications; 3 Credits Offered Spring Odd-Numbered Years. Introduction to scientific programming techniques and applications. This MATLAB-based course will cover programming fundamentals, input/output operations, and mapping and other data visualization techniques. Students will work with NetCDF and OpenDAP protocols and remote large-volume data repositories. No prior programming experience required. Prerequisites: Senior or graduate level standing.

This course provides a practical introduction to scientific computer programming techniques and applications. It is intended for students with little or no prior computer programming experience but who have a need to manipulate and graphically depict large volume datasets that fall beyond the capability of typical spreadsheet operations. We will cover programming fundamentals, file input/output operations, working with large-volume data repositories (such as oceanographic or atmospheric reanalysis numerical model output files) and data visualization techniques. Although we will base the instruction and homeworks on the MATLAB programming language, the techniques learned here can be applied to many other platforms.

### **Course Overall Goal:**

Students will gain a fundamental knowledge of computer programming techniques that will enable them to effectively handle, analyze, and visualize arbitrary data. Students will be well prepared for the programming requirements within MSL 620 Physical Oceanography and MSL F632 Oceanographic Data Analysis and Visualization.

### **Student Learning Outcomes:**

Students who take this class, participate, do the homework, and attend regularly can develop the following skills:

- Understand basic computer programming techniques including loops, conditional statements, sorting, and mathematical operations.
- Understand how to identify and fix programming errors.
- Understand how to read in ASCII and NetCDF data files.
- Understand how to access remote data archives using the OpenDAP interface protocol.
- Understand how to export the results of computations into a formatted ASCII output file.
- Improve student's ability to generate publication-quality data-based graphics.

- Understand how to make a map and plot spatially explicit data on the map.

### **Instructional methods:**

This course is based on lectures, which will cover the major topics, emphasizing and discussing the important points. They are not sessions to regurgitate material already written in the text. Your personal participation is important; asking questions will help you overcome conceptual roadblocks and coding problems. It will help you learn more efficiently if you read the assigned material before lecture. We do have required and recommended textbooks and programming reference books. Readings will be regularly assigned. The course will follow these readings although in class we will emphasize select topics. Material will be conveyed by the assigned readings, standard lecture, in-class discussions, and in-class presentations by students. Weekly homeworks will ensure that students keep up to date with learning the material at hand.

### **Supplies Needed:**

Access to a computer (PC, MAC, UNIX, or Linux) with MATLAB installed (student version available at bookstore) or access to a UAF computer lab that has MATLAB. Note that *one copy of the student version of MATLAB costs \$100* but this copy holds licenses for installation on *two* computers so you can share with a classmate and cut your costs in half if you decide to purchase a copy. Students who do not wish to purchase MATLAB can use a computer that has a UAF site license. Within SFOS, there are four computers in the 2<sup>nd</sup> floor O'Neill undergraduate lab, four computers in the SFOS Arctic Health computer lab and three computers in the 1<sup>st</sup> floor O'Neill oceanography commons.

### **Required Text:**

Author: Attaway, S.,

Title: MATLAB: A Practical Introduction to Programming and Problem Solving.

Edition/Publisher: Third Edition. Elsevier. San Francisco.

### **Recommended supplementary readings: basic texts and references**

- The MathWorks. The official website for MATLAB is at <http://www.mathworks.com/>. Various tutorials can be found under Support > MATLAB > Demos and Webinars. User guides can be found under Support > MATLAB > Documentation.
- Hahn B., and D. Valentine, 2013. Essential Matlab for Engineers and Scientists: 5<sup>th</sup> Edition, Academic Press. Waltham, MA. Available at the UAF Bookstore.
- MATLAB online glossary: [http://people.sc.fsu.edu/~jburkardt/html/matlab\\_glossary.html](http://people.sc.fsu.edu/~jburkardt/html/matlab_glossary.html)
- Hart, David and Clinton Wolfe, 1999. "Getting Started with MATLAB," Indiana University, University Information Technology Services, available online at: <http://www.indiana.edu/~statmath/support/bydoc/>.
- Still have a programming problem? A Google search is your friend.

