GRAPHIC DESIGN & DIGITAL MEDIA
CITA/CSCI-140 ♦ FALL 2021

Contact Information

Instructor: Mrs. Christine Moore
Email: moorec@cofc.edu
Phone: (843) 953-4997
Office: Harbor Walk East, Rm 316
Website: OAKs & moorec.people.cofc.edu

Office Hours:

MWF: 02:30pm - 03:30pm
Thursday: 09:30 - 12:30pm
Appointments at other times are welcome

Inside this Syllabus

> *Come go with me. Let's make beautiful things happen with code and design.*
>
> — *Christine Moore* —

Course Overview

A great way to begin describing this course is that it involves "drawing with code." But wait, there's more! Not only will you learn the fundamentals of computer programming by creating visually stimulating designs, but you will also learn to apply popular design theory to aid in communicating the desired messages through digital media.

The course is taught primarily using Processing, a language for programming images, animation, sound, and games. We start simply, with creating static sketches in order to get comfortable with coding basic shapes, colors, and more complex compositions. Then as you begin to learn programming constructs such as variables and loops, your programs even more alive with animation and interactivity

Around Midterm, you'll have a project in which functions or objects are used to create more intricate animations of your choosing. During a two-week period, there will be small group presentations on principles of design. During the same time, we will take a pause from programming and use a graphic editing software such as Photopea as we take a closer look at how images are produced and edited in such applications.

In the Final Project, you'll design an interactive art composition or a game, while demonstrating a composite of knowledge and skills gained during the semester. The project will be judged on the effective implementation of design principles in the code.

Required Resources

Textbook:

Learning Processing, Second Edition: A Beginner's Guide to Programming Images, Animation, and Interaction by Daniel Shiffman (publisher: Morgan Kaufmann) ISBN: 978-0123944436

Laptop:

In conformity with the college laptop policy, all students are required to have a laptop with a working camera and microphone. We are hoping that all of our meetings will be in person this semester. However, should we ever need to move the class online, be sure to turn your camera on.

Chromebooks are not recommended because they cannot run all of the software you may need, and some models have issues connecting to our wireless network. The software that you will need to use for this class are free and do not require a lot of hard drive space or memory.

Other Resources

≈ Required readings from various free sources will be assigned from time to time.
≈ Processing Programming website: www.processing.org
≈ Here you can browse and share Processing sketches: www.openprocessing.org
≈ It is a good idea to back up your files on a storage device such as OneDrive, Google Drive, or a flash drive. In that way, if something goes awry with your laptop, your files will not be lost.

Tutoring: Computer Science now has a walk-in tutoring lab at CSL. The schedule for tutors will be set a few days after the semester starts.

Technology Assistance: If you are having trouble logging into MyCharleston, OAKS or your College email account contact the College of Charleston HelpDesk at (843) 953-3375. In addition, technology information and tutorials on many topics are available at the Student Computing Support page.
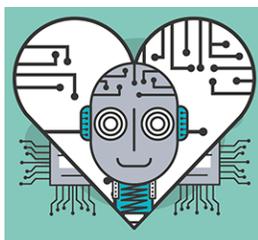
Catalog Course Description

A course introducing the creative side of computing in the context of applying graphic design principles in digital media. It emphasizes computer programming related to image processing. Students will produce raster and vector graphics, motion graphics, and interactive web applications while implementing the fundamentals of graphic design.

Inclement Weather or Substantial Interruption of Instruction

If in-person classes are suspended, I will set up and inform students of a change in modality to ensure the continuity of learning. The plans and nature of work will be commensurate with the type and length of interruption, and where we are in the course at that time.

All students must have access to a computer equipped with a web camera, microphone, and Internet access. Resources are available to provide students with these essential tools.


Grading Scheme and Scale

Evaluation Scheme:

Assignments and Class Work 40%

Projects 25%

Tests & Assessments 25%

Attendance & Participation 10%

Grading Scale

A	93 - 100
A-	90 - 92
B+	88 - 89
B	83 - 87
B-	80 - 82
C+	78 - 79
C	73 - 77
C-	70 - 72
D	60 - 69%
F	Less than 60%

