Procedural Expression

USC School of Cinematic Arts, CTIN 506, 2 units

Thursdays 7:00 to 8:50pm

Room: SCB 104

Professor	Student Assistant
Mitch Mastroni (he/him)	Albert Liang (he/him)
mastroni@usc.edu	youliang@usc.edu
Office Hours: Mondays 10am-12pm, Fridays 10am- 2pm, by appointment [here](#)	Office Hours: Tuesdays 9:30am-11:30am by appointment [here](#)

Course Description

This course is an introduction to the practice of the creative coder. Today, computation - by way of software - touches nearly every human endeavor, and, in particular, has profoundly altered creative production and dissemination. But what distinguishes interactive media from other expressive forms is that it is not just made with software, it is software. The creative coder explores the computer's unique expressive potential by harnessing its unprecedented ability to execute rules. This will be our practice - writing instructions to make complicated systems out of simple processes. Further, the creative coder 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 for the game designer, 3D modeler, interactive writer, digital musician, and installation artist 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 carefully read computational media.   For example, one will recognize the patterns and behaviors of a given videogame as manifestations of general algorithms found in all sorts of software.

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.

During the latter weeks of the semester we will begin again with a new language. With the fundamentals of coding fresh in hand, we will learn the language, C# (within the Unity environment) starting with its core principles.

Finally, through readings, screenings, and discussions we will examine the portrayal of computer programmers as protagonists and as villains. Starting with the earliest "hackers", we will consider the ethos and attitude of the programmer in fiction and culture.

Student evaluation will be based on assignments, projects, and readings.

Learning Objectives

- Learn and practice the fundamentals of computer programming
- Map out an individualized plan for subsequent semesters - what classes at USC that will further your computer programming interests
- Observe a shared interactive logic and aesthetics among software (especially media creation software) with an appreciation for code as language
- Articulate computational media theory through art and media examples
- Hone a critical eye and a literacy for digital media
- Study the cultural influence of the computer programmer in creative cultures and technology industries as well as fictional portrayals in media and literature
- Play and deconstruct video games with an interest in practicing above concepts

Evaluation of Student Performance

Participation and Reading	20
Weekly Assignments	60
Final Project	20
Total:	100

Readings

Learning Processing by Daniel Shiffman
Life In Code by Ellen Ullman
Expressive Processing by Noah Wardrip-Fruin

additional readings to come

Course Content

Week 1, Aug 25

    Due for Class
- [Download](#) and [install](#) Processing
- [Complete the welcome survey](#)

    Lecture
- definitions of procedurality
- introduction to Processing
- colors in computers
- data types
- functions

Week 2, Sept 1

    Reading
- [A Brief Rant on the Future of Interaction Design](#) by Bret Victor
- Hello Processing videos
  - [Shapes](#)
  - [Interact](#)
  - [Questions](#)

    Due for Class: Avatar
- Make a social media icon in Processing (512x512 pixels)

    Lecture
- discuss Bret Victor essay
- download and install [GitHub Desktop](#)
- variables
- logical flow
- conditionals
- troubleshooting strategies

Week 3, Sept 8

Reading
- *Life In Code* by Ellen Ulman, Ch 1

Due for Grade: Circle Step Right
- Add to this template to make this.
- The line of circles has a 1 of 4 chance to 'step-down' at any given moment
- Submit by Wednesday at 9pm

Lecture
- discuss Ellen Ulman reading
- loops
- counters
- debugging

Week 4, Sept 15

Reading
- Expressive Processing by Noah Wardrip-Fruin, Ch 1

Due for Class
- Install Unity Hub and Unity 2021.3.x
- Create one example of a Wardrip-Fruin effect (based on above reading)

Due for Grade: Circle Grow in Steps
- Using this template, make this
- Submit by Wednesday at 9pm

Lecture
- discuss Noah Wardrip-Fruin reading and examples

Week 5, Sept 22

Due for Class: Explore the Unity Editor Layout

Due for Grade: Loops, Part I
- Study [this template](), in order to make [this]() and [this]()
- Submit by Wednesday at 9pm

