

Syllabus: Programing for Interaction

Barney Haynes
bhaynes@cca.edu

New Course Site

<http://interaction09.pbworks.com/>

- 1) This syllabus will be a living document and will change as the class evolves. Please check it regularly.
- 2) The best way to learn programming is experimenting with and writing code. Your assignments will be to post weekly sketches that reflects, leverages, recombines, and utilizes the coding elements presented in class each week. We will go over how to export applets and post them to a website the first day. One important thing to keep in mind. Every programmer, even the best ones "borrows" snippets of code. However if you fob off a sketch that is borderline plagiarism your grade will suffer accordingly.
- 3) This syllabus is organized into three sections; Learning Resources, a Synopsis, and an Extended Version with a list of each function and concept that will be covered.



Learning Resources:

In the CCA Library:

**Processing: A Programming Handbook
for Visual Designers and Artists**
Casey Reas and Ben Fry

Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction
Daniel Shiffman.

It's available for free online through Safari Books:
<http://my.safaribooksonline.com/9780080920061>
(if you sign up for a free trial)

Web Sites:

Processing's Getting started guide:
<http://processing.org/learning/gettingstarted/>

Processing's Tutorials:
<http://processing.org/learning/>

Processing Basics
<http://processing.org/learning/basics/>

Processing Reference Page:
<http://processing.org/reference/>

If you have your own machine and plan to use it in class download and install Processing if you haven't done so.
<http://processing.org/download/>

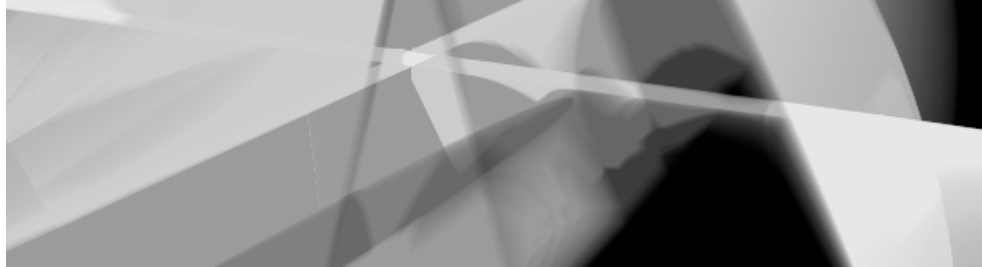
A comparison to other programming languages:

<http://processing.org/reference/compare/index.html>

Where to see Processing examples:

The Exhibition page on the Processing site:
<http://processing.org/exhibition/>

OpenProcessing
<http://www.openprocessing.org/>



Synopsis:

October 26

The Processing Environment (PDE)

Syntax

Processing's Functions

Coordinates and Primitives

Cartesian grid.

Primitive Shapes:

Order of Operation:

Color functions:

Drawing functions:

Looping:

Order of precedence:

Suggested Reading:

Processing's Getting started guide:

<http://processing.org/learning/gettingstarted/>

November 2

Primitive data types
Conversion functions
Variables
Math one

Mouse functions

System Variables (width,height), frameRate()

Chance and Emergence 1

Applying Variables to enable motion.

Operation shortcuts

Decisions/Conditionals

Applying Conditionals to Motion Control

Constraining and Mapping Numbers map() and constrain()
for loops()

Exporting to an applet.

November 9

Iteration

Formating code blocks

Programing functions

Recursion

November 16

Image:

Image Processing
Transformation
Angles:
Advanced Color

November 23

Text:

