Beyond the Mouse: Computer Programming and Automation for Geoscientists

GEOS 436/636
Fall 2014
Tu-Th 3:40-5:40, Reichardt 316 (G&G Computer Lab)

Instructor: Jeff Freymueller
x7286 Elvey 413B jeff.freymueller@gi.alaska.edu

TA: Kyle Smith
Elvey 307 or Reichardt 312 kksmith7@alaska.edu
Office Hours: To Be Determined

Last Updated: September 3, 2014

Contents

Introduction
Course Topics
Prerequisites
Textbook
Student Learning Outcomes
Grading Scheme
Policies and Make-up Labs
Detailed Schedule

Introduction

In the (geo)sciences - as in many other disciplines - we collect data which need to be analyzed in ways that depend on the problem posed. The ability to modify your working environment according to your needs instead of having it dictate how you approach a problem is invaluable. This is especially true in a setting that is supposed to generate fresh knowledge. Also, we do not want to waste time by repeating the same steps again and again, and ... again. Doing repetitive, boring tasks leads to errors. A computer (the machine, and earlier the person) exists to perform such routines rapidly, reliably and repetitively: it takes in data,
manipulates the data following your commands, and produces a result. The point of writing computer programs is to automate an intellectual challenge that has been solved and make it reusable at all times - for yourself and ideally for others. 21st century scientific research frequently involves manipulation or analysis of very large data sets, or the development of numerical models; this work can only be used effectively by scientists who can make software tools themselves. Accordingly, the geophysics graduate curriculum now expects students to be able to write simple computer programs. This course will teach you the basic techniques and skills to do this.

This course will teach you how to make simple tools that will allow you to read in and massage data in exactly the way you want, and plot or visualize the results. We will start out manipulating your thinking, introduce you to programming in general, and then take off into specific working environments namely Unix/Linux and Matlab while teaching you how to map your data using GMT and create simple web pages by writing the HTML yourself. All of this is easier than you might think - you simply have to get up over the initial part of the learning curve. We will cover many things in a short amount of time, which means that we will give you many pointers that you can follow up on depending on your needs. There is a tremendous amount of reference material (and examples to adapt) available on the web. We encourage you to play with the tools we are teaching you to use beyond the course assignments, and do things with them that are fun for you. The more you do, the more you will learn.

The use of scripting and well-documented tools is not just my idea; it is widely accepted as a good practice. The paper "Ten Simple Rules for Reproducible Computational Research" by Sandve et al. (2013) is a good example.

**Course Topics**

- Basic computer programming concepts, using MATLAB
- Unix tools to enhance automation, including making figures and maps with GMT
- HTML and creating your own web page

**Prerequisites**

- GEOS 436: Senior standing or permission of instructor.
- GEOS 636: Graduate standing.

**Textbook**

Essential MATLAB For Engineers and Scientists, 4th edition.

**Student Learning Outcomes**

http://www.gps.alaska.edu/jeff/Classes/GEOS636.html
By actively participating in this course you will become significantly more proficient at:

- Breaking problems down into a series of steps.
- Organizing data and tools to make automated work easier.
- Writing and understanding how to read computer programs in MATLAB.
- Writing and understanding how to read Unix/Linux shell scripts.
- Making publication-quality maps and figures using GMT (Generic Mapping Tools).
- Using HTML and CSS for web pages.

Grading Scheme

This 2 credit class is pass/fail. The class assignments are primarily lab exercises, specifically computer programs written in the computer lab. We use software that is available to students at no cost (for use within the UAF network), so all students could also install and use it on their own computer if they wish. The computer lab is also available for students to use at other times, if they need to finish an assignment outside of lab. During the first third of the semester, additional short homework assignments will be given outside of lab.
lab (these do not require any particular computer or software).

Grading is based on weekly lab exercises, homework assignments, a final project, and the presentation of that project in the form of a web page or pages. There will be a total of 12 graded lab assignments, equally weighted, and all other assignments except for the final project itself are scored points equivalent to a lab assignment or a fraction of that.