Learning Outcomes

- To understand and apply basic algorithms for generating, manipulating, and representing digital information in the creation of digital media.
- To apply the software development process in program development.
- To apply various data types to represent information.
- To apply variables in program development.
- To design expressions using arithmetic, relational, and logical operators.
- To design selection statements.
- To design repetition statements.
- To design simple data structures using lists.
- To design and use functions.
- To apply event-driven programming
- To understand and apply principles of graphic design.
- To process and create composite digital images and digital audio.
- To apply the concept development process.
- Apply conceptual, aesthetic, and computational skills in the creation and critique of designs.
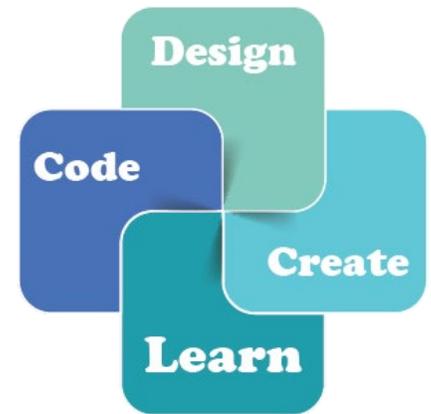
Participation and Attendance

Success in the course requires reading and reviewing course material BEFORE class. As well, you should work through the textbook exercises and practice questions for greater comprehension.

Excessive absences will result in the lowering in the Participation component of your grade. As well, absences may result in you missing in-class assignments that cannot be made up. To the extent possible, I will make reasonable accommodations for absences related to COVID19 and other absences that are beyond your control.

If you need to miss class for any unavoidable reasons, please let me know as soon as possible. You won't have to share private information. As for missing class notes and such, it is a good idea to connect with a few **study buddies** ahead of time. In that way you can have mutually supportive roles in studying together, sharing notes, and troubleshooting code.

Above all, please remember that you are responsible for course content and assignments whether or not you are in attendance. To the greatest extent possible, you will need to take responsibility for staying current.

Late Assignments

Assignments are due at the beginning of the class period on their due date. Thirty (30) points will be deducted if an assignment is late. Late assignments must be submitted no later than the next class period. Due dates for assignments will be strictly enforced.

Tests & Assessments

Tests and exams are to be taken during the specified time. Make-up tests will only be given for compelling reasons for absence, such as sickness or death.

Computer & Electronic Device Policy

When we are in the classroom, everyone has a computer in front of them, and a large part of our time will be spent working on them. However, while we are having class discussions or presentations, I do not expect you to use your computers for wasteful activities. Please do not text, use social media, or use electronic devices in any way that is disruptive to your learning.

Disability Accommodation

Any student eligible for and needing accommodations because of a disability is encouraged to speak with me during the first two weeks of class or as soon as you have been approved (Center for Disability Services/SNAP). In that way, we can talk about accommodations appropriate to your needs.

Academic Integrity

Academic integrity is taken seriously in this course. Please be aware of the specific requirements of the Honor Code at the College of Charleston.

You may be wondering what "academic integrity" means in the context of programming. In this course, you may consult with other students for conceptual and debugging help while working on your code, but unless otherwise specified on the assignment the final code you submit should be written, tested, and documented by you. This means that if two students submit code that is substantially the same, we will consider this a likely academic violation. All assignments will be automatically scanned for similarity.

It is a usual practice for real-world programmers to find and adapt publicly available code written by others in their own projects, and you may also do this in this class. If you use "found code" from online sources, you must bring that code up to the standards expected in this course. You must also identify which code is "found" and document its source with a comment in your code, just as you would for an academic citation in a written paper. Unless otherwise specified for a particular assignment, found code may not take up more than 10% of the code that you submit (counting by correctly formatted lines).

Mental & Physical Wellbeing

At the college, we take every student's mental and physical wellbeing seriously. If you find yourself experiencing physical illnesses, please reach out to student health services (843.953.5520). And if you find yourself experiencing any mental health challenges (for example, anxiety, depression, stressful life events, sleep deprivation, and/or loneliness/homesickness) please consider contacting either the Counseling Center (professional counselors at counseling.cofc.edu or 843.953.5640 3rd Robert Scott Small Building) or the Students 4 Support (certified volunteers through texting "4support" to 839863, visit counseling.cofc.edu/cct, or meet with them in person 3rd Floor Stern Center). These services are there for you to help you cope with difficulties you may be experiencing and to maintain optimal physical and mental health.

Schedule of Topics

- ▶ Processing, Drawing Primitive Shapes
- ▶ Flow of Control and Interaction
- ▶ Variables & Control Structures
- ▶ Loops and Concept of Iteration
- ▶ Functions and Reuse
- ▶ Classes and Objects
- ▶ Image Editing with graphic editors
- ▶ Visual Design Principles
- ▶ Image Processing with Code
- ▶ Arrays
- ▶ Debugging
- ▶ Algorithms
- ▶ Text & Data Input
- ▶ Translation and Rotation (3-D)
- ▶ Other topics as permitted