### **Course Policies:**

#### **Homework:**

There will be weekly programming-based homework sets. You are encouraged to work with others on the homework, but please make sure that you understand the problems that you hand in. I will randomly ask students to present the homework on the board and the board presentation of the problems will be a part of your participation grade. You will hand in your homework

papers at the beginning of class and we will discuss problems after they have been graded and returned in the following class period.

### Complaints and Concerns:

You are always welcome to talk to me to express complaints and concerns about the class.

### Evaluation:

The course grade will consist of the following components. Final letter grades will be based on a standard scale: A=90 to 100%, B=80% to 89%, C=70% to 79%, D=50% to 69%, and F≤50%. As of Fall 2006, UAF has instituted a +/- scale to the grades, so the bottom and top 3 percentage points will fall within the '-' and '+' ranges, respectively. For example: 90-92% will be an A-, 93-96% will be an A, and above 97% will be an A+.

| Grade component | Fraction of final grade |
|-----------------|-------------------------|
| Attendance      | 15 %                    |
| Participation   | 15 %                    |
| Homework        | 30 %                    |
| Tests           | 30 %                    |
| Final Project   | 10 %                    |

This course is based on a combination of lectures, homeworks, and a final project. The instructor will evaluate participation and homeworks and assign grades appropriately. You will have one week to complete and return the homework sets to the instructor. *Late problem set grades will be reduced by 10% for each class day they are late.* Attendance will be used for evaluation with two missed classes excused. Evaluation of student participation in classes will be based on your answers to in-class questions regarding material covered in the previous class. There will be no formal exam during the final week. However, there will be three in-class tests (each worth 10% of your final grade) designed to test your comprehension of the subject matter. A missed test will be given a 0 grade but prior arrangement with the instructor can be arranged to reschedule the test at another time.

**General Advice:** Computer programming is not something one reads and memorizes, rather it is something you learn how to do through hands-on practice. Attending lectures but not spending time interfacing with the computer will be disastrous for your performance on tests and homeworks. Expect that each piece of code will require multiple attempts to get the syntax right, to correct typing errors, and to get the logical flow correct. ***Think of programming as a game in which you are trying to solve a puzzle.***

Try the following study and coding procedures:

- A. Read the assigned material so you get a second description of how to go about programming. Browse through the reading material so that you are familiar with the various tools (commands) that are available to help you solve your problem.
- B. Listen carefully to the lecture and take notes, ask questions and participate. This is a substantial part of your grade and could mean the difference between a letter-grade in the end. Pay particular attention to the programming examples given in the lectures.
- C. There is a three-step process in coding. First you must think about an approach to solving the

programming “puzzle” at hand. It may be worthwhile to diagram the approach with a flow chart. Second, try writing code that implements the selected approach. Third, debug the program so that it actually works correctly.

**Support Services:**

The instructor is available for assistance during office hours (112 O’Neill Building, MWF 1-2 pm) or through individually scheduled appointments at other times. UAF Student Support Services Tutoring Center is located in 514 Gruening and is open Sunday-Friday. Walk-in and scheduled tutoring for one-on-one or small group (up to 3 students) is available. Additional information is at <http://www.uaf.edu/sss/tutoring-center/>.

**Disabilities Services:** The Office of Disability Services implements the Americans with Disabilities Act (ADA), and insures that UAF students have equal access to the campus and course materials. We will work with the Office of Disabilities Services (208 WHITAKER BLDG, 474-5655) to provide reasonable accommodation to students with disabilities. If you need assistance with completion of any form, contact UAF Disability Services by email at [uaf-disabilityservices@alaska.edu](mailto:uaf-disabilityservices@alaska.edu), by phone at (907)474-5655, or by TTY at (907)474-1827.

**Academic Conduct:**

Plagiarism and cheating are matters of serious concern for students and academic institutions. The UAF Student Code of Conduct defines academic standards expected at the University of Alaska Fairbanks, and these will be followed in this class. The UAF Student Code of Conduct can be found at: <https://www.uaf.edu/catalog/current/academics/regs3.html>.

