

# PROG 110 Art of Code

## SP2024

### Course Details

**Credits:** 3

**Prerequisites:** MATH 110 College Mathematics or MATH 115 Liberal Arts Mathematics or MATH 120 Quantitative Reasoning or MATH 150 Geometry in the Arts or MATH 155 Math in Art and Nature or MATH 160 Mathematical Ideas or MATH 165 Math for Marketing and Management or ACCU-T-67 EXAM-ACCUPLACER MATH MINIMUM SCORE = 67 or ACT-M-23 EXAM-ACT MATH MINIMUM SCORE = 23 or SAT-M-550 EXAM-SAT MATH MINIMUM SCORE = 550 or CMPS-M-67 EXAM-CMPS MATH MINIMUM SCORE = 67 or ALEK-1-46 EXAM-ALEKS MINIMUM SCORE = 46

**Time:** Thursday, 9-11:50 AM

**Place:** Building 33, 609

**Instructor:** Teerath Majumder

**Instructor Email:** [tmajumder@colum.edu](mailto:tmajumder@colum.edu)

**Office Hours:** Wednesday and Thursday, 3:30-5:30 PM

### Course Description

In this course, you will develop projects specific to your major as you learn a disciplined approach to problem-solving and algorithm development with programming. Topics covered throughout the semester include data abstraction, procedural structures, sequence control, repetition, and best practices.

### Learning Outcomes

A student who has successfully completed this course will have learned...

- the building blocks of text-based computer coding and how to use them to write programs
- logics of how a computer programming language works
- to solve problems using conditional and iterative processes
- basics of how to utilize computer programming to analyze, create and manipulate audio signals

### Evaluation

Students will be evaluated on their performance in the following:

- Coding assignment 1 - 15%
- Coding assignment 2 - 15%
- Coding assignment 3 - 15%

- Coding assignment 4 - 15%
- Coding assignment 5 - 20%
- Coding assignment 6 - 20%

All assignments must be submitted on Canvas. It is the student's responsibility to submit each assignment before its posted deadline. **If a student is unable to turn in their work before the deadline, they must inform the instructor in advance by email or Canvas message.** The instructor *may* grant an extension based on the student's rationale for submitting their work past the deadline. **Otherwise, 10% of the score will be deducted for every 24 hours past the posted deadline.** That means that if you submit your work ten or more days past the deadline, you will receive no points for the assignment. The assignment will remain open until ten days after the deadline.

## Grading Scale

93% ≤ A ≤ 100%	73% ≤ C < 77%
90% ≤ A- < 93%	70% ≤ C- < 73%
87% ≤ B+ < 90%	60% ≤ D < 70%
83% ≤ B < 87%	0% ≤ F < 60%
80% ≤ B- < 83%	I = Incomplete
77% ≤ C+ < 80%	

## Texts (Recommended)

- Downey, Allen (2015). *Think Python: How to Think Like a Computer Scientist*. 2nd ed. O'Reilly Media. 292 pp. isbn: 978-1491939369. url: <https://greenteapress.com/wp/think-python-2e/> (visited on 04/19/2020). Available as a free e-book online (.pdf, .html).
- Alvarado, Christine et al. (2019). *CS for All: An Introduction to Computer Science Using Python*. Franklin, Beedle & Associates Inc. 288 pp. isbn: 978-1590282908. Accompanying website: <https://www.cs.hmc.edu/csforall>.
- Guttag, John V. (2021). *Introduction to Computation and Programming Using Python: With Application to Computational Modeling and Understanding Data*. 3rd ed. Mit Press. 496 pp. isbn: 978-0262542364.
- Varoquaux, Gaël, Emmanuelle Gouillart, and Olav Vahtras, eds. (2017). *SciPy Lecture Notes*. url: <http://www.scipy-lectures.org> (visited on 02/08/2018). Downloadable version available in multiple formats (.pdf, .html).

