

# ACCAD 5102 – 30 40

PROGRAMMING CONCEPTS AND APPLICATIONS FOR ARTISTS AND DESIGNERS

## Course Overview

### **Instructor: Gaëtan Robillard**

Email: robillard.11@osu.edu

Office: Hayes Hall, Room

Office hours: 9h-12h am, on Wednesday and Friday (Zoom)

Preferred method of contact: email, Teams chat, Zoom

### **Location: Sullivant Hall 349A**

Credit hours: 3

## Course Description

Fundamental programming concepts are useful to artists and designers for creating algorithmic-based graphics and graphics tools.

Using computer programs is an inherent part of being a digital artist or designer. The programs we use are largely defined by programmers' understanding of artists' and designers' needs. Misconceptions about needs often result in programs that fail to properly address these requirements. There are two ways in which artists' needs may be better met: either programmers can learn more about art, or the artists can learn more about computer programming. The purpose of this course is to attempt to fulfill the second method. If artists know more about computer programming, they are better able to explain their needs to programmers. In addition, artists can use their programming skills to write their own programs.

Some examples of practical techniques that would be useful for artists and designers to be able to implement include parametric, iterative, or recursive design, hierarchical control, maintaining persistent state information, pattern matching, command generation, object-oriented design, data re-usability, data filtering and visualization, file and process management, template-based data generation, batch processing, and algorithmic drawing and animation.

The main objective of this course is to teach beginning to intermediate concepts of computer programming. The course will emphasize the application of programming methods that will best serve the artist. Each student will be expected to master fundamental, theoretical concepts in programming, thus allowing them to use software dependent upon those concepts more effectively and efficiently.

## Course Learning Outcomes

By the end of this course, students should successfully be able to:

- Apply basic programming concepts
- Program animated visual applications
- Implement common interaction techniques
- Create simple data visualizations
- Methodically locate problems when coding
- Teach themselves additional programming languages

## How this course works

### Mode of delivery

The course is in-person.

### Credit hours and work expectations

Use this statement for courses that meet for 2 hours and 40 minutes per week:

This is a **3-credit-hour course**. According to Ohio State policy ([go.osu.edu/credithours](https://go.osu.edu/credithours)), students should expect around 3 hours per week of time spent on direct instruction (instructor content and Carmen activities, for example) in addition to 6 hours of homework (reading and assignment preparation, for example) to receive a grade of (C) average.

### Participation requirements

#### Attendance

Design is a field that requires discipline, timely participation, and respectful and thoughtful communication. Timely and consistent engagements are critical in all formats used to deliver the content of this course. You are expected to come to class prepared and to participate actively in each class. Spontaneous and planned learning experiences and discussions are impossible to recapture or duplicate. You can anticipate that your instructor will take attendance.

The Department of Design recognizes that students may on occasion miss class due to extenuating circumstances such as illness, emergency, or other important matters. When this occurs, it is your responsibility to request updates and notes from a peer and to review any course material on Carmen that is associated with the class you missed.

#### Department of Design attendance expectations:

The Department of Design acknowledges that illness, family obligations, and other conflicts with your classes do occur from time to time. We do not distinguish between excused and unexcused absences.

For courses that meet twice per week, up to three absences are allowed for any reason during the semester without penalty. All absences from class will be counted, however, and if three absences have occurred, **you are required to initiate a meeting with your instructor to discuss what is preventing your attendance, and strategies for avoiding additional absences.** If you do not initiate this meeting prior to being absent a fourth time, you will receive an "E" grade for the

course when your fourth absence occurs. If you meet with your instructor before missing a fourth class, you *may* be allowed additional absences at your instructor's discretion, but ANY student who misses more than six class meetings in a semester will earn an "E" grade. Students who are registered with SLDS and have approved Flex Plans should follow the attendance requirements that are designated there.

For courses that meet once per week, up to two absences are allowed for any reason during the semester without penalty, but you will receive an "E" grade for the course when your third absence occurs. Students who are registered with SLDS and have approved Flex Plans should follow the attendance requirements that are designated there.

