

CS 106 Winter 2022

Introduction to Computer Science 2

Course Syllabus

About the Course

- LEARN Site <https://learn.uwaterloo.ca/>
 - This is the main site for the course. For announcements, course content, lab and assignment submissions, grades. Also used for Participation grade.
- Microsoft Teams Discussion Boards
 - For questions and clarifications about labs, assignments, and course content. Also used for Participation grade.
- Public Website <http://www.student.cs.uwaterloo.ca/~cs106>
 - For general information about the course.
- This is a fully online course with one exception, the Final Exam is in-person.

Course Description

This course, together with its predecessor CS 105, offers a comprehensive introduction to practical computer programming for students with no background in the subject, and who will not normally go on to further study in computer science. The course is required by students in the GBDA program, and available to students from other programs on campus.

The course is taught using the “JavaScript p5” (<https://p5js.org/>) programming environment. When working on assignments, the reference (<https://p5js.org/reference/>) section will be especially useful. We are using the Open Processing editing environment for creating JavaScript p5 sketches (<https://www.openprocessing.org/>)

While the main theme of CS 105 is to develop basic skills in imperative programming (variable declarations, control flow, loops, arrays, defining functions), in CS 106 we explore more general applications of programming in contexts of interest to visual artists and designers.

Course Objectives

The goal of CS 106 is to apply programming idioms in a practical context, using functionality available in built-in functions, and libraries that can be added on to it. Topics include input/output, user interface programming, procedural content generation, and text and structured data processing.

Textbook

- No required textbook, but there is a recommended textbook: Lauren McCarthy, Casey Reas, and Ben Fry. *Getting Started with p5.js*. Published October 2015, Maker Media. Paperback. An electronic version of the textbook is available through the University of Waterloo library at: https://ocul-wtl.primo.exlibrisgroup.com/permalink/01OCUL_WTL/5ob3ju/alma999986579998305162
- Anyone with an email address ending in uwaterloo.ca can access the above link. If you are asked to select your institution, then select “Not Listed” and log in using your uwaterloo.ca email address.

Lecture Handouts and Video Lectures available on LEARN

The lecture handouts contain the text and images of the lectures, as well as links to setches in Open Processing. Videos lectures including demonstration and explanations of the lecture handouts are also available. Lecture handouts and lecture videos are typically available on LEARN on Wednesdays at 3am. The lab and assignment associated with that content are normally due the following Tuesday and Friday respectively.

Course schedule

Week	Date	Lab	Lab Due 11:59pm	Asg Or Test	Asg or test Due 11:59pm
1	Wed Jan 5 – Fri Jan 7 Recap and Video	None	None	None	None
2	Mon Jan 10 – Fri Jan 14 Arrays and Strings	1	Tue Jan 11	1	Fri Jan 14
3	Mon Jan 17 – Fri Jan 21 Input and Output	2	Tue Jan 18	2	Fri Jan 21
4	Mon Jan 24 – Fri, Jan 28 Advanced Shapes	3	Tue Jan 25	3	Fri Jan 28
5	Mon Jan 31 – Fri Feb 4 Random and Noise	4	Tue Feb 1	4	Fri Feb 4
6	Mon Feb 7 – Fri Feb 11 HTML and CSS	5	Tue Feb 8	5	Fri Feb 11
7	Mon Feb 14 – Fri Feb 18 Text Processing	6	Tue Feb 15	Midterm Test	Fri Feb 18
	Mon Feb 21 – Fri Feb 25			Study Week (nothing is due)	
8	Mon Feb 28 – Fri Mar 4 Tables	7	Tue Mar 1	6	Fri Mar 4
9	Mon Mar 7 – Fri Mar 11 JSON/Trees	8	Tue Mar 8	7	Fri Mar 11
10	Mon Mar 14 – Fri Mar 18 JSON/Trees	9	Tue Mar 15	8	Fri Mar 18
11	Mon Mar 21 – Fri Mar 25 Geometric Context	None	None	Week 1 of Project	
12	Mon Mar 28 – Fri Apr 1	None	None	Week 2 of Project	Project due: Fri Apr 1, 11:59pm
13	Mon Apr 5 – Tue Apr 6	None	None	Final Exam	TBA

Course staff

All email addresses are @uwaterloo.ca.

