Azadeh Farzan

Teaching / CSC410

Software Verification

(Fall 2023, under construction)

General Information

- The course forum (access code will be provided on Quercus)
- Grades (of non-Markus items) are posted on Quercus.
- For textbooks and references see here.
- Markus (becomes active on September 15)

Instructor Contact

Azadeh Farzan: azadeh *at* cs dot toronto dot edu

(Read the communication guidelines before sending emails)

TA Team

Nick Feng
Logan Murphy
Avery Laird

Note: Do not email your TAs. It is not in their contract to answer emails. They are contracted to communicate with you on the course forum and during office hours and tutorials.

Class Time/Location

- Tuesdays 3-5pm in MP 137
- Wednesdays 3-5pm in MP 134
- Joint Tutorials: Fridays 3-5pm in MP 200 (typically the tutorial slot, but there will be exceptions)

Office Hours

Fridays 4:10-5, in class, if nothing else is anounced.

Grading Scheme

See here.

The Goal of the Course and Learning Objectives

In this course, we aim to expose you to a variety of techniques used to ensure software reliability. These techniques range proving programs correct using theorem provers (involves a lot of contribution from the user) to testing (fully automatic). We do this by:

- covering theoretical foundations of various verification and testing techniques, and
- putting your theoretical knowledge to practical use by using a variety of tools to solve small scale (classroom-sized) problems. This gives you a chance to practice the theory, and to get to know these tools that may be useful to you in your future careers.

This process will involve learning to use tools in a relatively short period of time (12 weeks). It may seem a bit overwhelming for some. Given the fact that most of you are about to graduate and start your new careers, this situation simulates what you will face very soon in your work environment. Computer science is a very dynamic field. New programming languages, environments, tools and techniques find their way to the market every day. As a computer scientist, you need to have to skill to catch up with the new technology quickly (in contrast to knowing one or two of these very well from your school days). The majority of students catch up and do extremely well. The small minority (about or under 20% of students) may give up trying because they think they cannot handle it. You can!

Also, in case you are taking this course as an elective, and do not like the general idea of it, then please reconsider. There are other elective courses. This course involves theory and proofs. It can be challenging for those who have a not-so-perfect background in logic and discreet math from the courses in year 1 and 2.

Communication Guidelines

Read before sending an email. *Any email that does not meet the following specification will be ignored and left unanswered*.

- Email your instructor for *personal matters*, or asking for *special considerations*. These are generally considered to be rare occasions.

- All technical (non-personal) questions should be posted on the course forum (on Piazza), so that everyone can benefit from the question and the answer.
- Always sign your full name (as it appears on university records) at the bottom of your emails.
- Always send your email from your university account.
- Always include CSC410 in the subject line so that the email goes to the right folder.
- Include your Markus handle in any Markus-related enquiries.
- Your questions will generally be answered within two working days (excluding weekends), although this may take longer during busy times (such as around assignment due dates). So, do not rely on getting a same-day answer.

Providing Feedback

Rather than wait until the official course evaluations at the end of the term, by which point it is too late to make a difference for your experience, please feel free to get in touch with your instructor at any point during the term with any suggestion about any aspect of the course. In particular, do not hesitate to let us know if there are aspects of the course that you particularly like, so that we can keep them that way, or if there are specific aspects that are challenging, so that we can provide more help before it is too late. This naturally does not apply to the technical content of the course (the syllabus) which is fixed. The course is an elective and designed to appeal to those who are interested in this particular topic.

Azadeh Farzan

Teaching / CSC410 / Announcements

Announcements

This page will be a continous feed in which all the upcoming information about the course will be posted. Quercus does not permit good formatting for longer content, and hence, I will use this space when the announcments are not short. It is therefore your responsibility to check it frequently and keep yourself up-to-date with the relevant course news.

For really important announcements (for example, as a class-wide extension on a due date or any anomaly in the class schedule), emails will be sent on Quercus.

The most recent announcmenet will always be at the top.

2023/08/20

Class Content

The name of this course is old and legacy. The actual content of the course is about different ways that programmers would reason about the reliability of the code they write. As such, the course webpage lists a title with "Verification" in it which is what this course is really about. Our first lecture will give an overview of the course content. If you are on the fence, attend that one and then decide if this course is for you.

