

CS247: Software Abstraction and Specification, Spring 2019

Course Times and Locations

Lectures:

MWF 10:30-11:20am
Location: MC 4020

Tutorials:

F 9:30-10:20am
Location: MC 4021

Course Personnel

Instructor:

Rob Hackman
DC 2128
r2hackma@uwaterloo.ca
Office hours:
Mondays 3:30-5pm
and by appointment
(subject to change).

Instructional Support Assistant (ISA):

Raghav Sethi
MC4065
cs247@uwaterloo.ca
Office hours: TBA, and by
appointment

Course Instructional Support Coordinator:

John Akinyemi
MC 4011
519-888-4567 x37627
john.akinyemi@uwaterloo.ca

Calendar Description

This course introduces systematic methods for designing, coding, testing, and documenting medium-sized programs. Major topics include abstraction, modularity, software modeling, object-oriented programming and design, generic programming, and testing and debugging.

Required Background

At the start of the course, students should be able to:

- Demonstrate knowledge learned in CS 137/138 and CS 241 by being able to program in C/C++ (structures, strings, procedural abstractions, pointers, addresses, recursion, classes, polymorphism, STL vector container).
- Explain container Abstract Data Types (lists, stacks, queues, trees) as they were used in CS 138 and CS 241.
- Write functional specifications (pre / post conditions) as they were used in CS 138.

Learning Objectives

At the end of the course, students should be able to:

- Design and implement data abstractions (ADTs, polymorphic objects, generic functions) in C++.
- Critique designs with regards to cohesion, coupling, generality, robustness, information hiding.
- Create software designs that are modular, general, robust, flexible.
- Express software designs and behaviour using basic UML models.
- Express the structure of an OO program as a class model.
- Express the state of a dynamic data structure as an object model.
- Express interactions between objects using a sequence diagram.
- Develop and test programs incrementally.
- Test and debug programs systematically.
- Use simple software tools effectively, including makefiles, `gdb`, version control.

Topic Overview

Module Design (12) hours

- ADT design, function and operator overloading
- Modules and interfaces
- Namespaces
- Interface specification

Object-Oriented Design and Programming (12) hours

- Composite objects
- OO design principles and patterns
- Multiple inheritance and mixins

Software Engineering and Tools (5) hours

- UML modelling
- Incremental development (`make`)
- Testing and debugging (`gdb`)
- Version control (e.g., `svn/git`)

Generic Programming (5) hours

- STL algorithms
- Iterators

Texts

These texts are optional, and available electronically; however, the material referenced in the readings is testable.

- Bruce Eckel, *Thinking in C++ Volume 1*, 2nd edition, Prentice Hall, 2000
- Bruce Eckel, Chuck Allison, *Thinking in C++ Volume 2*, Prentice Hall, 2003

Course Resources

Note that all important course announcements will be made to the newsgroup, so you are responsible for reading it on a daily basis. The course web page also contains some useful resource links.

Course web page: <http://www.student.cs.uwaterloo.ca/~cs247>

News group: piazza.com/uwaterloo.ca/spring2017/cs247

Course account: `cs247@uwaterloo.ca`

Lectures

Since the course notes/slides are deliberately incomplete, there will be some additional testable material presented in lectures verbally and/or on the board. Note that while the lecture material may seem simple, on the terms when such data was tracked, there was a direct correlation between lecture attendance and midterm grades. **Thus, your attendance in lecture is strongly recommended.**

It is also important that you pay attention during lecture, or at least not distract your classmates. Please turn off all devices that could ring, or at least set them to operate in silent mode.

Evaluation

Assignments (15%)

There will be three assignments, each consisting of short programming problems, design exercises, or software engineering tasks. *No group work is allowed on the assignments.* You are strongly advised to do the assignments as they are good preparation for the exams. Since part of the marks will come from automatically compiling and testing your code, it is *imperative* that your code compile and run in the student environment, `linux.student.cs.uwaterloo.ca`.