Instructor: **Kevin Harrigan**

Email: kevinh@uwaterloo.ca

Office: MS Teams

Office hours: Mondays 9:30am-11:00am

Instructional Support Assistants: **Lakshmi Vinoj, Aksh Patel**

Email: cs106@uwaterloo.ca

Office: MS Teams

Office Hours: **Lakshmi**: Tuesdays 3:00pm-5:00pm, Wednesdays 10:00am-12:00pm, Thursdays 4:00pm-6:00pm, Fridays 2:00pm-4:00pm

Office Hours: **Aksh**: Tuesdays 1:00pm-3:00pm, Wednesdays 12:00pm-2:00pm, Thursdays 12:00pm-2:00pm, Fridays 12:00pm-2:00pm

Instructional Assistants (IAs): **Adam Campbell, Benyamin Jamialahmadi, Ken Jen Lee, Ehsan Jahangirzadeh Soure**

Office: MS Teams

Office Hours: **Adam**: Wednesdays 2:00pm-4:00pm, Fridays 10:00am-12:00pm

Office Hours: **Benyamin**: Tuesdays 6:00pm-8:00pm, Thursdays 2:00pm-4:00pm

Office Hours: **Ken**: Tuesdays 9:00am-11:00am, Thursdays 9:00am-11:00am

Office Hours: **Ehsan**: Mondays 12:00pm-2:00pm, Thursdays 6:00pm-8:00pm

Instructional Support Coordinator: **Scott King**

Email: sfking@uwaterloo.ca

Office: MS Teams

Office hours: Scheduled by email

List of planned topics

- JavaScript p5
 - A review of programming concepts from CS 105, in terms of the basic structure of JavaScript p5, and the Open Processing development environment.
- Arrays
 - High-level operations on arrays, including appending, concatenation, and removal. Built-in array manipulation functions.
- Strings
 - Working with characters and strings. String comparisons. Printing and displaying text.
- Input and output

- Loading files in various formats (text, images, illustrations) into JavaScript p5, writing files.
- Advanced Shapes
 - Drawing fancy shapes with `beginShape()` and `endShape()`.
- User interfaces
 - The model-view-controller architecture. Direct manipulation interfaces. User interface toolkits. Creating interfaces using JavaScript p5 and the DOM.
- Geometric context
 - The use of `translate()`, `rotate()`, and `scale()` to modify a program's coordinate system. Building a hierarchy of transformations using `push()` and `pop()`. Order of operations.
- Randomness and Noise
 - The `random()` function in detail. Pseudo randomness. Applications of randomness. Introduction to noise).
- Text processing
 - Decomposing text into tokens. Regular expressions. Unicode. Working with dates and times.
- Structured data processing
 - Dealing with text, table-structured (CSV) and tree-structured (JSON) data. Processing live data acquired from web APIs.
- “Pure” or “Vanilla” JavaScript
 - Using JavaScript without the p5 library.

Student Expectations

Assignments

There will be 8 assignments (usually due at 11:59pm Friday nights via LEARN), consisting primarily of programming questions. All assignments must be completed individually by students. Absolutely no late assignment submissions are permitted. Each student's final assignment grade will be computed by dropping their lowest-scoring individual assignment.

Final Project

There will be one Final Project due at the end of the term.

Labs

There will be weekly labs (usually due Tuesday nights via LEARN), consisting of smaller exercises and coding walk-throughs. All labs must be completed individually by students, though students are invited to discuss lab questions and share ideas.

Tests

There is a Midterm Test (online) and a Final Exam (in person). The Final Exam is in-person.

Marking Scheme A

This is the default marking scheme:

9 Labs (best 8 count): 8%

8 Assignments (best 7 count): 21%

Final Project: 10%

Midterm test (online): 15%

Final Exam (in person): 40%

Weekly Participation: 6%

Marking Scheme B

This marking scheme is used **ONLY IF** an in person final exam is **not** allowable:

9 Labs (best 8 count): 16%

8 Assignments (best 7 count): 28%

Final Project: 10%

Midterm test (online): 20%

Final Exam (online): 20%

Weekly Participation: 6%

Course Policies

MOSS

MOSS (Measure of Software Similarities) is used in this course as a means of comparing current students' assignments as well as comparing with past comparing students' assignments and code on the internet, to ensure academic integrity. We will report suspicious activity, and penalties for plagiarism (i.e. cheating) are severe. Please read the available information about academic integrity very carefully.