Class Format

The class is somewhat inverted. You will get assigned readings and videos in advance of each class. Your role is to consume the material, and then show up for the class during which we focus on practical exercises to cement your knowledge of the material. The class will be meaningless to you if you have not at the very least watched the video in advance. If you are not the type of person who will keep up with material in advance of the class, do not take this class.

The time we have face-to-face is short. I will not use to repeat basic definitions of cocepts that already appear in videos. Instead, we try to make sure that we understand the material by applying the ideas in class. Sometimes, I may present a different angle (from the one presented in the video) on the material.

Class Organization

There are two sections in this course, and the two sections have a common 2-hour tutorial slot on Fridays. We will use this 2-hour slot for different roles through out the term. This could include content presented by me (your instructor), tutorial material presented by your TAs, office hours held by your TAs, in class Exams, and anything else for which we may want the two sections to come together.

Each section has its own two-hour class on a different day. We will do our best to keep these two sections in synch. But, naturally, things may not align 100%, because the class will flow with student questions and sometimes one section ends up being more active than another.

First Class

Our first opportunity to meet is our (joint) Friday slot on September 8. Since we can definitely not start the class with a tutorial, we will hold an introductory lecture for both sections combined on this day. I will introduce the course, and answer any questions about the course content and organization. Once done with that (about one hour of time), we turn our attention to the technical content in the second hour

We will then continue with the technical content on September 12/13 (depending on your section), when we meet for the second time. The video material for these two classes is already online. The plan for our first class is the following:

- Install Dafny by following the instructions here. You need to have Dafny on your computer to participate in the lecture. You can use Dafny in VSCode or run it commandline.
- Watch the videos posted for Weeks (1) and (2) here.
- We will use Dafny together to prove that the square root of 2 is not a rational number. Feel free to look up a proof for this online in advance of the class, if you have never seen one before.
- We will then switch to proving iterative programs correct. Technically, you should not need anything more than what you learned in CSC236. But, the videos will remind you of anything you have forgotten and demonstrate how Dafny is used for this.

If you come to the class unprepared, you will not get much out of it. I will not use precious class time to repeat what is already mentioned in the videos.

The idea of using this first tutorial as a full two-hour (joint between the two sections) lecture is to get ahead of the curve with our course. You will end up having precisely 12 of these two-hour lectures. This first additional one lets us not miss the tutorial slot, and we will not have any lectures during the last week of classes because the two sections will come together on Friday December 2nd for the second in-class exam as our last act. Hence, we will end up having done precisely 12 lectures together by then.

First Check-in (Logic) Quiz

We heavily rely on the basic knowledge that you learned in CSC165 and CSC236. You may want to consult your own old texts or use the reading that I have recommended here (under week (0)) to remind yourself of the material. Either way, consider this as one of the most important things you do for this course. Without fluency in basic logic and discrete math, you will have a very hard time succeeding in this course.

To make sure that everyone does this in a timely manner, we will have a short quiz (a few multiple choice questions) to cover this material on September 15 (synchronously) during our tutorial slot on Friday. It will be a quizz on Quercus. You can get it done from anywhere. It will require about 15 minutes, but we will dedicate 20 minutes to it to make it a relaxed experience. The questions will be very very elementary logic questions. Consider that date as your deadline to remind yourself about your knowledge of logic. The grade for this is not part of your original grading scheme. Together with participation, it can count as a bonus at the end of the term. So, the better you do for this, the more of a bump you will receive in the form of a bonus at the end of the term. Do your best!

Is this the right course for you?

This is an elective course. It is desgined to broaden your knowledge of CS in a way that is useful for any career path with an undergraduate CS degree. Yet, it is rather theoretical. It involves logic, proofs, and automata (i.e. the content of CSC165 and CSC236). It heavily relies on competency in the basic knowledge in these areas. Consequently, it may not appeal to everyone and it may prove to be difficult for those who have problems with this basic knowledge.

I have prepared and released the entire material for the first 3-4 weeks of the course upfront. This means you have access to all videos and lecture material. The first assignment which is worth 15% of this course’s mark will also be released on the day of the first class (not our first meet which is a tutorial slot).

You can effectively fastforward through this class using the material provided, to see if it is a good fit for you. This way, you do not have to waste 3-4 weeks and then reluctantly decide to drop this course.

