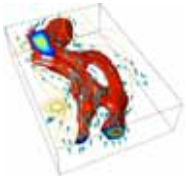
CIS 401: Applied Scientific Computing with MATLAB



Andrew Pershing
3134 Snee Hall
ajp9@cornell.edu
255-5552

Outline

- Course Description
- Details
- Policies
- Intro to CIS Tools Curriculum
- Role of Computing in Science and Engineering
- Basic Concepts

Course Goals

- This course will:
  - Introduce the basic functionality of MATLAB
  - Demonstrate its utility in scientific research
  - Identify interesting concepts and useful techniques in scientific computing

By the end of the course, you should have the skills necessary to apply MATLAB to your research and learn how to extend its capabilities

Syllabus

1. Course intro and basic concepts
2. Intro to Matlab: the workspace
3. Matlab fundamentals: arrays, & simple plots
4. Matlab programming: loops and conditionals
5. Text processing and a survey of Matlab
6. Improving performance
7. Statistics and simple plots
8. Applied Scientific Computing I: Simulation
9. Applied Scientific Computing II: ODE/Optimization
10. File I/0
11. Applied Scientific Computing III: Linear systems
12. Class projects

Course Ungoals

- This course will NOT:
  - Teach you how to program (try CS 100m)
    - You should be comfortable writing programs in some language and be familiar with the following concepts:
      - Programs vs. algorithms
      - Iteration
      - Conditionals and logic
      - Recursion
      - Subroutines, variables, and scope
  - Teach you numerical methods (CS 322,421, 62X)
  - Cover everything in MATLAB

Course Business:

- http://www.cs.cornell.edu/Courses/cs401/2003sp
  - Contains syllabus, lecture notes, examples, homework
- Office Hours
  - Tuesday & Wednesday, 3-4 in 3134 Snee (or by appointment)
- Registration: CIS 401 or COM S 401
  - get my signature or CS Undergrad office (303 Upson)
  - Number:641-522
  - S/U only, 1 credit
  - Last day to add/drop: Monday, September 8 !

Requirements

- Reference Text: Hanselman and Littlefield *Mastering Matlab 6*
  - No required reading, but this is a great reference
- Find a computer with MATLAB:
  - Check departmental labs--good site licensing for Cornell machines
    - http://www.cit.cornell.edu/software/licenses/matlab.html
  - ACCEL in Carpenter Hall
  - Upson, Carpenter, and Dickson Labs
  - Buy student version

Course Policies

- 4 assignments: 1 per week, due Wednesday, 5PM by e-mail
- If you complete each assignment on time and demonstrate a basic command of the material, you will pass!
- Course policies are strict:
  - A direct consequence of the "mini-course" format
- This course operates as a contract between you and me

The Contract

- I agree to:
  - Begin and end lecture on time
  - Put lecture notes on website prior to lecture (usually night before)
  - Be available during office hours
  - Make the assignments of reasonable length (2-4 hours) focusing on material from lectures

Assignments

- Assignments will be fairly short
  - "Essential Knowledge"
    - short answer questions from lecture material
  - A programming task
    - opportunity to apply Matlab
- Problem Set 4 will be special
  - You will have to come up with a programming task, relevant to your area of expertise
  - Proposals will be due with PS3, code with PS4
  - Brief presentation on last day

The Contract

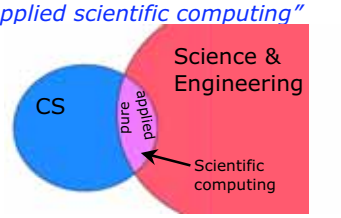
- By registering for the course, you agree to:
  - Arrive on time
  - Participate in the course by asking questions and coming to office hours
  - Turn in your assignments on time
    - Late work will not be accepted and will jeopardize you chance of passing!
    - The only exceptions are for documented, university-sanctioned reasons such as severe illness or by prior arrangement made w/ me 3 days before (includes religious holidays, sports, etc.)

CIS and FCI

- *Cornell University has recognized that computing and information science has emerged as a key enabling discipline vital to nearly all of its scholarly and scientific pursuits.*
- *The Faculty of Computing and Information is founded on the recognition that the ideas and technology of computing and information science are relevant to every academic discipline.*
- *We are united in the need to bring together a core of faculty in this field from across the traditional colleges.*

CIS Tools Curriculum

- CIS 401 is the first in a series of courses designed to teach
  "applied scientific computing"



CIS Tools Curriculum

- "Pure" Scientific Computing
  - Focus is on algorithms for general problems such as optimization, linear systems, differential equations
  - Concerned with accuracy, stability, and efficiency of these algorithms
- "Applied" Scientific Computing
  - How to apply general algorithms to solve scientific problems
  - Algorithms are "black boxes" that we string together to get our work done
  - Applied SC is also concerned with data

CIS Tools Curriculum

- Fall: MATLAB
  - 401: the basics
  - 402: visualization (starts September 30)
- Spring: General tools
  - 403: Developing scientific computer programs (compilers, debuggers, managing large projects, designing programs)
  - 404: Numerical libraries (compiling and linking, static vs. dynamic libraries, BLAS, LAPACK, MPI)

