

SOFTWARE ENGINEERING PRINCIPLES SPRING 2021

CS 247

Published Jun 16, 2021

CLASS SCHEDULE

Section	Location	Time	Instructor(s)
CS 247 041 [LEC]	No Location or Online		Werner Dietl werner.dietl@uwaterloo.ca
CS 247 241 [TST]			Scott King sfking@uwaterloo.ca
This table is generated automatically			

INSTRUCTOR / TA INFORMATION

INSTRUCTOR

- Werner Dietl
<https://ece.uwaterloo.ca/~wdietl/contact> (<https://ece.uwaterloo.ca/~wdietl/contact>)
Office hours: By appointment only

INSTRUCTIONAL SUPPORT ASSISTANT (ISA)

- SK Sadman Sayeed cs247@uwaterloo.ca (<mailto:cs247@uwaterloo.ca>)
- **Office hours: By appointment only**

TEACHING ASSISTANTS

- Sayantan Gupta s343gupt@uwaterloo.ca (<mailto:s343gupt@uwaterloo.ca>)
- Jubilee Imhanzenobe jimhanze@uwaterloo.ca (<mailto:jimhanze@uwaterloo.ca>)
- Samapti Kundu s25kundu@uwaterloo.ca (<mailto:s25kundu@uwaterloo.ca>)
- **Office hours: By appointment only**

COURSE INSTRUCTIONAL SUPPORT COORDINATOR

- Scott Freeman King sfking@uwaterloo.ca (<mailto:sfking@uwaterloo.ca>)

COURSE DESCRIPTION

Calendar Description for CS 247

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, testing and debugging.

Prereq: CS 241; Software Engineering students only. Antireq: CS 246, MSCI 342, SYDE 322

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.

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 mix-ins

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

COURSE RESOURCES

All lecture and tutorial modules will be made available on Learn and the course web page, and important course announcements will be made directly through Learn or the course webpage as well. You are responsible for checking these resources regularly. The course web page also contains some useful resource links.

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

Course account: cs247@uwaterloo.ca (<mailto:cs247@uwaterloo.ca>)

LECTURES

This version of CS247 is customized to be delivered remotely through pre-recorded lecture videos in response to the COVID-19 lockdowns. The course notes/slides are deliberately made incomplete, and additional testable material will be presented in the lecture video clips verbally. 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 exam grades. Thus, your weekly commitment to completing the lecture modules and tutorial materials are strongly recommended.

LEARNING OUTCOMES

By the end of this 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.

TENTATIVE COURSE SCHEDULE

Week	Lecture Topics	Tutorial Topics
1	Intro to OOD in C++, Abstract Data Types	No Tutorial
2	Entity v.s. Value, ADT Special Functions	Review of C++
3	Modules, Interface Specifications	ADT Example - Linked List
4	Exceptions, RepInvariant	Makefiles, Conditional Compilation

5	Project Sprint 1 (No Lectures)	Project Sprint 1 (No Tutorial)
6	UML Diagrams, Design Patterns Part 1	UML Tools, A2 Support
7	Design Patterns Part 2, Refactoring	More on Exceptions, Smart Pointers
8	OOD Principles, Design Patterns Part 3	Decorator, Non-Virtual Interface
9	Design Patterns Part 4, Part 5	Factory and Strategy, A3 Support
10	Project Sprint 2 (No Lectures)	Project Sprint 2 (No Tutorial)
11	STL Containers and Algorithms	Type casting in C++, More on STL
12	Lambda Functions, Templates	Project Sprint 3 (No Tutorial)
13	Final Review	No Tutorial
14	Final Exam	

Lecture and tutorial topics are subject to adjustment based on course progress.

TEXTS / MATERIALS

Title / Name	Notes / Comments	Required
Bjarne Stroustrup, A Tour of C++	Texts are optional, and available electronically; however, the material referenced in the readings is testable	No
Bruce Eckel, Thinking in C++ Volume 1, 2nd edition, Prentice Hall, 2000	Texts are optional, and available electronically; however, the material referenced in the readings is testable	No
Bruce Eckel, Chuck Allison, Thinking in C++ Volume 2, Prentice Hall, 2003	Texts are optional, and available electronically; however, the material referenced in the readings is testable	No

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).

STUDENT ASSESSMENT

Component	Value
Assignment 1	12
Assignment 2	12
Assignment 3	12
Project Part 1	12
Project Part 2	12
Project Part 3	15
Final Assessment	25

ASSIGNMENTS (36%)

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 course project and the final exam. 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: Fr 28.05.2021 @ 23:59
- A2: Fr 18.06.2021 @ 23:59
- A3: Fr 09.07.2021 @ 23:59

PROJECT (39%)

The project component of the course will be run in teams of two or three students. With the remote delivery arrangement, students are required to collaborate via GitHub. More grouping details will be provided in the Week 1 lectures. It is *imperative* that your code compile and run in the student environment, `linux.student.cs.uwaterloo.ca`.

The project is divided into three phases, each with a deliverable due on the following tentative dates:

- P1: Fr 11.06.2021 @ 23:59
- P2: Fr 16.07.2021 @ 23:59
- P3: Tu 03.08.2021 @ 23:59

MIDTERM (0%)

No midterm exam for the remote delivery version of the course.

FINAL EXAM (25%)

The final exam is worth 25% of your final grade. You must achieve a passing grade on the exam component of the course 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 2 or 3.

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.

See note on the use of MOSS, Marmoset, MarkUs, and Turnitin below.

LATE POLICY

Anything handed in after the due date without a pre-approved extension (see below) will receive a grade of 0.