Collaboration

Group work is disallowed unless otherwise specified. Any excessive collaboration will be treated as a violation of academic integrity.

You are responsible for writing all the code in your assignments and labs. You may not receive code from anybody else, whether by email, by copying-and-pasting, by dictation, by having them type it into your Open Processing IDE, or by any other means.

When working on assignments, please do not show your code to other students, or allow them to see it accidentally. You can discuss assignment problems in general terms, and perhaps work together on strategies for solving them, but do not go so far as to develop code in pairs or groups.

When working on lab exercises, we're more tolerant of students looking over each other's shoulders and seeing ideas for solutions. But in the end, the work you submit must still be your own.

Late assignments and labs

Late assignments will receive no credit; consequently, you should aim to finish early, to allow for unexpected delays. After an assignment or lab due date has passed, you may still submit your work for feedback only (no marks) to the CS 106 ISAs by email and request for feedback.

Missed assignments and labs due to illness

With appropriate, authorized documentation (provided via a [Verification of Illness Form](#)), assignment and lab work may be excused. Please contact the ISC (Scott King) to see if this applies. If a missed assignment or lab is excused, its weight is distributed over the remaining unexcused assignments or labs. In the interest of understanding the course material for future assignments and exams, students who miss work are encouraged to complete it anyway, and submit it for feedback from the ISAs.

Remarking

If you have problems with the marking of an assignment, please contact the ISAs within two weeks of the date the assignment's mark was made available on LEARN. The email must include your name, student number, Quest user ID, and assignment number. We also require that you list the questions you feel were marked incorrectly, and, for each of those questions, why you feel your mark should be changed. Please be aware that the assignment will be remarked in its entirety.

If you have problems with the marking of a midterm exam, please fill out a re-mark request form. Details will be provided after the midterm.

University Policies

Academic integrity: In order to maintain a culture of academic integrity, members of the University of Waterloo community are expected to promote honesty, trust, fairness, respect and responsibility. [Check [the Office of Academic Integrity](#) for more information.]

Grievance: A student who believes that a decision affecting some aspect of his/her university life has been unfair or unreasonable may have grounds for initiating a grievance. Read [Policy 70, Student Petitions and Grievances, Section 4](#). When in doubt, please be certain to contact the department's administrative assistant who will provide further assistance.

Discipline: A student is expected to know what constitutes academic integrity to avoid committing an academic offence, and to take responsibility for his/her actions. [Check [the Office of Academic Integrity](#) for more information.] A student who is unsure whether an action constitutes an offence, or who needs help in learning how to avoid offences (e.g., plagiarism, cheating) or about "rules" for group work/collaboration should seek guidance from the course instructor, academic advisor, or the undergraduate associate dean. For information on categories of offences and types of penalties, students should refer to [Policy 71, Student Discipline](#). For typical penalties, check [Guidelines for the Assessment of Penalties](#).

Appeals: A decision made or penalty imposed under [Policy 70, Student Petitions and Grievances](#) (other than a petition) or [Policy 71, Student Discipline](#) may be appealed if there is a ground. A student who believes he/she has a ground for an appeal should refer to [Policy 72, Student Appeals](#).

Note for students with disabilities: [AccessAbility Services](#), located in Needles Hall, Room 1401, collaborates with all academic departments to arrange appropriate accommodations for students with disabilities without compromising the

academic integrity of the curriculum. If you require academic accommodations to lessen the impact of your disability, please register with AccessAbility Services at the beginning of each academic term.

Turnitin.com: Text matching software (Turnitin®) may be used to screen assignments in this course. Turnitin® is used to verify that all materials and sources in assignments are documented. Students' submissions are stored on a U.S. server, therefore students must be given an alternative (e.g., scaffolded assignment or annotated bibliography), if they are concerned about their privacy and/or security. Students will be given due notice, in the first week of the term and/or at the time assignment details are provided, about arrangements and alternatives for the use of Turnitin in this course.

It is the responsibility of the student to notify the instructor if they, in the first week of term or at the time assignment details are provided, wish to submit alternate assignment.