CS 241 Organizational Information Winter 2026

Contents

1 Outline

The high-level organization of this course is discussed in the course outline: https://outline.uwaterloo.ca/viewer/view/nbrgvt. This document discusses more specific details of the day-by-day organization and schedule.

2 Office Hours

NOTE: The course content is the same in all sections. You can attend any office hours, and do not need to specifically attend the office hours of your instructor.

Instructors	**Email**	**Office Hours**	**Location**
Gregor Richards	gregor.richards@uwaterloo.ca	Tues,Thurs 12:30-2:00PM	MC 6457
Mark Petrick	mark.petrick@uwaterloo.ca	TBD	MC 6467
ISA			
Ike Yang	cs241@uwaterloo.ca	(TENT)3:00-4:00PM	Hybrid (MC 4065)
Mark Gasior	cs241@uwaterloo.ca		
IAs			
Olivier Lamonde		TBD	TBD
Instructional Support Coordinator			
Deven Wolff	dbwolff@uwaterloo.ca	Email for appointment	

3 Schedule

Week	Monday	Tuesday	Wednesday	Thursday	Friday
Week 1 Jan 5–11	**Classes begin**	Lecture 1 Coure intro & M1: Data representations		Lecture 2 M2: Machine language	A1 out
Week 2 Jan 12–18		Lecture 3 M3: Regular languages (part 1)		Lecture 4 M3: Regular languages (part 2)	A2 out
Week 3 Jan 19–25		Lecture 5 M3: Regular languages (part 3), assemblers		Lecture 6 M4: Assembly language (part 1)	A3 out **A1 due** Jan 23, 5PM
Week 4 Jan 26–Feb 1		Lecture 7 M4: Assembly language (part 2)		Lecture 8 M4: Assembly language (part 3)	A4 out **A2 due** Jan 30, 5PM
Week 5 Feb 2–8		Lecture 9 M5: Formal languages CFLs		Lecture 10 M6: Top-down parsing (part 1)	A5 out **A3 due** Feb 6, 5PM
Week 6 Feb 9–15		Lecture 11 M6: Top-down parsing (part 2)		Lecture 12 M7: Bottom-up parsing (part 1, LR(0))	
Reading week					
Week 7 Feb 23–Mar 1	**A4 due** Feb 23, 5PM	Lecture 13 M7: Bottom-up parsing (part 2, SLR(1))		Lecture 14 M8: Context-sensitive analysis (part 1, types)	A6 out
Week 8 Mar 2–8		Lecture 15 M8 p.2: More types M9: Code generation (p.1)	**Midterm exam** Mar 4 4:30PM	Lecture 16 M9: Code generation (part 2)	A7–A8 out
Week 9 Mar 9–15		Lecture 17 M9: Code generation (part 3)		Lecture 18 M9: Code generation (part 4, pointers)	**A5 due** Mar 13, 5PM
Week 10 Mar 16–22		Lecture 19 M9: Code generation (part 5, procedures)		Lecture 20 M10: Linking & loading (part 1)	**A6 due** Mar 20, 5PM
Week 11 Mar 23–29		Lecture 21 M10: Linking & loading (part 2)		Lecture 22 M8b: Optimization	**A7 due** Mar 27, 5PM
Week 12 Mar 30–Apr 5		Lecture 23 M11: Memory		Lecture 24 Summary and recap	GOOD FRIDAY NO CLASSES
Week 13!? Apr 6	**A8 due** Apr 6, 5PM **Classes end**				
Final exam (scheduling TBD)					

The scheduling of particular lecture topics is tentative and may change without notice.

4 Lectures and Tutorials

Lectures	Days	Time	Location	Instructor
LEC 001	Tue & Thur.	8:30–9:50am	MC 4045	Gregor Richards
LEC 002	Tue & Thur.	10:00–11:20am	MC 4045	Gregor Richards
LEC 003	Tue & Thur.	10:00–11:20am	MC 2065	Mark Petrick
Tutorials				
TUT 101	Wednesday	10:30–11:20am	MC 2065	
TUT 102	Wednesday	11:30am–12:20pm	MC 2065	
TUT 103	Wednesday	12:30–1:20pm	RCH 207	
TUT 104	Wednesday	2:30–3:20pm	MC 4040	

In the first week of the term, there will be no tutorials.

5 Web Resources

Course Web Site: https://student.cs.uwaterloo.ca/~cs241/ includes the course syllabus, assignment specifications, and course text.
Piazza: https://piazza.com/uwaterloo.ca/winter2026/cs241 is used for communication among students and instructors, and will also be used for course announcements.

6 Course Notes

The text for this course is course notes available from the course web site. The content of the course is defined by the notes; that is, anything mentioned in lectures that is not mentioned in the notes is not a component of the course, and anything mentioned in the notes that is not mentioned in the lectures is nonetheless something students are expected to learn. The instructors will of course endeavour to cover all course content in the lectures, and to specify what they didn't cover so that students know what part of the notes to focus on. Students are advised to read the course notes regardless.