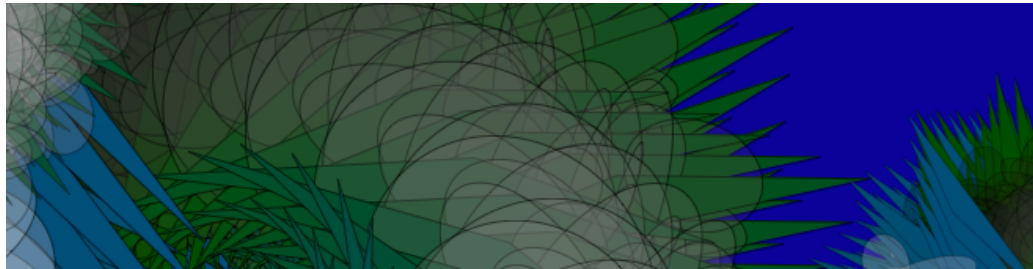
Strings
PFont
Text motion and response

November 30

Math 2
Curves and Trig
Arrays
Classes

December 7

Output Images High Resolution PDFs
External Libraries
Where to install them (Warning this has changed since version 1)



Extended Version: (This will change in case we go on a pedagogically rewarding tangent.)

October 26

The Processing Environment (PDE)

File Menu
PDE GUI
Sketchbook location

Syntax

```
// (comment)
/* */ (multiline comment)
; (statement terminator),
" " (comma)
print()
println()
```

Processing's Functions

Functions appear as colored text which means they are keywords or reserved
Some need parameters or arguments, some don't.

Case Sensitive

When the name of a function is a conjunction then second word is capitalized

White Space

Console println() and print()
Errors

Coordinates and Primitives

Cartesian grid.
Processing's zero points are at the upper left hand corner.

Primitive Shapes:

Points:
points(x,y)

Line:
line(x1, y1, x2, y2)

Ellipse:
ellipse(x, y, width, height)
ellipseMode(CORNER,CENTER, RADIUS)
Default mode is CENTER

Rectangles:
rectMode(CORNER,CENTER, RADIUS)
Default CORNER

Triangles
triangle(x1, y1, x2, y2, x3, y3)
works clockwise

Order of Operation:

Basic Color
color parameters range from 0 -255
RGB
Hex
HSB

The color selector in the File Menu

Color functions:

fill()
noFill()
stroke();
noStroke()

Drawing functions:

smooth()
noSmooth()
strokeWeight()
strokeCAP()

Primitive data types

int
float
boolean
true
false
char

Conversion functions

int()
float()

Variables

=
declaring int x; or declaring and initializing int x = 100;

global or local

Looping:

```
void setup(){  
  code here once  
}
```

```
void draw(){  
  code here repeats till the sketch closes  
}
```

Math one

+ (add), - (subtract), * (multiply), / (divide), % (modulus)

More mouse functions:

mouseX
mouseY
mouseButton
mousePressed
mouseReleased()
mouseMoved()
mouseDragged()

Exporting to an applet.

Suggested Reading:

Processing's Getting started guide:

<http://processing.org/learning/gettingstarted/>

November 2

Variables and Math amplified.

Order of precedence

Modulus %

System Variables (width,height), frameRate()

Constraining and Mapping Numbers map() and constrain()

Chance and Emergence

random()

noise()

noiseDetail()

Probability

Applying Variables to enable motion.

Operation shortcuts ++ (increment), -- (decrement), += (add assign), -= (subtract assign)

*= (multiply assign), /= (divide assign)

Decisions

> (greater than), < (less than)

>= (greater than or equal to), <= (less than or equal to)

== (equality), != (inequality)

Conditionals

if, else, {} (braces)

|| (logical OR), && (logical AND), ! (logical NOT)

Applying Conditionals to Motion Control

Bouncing shapes within constraints

November 9

Iteration

for() loops

Nested for() structures

Nesting conditionals into for loops

Formating code blocks

Programing functions
Recursion

November 16

Image:
PImage
loadImage()
image()

Image Processing
tint()

Transformation

Angles:
radians()
degrees()

translate()
rotate()
scale()
pushMatrix()
popMatrix()

Advanced Color
colorMode()
color()
hex()
red()
green()
blue()

November 23

Text:
Strings

PFont
loadFont()
textFont()
text()
createFont()

Text motion and response

November 30

Math 2
Curves and Trig
Arrays
Classes

December 7

Output Images High Resolution PDFs
External Libraries

Where to install them (Warning this has changed since version 1)