If you have an illness, conflict, or absence, then you need to consult with the Instructional Support Coordinator (Scott Freeman King, sfking@uwaterloo.ca (mailto:sfking@uwaterloo.ca)) as soon as possible in advance to make alternative arrangements for fulfilling the assignment/project 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 at the end of the term will be shredded.

RE-MARKING POLICY

Assignments: email the ISA (cs247@uwaterloo.ca (mailto:cs247@uwaterloo.ca)) with the subject "CS247 A Re-mark Request", replacing 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.

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 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 final exam is illness substantiated by a doctor's note. You must present a valid doctor's note to the course Instructional Support Coordinator (Scott Freeman King, sfking@uwaterloo.ca (mailto:sfking@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 final exam of CS247 in the subsequent term of offering.

ASSIGNMENT SCREENING

Text matching software (Turnitin) will be used to screen assignments in this course. This is being done to verify that use of all material and sources in assignments is documented. In the first week of the term, details will be provided about the arrangements for the use of Turnitin and alternatives in this course. See Administrative Policy below for more information and links.

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

Discipline cases involving any automated marking system such as Marmoset or MarkUs include, but are not limited to, printing or returning values in order to match expected test results rather than making an actual reasonable attempt to solve the problem as required in the assignment question specification.

ADMINISTRATIVE POLICY

Mental Health Support: The Faculty of Math encourages students to seek out mental health support if needed.

On-campus Resources:

- Campus Wellness <https://uwaterloo.ca/campus-wellness/> (https://uwaterloo.ca/campus-wellness/)
- Counselling Services: [counselling.services@uwaterloo.ca/](mailto:counselling.services@uwaterloo.ca) (mailto:counselling.services@uwaterloo.ca/) 519-888-4567 ext 32655
- MATES: one-to-one peer support program offered by Waterloo Undergraduate Student Association (WUSA) and Counselling Services: mates@wusa.ca
- Health Services: located across the creek from the Student Life Centre, 519-888-4096.

Off-campus Resources:

- Good2Talk (24/7): Free confidential help line for post-secondary students. Phone: 1-866-925-5454 (Ontario and Nova Scotia only)
- Here 24/7: Mental Health and Crisis Service Team. Phone: 1-844-437-3247 (Waterloo Region only)
- OK2BME: set of support services for lesbian, gay, bisexual, transgender or questioning teens. Phone: 519-884-0000 extension 213 (Waterloo Region only)

- EMPOWER ME 1-833-628-5589 for Cdn./USA other countries see:
http://studentcare.ca/rte/en/IHaveAPlan_WUSA_EmpowerMe_EmpowerMe
(http://studentcare.ca/rte/en/IHaveAPlan_WUSA_EmpowerMe_EmpowerMe)
 - EMPOWER ME in China:
China North 108007142831
China South 108001402851

Diversity: It is our intent that students from all diverse backgrounds and perspectives be well served by this course, and that students' learning needs be addressed both in and out of class. We recognize the immense value of the diversity in identities, perspectives, and contributions that students bring, and the benefit it has on our educational environment. Your suggestions are encouraged and appreciated. Please let us know ways to improve the effectiveness of the course for you personally or for other students or student groups. In particular:

- We will gladly honour your request to address you by an alternate/preferred name or gender pronoun. Please advise us of this preference early in the term so we may make appropriate changes to our records.
- We will honour your religious holidays and celebrations. Please inform of us these at the start of the course.
- We will follow AccessAbility Services guidelines and protocols on how to best support students with different learning needs.

UNIVERSITY POLICY

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](https://uwaterloo.ca/academic-integrity/) (<https://uwaterloo.ca/academic-integrity/>) for more information.]

Grievance: A student who believes that a decision affecting some aspect of their university life has been unfair or unreasonable may have grounds for initiating a grievance. Read [Policy 70, Student Petitions and Grievances, Section 4](https://uwaterloo.ca/secretariat/policies-procedures-guidelines/policy-70) (<https://uwaterloo.ca/secretariat/policies-procedures-guidelines/policy-70>) . 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 their actions. [Check [the Office of Academic Integrity](https://uwaterloo.ca/academic-integrity/) (<https://uwaterloo.ca/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](https://uwaterloo.ca/secretariat/policies-procedures-guidelines/policy-71) (<https://uwaterloo.ca/secretariat/policies-procedures-guidelines/policy-71>) . For typical penalties, check [Guidelines for the Assessment of Penalties](https://uwaterloo.ca/secretariat/guidelines/guidelines-assessment-penalties) (<https://uwaterloo.ca/secretariat/guidelines/guidelines-assessment-penalties>) .

Appeals: A decision made or penalty imposed under [Policy 70, Student Petitions and Grievances](https://uwaterloo.ca/secretariat/policies-procedures-guidelines/policy-70) (<https://uwaterloo.ca/secretariat/policies-procedures-guidelines/policy-70>) (other than a petition) or [Policy 71, Student Discipline](https://uwaterloo.ca/secretariat/policies-procedures-guidelines/policy-71) (<https://uwaterloo.ca/secretariat/policies-procedures-guidelines/policy-71>)

may be appealed if there is a ground. A student who believes they have a ground for an appeal should refer to [Policy 72, Student Appeals](https://uwaterloo.ca/secretariat/policies-procedures-guidelines/policy-72) (https://uwaterloo.ca/secretariat/policies-procedures-guidelines/policy-72) .

Note for students with disabilities: [AccessAbility Services](https://uwaterloo.ca/disability-services/) (https://uwaterloo.ca/disability-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.