Processing
Software Sketchbook Demonstration

Contact Information:
Woody March-Steinman
wmarchsteinman@math.arizona.edu

August 10, 2021

1 Quick Rundown

Processing is an IDE for quick graphical programming. You can use Python and R. It's fun to use and great for teaching, presentations, and visualization prototypes.

2 Introduction

Processing (https://processing.org) is a graphical library/IDE that allows for rapid prototyping of visualizations. It's been around in various iterations since 2001; below are a few resources from the extensive trove of material that's since been developed. Environments are available for a variety of languages (including Python and R), with install-able libraries ranging from OpenCV to algorithmic music.

While Processing has proven to be excellent for prototyping classroom demonstrations, it also has strong potential to be of use for presentations and quick, *interactive* visualization prototyping. The learning curve is reasonably shallow, as well, with plenty of room to develop creative solutions to problems.

As a note, there is a JavaScript library called p5.js that is a direct port of Processing. If you prefer this language (in addition to maintaining an easily accessible online directory for projects), check out https://openprocessing.org and https://p5js.org.

A couple caveats:

- The Processing IDE is implemented using Jython, which effectively means that common Python libraries like numpy and scipy do not work (yet).
- There is a p5 port for Python that works natively with Cython – it looks to have very similar functionality, but I have not yet been able to test it due to issues with OpenGL – the resource repository will be updated with examples when this happens.

3 Getting Started – Processing IDE

- Download Processing from https://processing.org
- In the corner, there is a drop down menu that says "Java". Click on this, go to "Add Mode," and select the languages of your choice.
- Create a new sketch and save it. You have the option to make tabs associated with the project; these work well for separate classes.
- Declare globals.
- Make a function called setup() to initialize globals and create a canvas using the size(dimX, dimY) function.
- Make a function called draw(). This is where most animation action happens (this is looped elsewhere to act as a frame of animation; looping behavior can be halted with the noLoop() command).
- Make other functions and classes as necessary. See reference below for built-in functionality.
- Optionally, implement interactivity with keyPressed()/mouseClicked() or similar functions.

4 Templates/Sample Code

All templates/samples covered today are available in a git repository or as links below:

GitHub (Python/Java)

- https://github.com/wmarchsteinman/SoftwareWorkshop2021

OpenProcessing Visualizations (JavaScript)

- Motion Grid
- Epidemic Sandbox

5 More Robust Resources

- Official Processing Documentation: `https://py.processing.org/reference/`
- p5 for Python installation instructions/documentation: `https://p5.readthedocs.io/en/latest/`
- Dan Shiffman YouTube Playlists – these range from science and machine learning to art: Coding Train