Forum

The forum has been created for you to have community. You are strongly encouraged to answer each other questions, share tips, and pointers. It will be monitored by your TAs so that questions that cannot be answered by other students are answered by them.

In a normal year, I might have partially monitored the forum myself from time to time. This year, the course is for the first time cross-listed for graduate students, and I have extra responsibilities to get that aspect of the work off-the-ground. Therefore, I will not be present there. But, your TAs keep me informed.

We expect respect for students and TAs. We want a warm and friendly place that people can come to share technical ideas. If anyone digresses from this ideal, we will not hesitate in removing them from the forum.

Code From Videos

Why is the code used in the videos of the first three weeks of lectures not released online? It is a conscious decision to have you (as a student in training) follow along with the lecture and type things in your own Dafny editor to learn. We will have other examples during class and the codes for those examples will be released online after the lectures.

Participation

I highly value a **community** for the class. I would love to see the Piazza forum turn into such a community rather than merely a place that students ask questiosn and TAs answer. Therefore, I have dedicated a bonus participation mark for the course which is granted through activity on the forum (of any kind other than asking questions directly related to assignments) **and/or** lively class participation.

So, you can gain your participation mark by actively participating in class by asking questions or answering my questions, and/or by having an active online presence on Piazza by answering other students' questions or sharing interesting facts/ideas about the class material. Asking direct questions about assignments/exams on any platform (though highly encouraged) does not count towards your *participation*.

If you are an active member of our community, at the end of the term you can reach out to me personally to claim a %1-%2 bump in your grade for being a good citizen.

We had a wonderfully active class in Fall 2022. I would love to see the Fall 2023 class beat last year’s class out of the park!

Welcome Email

A welcome email was sent to the class on September 1st. If you have not received this email, you should fix your Quercus settings. I will use the same channel to send all future emails whenever there is important information to be communicated with the class. The info will appear on this page as well. Make sure you are receiving the emails and check this page frequently.

All private (to the class) information, such as a Piazza participation code, will be on Quercus as an announcement. The only critical part of the welcome email is the link that gets you to this page. If you are already here, you have not missed anything!

Azadeh Farzan

Teaching / CSC410 / Assignments

Assignments

Submit on Markus.

All assignments are to be completed by groups of **up to** 4 students. Only one assignment needs to be submitted by each group. The assignments are simple but involve a degree of learning to work with a new tool. The team format is meant to provide you with a local support group to quickly learn the necessities together. You do not have to stick with the same group throughout the term. Feel free to switch groups. You can form groups with people from a different section of the course. The only limitation is that graduate students can only be in a group with other graduate students.

Everything is due at 11:59:59 pm on the day of its due date.

Below is an estimate for assignment release and due times. The times will be adjusted during the semester to ensure that an assignment is released when the topic is fully covered in class, and this can vary slightly by the number of questions asked during the class and the pace at which the class is comfortable to make progress.

- Assignment (1): (15%) Program Correctness, dafny files
  - Release **9/14**, Due **9/28**
- Assignment (2): (15%)
  - Release 10/4, Due 10/18
- Assignment (3): (15%)
  - Release 10/25, Due 11/16
- Assignment (4): (15%)
  - Release 11/16, Due 12/6

Bold dates are fixed. Other dates are approximate but will not change by more than a few days.

Azadeh Farzan

Teaching / CSC410 / Exams

Exams

Since the course assignments and the project are all group work and mostly focus on practical skills, there will be 2 exams, worth 20% each to cover the material from the more theoretical aspects of this course and allow for some individual distinction. We will go with 2 exams instead of one, mostly to avoid having one stressful event with too much weight assigned to it.

Exam Dates:

- Exam 1: Friday 10/20 (length: 110 minutes, weight: 20%)
- Exam 2: Friday 12/1 (length: 110 minutes, weight: 20%)

The exams will be given during the joint tutorial slots. There will be no make-up exams. If you cannot make it to a test due to a (documented) illness or personal emergency, then

1. Contact your instructor in advance of the test time.
2. After the test, bring your documentation (which is approved and stamped by the registrar) for why you missed the test.
3. Under legitimate circumstances, the weight of the remaining exam will be increased to cover the grade of the missed exam.
4. You cannot miss both exams. If you miss one, you must show up to the other one, because group work cannot be reweighed to cover individual work.