Role of Computing in Science and Engineering

- Scientists have been computing for centuries, well before digital computers
- Digital computers allow us to do thing faster, but often the ideas are ancient
- Example: Velocity from pressure data

Geostrophy

- Measuring the velocity of atmosphere and ocean is difficult, but observing pressure is easy
- Fortunately, velocity can be determined from pressure using the *geostrophic* relationship:

Pressure gradient = Coriolis force (Earth's rotation)

$$1/\rho \nabla p = f \begin{bmatrix} v \\ -u \end{bmatrix}$$

$$f = 2\Omega \sin\phi \qquad \Omega = 7.29\text{e}-5$$

Geostrophy

- An alternative to pressure is sea-surface height
- SSH can be measured by satellites

Geostrophy

- Use hydrostatic equation:

$$p = \rho g Z$$

to introduce SSH (Z) into geostrophic equation:

$$g \nabla Z = f \begin{bmatrix} v \\ -u \end{bmatrix}$$

Geostrophy in MATLAB

- MATLAB allows us to compute the velocity:

$$u = -g/f \frac{\partial Z}{\partial y} \qquad v = g/f \frac{\partial Z}{\partial x}$$

$$f = 2\Omega \sin\phi \qquad \Omega = 7.29\mathrm{e}-5$$

in only a few lines

- Can examine results graphically

So, what's the point?

- Theme of the FCI is that computing is general
- The geostropic calculations are a specific instance of the general scientific computing process.

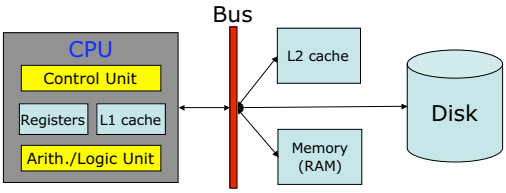
Scientific Computing Process

	Data →	Program →	Output
Currents	SSH	Geostropic eq.	U,V,plot
Weather	T,V,M	Finite diff.	T,V,M in future
Bioinfomatics	ATCGCGTA…	Search for genes	Location of genes
Electronics	Signal	FFT	Plot of spectrum
Economics	Historic Stock Prices	Trading Strategy	Money Made/Lost

- It is possible to do all of these things in Matlab, and most of them are easier.

Computer Science Review

- What is a computer?

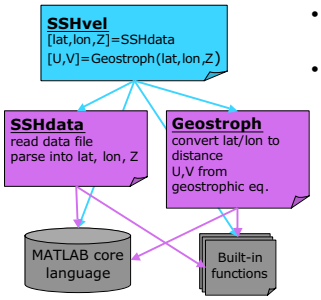


  - Program is a series of instructions that tell the processor how to manipulate system
    - read/write to memory, disk
    - calculate

Basic Concepts

- Algorithm vs. Program
  - A program is a series of commands on a computer
  - An algorithm is a step-by-step procedure to compute something
    - A program should be an algorithm
    - An algorithm does not require a computer
      - Ex: To get to my office,
        1) Stand up and leave this room
        2) Walk west to stairs
        3) Go down stairs to door
        4) Turn left, then down stairs to left
        5) Cross parking lot to right & enter Snee Hall
        6) Go up two flights of stairs to 3rd floor
        7) Turn left out of stairway, my office is first on left

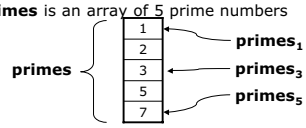
Basic Concepts



- Programs are constructed from subroutines (or classes or functions)
- Subroutines are constructed from
  - variables--name given to a specific region of memory
  - commands--instructions that change value of variables

Variables

- have a name
- have value (numeric, text)
- have type (integer, floating point, char, etc.)
- Variables can also be data structures (arrays, structs)
  - Arrays: group of data that can be accessed through subscripts
  - Ex: **primes** is an array of 5 prime numbers



Commands

- Simplest command is assignment
  - a=1; %value of a is 1 after this statement
  - b=2+a; %value of b is constructed from a
  - a=2; %changes value of a, what is value of b?
- Control (if-then-else) or iteration (for, while)
- Call another subroutine

Commands

- Programming languages differ in syntax
  - syntax=how commands are written
  - Java:
    - for(int j=0;j<n;j++){ //code }
  - Matlab:
    - for j=1:n; %code; end
- But, all programming languages do the same thing
  - If you know one language, you can understand programs written in most others.

Developing programs

- Problem: you have a task you want your computer to do, and you need to develop a program to perform it
- Several approaches, iterative refinement is a simple and effective technique
  1. Describe--what will your program/subroutine do?
  2. Divide--what are the essential tasks?
  3. Repeat--subdivide tasks if possible.
- The idea is to move from some description of the task (English, math, pictures) to a series of commands

Iterative Refinement

I. Do Laundry
   A. Wash
   B. Dry
   C. Fold

Do Laundry

A. Wash
   1. Get clothes
   2. Place in washer
   3. Configure washer
   4. Start
B. Dry
   1. Move from washer to dryer
   2. Configure dryer
   3. Start
C. Fold
   1. Remove from dryer
   2. If (shirt) then
      a. Fold shirt

   else
      b. Fold pants

Main Point of Design

- Think before you code!