Our approach to attendance is summarized in this table:

	ABSENCES	CONDITION	CONDITION
<b>CLASS MEETS ONCE PER WEEK</b>	1-2	None	No effect on grade
	After 2		E grade assigned if maximum exceeded
<b>CLASS MEETS TWICE PER WEEK</b>	1-3	None	No effect on grade
	Before 4th	Must meet with instructor	Number of allowed absences determined
		If meeting not held	E grade assigned for 4 <sup>th</sup> absence
	4-6	# of additional absences approved at instructor's discretion of allowed by SLDS accommodations	E grade assigned if maximum exceeded

Making productive use of class time and meeting in-progress deadlines are also factors of the professional behavior this course is intended to cultivate. Students are expected to be present and active for entire class periods. Frequent tardiness or leaving early can, therefore, impact your project/course grades in a detrimental manner.

#### Use of learning management system (Carmen)

You will use Carmen to upload digital files of completed assignments for evaluation unless otherwise specified for an assignment or exercise. Carmen may also be used for class discussions and providing feedback (including scores/grades). Announcements in Carmen will inform all students about

schedule modifications or alternative planning. Teams can also be used for communication and for sharing information and files, if specified by your instructor.

### Office hours (optional)

My office hours are a great time to receive one on one instruction or conversation with me, but they are optional.

## Grading and instructor response

### How your grade is calculated

Category	Points and/or Percentage
1. Drawing with Code: getting familiar with the programming tools and workflow, basic drawing functions, variables, math, logic and conditional statements.	15
2. Parametric Patterns: using functions, arguments, and iteration to generate visual patterns controlled by numbers.	15
3. Interactive Data Visualization: loading, representing, remapping, and visualizing data.	20
4. OOP and Simulation: representing and visualizing classes of interacting entities.	20
5. Final Project: explore, extend, and/or integrate one or more of the previous concepts or projects.	20
Class participation	10
Total	100

*See the assignment pages online in Carmen for due dates and detailed requirements.*

### Description of major course assignments

*While all five assignments are outlined here, additional details for each can be found on the course webpage.*

### Assignment #1 draw with code

This first assignment is intended for you to quickly gain experience with the iterative process of writing, testing, and modify code. You'll gain experience learning new tools, making appropriate use of example code and documentation, and get a first taste of "parametric design". Demonstrate breadth of experimentation (try a bunch of things!) instead of crafting One Perfect Masterpiece from one or two things. Document your effort (show your work, not just your final product.)

### **Assignment #2 functions, Parameters, and Iteration**

While the first assignment asked you to draw one thing, the second assignment challenges you to create a collection of similar things. You will use functions, arguments, and iteration, to generate a visual collection of objects/items/sketches/etc, controlled by parameters. E.g., a collection of buildings, faces, animals, etc. Each is built from a set of components but sizes/parts/colors/relationships are each different.

### **Assignment #3 data input and interactivity**

The third assignment challenges you to create a basic interactive data visualization.

You will need to:

- 1) load from a file, store, access, and use data values (e.g. lists of numbers, lines of text, tables of data, etc.)
- 2) remap these values to visual attributes such as size, color, shape, rotation, position, etc.
- 3) change visual attributes interactively based on mouse input (e.g. cursor position, speed, button presses)

In your final submission, you should NOT be using data values that are hard coded into your program (e.g. by typing it into an array in your code.) You should instead be loading the data from a file and then using the data that you loaded. The intent is that you can write a program that can load different files to show different things based on the data loaded.

### **Assignment #4 Object Oriented Programming and Simulation**

This assignment challenges you to represent and visualizing classes of interacting entities.

You will need to:

- create object classes and instances, constructors, properties, and methods
- use these objects to run a visual, animated system
- document how changes in the objects' properties modify the system

### **Assignment #5 final project**

The final assignment is your opportunity to work with one or more techniques that you have not yet had the opportunity to investigate, or to push something a bit further that you've already started.

