

Syllabus:

Course Title, Prerequisites, and Description:

Procedural Animation, 5 credits

Prerequisites: ARTS COL 750 or 751 or 752, or permission of instructor

Students create computer animation using procedural methods for modeling, rendering and animating 3D computer generated objects, environments, or characters. The use of techniques such as parametric modeling, particle systems, data-driven design, dynamic simulation, and behavioral animation will be studied.

Course Objectives and/or Student Learning Outcomes:

This course will broadly investigate ways to analyze, design, and implement procedural approaches for modeling and animating 3D objects and environments, as well as their behaviors. Instead of sculpting, key-framing, and painting computer graphics assets, procedural techniques generally involve the construction of processes for generating design solutions. Students will gain experience identifying and understanding opportunities for creating efficient parametric representations of designs so that changing a handful of numeric values modifies their model or motion's creation process. Students will be exposed to current technologies that enable the creation of complex procedural relationships and generative systems without requiring extensive programming experience. They will gain an awareness of the relative computational expenses involved in managing complexity.

Course Methodology:

Students will be introduced to an evolving set of procedural techniques and concepts through design exercises and labs. The course will survey recent advances in generative interfaces and representations. Students will be shown existing procedural work primarily from film effects, design, art, and architecture. Examples of varying levels of complexity will be presented to demonstrate different techniques. Students will be required to use the techniques and examples presented to design their own procedural systems. Previous experience with creating and presenting digital images, video, 3D animation, and web pages will be assumed. The assignments will be flexible enough to allow students from different disciplines to create projects appropriate for their disparate fields and goals (e.g. architecture, art, CS, design, etc.) Collaboration will be encouraged.

While images, videos, and web sites illustrating different approaches will be presented throughout the course, students will learn primarily by creating and experimenting with their own projects. The class format will take on a variety of styles as the disparate subjects dictate. Examples will be presented in lectures and demonstrations, and in-class hands-on labs will allow students to work together on problem solving. Assignment results will be presented in group critique sessions.

Students must demonstrate satisfactory achievement of course objectives through fulfillment of course projects and by contributing to class discussions and critiques. Course projects will require students to use a variety of software and equipment at ACCAD. Collaboration between students in the course and other faculty, staff and students at ACCAD is encouraged. Course evaluation will be based on the following:

Projects one through three:	20% each
Final Project:	30%
Class Participation:	10%

Attendance and Conduct Policies:

All students are required to be on time and in attendance for each and every class. Students arriving to class more than 10 minutes late will be counted as absent. Two unexcused absences (and each absence thereafter) will lower a final grade by 1/3 a letter. 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. 10% per day will be subtracted from late assignments.

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 each class. Problems with home systems and/or incompatibility will not be an acceptable excuse for missed goals. Technical problems will happen frequently during the quarter and students may have trouble accessing the computer lab during "prime time" hours. Students must make their own arrangements for overcoming these difficulties and submitting their work on time. Unless there is a complete system failure in a computer-related course, technical difficulties are not an acceptable excuse for not meeting a deadline. Students should plan their time and work so as to anticipate the technical hurdles that are a part of this profession.

Academic Misconduct (rule 3335-31-02) is defined as "any activity which tends to compromise the academic integrity of the institution, or subvert the educational process." Please refer to rule 3335-31-02 in the student code of conduct for examples of academic misconduct.

To register a documented disability, please call the Office of Disability Services (located in 150 Pomerene Hall) at 292-3307; or 292-0901 TDD, and notify the professor.

Topics and Assignments:

While exact topics will vary from year to year based on student background and interests, as well as emerging software capabilities, the topics will be a subset of:

- Interactive tools for building procedural systems
- Construction history based hierarchical parametric modeling, animation, dynamics, and behavior
- Procedural geometry: e.g. L-systems, subdivision, isosurfaces, superquads, fonts, expressions
- Cross-context procedural mapping: e.g. shape drives shading, motion affects form, etc.
- Dynamics: rigid/soft body, fluid, wire, cloth
- Behavioral animation

Example weekly course calendar:

1. Procedural interface (workflow, assets, "takes")
2. Parametric modeling: generative geometry, construction history
3. Parametric modeling: hierarchical structure, form drives surface
4. Generative animation: functions, algorithmic manipulation, variation control
5. Particle systems: creation, control, modeling
6. Particle systems: effects – fire, water, earth, air
7. Behavioral animation: attraction/repulsion, conditions and state change
8. Rigid body dynamics: forces, collisions, fracture
9. Soft body dynamics: geometry, wires, cloth
10. Fluid simulation and custom dynamic systems

Assignments during the class will require the use of the procedural techniques covered in lectures to generate images and animations. A final project allows students to choose which area is most applicable to their interests for further in-depth study.

Example Assignments:

1. **Parametric Model:** construct a geometric, hierarchical structure; provide a set of both high and low level parametric controls for adjusting the structure's form which should be generated using the techniques discussed in class.
2. **Particle Systems:** render an animation showing a range of behaviors that demonstrate several of the particle system techniques and strategies presented in class.
3. **Dynamics:** create a short animation showing a combination of three of the different dynamics systems covered in class.
4. **Final Project:** explore, extend, and/or integrate one or more of the previous concepts or projects.

Reading List:

Software documentation, tutorials, and examples are provided with the software installation and are available for download. Blogs, artist web sites, and so forth will be updated annually to reflect current technologies.

Bibliography and Resources:

Fry, Ben and Casey Reas. "Processing", <http://processing.org>

Hodgin, Robert. "Flight404", <http://www.flight404.com>

Prudence, Paul. "Data is Nature", <http://www.dataisnature.com>

Side Effects Software. "Houdini", <http://www.sidefx.com>

Universal Everything. "Advanced Beauty", <http://advancedbeauty.org>

Watz, Marius. "Generator.x", <http://www.generatorx.no>

Zerouni, Craig. Houdini On the Spot. Focal Press, 2007