Short Course in MATLAB

CS 1132 – Fall 2021

Dr. Curran D. Muhlberger

TA: Aravind Suresh Babu

https://www.cs.cornell.edu/courses/cs1132/
Who is this course for?

- *Quickly* get up-to-speed with the Matlab *programming environment*
- How to make Matlab do things it doesn't currently do on its own

Example: visually check your math
Who is Dr. Muhlberger?

And why should scientists learn to code?
Today’s agenda

- Course goals
- Logistics
- Matlab desktop
- Script vs. function
Course goal

Learn how to program in Matlab!

Upon learning this material, you will be able to:
- Translate a problem's solution into an algorithm
- Implement algorithms in Matlab syntax
- Visualize data and simulations
Tools of the trade

Topics:
- Matlab basics (environment, built-in functions)
- Arrays (vector, matrix)
- Vectorized computation
- Control flow (if-else, loops)
- User-defined functions
- Strings and cell arrays
- Graphics
- Basic I/O (including file I/O)
Programming Fundamentals …

… that you should practice

- Top-down design
- Modular program development
  - Reduce redundancy
- Useful documentation
- Thorough testing

Learning is something you do; we can only facilitate
How will you learn?

- Read textbook, complete activities
- Attend lectures (MF, 7 weeks), take notes, participate
- Attend discussion section (W, Upson 225)
- Complete exercises
- Complete assignments
- Ask questions in office and consulting hours
  - Times/locations on website: https://www.cs.cornell.edu/courses/cs1132/
How to assess your learning

- Completed exercises (5%)
- Completed assignments (10%, 20%, 25-35%)
- Completed textbook activities (0-10%)
- In-class exam (30%)

Assessment → more learning via feedback loop
Must-know facts about CS 1132

- Assignments
  - If first submission is not perfect, one re-submission is allowed without penalty. One additional allowed re-submission incurs a 10% deduction.
  - Late (re)submission allowed up to 24 hrs for a 10% penalty
  - Penalties accumulate from (re)submission to resubmission

- Test
  - You may take the test second time if you wish—a different version will be given
  - Final score is the most recent score for the test

- Course ends after 7 weeks: October 19th
- 2 credits, S/U
- Requires mastery of material
  - “B level” (Course score of >85) is required to pass the course
Related course: CS 1112

- 4 credits, full semester, “Intro to Computing Using MATLAB”
- Covers more content than CS 1132; focusing on fundamental programming concepts
- More beginner friendly than this short 7-week course because it starts more slowly

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

- Electronic submission does not alter the University standards on academic integrity
- Your individual work is required
  - Do not copy code from any source (friend, published work, Internet, …)
  - You can discuss general strategy with others, but do not share any code, whether written, electronic, or verbally
  - We use MOSS or similar software to check your submitted programs
Matlab Demo

- Command window, memory (Workspace) window
  - Variables are not declared: create a variable simply by assigning a value to a variable name. A number has the type `double` by default

- Built-in functions
  - `rand()`, `floor()`, `ceil()`

- Current folder box

- Example script `diffArea`

- Turn a script into a function
Input & simple output

- `variable = input('prompt')`

  ```
  r = input('Enter radius: ')
  ```

- `disp('message to print')`
- `disp(variableNameOrExpression)`

  ```
  disp('Hello there!')
  s = rand*pi; disp(s)
  ```
Exercise

Modify this script to calculate the increase in surface area [mi$^2$] given an increase in the radius [in] of the sphere.

Reminder: 1 mi = 5280 ft, 1 ft = 12 in

```matlab
radius = input('Enter radius [mi]: ');
area = 4*pi*r^2;
disp('Surface area [mi^2]: ')
disp(area)
```
Tips for writing a program

- Check that you know what is given (or is input, or is assumed)
- Be *goal-oriented*: start by writing the last statement(s) for the program output
  - What is the program supposed to produce? *You know this from the problem statement*
  - Allows you to work backwards from the results
- Name as a variable what you don’t know
  - Helps you break down the steps
  - Allows you to temporarily skip over any part that you don’t know yet how to do