Azadeh Farzan

Teaching / CSC410 / Grading

Grade Distribution

Summary

- 4 Assignments, worth 60% in total
- 2 Exams (in class), worth 40% in total
- Bonus: up to 5% of bonus marks in participation and the initial quizz

Grading for Graduate Students

On each assignment and each exam, there will be problem(s) that are clearly marked as being for graduate students only. These will be slightly more challenging problems, and will only count towards the grade of a graduate student taking this course.

Policy on special considerations

If you are unable to complete an assignment due to major illness or other circumstances completely outside of your control, please contact your instructor immediately in order to receive special consideration. Note that special consideration will be considered on an individual basis and will not be given automatically—in other words, you risk getting a mark of zero for missed work unless you contact your instructor promptly, and before the due date of an assignment. Unless you are physically incapable of doing so due to extreme illness, contact your instructor before the due date and ask for an extension.

Policy on remarking requests

All remarking requests must be received within two weeks of the date when the assignment was returned. It is your responsibility to check for your posted grade or your returned assignment (electronically, during lecture or tutorial or office hours if you miss the distribution date).

- If there is a simple addition mistake, show it to your instructor (not your TA).
- There is no reconsideration for auto-graded code assignments. Yes, it is unfair to lose the entire assignment grade due to a small mistake. But, you are a computer scientists now; small coding mistakes can lead to people losing their lives. Learn not to make (many of) them!

- If your request is about reevaluating your answers for a higher grade, email your instructor with the details of your request. Note that your mark may decrease if we see that you have been incorrectly awarded too high a mark. It will take us a while (average two weeks) to identify the marker, have them reconsider and assign a new mark. So, be patient with these requests.

Azadeh Farzan

Teaching / CSC410 / Lectures

Topics

You will find a list of topics and an order in which we will explore them below. Lectures will follow this order.

- Introduction to the course
- Program proofs: The Inductive Assertion Method and Hoare Logic
- Program Analysis
- Decision Procedures: Satisfiability
- Symbolic Testing: Reachability
- Introduction to Temporal Logics (LTL and CTL)
- Model Checking

Assigned Reading

Here is a week by week guide of how you should organize your reading (of the text material) with the course. This reading is essential to stay on top of the course material, in addition to the videos for the lectures and the slides. If you do not do this reading, you risk your grade.

- Week (0): review Chapters 1-4 of this book for basic logic and induction. You have learned the material in CSC 168 and CSC236. This is just to remind yourself of the basic material that is essential for this course. If you do not know this material, you will struggle in this course.
- Weeks (1,2): Read chapter 5 of this book to prepare for weeks 1 andn 2.
  - The emphasis should be on sub-chapters: 5.2 and 5.3.
  - The main learning objective: understand the difference between *an invariant* and *an inductive invariant*.
- Week (3): Dataflow Anlysis: there are several books that cover this, but I do not know of one that is freely available. These notes by a leading researcher in static analysis is arguably as good as a book chapter. Our focus is on Chapters 4 and 5.

Lectures Material

Below you will find our (tentative) lecture schedule for the term. Lecture slides and other material related to lectures are posted here after each lecture takes place. Note that due to inverted nature of the class, the week structure is a bit more fluid compared to the traditional class, because the time spent is guided by student questions and comments in class. We will do our best to keep the two sections in synch, and stay close to the schedule below:

- Weeks (0):
  - Introduction to the course and program correctness (video slides for both Introduction parts)
  - Examples of mathematical proofs in Dafny
    - irrationality of square root of 2
- Week (1):
  - Program Proofs: Iterative and Recursive (video slides)
  - An alternative tiny introduction to program correctness in class
    - Max
    - Bubble Sort
    - Simple Termination
- Week (2):
  - Examples of recursive program correctness in class
    - Binary Search
    - Stooge Sort
    - Tricky Termination
- Week (3): Dataflow Analyses
- Week (4): Dataflow Analyses
- Week (5): Decision Procedurs (SAT)
- Week (6): Decision Procedures (SMT)
  - Exam 1
- Week (7): Symbolic Reachability
- Week (8): Model Checking and Linear Temporal Logic
- Fall Reading Week
- Week (9): Computation Tree Logic
- Week (10): Model Checking Algorithms
- Week (11): Model Checking and course wrap up
- Week (12): Exam2