Syllabus

Fall 2015
DESGN-629-01 Processing: Interaction Design

Instructor

Wendy Ju, wju@cca.edu

Synopsis

This course will look at how interaction designers use computation, to create rich and delightful experiences. We will use the Processing programming language as an introduction to programmatic thinking for design. The class will offer students a range of projects and explorations towards a secure understanding of the how to use this tool.

.

Organization

This course will center around three units.

The first is aimed at providing everyone with a working background for getting up and running with basic programming.

The second unit focuses on programming for interaction, and the use of libraries and external tools such as OpenCV to extend the capabilities of your programs.

The final unit incorporates hardware, electronics, and advanced networking capability so that your programs can be connected to the cloud and situated in the world.

This class will be taught studio-style. There will be a few limited lectures that demonstrate about critical components of programming and interaction design. Largely class time will be devoted to sharing work, with opportunities to program and debug in-class. Within the scope of the larger class projects, students have a lot of room to explore topics and features of the Processing programming environment that they are particularly interested in. Topics are subject to change based on student need and demand.

Readings

Readings will be from Casey Reas and Ben Fry's book *Processing: A Programming Handbook for Visual Designers and Artists*, c. 2007 published by MIT Press. This book is available in hardcover and electronic form via Amazon.com. You will not be tested on the reading; however, most of the basic knowledge you need is in the book, which you should read and program along with. We will answer questions you have and address issues you run into in class, and otherwise focus on approach and technique.

Outcomes

Learning outcomes of this class are:

- understanding of programming fundamentals (variables, functions, data types, control operators, mathematical operators, input and output, libraries and modules.),
- ability to iteratively develop interactive programs through processes of ideation, sketching, critique, programming, debugging, and critique,
- comprehension of technical underpinnings of a wide variety of interactive products and programs,
- familiarization and engagement of larger open-source programming community as a resource, and
- ability to express creative ideas through the art of programming.

Evaluation

Feedback on work will be primarily given as in-class critique, although a numeric grade will be given to each project. Grades will be based on your project work. Please feel free to contact the instructor at any time for a more detailed qualitative evaluation of your performance and progress.

Project 1	20%
Project 2	20%
Project 3	20%
Final Presentation	20%
Participation	20%

Schedule

Week		
1	September 1 *(Wendy remote)* Introductions Course overview Hello World	PROJECT 1: Like Clockwork READING: R&F, pages 1-172, 245-250.
2	September 8 *(Wendy remote)* TOPIC: Programming fundamentals What Lies Beneath The Art of Programming	.
3	September 15 TOPIC: Inputs, Outputs and Interaction	

4	September 22 ::Project 1 Presentations::	PROJECT 2: Music Boxes READING: R&F, pages 173-394.
5	September 29 TOPIC: Sound, Video and other libraries	
6	October 6 TOPIC: OpenCV, TUIO	
7	October 13 TOPIC: Networked	
8	October 20 ::Project 2 Presentations::	PROJECT 3: Eye Spy READING: R&F, pages 395-518.
9	October 27 -> moved to October 23rd *(Guest lecture)* TOPIC: Arduino, Electronics	
10	November 3 TOPIC: Text parsing, text-to-speech and other fun with data	
11	November 10 ::Project 3 Presentations::	
12	November 17 -> moved to November 14th *(Guest lecture)* TOPIC: Single-board computers	
13	November 24 Studio Time	
14	December 1 :: Penultimate Presentations ::	
15	December 8 :: Final Presentations ::	