ACCAD 5102: Programming Concepts for Artists and Designers

Tue,Thur 1.50-4.30 (3 credits) - 1224 Kinnear, Room 205

Instructor

J Eisenmann :: jeisenma@accad.osu.edu (Office hours can be set up via e-mail.)

Website

http://accad.osu.edu/~jeisenma/teaching/pcad.py

Description

This class will provide students with the opportunity to explore programming principles through the creation of interactive works. Students will work with programming tools including Scratch, Python and Processing for creating interaction while building an understanding of programming as an artistic tool. Students will have the opportunity to create applied projects in their personal area of interest. (e.g. data visualization, interactive music, games, generative art, etc.).

Prerequisites

No prior programming experience is required.

Learning Goals

By the end of this course you should be able to:

- Understand basic programming concepts such as:
  - variables and data types
  - conditional statements and loops
  - functions and objects
  - events
  - arrays and data streams
  - libraries and APIs
- Create interactive media (i.e. games, visualizations, interactive art, nonlinear narratives, etc)
- Understand and modify other people's code to make it work for you
- Collaborate and share ideas more easily with programmers in your field

You should leave the course with a sketchbook of small interactive projects (called “sketches”) and with one final project that has evolved into a polished interactive piece.

Course Materials (all free!)

Texts: Learning Python the Hard Way, Processing's Learning Page

Software: Scratch, Processing

Other links posted on the course website

Methodology

Processing will serve as our primary sandbox for experimentation, and python will be the primary programming language that we use, but the goal is to get you to think abstractly about ways to solve problems by programming in whatever context you find yourself. Lectures may include comparisons to or idiosyncrasies of other languages and environments to which you may eventually be exposed near the end of the course. By the end of the course, you should be able to work in almost any language by transferring the universal principles that you have learned in python.

Class time will consist of lectures, examples/demos, workshop time for you to get input from both me and your peers, and presentations/critique of your work. Assignments will be short and to the point, while the projects will provide opportunities for you to apply what you have learned to your own research problems. The option will be given to pursue projects in other languages or environments (i.e Processing (Java/JavaScript), JavaScript+Canvas, Unity, PyGame, Maya, etc.)

Attendance Policy

All students are required to be on time and in attendance for each and every class. Students arriving to class more than 15 minutes late will be counted as absent. Two absences will lower a final grade by 1/2 a letter. Three absences will lower a final grade by one letter. Four absences will result in failure of the course.

Grading Policy

OSU Graduate School policy does not allow Graduate Associates to evaluate and assign grades to students enrolled for graduate credit. Grade evaluations for this course will be given by approved ACCAD staff in consultation with the instructor. Questions about evaluations or other course matters should not be directed to staff or to the director of ACCAD unless the instructor has first been consulted and the matter was not resolved.

Adherence to deadlines is expected. It is the individual student's responsibility to keep track of deadlines and to present the work to the class and instructor on the specified dates. Revision is part of the process in preparing work for the final due date, not after. Make use of production time in class to receive feedback on work in progress from the instructor and classmates. Work presented

late will be marked down one grade letter for each class meeting missed.

Students choosing to use "at home" hardware and software must have their current working files on the system and available for review at the beginning of every class. Problems with home systems and/or incompatibility will not be an acceptable excuse for missed goals. Technical problems may happen frequently and students may have trouble accessing the computer lab during "prime time" hours. Students must make their own arrangements for overcoming these difficulties to submit their work on time. Unless there is a complete system failure at ACCAD, technical difficulties are never an acceptable excuse for not meeting a deadline. Students should plan their time so as to anticipate the technical hurdles that are a part of this profession.

Grade Breakdown

Participation	10% (taking part in discussions/critique)
Assignments	40%
Data Viz Project	20%
Final Project	30%

Academic Dishonesty

Any and all suspected cases of academic dishonesty will be dealt with according to university procedures. Students are referred to the student handbook for further information on academic dishonesty and the accompanying procedures and penalties. Students can read the code of student conduct at: http://studentaffairs.osu.edu/csc/

Accommodations for Students with Special Rights

Any student who feels s/he may need an accommodation based on the impact of a disability should contact the instructor privately to discuss their specific needs. Please also contact the Office for Disability Services at 614-292-3307 in room 150 Pomerene Hall to coordinate reasonable accommodations for any documented disabilities.

Schedule

Subject to change based on class progress. See the website for an up-to-date schedule.

Week	**Topics** *(cumulative!)*	**Applications**	**Technologies**
1	intro, interaction, sequencing, variables, conditions, loops, events	simple games	Scratch, Python
2	mod-ing, data types, data streams	gameMod, procedural drawing	Python, Processing
3	sequences, loops, randomness	animation	Python, Processing
4	objects	generating variants (i.e. “copy stamping”)	Python, Processing
5	Guest lecture/workshop		
6	file I/O, libraries	GUI, data viz	Python, Processing
7	APIs, publishing, translation	mobile and gestural interaction, generating audio	Java, JS, Python Processing, Houdini, Maya, Unity, etc...