Course notes will be made available at least a week in advance of the lecture that covers the material. Students are expected to read the course notes, but it's up to an individual student's prerogative whether they read them before or after the lectures that cover the same content.

The course notes include supplementary videos. Videos are supplementary and are not a required part of the reading, but you may find them very useful for your learning.

7 Assignments

There will be 8 assignments. Each assignment is worth 5% of your final grade. The entire grading scheme is detailed on the outline.

For most students, the course material can only be learned well by carefully working through each assignment. Students are expected to compile, run, and test their own code. Feedback on the correctness of your work is provided by the Marmoset submission and testing server, https://marmoset.student.cs.uwaterloo.ca. All assignments must be submitted electronically to Marmoset. A link to the Marmoset system may also be found on the course web page.

We recommend that you start working on the assignments early. Use Marmoset to assess your progress (and grade) after convincing yourself of correctness using your own self-designed tests. By the time you submit to Marmoset, you should be convinced by your own thorough testing that your program is perfect. (Hint: the test suites and automated testing that you were introduced to in CS246 would also work very well in CS241. Consider using them here.)

Most “high-level programming” questions on assignments can be done either in C++ or Racket. However, occasionally there might be questions that restrict you to use only C++. There will also be some “low-level programming” questions that use MIPS assembly language, which you will learn about in Module 2.

There will be no hand-marking or “style marks” for assignments. Assignment marks are based entirely on passing the Marmoset tests.

7.1 Assignments by module

Module	Topic	Practiced in
1	Data representations	Assignment 1
2	Machine language	Assignments 1, 2
3	Regular languages	Assignments 2, 4
4	Assembly language	Assignment 3
5	Context-free languages	Assignment 5
6	Top-down parsing	Exams
7	Bottom-up parsing	Assignment 5
8	Context-sensitive analysis	Assignment 6
9	Code generation	Assignments 7, 8
10	Linking and loading	Assignment 8, exams
11	Memory management	Exams

7.2 Assignment Slip Days/Extensions

Each student will have the ability to use up to two “slip days”—24-hour extensions—for assignments. These can be used to make late submissions on Marmoset count for marks.

These slip days are *in addition to* the 48-hour no-reason short-term absence provided to all undergraduate students by University policy: https://uwaterloo.ca/registrar/current-students/undergraduate-studen
Thus, you have a total of four extension days: two slip days and two short-term absence days.

If you submit late to Marmoset, this does not automatically use an extension. You must request an extension on our web form for the late submission to count:
https://student.cs.uwaterloo.ca/~cs241/cgi-bin/extension/request.cgi

If the extension is for a University short-term absence, you must use the above form *as well as* submitting the standard short-term absence form to the University.
Extensions can be requested *after* making your late Marmoset submission. However, you must request the extension within 72 hours of the assignment deadline.

These extensions are meant to handle cases not covered by the University’s normal illness and emergency policies, ranging from simple time management problems to minor illnesses. Budget them carefully; do not waste them. If you end up using up all your assignment extensions, you will not be granted any additional extensions.

If you are using assignment extensions, you must use a whole number of them on an assignment (Example: you cannot use 1.5 assignment extensions; you must use 2 if you are 36 hours late). Furthermore, the University short-term absence cannot be split into two separate 24-hour absences.

Assignment extensions can be used all together on one assignment, or split across different assignments.

You do not need a reason to use assignment extensions. You may use them as you please.

Once you have made an extension request, you cannot take it back. If you make a request in error, contact the ISA at cs241@uwaterloo.ca.

If you have a standard, University-approved reason to request an extension, such as an illness with a Verified of Condition form, you do not need to use slip days or the short-term absence. Because handling University-approved extensions is sometimes slow, you may use slip days as a "stopgap" in such situations and then request that they be refunded. If you do so, please mention it when requesting the extension.

7.3 Missed Assessments

You must notify the instructor of any severe, long-lasting problem that prevents you from completing an assessment. There will be no deferral, make-up, or extra credit if you miss an assessment. Under extenuating circumstances that are pre-approved within a week of the missed assessment, the instructor may shift the weight of the missed assessment to later assessments. To be considered for this option, rules instituted by the Math Faculty regarding Verification of Illness Forms will be applicable. The notification of an illness is not a guarantee that an accommodation will be made.

If a student requires an exemption from more than two assignments, they will either receive a DNW or INC depending on their performance in the rest of the course.

7.4 Marmoset downtime

If Marmoset fails to accept submissions for more than two of the six hours immediately prior to the deadline, or is down at the deadline, a 12-hour extension will be granted. For an extension to be granted, Marmoset must fail to accept submissions; failure or delay in displaying results is not grounds for extension. It is bad practice, and risky, to rely on Marmoset as your primary means of testing. The failure must be due to a problem with Marmoset, the University's network, or a widespread network failure. Your home connection is your own responsibility; if it fails, you have access to several University computer labs.

