Introduction to Computing
Using Matlab

CS 1112
Dr. Curran Muhlberger
Dr. K.-Y. Daisy Fan

http://www.cs.cornell.edu/courses/cs1112/

Discussion starts *this* week
in Upson 225 lab (not classroom listed in Student Center)
Who is Dr. Muhlberger?

And why should scientists learn to code?
Who is Dr. Daisy Fan?

- Interest in **optimization**—what is the “**best**” way to operate a system given constraints and uncertainties?

- Other courses:
  - Intro to computing using Python
  - Optimization with metaheuristics

- **Author:** *Insight Through Computing: A Matlab Introduction to Computational Science and Engineering* with C. F. Van Loan

About you … in CS1112

- Undergraduates, graduates, researchers, and professionals who want (need) to learn computing
- No prior programming experience needed but some “mathematical maturity”
- You will …
  - Learn programming concepts and good programming habits
  - Practice problem analysis and decomposition
  - Become a code detective—find out “whodunit”
- Develop a “spirit of experimentation”
  - Not thoughtless trial-&-error but purposeful try-then-analyze
- Why should you learn computing?
Today’s lecture

- An illuminating problem
- What is computer programming?
- CS1112 philosophies & syllabus
- Choosing between CS1112 & CS1110
- Course logistics/policies (highlights)

Discussion starts this week in Upson 225 lab (not classroom listed in Student Center)
An illuminating problem: computing square roots

- Suppose $A > 0$

- **Observation:** If $A$ is the area of a square ... then I can just measure the side length—that is $\sqrt{A}$

- **Solution idea:** Make a square with area $A$

- **Real task:** Make a sequence of increasingly square rectangles, each with area $A$
How to make a rectangle “more square”?

- If a square and a rectangle both have area $A$ ...

- then $\sqrt{A}$ is between the length and width of the rectangle
An improvement strategy

Current:

Recipe: \[ L_{\text{new}} = \frac{L + A/L}{2} \]

Next:

The average of the length and width.
A Matlab program to make “increasingly square” rectangles

% The first rectangle...
L1 = A;
W1 = 1;

% The second rectangle...
L2 = (L1+W1)/2;
W2 = A/L2;

% The third rectangle...
L3 = (L2+W2)/2;
W3 = A/L3;

% and so on...
Demo!
Some conclusions from square root finding problem

- It paid to have a geometric sense
- A complicated computation was reduced to a sequence of elementary calculations
- A program is like a formula (or sequence of formulas)
Course Goals

- Develop your “computational senses,” senses that you need in computer problem-solving

- Develop a facility with the Matlab programming environment
A sense of geometry
A sense of complexity

What is the best itinerary to visit Boston, Miami, LA, Dallas?

3! = 6 possibilities

Add Seattle, NYC Austin, Denver

7! = 5040

If a computer can process 1 billion itineraries a second, how long does it take to solve a 20-city problem?

Nearly a century...
A sense of approximation & error

\[ \frac{1}{3} = .33333... \]
A sense of randomness and probability

Random walk
Brownian motion in water
Course Goals

- Develop your “computational senses,” senses that you need in computer problem-solving
- Develop a facility with the Matlab programming environment
Computer problem-solving

Key: Algorithmic thinking

Algorithm:
A step-by-step procedure that takes you from a prescribed set of inputs to a prescribed set of outputs

Program:
The algorithm expressed in a specific language, e.g., Matlab
Computer problem-solving — Programming

- Developing instructions for the computer to execute (in order to solve some problem)
- The steps must be logical
- Use a particular language and follow the rules of the language (grammar/syntax)
Example: *Adding songs from the internet to your music library*

- Find a website with MP3 or other audio files
- Register with the music site, if required for music downloading. *(Don’t steal music.)*
- Click on the music file to download it onto your computer
- Drag the file to your library

Reference: iTunes
Example: *Adding songs from the internet to your music library*

- Drag the file to your library
- Click on a music file to download it onto your computer
- Find a website with MP3 or other audio files
- Register with the music site, if required for music downloading.
  (Don’t steal music.)