### **Academic integrity and collaboration guidelines**

## What should I submit to Carmen (and how)?

- 1) Submit all of the html, javascript, data, and other files that I would need to run your work. If you are using the p5 editor for your work, a convenient zip file containing all of these files can be downloaded from "File/Download" (if Download is not in the menu see if you are logged in.)
- 2) Give me one or more images showing your results (jpegs are preferred).
- 3) If your project result is time-based instead of just an image (e.g., involving animation, sound, and/or interaction) then submit a video file (mp4 or mov) using your favorite screen capture solution (e.g. OBS or quicktime player). Please keep the file size under control.
- 4) Write a little bit about your intent, difficulties, and accomplishments. Tell me how you feel about your results. Doc, txt, or pdf file please. Less than a page is fine.
- 5) Submit any additional \_process\_ documentation you'd like me to see: notebook sketches, failed ideas you tried, steps along the way, work diary, etc.
- 6) ALL of the files mentioned above that you submit to me in Carmen should be collected inside ONE \_zipped\_ folder. Not rar or tar. Zip. So you will be only uploading ONE file to Carmen, named something like myCoolProject.zip or bob\_project\_2.zip. All of the things mentioned above should be inside this zip file, hopefully well organized and named in subfolders, etc.
- 7) When I unzip the (ONE) zip file you submitted to Carmen and look inside the resulting folder containing all of your files, it should be obvious what I should look at to see and run your work. (Please don't submit 10 things all named like tempFinal3final7.html)
- 8) Your project should run without issues. If files it needs to run are missing, or make global path references to your C: drive, then it won't run for me.
- 9) Do not submit your work as just links to web sites (e.g. don't direct me to youtube, notion, miro, p5.js editor, github, etc). I would like the actual files for all of the above.

Did you understand each of those? If not ask me!

The most common mistakes are:

- \* submitting several files, instead of one zip file containing all your files
- \* forgetting to do number 2 or 3
- \* just describing what you made with no reflection, for number 4

## ## Sharing

- 1) Post some documentation: images, video, links, code, questions, comments, etc (whatever you'd like to share) in the Teams discussion thread for the presentation day, so your classmates can see, review, respond, and discuss.
- 2) Everyone will be given ~5 minutes in class to say and show what they did. Prepare! Note this is not enough time for a line-by-line code review. Show what you did, then briefly mention highlights and challenges.
- 3) Everyone is welcome to review, comment, and respond in the Teams discussion thread. Unless specific types of critique are requested, please err on the side of communicating what worked well and ask questions about things you want to know more about.

## ## Grading Rubric

All requirements minimally completed will result in a "B" base grade.

Grades can be raised by doing the following:

- \* exceptional commenting and documentation
- \* exceptional attention to visual design
- \* exceptional (successful) creative problem solving
- \* exceptionally interesting ideas

Grades will be reduced if:

- \* a part is broken in a way that impacts requirements
- \* didn't satisfy or ignored a requirement
- \* messy hard to follow code ("spaghetti code")
- \* work shown not appropriate for time given (document your effort)
- \* you submit work you didn't create and don't provide references (e.g. copied code)

### **Late assignments**

In design courses at the 2000 level and in non-major courses, students are granted a two-day grace period after due dates with no penalty for lateness (no matter the cause), as long as they have notified their instructor that a submission will be late prior to its deadline. Emailed notification is preferred.

Total scores for an assignment submitted more than 48 hours after a deadline will be reduced by 10%. Scores for any work submitted more than seven (7) days after a deadline will be reduced by 50%. All course work must be submitted by the last regular class meeting of the semester to be eligible for evaluation/grading. Flex plans for students registered with SLDS will be honored.

In design major courses at the 3000, 4000, and 5000 levels, no grace period is provided and total scores for assignments submitted up to 7 days after a deadline will be reduced by 10%. Total scores for assignments submitted more than 7 days after a deadline will be reduced by 50% as long as they are received by the last regular class meeting of the semester. Flex plans for students registered with SLDS will be honored.