Week 6, Sept 29

Due for Class: [See How to Create a Unity Project]()

Due for Grade: Loops, Part II
- Notice how loops augment [this](), into [this]().
- Make [this]() (details on page)
- Submit by Wednesday at 9pm

Week 7, Oct 6

Due for Today: [Watch How Scripts are Used as Behavior Components]()

Due for Grade: Meme Maker
- *Details to come*

Week 8, Oct 13

HOLIDAY, NO CLASS

Week 9, Oct 20

Due for Grade: Bouncing Ball
- *Details to come*

Week 10, Oct 27

Due for Grade: Solar System Musical Magic
- *Details to come*

Week 11, Nov 3

Due for Grade: Absurd Popcorn Arena
- *Details to come*

Week 12, Nov 10

Due for Grade: Mad Libs Therapy
- *Details to come*

Week 13, Nov 17

Due for Grade: Final Project, Alice in Wonderland Moment.
- *Details to come*

Lecture
- Process the Semester, Part 1
- Play: Baba Is You, Teardown, Desert Golfing, Noita

Week 14, Nov 24

HOLIDAY, NO CLASS

Week 15, Dec 1

Reading
- [Game Over](#) by 99 Percent Invisible

Lecture
- Process the Semester, Part 2
- Play: Plug & Play, Thomas Was Alone

Weekly Assignments:

Students will typically have 1 or 2 assignments due every week, to be turned in through GitHub the night before class.

Missing an Assignment Deadline, Incompletes:

The only acceptable excuses for missing an assignment deadline or taking an incomplete 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.

Additional Information

Academic Conduct

Plagiarism – presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in *SCampus* in Section 11, *Behavior Violating University Standards* https://scampus.usc.edu/1100-behavior-violating-university-standards-and-appropriate-sanctions/. Other forms of academic dishonesty are equally unacceptable. See additional information in *SCampus* and university policies on scientific misconduct, http://policy.usc.edu/scientific-misconduct/.

Discrimination, sexual assault, and harassment are not tolerated by the university. You are encouraged to report any incidents to the *Office of Equity and Diversity* http://equity.usc.edu/ or to the *Department of Public Safety* http://capsnet.usc.edu/department/department-public-safety/online-forms/contact-us. This is important for the safety whole USC community. Another member of the university community – such as a friend, classmate, advisor, or faculty member – can help initiate the report, or can initiate the report on behalf of another person. *The Center for Women and Men* http://www.usc.edu/student-affairs/cwm/ provides 24/7 confidential support, and the sexual assault resource center webpage sarc@usc.edu describes reporting options and other resources.

Support Systems

A number of USC's schools provide support for students who need help with scholarly writing. Check with your advisor or program staff to find out more. Students whose primary language is not English should check with the American Language Institute http://dornsife.usc.edu/ali, which sponsors courses and workshops specifically for international graduate students. The Office of Disability Services and Programs http://sait.usc.edu/academicsupport/centerprograms/dsp/home_index.html provides certification for students with disabilities and helps arrange the relevant accommodations. If an officially declared emergency

makes travel to campus infeasible, USC Emergency Information http://emergency.usc.edu/will provide safety and other updates, including ways in which instruction will be continued by means of blackboard, teleconferencing, and other technology.

Disruptive Student Behavior:

Behavior that persistently or grossly interferes with classroom activities is considered disruptive behavior and may be subject to disciplinary action. Such behavior inhibits other students' ability to learn and an instructor's ability to teach. A student responsible for disruptive behavior may be required to leave class pending discussion and resolution of the problem and may be reported to the Office of Student Judicial Affairs for disciplinary action.

Syllabus Updates:

This syllabus is liable to change up to the beginning of class and possibly over the semester. Please check the posted syllabus regularly, and note all changes that are shared by the instructor in class.