

Fundamentals of Procedural Media

USC School Cinematic Arts, CTIN 499, 2 units

Instructor:

Peter Brinson

peterbrinson AT gmail.com

10-12:50PM Th - RCZ 119 (the XML)

Course Description:

This course is an introduction to the procedural nature of interactive media. What distinguishes interactive media from other expressive forms is based on the computer's unique ability to execute rules. This is the practice of the computer programmer - writing instructions to make complicated systems out of simple processes. The interactive artist writes code that creates meaning and representation, as opposed to the film director, the playwright, and the novelist, who author the representation itself. This course is not focused on learning a specific language, but to help the future game designer, 3D modeler, interactive writer, and, indeed, software engineer to become procedurally literate.

Therefore, this class involves coupling a structured orientation of the programming language, Processing, with a weekly examination and discussion of relevant computational media and texts. In particular we will investigate what aesthetics and representations lend themselves to, and conversely, influence both the inherent structures of programming and the practices of the programmer. Learning to program has clear value in and of itself, but procedural literacy imbues the student with the ability to more thoroughly read computational media. For example, one will recognize the patterns and behaviors of a given game as manifestations of general algorithms found in all sorts of game genres as well as other interactive forms.

The open source programming language Processing was developed to address this need. It is powerful and simple. It provides the advantages of both a scripting and object oriented language, so that the Processing programmer can focus on fundamentals while learning to create flexible building blocks. By nature of being open source, it is free, works on multiple operating systems, and is well documented.

Student evaluation will be based on two tests and two projects. These open book tests include a number of programming assignments in which students have two hours to build their own solution in Processing. Any materials can be used, but there will be no collaboration or communication. Later in the semester, students will build creative projects in and outside of class.

Required Readings:

Expressive Processing - Noah Wardrip-Fruin

Learning Processing - Daniel Shiffman

Educating the Fighter: Buttonmashing, Seeing, Being - Kurt Squire

Evaluation of student performance:

Participation and 3 blog posts	10
Test 1	20
Test 2	20
Project 1	25
Project 2 (Final Project)	25
Total:	100

Course Content

Week 1: Introduction/Overview

- definitions of procedurality
- pixels, color, shapes
- functions
- Read for next week: Shiffman, Ch 1, 2, 3
- post a pattern

Week 2

- logical flow
- conditionals
- Read for next week: Shiffman, Ch 4, 5
- Read for next week: Wardrip-Fruin (pp 1-21)

Week 3

- loops
- discussion: Expressive Media
- Read for next week: Shiffman, Ch 6

Week 4

- * Test
- Play Eliza: <http://users.design.ucla.edu/~acolubri/processing/eliza/applet/index.html>
- Read for next week: Wardrip-Fruin (pp 23-40) and post (images of) examples of an Eliza-like breakdown
- Read for next week: Process Intensity: <http://www.manifestogames.com/node/2348>

Week 5

- review test
- review loops
- discussion: The Eliza Effect

Week 6

- Meet on Wednesday at 6pm in ZML. Casey Reas talk.
- Optional - study hall on Thursday 10-1pm

Week 7

- functions
- objects
- Read for next week: Shiffman, Ch 7, 8, 10 (end on page 177)
- not graded homework given ("CrazyShips")

Week 8:

- objects and algorithms (first half of chapter 10)
- discuss homework

Week 9:

- * Test
- Read for next week Shiffman, Ch 9

SPRING BREAK

Week 10

- Review Test 2
- Arrays
- Read for next week: Wardrip-Fruin (pp 299-352)

- Read for next week: "Educating the Fighter: Buttonmashing, Seeing, Being" and post (images of) examples

Week 11

discuss project 1
discussion: Repetition
discussion: The Sim City Effect

Week 12

* Due - project 1 - Represent Time In an Unusual Way
critique
discuss pattern posts from week 1

Week 13

discuss project 2

Week 14

work on project in class

Week 15

* Present project 2 - A Digital Toy

Week 16

Final critique of project 2 – A Digital Toy

Explanation of Tests:

The two tests will follow the same format. Students will be given two hours to complete a number of programming tasks that they will implement in the Processing environment. An example task could be, "Write a program that lays 10 spheres on the screen at random locations, except the spheres cannot be touching each other. Using a for loop is recommended." A second example could be, "Add to the project on page 67 of our book so that the lines gradually fade away rather than disappear immediately."

The concepts covered in each test will be based on class lectures from all previous weeks. Students may use any notes, books, or online documentation, but must work alone. At the end of the test they will post their test to blackboard.

Explanation of Project 1:

"Represent Time in an Unusual Way" asks the student to develop a non-interactive project that presents the current time in an expressive way. Examples might include a clock that represents the time by means other than hands or numbers, or a screen saver that changes based on the time of day.

Explanation of Project 2:

"A Digital Toy" is the most open-ended project assignment. This interactive project is like a digital game without a clear outcome. In other words, a focus is placed on giving the user one or more digital objects to manipulate but there is no winning or losing involved. Students are encouraged to use many of the programming concepts taught during the semester. The projects should be interactive, emergent, and playful.

Missing an Assignment Deadline, Incompletes:

The only acceptable excuses for missing an assignment deadline or taking an incomplete in the course are personal illness or a family emergency. Students must inform the professor before the assignment due date and present verifiable evidence in order for a make-up to be scheduled. Students who wish to take incompletes must also present documentation of the problem to the instructor or teaching assistant before final grades are due. Incompletes are only available after the 12th week withdrawal deadline.

Statement for Students with Disabilities

Any student requesting academic accommodations based on a disability is required to register with Disability Services and Programs (DSP) each semester. A letter of verification for approved accommodations can be obtained from DSP. Please be sure the letter is delivered to me (or to TA) as early in the semester as possible. DSP is located in STU 301 and is open 8:30 a.m.–5:00 p.m., Monday through Friday. The phone number for DSP is (213) 740-0776.

Statement on Academic Integrity

USC seeks to maintain an optimal learning environment. General principles of academic honesty include the concept of respect for the intellectual property of others, the expectation that individual work will be submitted unless otherwise allowed by an instructor, and the obligations both to protect one's own academic work from misuse by others as well as to avoid using another's work as one's own. All students are expected to understand and abide by these principles. *Scampus*, the Student Guidebook, contains the Student Conduct Code in Section 11.00, while the recommended sanctions are located in Appendix A:

<http://www.usc.edu/dept/publications/SCAMPUS/gov/>. Students will be referred to the Office of Student Judicial Affairs and Community Standards for further review, should there be any suspicion of academic dishonesty. The Review process can be found at: <http://www.usc.edu/student-affairs/SJACS/>.