**Graduate Students**

- Labs+Homework+Project Presentation 70% of total
- Final Project 30% of total
- Each Lab assignment 1 Lab
- Each Homework assignment 1/2 Lab
- Final Project Presentation 1 Lab

**Undergraduate Students**

- Labs+Homework+Project Presentation 70% of total
- Attendance and Participation 30% of total
- Each Lab assignment 1 Lab
- Each Homework assignment 1/2 Lab
- Final Final Presentation 1 Lab

Passing >= 65%

The homework and lab exercises consist of basic application of methods and practices presented in class. The labs help you apply things taught in class. The complexity of the labs varies. Usually they consist of a simple introduction problem to get you used to the environment, understand new commands, etc. In a second part you will apply this in a slightly more complex way to data, or simply write more complex code.

The final project will (hopefully) be specific to your research project. We want to encourage you to set up an efficient and safe environment in which you apply the methods and tools introduced in class.

Graduate students are expected to carry out a complete project within their own field of specialization (this can and should be something that helps them in their own research). The project will be presented in the form of a web page or pages, for which the student will write the HTML using the templates provided in class and used in one of the labs. Undergraduates will substitute a presentation of some of their own work from the labs in place of an independent project, also presented in the form of a web page or pages.

There are several styles of project that a student could take on, depending on their needs. Flexibility in this regard is beneficial for the students, as they learn more by doing more, and do more when they are excited about and see the relevance of the project. The project must be implemented in code using one of the tools used in the class (or a different tool with instructor permission). The students must turn in complete code, raw data files, etc, so that the instructor could run their code and replicate their results. Code must be adequately commented. All of the code and data files should be linked on the web page or pages. Sample projects include one of these, at a minimum: (a) reading in data and doing useful manipulation and visualization of the data; (b) constructing a coherent suite of scientific figures or visualizations of data; (c) developing and running a numerical model; (d) writing a program or programs to automate a task that must be done...
repeatedly (for example, a data processing or analysis task), and using this program to run a substantial amount of data.

In the beginning of the semester you will provide us with a snapshot of your project directory (If you have one). Send rudimentary data files, and any scripts/programs should be executable. If you do not have such a project directory, make one up! (Tell us it is made up). In that case, tell us how you would organize and name files, what kind of data they contain, and how you would store other information. You will do the same at the end of the term through your final project, and tell us how you improved or changed the organization to make working with your data easier to automate. If your project involved doing something totally new, you will tell us why you chose to organize things as you did.

Note: Everything will be done for this course as of December 11, just before Finals Week.

Policies, late assignments, and makeup-labs

You are subject to the UAF Student Code of Conduct. We will work with the Office of Disabilities Services (203 WHIT, 474-5655) to provide reasonable accommodation to student with disabilities.

Late assignment will be penalized by 25% off for each class period after the due date. Exceptions can be granted in advance for students who have University sponsored activities that will force them to miss class (for example, fieldwork or attending a conference for graduate students). Arrangements for this need to be made in advance.

Makeup versions of labs will be provided if we have a convincing reason to do so. All MATLAB lab assignments must be turned in before we begin the Unix shell and GMT parts of the course. Any makeup or late assignments must be completed prior to the final project presentations.

Detailed Schedule

Each lecture title is a link to a web page for that lecture or lab.