Our approach to accepting late submissions is summarized in this table:

	<b>TIME FRAME</b>	<b>PENALTY</b>	<b>CONDITION</b>
<b>2000 LEVEL COURSES AND COURSES FOR NON-</b>	48-hour grace period	No penalty	Notify instructor before official deadline
	3 - 7 days late	10% deduction	

<b>DESIGN MAJORS</b>	8 days late – last class meeting of the semester	50% deduction
<b>3000-5000 LEVEL COURSES</b>	0-7 days late	10% deduction
	8 days- last class meeting of the semester	50% deduction

### Departmental Grading Scale

**A** (93–100) Work, initiative, and participation of exceptional quality

**A-** (90–92.9) Work, initiative and participation of very high quality

**B+** (87–89.9) Work, initiative and participation of high quality which reflects higher than average abilities

**B** (83–86.9) Very good work, initiative and participation that satisfies the goals of the course

**B-** (80–82.9) Slightly above average work, initiative and participation that satisfies the goals of the course

**C+** (77–79.9) Average work, initiative and participation which reflects an understanding of course material

**C** (73–76.9) Adequate work; student has a less than average level of initiative and participation

**C-** (70–72.9) Passing but below good academic standing; student has a less than average level of work, initiative and participation

**D+** (67–69.9) Below average work, initiative and participation

**D** (60–66.9) Well below average work, initiative and participation

**E** (59.9–0) Failure; no credit. Unsuccessful completion of work. Limited or no participation. Objectives of the assignment are not met or are met in a significantly limited way.

### Instructor feedback and response time

Project grading and feedback can generally be expected within two weeks of the submission of work. You can expect to receive an update on your performance status in my course before the middle of Week 4 and the end of Week 10 (at minimum). As your instructor, I am here to help and support you. Please engage with me either via Carmen, email, TEAMS, or in person by scheduling an appointment. I will make every effort to reply to emails within 24 hours M-F, but do not guarantee a response between 8pm and 7am or on weekends.



## Course Materials and Technologies/Tools

### Required Supplemental Materials

There are no required textbooks, specific software, or specific hardware. Software documentation, tutorials, and examples are provided with software installation or are available for download or online. Links to blogs, web sites, etc. are updated through the semester via Carmen, reflecting emerging technologies

### Communication Tools

Optional video conferencing with your instructor requires the use of a video call using Zoom. Therefore, it is required that you install the Zoom application. Be sure to log in with the SSO (osu.zoom.us). You will also need a microphone for voice connection. Video connection is optional, but highly recommended to ensure optimized communication.

### Documentation Tools

You are not required to print anything for this course. All of your work will be uploaded as a digital file. In some cases, images may be included in your work, and you will be required to upload them to Carmen. I will be happy to guide you through this process (as well as file size management) if you require help.

### Optional Materials

N/A

### Carmen access

You will need to use BuckeyePass ([buckeyepass.osu.edu](https://buckeyepass.osu.edu)) multi-factor authentication to access your courses in Carmen. To ensure that you are able to connect to Carmen at all times, it is recommended that you take the following steps:

- Register multiple devices in case something happens to your primary device. Visit the BuckeyePass
- Request passcodes to keep as a backup authentication option. When you see the Duo login screen on your computer, click **Enter a Passcode** and then click the **Text me new codes** button that appears. This will text you ten passcodes good for 365 days that can each be used once.
- Download the Duo Mobile application to all of your registered devices for the ability to generate one-time codes in the event that you lose cell, data, or Wi-Fi service

If none of these options will meet the needs of your situation, you can contact the IT Service Desk at 614-688-4357(HELP) and IT support staff will work out a solution with you.

## General Class and Studio Policies

**Writing style.** Any written submissions should follow the standard English guidelines for using proper grammar, spelling, and punctuation. Informality is fine for non-academic topics.

**Tone and civility.** In verbal exchanges and in writing, let's maintain a supportive learning community where everyone feels safe and where people can disagree amicably. Remember that sarcasm doesn't always come across online or in writing.