The assignment due dates are tentatively set to be:

- A1: Friday, May 24, 5:00:00 PM (5%)
- A2: Friday, June 14, 5:00:00 PM (5%)
- A3: Tuesday, July 30, 5:00:00 PM (5%)

Project (15%)

There will be a course project that can be done in teams of two or three students. Note that there is no advantage in terms of work load to doing the assignment on your own, or in a smaller group. You will also be required to demonstrate the final project deliverable to the marker since it is graphical in nature and thus cannot be marked automatically. Time slots for the demonstrations will be set aside in one of the computer labs, likely July 19-21. It is *imperative* that your code compile and run in the student environment, `linux.student.cs.uwaterloo.ca`.

Groups this term will be given the option to *propose their own project*. Students will still be given the option to build a project with a provided specification, but they may choose to propose their own project. There is no bonus for proposing your own project, and it means more work, however you get to create something entirely of your own. In order to propose your own project, you have to submit an extra deliverable early (Friday June 7th), which is worth 0% but is your project proposal. Submitting a proposal **does not guarantee** you will get to build your proposal as your project, as your proposal may be rejected. See the project proposal specification (when released) for more information.

The project is divided into two or three deliverables, depending on the option your group takes. The due dates are tentatively set to be:

- Friday June 7, 5:00PM (0%) **(Mandatory for groups proposing original project)**
- Friday, June 28, 5:00 PM (7%)
- Tuesday, July 16, 5:00 PM (8%)

Midterm (20%)

The midterm is worth 20% of your final grade. **Since there is a strong correlation between exam performance and lecture attendance, you are strongly encouraged to attend all lectures since the course notes/slides are not complete.**

Final Exam (50%)

The final exam is worth 50% of your final grade. You must achieve a passing grade on the exam component of the course (the midterm and final exam, weighted accordingly) in order to pass the course. Otherwise, your final grade will be the minimum of the computed final grade and the exam component grade.

Course Policies

Group Work Policy

No group work is permitted on assignments. Students may work on the course project in teams of size 1 or 2.

Collaboration Policy

Some degree of collaboration is beneficial: you can learn a lot from others; you can avoid getting stuck; and teaching someone else can be the best way to cement your own understanding of a difficult concept. That said, your understanding of the course material will be deeper and more ingrained if you solve problems on your own. Thus, we permit and encourage *discussion* but not outright *collaboration* (except where explicitly permitted, such as teamwork on the course project).

Specifically, you are allowed (and even encouraged) to discuss course concepts, assignments, and projects with other CS247 students *but only under the following restrictions*:

- No materials you bring to or take away from such meetings may be incorporated into your solution. In particular, you may not bring any of your code or solutions to the meeting.
- You may only bring away information in your long-term memory. You must destroy any materials that you and others create during the meeting. Then, you must spend 30 minutes without thinking about CS247 (e.g., watching a mindless TV show). After that, you can use whatever you still remember.
- You must write up anything you submit on your own. Your code (which includes tests and documentation), problem answers, etc. must represent your own understanding, as explained solely by you.
- You may not view other people's code or solutions. You may not share any of your own code (including, as always, tests and documentation) with others, including bringing it to a meeting with others. You may not allow anyone except the course staff access to your CS247 course directory, and you must restrict access to any Git repositories containing your CS247 work, or any other location where you keep CS247 solutions. Don't post large amounts of your code (more than about 5 lines) to a public forum (i.e., in a public Piazza post, to a Facebook group, to reddit, etc.)
- You must give credit where credit is due. Within each assignment/project submission, list everyone with whom you've had substantive discussions. Likewise, if you obtained a key idea from some other resource, such as a textbook or a website, then you should credit it (e.g., in comments at the top of your program submission).

- You may not view and/or use any substantive material or solutions from similar assignments this term or previous terms at UW or elsewhere, including anywhere on the Internet, transcribing solutions from any other source, etc.

We will use technological and other means to detect cheating.

Late Policy

Up to one assignment and one project deliverable may be handed in, without penalty, up to 48 hours late. This late policy is meant to cover all minor reasons for missing a due date (e.g., minor medical illness, conflict with another assignment, etc.). There is no special process for invoking the late policy—simply hand in your assignment late.

Anything handed in after the late-policy due date without a pre-approved extension (see below) will receive a grade of 0. If more than one assignment/project deliverable is handed in after the late-policy due date, one of the late submissions will receive a grade of 0.