<table>
<thead>
<tr>
<th>Day</th>
<th>Date</th>
<th>Lecture Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thu</td>
<td>Sep 4</td>
<td>0. Getting set up, brief intro; READING: none</td>
</tr>
<tr>
<td>Tue</td>
<td>Sep 9</td>
<td>1. Thinking Programs; READING: Chapter 1, Skim Chapter 3.</td>
</tr>
<tr>
<td>Thu</td>
<td>Sep 11</td>
<td>1. Thinking Programs Lab;</td>
</tr>
<tr>
<td>Date</td>
<td>Event</td>
<td>Reading</td>
</tr>
<tr>
<td>------------</td>
<td>------------------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>Tue Sep 16</td>
<td>2. Variables; Guest Lecturer: Steven Holtkamp; Reading: Chapter 2 through 2.6, Chapter 6.1, 6.2, 6.3.</td>
<td></td>
</tr>
<tr>
<td>Thu Sep 18</td>
<td>2. Variables Lab; Reading: Chapter 2 through 2-6, Chapter 6.1, 6.2, 6.3.</td>
<td></td>
</tr>
<tr>
<td>Tue Sep 23</td>
<td>3. Variables and Functions; Guest Lecturer: Steven Holtkamp; Reading: Chapter 10.1-10.3, Chapter 11.1, 11.2, 11.4, 11.5.</td>
<td></td>
</tr>
<tr>
<td>Thu Sep 25</td>
<td>3. Variables and Functions Lab; Reading: Chapter 10.1-10.3, Chapter 11.1, 11.2, 11.4, 11.5.</td>
<td></td>
</tr>
<tr>
<td>Tue Sep 30</td>
<td>4. Control Structures; Reading: Chapter 2.7, 2.8, Chapter 5, Chapter 8.</td>
<td></td>
</tr>
<tr>
<td>Thu Oct 2</td>
<td>4. Control Structures Lab; Reading: Chapter 2.7, 2.8, Chapter 5, Chapter 8.</td>
<td></td>
</tr>
<tr>
<td>Tue Oct 7</td>
<td>5. MATLAB I/O 1; Reading: Chapter 2.10, 2.11, Chapter 4.</td>
<td></td>
</tr>
<tr>
<td>Thu Oct 9</td>
<td>5. MATLAB I/O 1 Lab; Reading: Chapter 2.10, 2.11, Chapter 4.</td>
<td></td>
</tr>
<tr>
<td>Tue Oct 14</td>
<td>6. MATLAB Plotting and Graphics; Reading: Chapter 7, Chapter 12.</td>
<td></td>
</tr>
<tr>
<td>Thu Oct 16</td>
<td>6. MATLAB Plotting and Graphics Lab; Reading: Chapter 7, Chapter 12.</td>
<td></td>
</tr>
<tr>
<td>Tue Oct 21</td>
<td>Live MATLAB 1; Guest Coders!; Reading: none.</td>
<td></td>
</tr>
<tr>
<td>Thu Oct 23</td>
<td>Live MATLAB 2; Guest Coders!; Reading: none.</td>
<td></td>
</tr>
<tr>
<td>Day</td>
<td>Date</td>
<td>Topic</td>
</tr>
<tr>
<td>-------</td>
<td>----------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>Tue</td>
<td>Nov 4</td>
<td>8. Unix Tools 2</td>
</tr>
<tr>
<td>Tue</td>
<td>Nov 11</td>
<td>9. GMT 1</td>
</tr>
<tr>
<td>Thu</td>
<td>Nov 13</td>
<td>9. GMT 1 Lab</td>
</tr>
<tr>
<td>Tue</td>
<td>Nov 18</td>
<td>10. GMT 2</td>
</tr>
<tr>
<td>Thu</td>
<td>Nov 20</td>
<td>10. GMT 2 Lab</td>
</tr>
<tr>
<td>Tue</td>
<td>Nov 25</td>
<td>11. Live Shell. No material to download; READING: none.</td>
</tr>
<tr>
<td>Thu</td>
<td>Nov 21</td>
<td>NO CLASS: Thanksgiving; Gobble gobble.</td>
</tr>
<tr>
<td>Tue</td>
<td>Dec 2</td>
<td>12. Debugging</td>
</tr>
<tr>
<td>Thu</td>
<td>Dec 4</td>
<td>13. HTML</td>
</tr>
</tbody>
</table>
Tue Dec 9
13. HTML Lab; READING: none.

Thu Dec 11
Student Presentations; Graduate student projects presented via website.

END OF CLASSES

Dr. Jeffrey T. Freymueller  
Professor of Geophysics  
Geophysical Institute  
University of Alaska, Fairbanks  
Fairbanks, AK 99775-7320

jeff.freymueller@gi.alaska.edu  
Phone 907-474-7286  
Fax 907-474-7290  
Office 413B Elvey