## MSL494 SPRING 2015 CLASS CALENDAR:

- F 1/16:** Class introduction, syllabus; Description of Matlab Environment, Variables, Use of MATLAB help. Basic dos and don'ts. Read Chapter 1 of text.
- M 1/19:** No class (Civil Rights day)
- W 1/21:** HW1 assigned. Relational and logical operators, Conditional statements. Read Ch. 2.
- F 1/23:** Conditional statements continued, useful built-in functions.
- M 1/26:** HW1 due, HW2 assigned. Adopting good programming habits, commenting, and error checking. Read Ch. 3.
- W 1/28:** Review HW1 issues. Creating and manipulating matrices.
- F 1/30:** Matrix math, Matrix functions, colon operator, dot multiplication, cross products.
- M 2/2:** HW2 due, HW3 assigned. 2-D plots: line, scatter, subplots, log-log. Read Ch. 4.
- W 2/4:** Review HW2 issues. Creating publication quality graphics, control of figure characteristics.
- F 2/6:** Saving plots and controlling graphic export characteristics.
- M 2/9:** HW3 due. Loops. Debugging loops. Read Ch. 5.
- W 2/11:** Review HW3 issues. Multiply embedded loops. Review for Test 1.
- F 2/13:** \*\*\* TEST #1 \*\*\*
- M 2/16:** HW4 assigned. More Advanced uses of matrices: indexing, finding subsets. Read Ch. 11.
- W 2/18:** 3-D arrays, meshgrid, shiftdim.
- F 2/20:** 3-D plots (mesh, surf, contour), More control of figure characteristics, controlling color schemes.
- M 2/23:** HW4 due, HW5 assigned. Importing and exporting data. Read Ch. 6.
- W 2/25:** Review HW4 issues. Reading and writing formatted ASCII files.
- F 2/27:** Checking algorithms: developing test cases. User-defined functions.
- M 3/2:** HW5 due, HW6 assigned. Graphical User Interfaces. Read Ch. 13.
- W 3/4:** Review HW5 issues. More about GUIs and GUI control.
- F 3/6:** Data types and data type conversions. Working with character strings.
- M 3/9:** HW6 due. Reading NetCDF files. Read Ch. 8.
- W 3/11:** Review HW6 issues. Writing NetCDF files. Review for Test 2.
- F 3/13:** \*\*\* TEST #2 \*\*\*
- Week of 3/16/2015: SPRING BREAK - NO CLASSES**
- M 3/23:** HW7 assigned. Mapping with M\_MAP: introduction, installing, projections. Read M-Map handout.
- W 3/25:** Mapping with M\_MAP: using hi-resolution bathymetry and coastline datasets.
- F 3/27:** Mapping with M\_MAP: plotting vectors and contours on maps

- M 3/30:** HW7 due, HW8: propose final project. Animations: Flat graphics files with Javascript controls. Read Ch. 9.
- W 4/1:** Review HW7 issues. Animations: creating and structure all within Matlab
- F 4/3:** Animations: controlling, and exporting.
- M 4/6:** Final project proposal due, HW9 assigned. Accessing data through the internet: urlwrite. Read Ch. 7.
- W 4/8:** Review HW8 issues. Accessing data through the internet: OpenDAP.
- W 4/10:** Accessing data through the internet: OpenDAP, continued.
- M 4/13:** HW9 due. Homework: work on final project. 1- and 2-D interpolations. Read Ch. 14.
- W 4/15:** Review HW9 issues. Regressions and correlations.
- F 4/17:** Fitting polynomials to data.
- M 4/20:** Homework: work on final project. Embedded control statements (eval). Read Ch. 12.
- W 4/22:** Working with dates and times. Review for Test 3.
- F 4/24:** \*\*\* TEST #3 \*\*\*
- M 4/27:** Homework: work on final project. Other MATLAB tools and resources. Read Ch. 10.
- W 4/29:** Useful tricks and techniques: some final miscellaneous helpful hints.
- F 5/1:** Final project presentations (10 min. ea).
- M 5/4:** Final project presentations, continued. Class wrap-up.