Introduction to Computing
Using Matlab

CS 1112
(CS1142)
Dr. K.-Y. Daisy Fan

http://www.cs.cornell.edu/courses/cs1112/
Today’s lecture

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

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

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...
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 100-city problem?
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 100-city problem?

About a century…
A sense of approximation & error

\[
\frac{1}{3} = .33333\ldots
\]
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.)
- Click [ ] to download [ ]
- 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…
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.
Engineering students take one of these courses:

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

Each course satisfies the Engineering Computing Requirement. In **1112** 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

Path distance tool (like that in Google Earth)

Draw the random Mondrian

Draw the “Betsy Ross Flag”
Switched from Java to Python because Python is a friendlier and more modern object-oriented language.

Python is more relevant to non-computer scientists than Java—numerical libraries are available.

Matlab and Python are just different vehicles we use to travel the “computational landscape.”

Different scenery along the way
Both vehicles can get you there
<table>
<thead>
<tr>
<th>CS1112</th>
<th>CS1110</th>
</tr>
</thead>
<tbody>
<tr>
<td>- No prior programming experience</td>
<td>- No prior programming experience</td>
</tr>
<tr>
<td>- One semester of Calculus</td>
<td>- No Calculus</td>
</tr>
<tr>
<td>- Focus on computational science &amp; engineering</td>
<td>- Focus on software development</td>
</tr>
<tr>
<td>- Matlab</td>
<td>- Python</td>
</tr>
</tbody>
</table>
CS1112 requirements

- Attend lecture
- Attend discussion—get individual attention/help on weekly exercises!
- Monitor course announcements on website
- Do homework: best 5 of 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
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 quizzes (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

- MATLAB Student Version (2008 or later) optional because you can use it in the public labs
MATLAB software: to buy or not to buy, that is the question

<table>
<thead>
<tr>
<th></th>
<th>Convenience</th>
<th>Speed</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Use public lab</strong></td>
<td>😞 Need to go to the lab (most labs are open 24-7)</td>
<td>☺ Some lab machines are fast; some are slow</td>
<td>☻☺ Free!</td>
</tr>
<tr>
<td><strong>Your own copy</strong></td>
<td>☻ Available on your own machine (5+ months)</td>
<td>☻ Depends on your machine</td>
<td>☹☹ You have to purchase (at least $49)</td>
</tr>
<tr>
<td><strong>Remote access</strong></td>
<td>☹ Extra work to download and use supporting software and deal with file transfer</td>
<td>☹ Slow</td>
<td>☻☺ Free!</td>
</tr>
<tr>
<td></td>
<td>☻ Available on your machine with Internet connection</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Consulting & Computing

- Consulting in ACCEL Green Room (Engineering Library, Carpenter Hall). Check course website for hours.

- Some public labs that have Matlab:
  - Hollister 464 lab
  - ACCEL
    - (Carpenter Hall, former Engrg Lib)
  - North campus: RPCC
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 (HLS 464 lab) on Tues/Wed
- You must 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>HLS 464 lab &amp; HLS 314</td>
</tr>
<tr>
<td>202</td>
<td>T 1:25-2:15p</td>
<td>HLS 464 lab &amp; HLS 401</td>
</tr>
<tr>
<td>203</td>
<td>T 2:30-3:20p</td>
<td>HLS 464 lab &amp; HLS 401</td>
</tr>
<tr>
<td>204</td>
<td>T 3:35-4:25p</td>
<td>HLS 464 lab &amp; HLS 401</td>
</tr>
<tr>
<td>205</td>
<td>W 10:10-11:00a</td>
<td>HLS 464 lab &amp; HLS 401</td>
</tr>
<tr>
<td>206</td>
<td>W 11:15a-12:05p</td>
<td>HLS 464 lab &amp; HLS 401</td>
</tr>
<tr>
<td>207</td>
<td>W 12:20-1:10p</td>
<td>HLS 464 lab &amp; HLS 401</td>
</tr>
<tr>
<td>208</td>
<td>W 1:25:2:15p</td>
<td>HLS 464 lab &amp; HLS 401</td>
</tr>
<tr>
<td>209</td>
<td>W 2:30-3:20p</td>
<td>HLS 464 lab &amp; HLS 401</td>
</tr>
<tr>
<td>210</td>
<td>W 3:35-4:25p</td>
<td>HLS 464 lab &amp; HLS 401</td>
</tr>
</tbody>
</table>

**Discussions are held in HLS 464 lab the first two weeks**