These steps are out of order! Illogical!
Example: *Adding songs from the internet to your music library*

- Find a website with MP3 or other audio files
- Register with the music site, if required for music downloading. (Don’t steal music.)
- Click on the music file to download it onto your computer
- Drag the file to your library

Reference: iTunes
Example: *Adding songs from the internet to your music library*

- Find a website with MP3 or other audio files
- Register with the music site, if required for music downloading. (Don’t steal music.)
- To download the computer onto your music file click it on
- file Drag your librAry to

*Bad grammar (syntax)!*
Computer programming is …

- a tool used by computer scientists, engineers, and other professionals
- not all of computer science

Think about astronomy: Telescope is a tool used by astronomers; astronomy is not about telescopes…
Course Goals

- Develop your “computational senses,” senses that you need in computer problem-solving

- Develop a facility with the Matlab programming environment
Matlab is the vehicle we use

With the Matlab environment, you can easily
- Develop programs
- Display results & ideas graphically
- Interact with large data sets (process text, image, and other files)

Matlab has extensive libraries of mathematical, statistical, simulation, and other tools. It is heavily used in engineering & sciences, both in industry and academia.
Demo!
Engineering students take one of these courses:

- **CS1112** – this course, Matlab
- **CS1110** – Python

Each course satisfies the Engineering Computing Requirement. In **CS1112**, you will learn procedural programming in depth and be introduced to object-oriented programming.

Each course can serve as the prerequisite for **CS/ENGRD 2110 Object-Oriented Programming & Data Structure**
CS1112 has a focus on *computational science & engineering*

Approximation, randomness, model building, sensitivity of models

- Lecture examples and homework illustrate above themes
  - Edge detection
  - Ranking web pages
  - Congressional apportionment
Some past programming assignments

- Find the US population center from census data
- Organize protein data using structure arrays
- Mozart’s musical dice game

Root finding tool

Draw the random Mondrian

Draw the “Betsy Ross Flag”

Path distance tool (like that in Google Earth)
CS1112
- No prior programming experience
- One semester of Calculus
- *Focus on computational science & engineering*
- Matlab

CS1110
- No prior programming experience
- No Calculus
- *Focus on software development*
- Python
CS1112 requirements

- Attend lecture
  - Laptops **not** required – stage-right section is screen-free zone

- Attend discussion—get individual attention/help on weekly exercises!

- Monitor course announcements on website

- Do homework: 6 programming projects

- Take 2 prelims and a final exam at their scheduled times

- Answer in-class quizzes (use your clicker)

- Adhere to the Code of Academic Integrity

4 credits → 4x3 = 12hrs/week

In class: 2hr lec + 1 hr dis = 3 hrs/week

**Outside class: 9 hrs/week**
Learning and integrity

- Learning is something *you* do; we can only facilitate
- Computers facilitate duplication; duplication does not facilitate learning
  - No value in being a delivery vehicle for the write answer
  - In real engineering, there is no “right answer” to copy
- Respect yourself and others
  - Craft your own programs; build confidence in your own answers
Grading

- **Best five* of six projects (25%)**
  - Your lowest-scored project is eligible to be dropped only if you scored at least 50% on it

- Discussion exercises (4%)

- In-class polling (1%)

- Prelim 1 (20%)

- Prelim 2 (20%)

- Final exam (30%)
Course Materials

- **Insight Through Computing**
  A Matlab introduction to Computational Science and Engineering

- **An iClicker clicker**
  (or mobile app)

- **MATLAB Student Version**
  Download your own copy, use *MATLAB Online* (web browser based), or use public computer labs (Engineering Quad and RPCC)

FREE for students!
What to do now?

- Pick a course
  - Take CS1112 or CS1110
    (add/drop: lecture and discussion and optional AEW)
- Check course website
- Start reading (see listing on course website)
- Attend discussion in the lab (Upson 225) today or tomorrow
- Attend the discussion in which you are enrolled!
# CS1112 Discussion Sections – start today

<table>
<thead>
<tr>
<th>Sec #</th>
<th>Time</th>
<th>Room</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>T 12:20-1:10p</td>
<td>UPS 225 lab &amp; HLS 401</td>
</tr>
<tr>
<td>202</td>
<td>T 1:25-2:15p</td>
<td>UPS 225 lab &amp; HLS 401</td>
</tr>
<tr>
<td>203</td>
<td>T 2:30-3:20p</td>
<td>UPS 225 lab &amp; HLS 401</td>
</tr>
<tr>
<td>205</td>
<td>W 10:10-11:00a</td>
<td>UPS 225 lab &amp; HLS 401</td>
</tr>
<tr>
<td>206</td>
<td>W 11:15a-12:05p</td>
<td>UPS 225 lab &amp; HLS 401</td>
</tr>
<tr>
<td>207</td>
<td>W 12:20-1:10p</td>
<td>UPS 225 lab &amp; HLS 401</td>
</tr>
<tr>
<td>208</td>
<td>W 1:25:2:15p</td>
<td>UPS 225 lab &amp; HLS 401</td>
</tr>
<tr>
<td>209</td>
<td>W 2:30-3:20p</td>
<td>UPS 225 lab &amp; HLS 401</td>
</tr>
</tbody>
</table>

Discussions are held in UPS (Upson) 225 lab the first two weeks.