If you have a serious (multiple-day) illness or absence, then you need to consult with the instructor in advance to make alternative arrangements for fulfilling the assignment deliverables of the course.

Assignment Submission and Pickup

All assignments in this course will be submitted electronically. **Do not wait until the last minute to submit**, just in case there are problems with your internet connection, Marmoset or your submission. It is better to submit early and often!

Marked assignments will be handed back in tutorials or returned via MarkUs/CrowdMark. After that, arrangements must be made to collect them from the course ISA. Anything left unclaimed (including midterms) at the end of the term will be shredded.

Re-marking Policy

Assignments: email the ISA (cs247@uwaterloo.ca) with the subject "CS247 A<N> Re-mark Request", replacing <N> as appropriate. Clearly state the questions you want re-marked and include any supporting evidence for your case. Requests that include code changes to fix failing test cases will be ignored since you are expected to test your code thoroughly in advance of the submission.

Midterm: Email cs247@uwaterloo.ca with the subject "CS247 Midterm Remark Request", clearly stating the questions you want re-marked. Include any supporting evidence for your case.

Deadline for all re-mark requests: You have two weeks after handback to deliver the request to the ISA. If for some (valid) reason you will be unable to make your request within this time period, you must make alternate arrangements with the course instructor. All requests will be processed after the deadline to ensure fairness and consistency in marking.

Note: We will examine your entire assignment/midterm when remarking it. It is possible that you will receive a lower mark than your current mark.

Missing an Examination

The only valid excuse for missing the midterm is illness substantiated by a doctor's note. In such a case, the weight of the midterm is added to the weight of the final exam. The only valid excuse for missing the final exam is illness substantiated by a doctor's note. In both cases, you must present a valid doctor's note to the course Instructional Support Coordinator (John Akinyem, john.akinyemi@uwaterloo.ca) as soon as possible—preferably before the exam.

In the case of a missed final exam with suitable documentation, you will receive an INC in the course, and you will have to make up the incomplete by writing the CS246 final exam in the F17 term.

Intellectual Property.

Students should be aware that this course contains the intellectual property of their instructor, TA, and/or the University of Waterloo. Intellectual property includes items such as:

- Lecture content, spoken and written (and any audio/video recording thereof);
- Lecture handouts, presentations, and other materials prepared for the course (e.g., PowerPoint slides);

- Questions or solution sets from various types of assessments (e.g., assignments, quizzes, tests, final exams); and
- Work protected by copyright (e.g., any work authored by the instructor or TA or used by the instructor or TA with permission of the copyright owner).

Course materials and the intellectual property contained therein, are used to enhance a student's educational experience. However, sharing this intellectual property without the intellectual property owner's permission is a violation of intellectual property rights. For this reason, it is necessary to ask the instructor, TA and/or the University of Waterloo for permission before uploading and sharing the intellectual property of others online (e.g., to an online repository).

Permission from an instructor, TA or the University is also necessary before sharing the intellectual property of others from completed courses with students taking the same/similar courses in subsequent terms/years. In many cases, instructors might be happy to allow distribution of certain materials. However, doing so without expressed permission is considered a violation of intellectual property rights.

Please alert the instructor if you become aware of intellectual property belonging to others (past or present) circulating, either through the student body or online. The intellectual property rights owner deserves to know (and may have already given their consent).

Students with disabilities

[AccessAbility Services](#), located in Needles Hall, Room 1132, 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 the office at the beginning of each academic term.

Integrity and Discipline

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 <http://www.uwaterloo.ca/academicintegrity/> 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, <http://www.adm.uwaterloo.ca/infosec/Policies/policy70.htm>. 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 <http://www.uwaterloo.ca/academicintegrity> to avoid committing an academic offence, and to take responsibility for his/her actions.

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, <http://www.adm.uwaterloo.ca/infosec/Policies/policy71.htm>. For typical penalties check Guidelines for the Assessment of Penalties, <http://secretariat.uwaterloo.ca/guidelines/penaltyguidelines.htm>.

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) <http://www.adm.uwaterloo.ca/infosec/Policies/policy72.htm>.