Welcome to CS 6452!
Prototyping Interactive Systems
Agenda

- Introductions
- Motivation for the class
- Structure of the class
- Class content
- Getting started with Python
Now it is your turn

- Name
- Degree program
- Your experience programming
- Other relevant experience related to computing
- Why are you interested in this class?
- What one thing did you do this summer that you never did before?
Big Decision for today

- Do you NEED to be in this class?
- Do you WANT to be in this class?
- Ask questions in class TODAY to help you figure this out.
Class Website

http://www.kedwards.com/cs6452
T-square site
Preliminaries

- Laptops
  - Bring to each class
  - Will use in lab exercises, etc.
  - Please leave closed otherwise

- You are responsible for checking on t-square and course website regularly for assignments and class preparation
What does Prototyping Interactive Systems mean, anyway?

- Rapid creation of interactive systems through programming

- Exposure to a variety of modern programming activities

- Some theory, mostly pragmatic, with some exposure to developing skills to describe and defend design choices
Original motivation for 6452

- Computing requirement for HCC Ph.D. programming
  - Basic level of competency in building programs yourself
  - Confidence in assessing and learning new programming paradigms, tools and languages
  - Learn about software architectural design considerations (e.g., how solutions are put together and assessing options)
  - Should help you satisfy the computation portfolio requirement
    - Technical reading/writing/doing/talking
Class is now popular beyond HCC

- Mostly masters students

- Exposure to programming if you haven’t had any

- Lots of variability in experience/skill with programming
  - That’s OK, but think about whether you will be “bored” by this class
  - There is a more advanced UI programming course offered
1. Programming in Python
   Coding, data structures, tools, etc.
   Data access, manipulation, and visualization
   Manipulations on the web

2. Object-oriented programming in Java/Processing
   Coding, tools, etc.
   Event-driven GUI programming

3. Physical prototyping and hardware
   3D printing and laser cutting
   Arduino

… Maybe a few extras along the way
## Grading Criteria

<table>
<thead>
<tr>
<th>Category</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>In-class participation</td>
<td>20%</td>
</tr>
<tr>
<td>Readings</td>
<td>10%</td>
</tr>
<tr>
<td>Programming assignments</td>
<td>70%</td>
</tr>
</tbody>
</table>
Style of class

- Some/most of classes will have a studio model to them
  - Review of a reading or a quick idea
  - In-class, hands-on activity (individual or small group)
  - Reflection
Quick poll. Who has experience with …

- Python
- Processing
- JavaScript
- Java
- Swing or JavaFX
- iOS app development (Swift)
- 3D printing, laser cutting
- Arduino
Python

- Our first programming language to examine
- Interpreted, interactive
- Includes object-oriented capabilities, but not what many Python programs use (not what it’s known for)
- Loads of libraries
- Useful for web tasks
Python

- “In-between” versions 2 and 3
  - Some subtle but important changes to the language
Resources

- Borrow or buy a book (any good one will do)
- Free online text

Informal HW

Getting ready for next time

- Install Python on your laptop
  - Mac folks – Should already have version 2.7
  - Windows folks – Get version 2.7 from web