- Downey, Allen B. (2016). *Think DSP: Digital Signal Processing in Python*. 1st ed. O'Reilly Media. 168 pp. isbn: 978-1491938454. url: <http://greenteapress.com/wp/think-dsp/> (visited on 05/14/2019). Available as a free e-book online (.pdf, .html).
- Lee, Kent D. and Steve Hubbard (2015). *Data Structures and Algorithms with Python*. Springer. isbn: 978-3-319-13071-2. doi: 10.1007/978-3-319-13072-9.

Most of these titles, as well as additional supplementary readings for this class, are also available through the Columbia College bookstore.

## Hardware Requirements

You will need reliable access to a laptop computer for work in class and at home. It can run any of the common operating systems (macOS, Windows, Linux). Please bring your laptop to every on-campus class meeting.

## Software Requirements

We will use the Anaconda Distribution as an environment for Python coding this semester. It is freely available for all major platforms (macOS, Windows, Gnu/Linux) and includes everything you will need: the Qt Console interactive IPython shell, the Spyder integrated development environment (IDE), and the Jupyter Notebook that will allow you to run the interactive slides for this class on your local computer. The Thonny ide, which is also freely available for all major platforms, is a useful alternative for students who run into problems installing Anaconda. It also includes an interactive Python shell (but no Jupyter kernel). Installation instructions for both packages will be provided in class.

## Jupyter Notebooks for This Class

The GitLab repository at <https://gitlab.com/floholl/art-of-code> includes so-called 'Jupyter notebooks' for (almost) all class meetings of the semester. You can think of these as interactive slides that include Python code, which you can conveniently run and edit directly in the slides. You can download the notebooks and run them locally on your computer, or you can launch them online in your browser through Binder. For details, see the instructions on the repository's landing page.

## Art\_of\_Code Online Python Course at the Runestone Academy

As part of this class, you will complete the Art\_of\_Code online Python course at the Runestone Academy for one of your assignments. Access instructions for this course will be shared through an announcement on Canvas during the first few weeks of the semester. Please stick to those rather than attempting to register for the course yourself.

## Communication

Students are encouraged to reach out to the instructor with any questions regarding the course through Canvas messages. **Please include in your message your full name, student ID number, and the course you are contacting about.**

## Academic Honesty

Collaboration between students in this course is strongly encouraged. Students are expected to exchange ideas, opinions, and information constantly, and to help each other with research and projects. However, each student is responsible for the completion of their own assignments.

In this class, you will be expected to attribute due credit to the originator of any ideas or words that you incorporate into your own work. **Any borrowed text, code, and sound must be cited.**

## Disability

If there are conditions that prevent a student from attending classes or participating fully in academic activities, the student is encouraged to consult [Services for Students with Disabilities](#) as soon as such conditions present themselves.

## General Reference

**School of Media Arts:** Tom Dowd, Interim Dean - [tdowd@colum.edu](mailto:tdowd@colum.edu)

**Department of Audio Arts and Acoustics:** Ben Sutherland, Chair - [bsutherland@colum.edu](mailto:bsutherland@colum.edu) (department phone: 312-369-8820)

## Semester Schedule

Week	Date	Lecture Topic	Assignment Due
1	01/25/24	Introduction	
2	02/01/24	Flow control: branching and iteration	Coding assignment 1
3	02/08/24	Collections and scripts	
4	02/15/24	Functions, namespaces and scopes	
5	02/22/24	Better code: read and write documentation; debugging	Coding assignment 2
6	02/29/24	Numeric arrays with NumPy	
7	03/07/24	Recording, playing and visualizing sound in Python	Coding assignment 3
8	03/14/24	Sound processing in Python	
9	03/21/24	Classes: object-oriented programming	Coding assignment 4
10	04/04/24	Sound synthesis in Python	
11	04/11/24	Better code: organizing larger projects and testing	
12	04/18/24	Sound analysis in Python	Coding assignment 5
13	04/25/24	Fast Fourier Transform (FFT)	
14	05/02/24	Final project presentations	
15	05/09/24		Coding assignment 6