**Citing your sources.** In any written form of academic submission, please cite your sources to back up what you say, for any images you use, and for any existing code you use. If you use code or if it is particularly inspired by another work and wish to include, mimic, or apply any part of it to your work, cite it. While use of precedents (existing designs/ideas) is expected to inspire new iterations and build skills, you are expected to credit your sources. For published or online written or visual materials, please use the **APA citation style**. If you are not accustomed to writing in APA style (which is preferred by the disciplines encompassed in this course), you can use a citation generator such as <https://www.mybib.com/tools/apa-citation-generator> to help you get it right.

**Artificial Intelligence and Academic Integrity.** Generative AI tools can be helpful for certain tasks, such as writing or analyzing code. However, using these tools to complete software code for assignments is not permitted. Exceptions may be granted only when explicitly authorized by the instructor, for example, to debug existing code or translate code between programming languages.

**Protecting and saving your work.** Consider composing anything you submit for this course using software that allows you to save your work separately. I recommend that you copy into the Carmen drop box for submission. Please do not submit one-of-a-kind material.

**Communication tool: Carmen.** Carmen ([carmen.osu.edu](https://carmen.osu.edu)) will be used for all communication specific to individual sections through announcements (for example: if I need to start class late or need to update you on a specific detail). Carmen will also be where all grades, readings, and lecture PDFs are posted. Assignment details will be shared on Carmen on each assignment page.

**Communication tool: Email.** Email through Carmen's inbox function or through your BuckeyeMail will be the only source of private and secure conversations. Information general personal matters, assignment or class inquiries or other similar topics should be addressed using these two sources.

All university correspondence is sent to your BuckeyeMail email address, and all email sent to faculty and staff should be sent from your BuckeyeMail email address.

Ohio State will never ask for your Ohio State username or password. Do not reply to any email asking for your Ohio State username, password, or other personal information. Report such messages to [report-phish@osu.edu](mailto:report-phish@osu.edu).

**Communication tool: Teams.** The course will make use of Teams for file sharing or communication. Your instructor may allow DMing (and have notifications on at certain times) where

you could contact them for clarifications or questions. However, the DM function should never be used for private information (such as grades).

**Reusing past work.** In general, you are prohibited in university courses from turning in work from a past class to your current class, even if you modify it. If you want to build on past research or revisit a topic you've explored in previous courses, please discuss the situation with your instructor at the start of the assignment/project.

**Grade Forgiveness.** The Grade Forgiveness Rule allows undergraduate students to petition to repeat up to three courses. The grade in the repeated course will permanently replace the original grade for the course in the calculation of the student's cumulative GPA. Only a first repeat can be used this way; all other repeats of the same course will be included under the general course repeatability rule.

The original grade will remain on the student's transcript and some graduate/professional school admission processes will re-calculate the student's GPA to include the original grade. See: <https://advising.osu.edu/grade-forgiveness-0> for more information.

## Academic Policies

Please see the Office of Undergraduate Education's Syllabus Policies and Statements for course policies on Academic Misconduct, Student Life – Disability Services, Religious Accommodations, and Intellectual Diversity. Optional statements explaining university policy related to copyright, content warnings, Counseling and Consultation Services/Mental Health, and services for military-connected students can be found here.

# CALENDAR

WEEK/ MODULE	TOPIC	DUE DATES/DEADLINES
1	Overview, introductions, tools, drawing, variables, conditionals	
2	Reading, web, debugging Project 1 presentations	Project 1 due
3	Functions, arguments, loops, nesting	
4	Loops, recursion Project 2 work	
5	Project 2 presentations Interaction, events, remapping	Project 2 due
6	Arrays, remapping File I/O	
7	Project 3 work Project 3 work	
8	Project 3 presentations Object oriented programming	Project 3 due
9	Simulation Image representation	
10	Spring break Spring break	
11	Libraries Project 4 work	
12	Project 4 work Project 4 presentations	Project 4 due
13	APIs DOM, instantiation	
14	Languages Project 5 work	
15	Project 5 work Project 5 presentations	Project 5 due

*See web pages for more detail.*