8 Submitting Assignments: Marmoset

Use Marmoset to submit and test your CS241 assignments.

There are three kinds of Marmoset tests:

- **Public tests:** These tests are "sanity checks": does your code compile and run. The results of these tests are visible as soon as the tests finish running by clicking the "view" link in the "detailed test results" column on the submission log. These tests are worth points, but there are usually few of them.

- **Release tests:** These tests are detailed tests of system behaviour. If you pass the public tests, you will be given the option to view the results of these tests. This option is on the ”detailed test results” page, at the bottom. However, releasing the results costs one “release token”. If you fail more than one release test, you will only be shown the first. **On most assignments, release tests are not worth points. In this case, they are to aid you in considering test cases and developing your own tests.**
- **Secret tests:** These tests are also detailed tests of system behaviour. On assignments problems in which release tests are not worth points, most secret tests are slight modifications of release tests (i.e., they are testing the same concepts as the release tests). These tests are completely hidden from you. You will not see any indication that they exist on Marmoset until three days after the assignment deadline, when they will become visible. The assignments specify how many points are secret tests; on assignments where release tests are not worth points, secret tests will be the majority.

This layout of tests is designed to encourage you to perform your own testing. For most students, it will not be possible to achieve a passing grade on the assignments without writing your own tests.

Some guidelines regarding the use of Marmoset:

a. If your submitted program does not compile or run successfully on its own, your submission will receive a result of “did not compile” and the detailed test results will contain something similar to the error message you get if you ran your program yourself. In this case, your submission will not be tested with any of the tests.

b. If your submitted program runs successfully on its own, it will be tested with all of the tests: public, release, and secret. However, you will only see the results of public tests initially.

c. If your submission fails any public test, the detailed test results will display an error message for that public test. In this case, you will not be able to view release test results for your submission until your submission passes the public tests.

d. If it passes all of the public tests, you will have the option to see information for the release tests. If you do so, you will use up one of your “release tokens” for that question to see the results of (some of) the release tests. Normally, for every assignment question, you have one release token. After using it, it will regenerate after one hour. Start your work early if you want to have more chances to see the results of the release tests. If the deadline will expire before your token regenerates, you can still submit, though you will not be able to tell how your submission did on the tests.

e. Marmoset automatically tests each submission with all of the release tests, in some order specified by the course staff. If your submission fails a release test and you use a token to see the results, you will only see the passing release tests prior to that test and the failing test in the detailed test results.

f. If you fail a release test, the information we are willing to give you for that test will be displayed by Marmoset. This may include things like the test input or the reason for the test failure. Some release tests may be *blind*, i.e., no additional information is provided. *Do not ask about or speculate about the test cases on Piazza.* The correct action when failing a release test is to re-examine your own test suite and redesign it to find the error in your code or your assumptions.

g. You can continue to submit and see the results of tests after the deadline has passed. This will not affect your mark unless you use slip days to adjust your deadline appropriately. It is a good idea to finish questions on which you ran out of time so that you can learn from the process, as most assignments build on the concepts of previous assignments.

h. Release tokens are provided as a courtesy to supplement your own testing. They are not something to which you are entitled. Release tokens can go away at any time, either as a result of Marmoset malfunctioning, or deliberately (for example, in response to widespread abuse). Loss of release tokens will not be considered grounds for assignment due date extensions.

i. Secret test details will only become visible 72 hours after the assignment deadline, to account for students who use slip days. Before this, secret tests are hidden from students completely; you will not see your score on the tests or even any indication at all that the tests exist. However, if a question has secret tests, this will be mentioned in the assignment specification.

9 Tutorials

The tutorials cover examples of the types of questions you would find on the midterm or final exam, which will aid in your understanding of the course material and help you prepare for these exams. Attendance is not mandatory but is strongly encouraged.

10 Office Hours

Some office hours are held online using MS Teams, and some are held in person. At the start of the term, make sure that you can connect to Teams. Contact CSCF if you encounter issues.

11 Discussion Forum

CS 241 will be using Piazza to make announcements and answer questions about course material and the assignments. **You are expected to check the forum regularly,** at least once per day. Important course information will appear in pinned posts. Any information that appears in a pinned post is considered to be disseminated and we will assume that you have read it.

11.1 Rules for using Piazza

a. When asking about a particular problem on an assignment, make sure to use the appropriate folder based on the assignment number.

b. Before posting a question, read all relevant existing posts. Your question might already have been answered.

c. You may post private questions which are only visible to instructors. Note that students can show up as anonymous to other students, but not to instructors.

d. **Do not post any questions asking for hints or help with failing Marmoset release test cases.** In order to pass these test cases you should be rereading the assignment question, consulting the reference material and creating your own test cases. The instructors and staff for CS241 will never give any hints for Marmoset release test cases, and students are *strictly forbidden from doing so* as well.