Formal Methods in Software Engineering (CISC/CMPE 422, CISC835):

Syllabus

Juergen Dingel
School of Computing
Queen's University

August 2018

1 Course and Lecture Information

Course term: Fall 2018

Course web page: `www.cs.queensu.ca/~cisc422`

Time and locations of lectures:

Time: Tuesday, 11:30am; Wednesday, 1:30pm; and Friday, 12:30pm

Location: Kinesiology 100

2 Teaching Staff Information

Instructor: Juergen Dingel

Email: dingel@cs.queensu.ca

Office hours: see course web page

Web page: `www.cs.queensu.ca/~dingel`

Teaching assistants:

See course web page

3 Intended Student Learning Outcomes

To complete this course students will demonstrate their ability to

1. use and explain formal specification languages based on, e.g., propositional logic, predicate logic, relational calculus, and finite state machines,
2. use and explain notations and techniques to define the semantics of a language precisely,
3. use and explain analysis techniques for formal specification languages such as theorem proving, satisfiability checking, and exhaustive state space exploration together with their capabilities and limitations,

4. use and explain tools supporting formal specification languages together with their capabilities and limitations,
5. design, construct, and analyze small formal specifications, and
6. explain the advantages and disadvantages of formal specification languages and tools, and
7. explain the role and potential uses of formal methods for different software development activities.

Students in CISC 835 will additionally

1. demonstrate potential for independent research and development via a course project,
2. describe the process and results of their independent project in a written report, and
3. summarize the motivation and key results of their independent project in an oral presentation.

4 Course Outline

Modern software development inevitably requires the design and analysis of a number of different artifacts. Formal methods allow the mathematically precise formulation of some of these artifacts. For instance, formulas in predicate logic capture operational requirements, state machines describe the behaviour of code fragments and protocols, and object models capture static designs. The advantage of using these formal notations is that they typically improve the overall quality of the artifacts by removing ambiguities and imprecisions, and enabling automatic analyses that establish desirable properties or uncover undesirable properties. Consequently, the use of formal methods is indicated in domains in which the software has to meet very high quality standards and failure cannot be tolerated such as air-traffic control. Moreover, the abstraction and automation capabilities of some formal techniques present a powerful weapon against the ever-increasing complexity of software. Indeed, in Model-Driven Development (MDD), a development methodology advocated by, for instance, the OMG and IBM, formal models of the software and its requirements form the primary artifacts from which the code is automatically generated.

CISC422 is an introduction to the use of formal methods for the specification, design, and automatic analysis of software systems. The course will present a variety of specification notations (propositional and predicate logic, Z, Alloy, UML/OCL, temporal logic), and discuss corresponding analysis techniques (theorem proving, constraint checking, animation, model checking) using existing commercial and research tools (Jape, Z/Eves, Alloy, USE, SMV). The course is most suited for students with a general background in computer science or electrical engineering and in interest in the theory and practise of software development.

5 Textbooks and Readings

The course has a courseware package that is available at the bookstore. Unless explicitly stated otherwise, this package is required reading. There is no required text book. However, parts of the following book might constitute beneficial supplemental reading:

> M. Huth and M. Ryan. *Logic in Computer Science: Modeling and Reasoning about Systems.* Cambridge University Press. 2nd Edition. 2004.

Additional material such as sample specifications used in class will be available from the course web page.

6 Late Policy

There is no late policy for assignment submission. Submissions after the stated due date of an assignment will not be accepted.

7 Grading Scheme and Grading Method

The grade of students in CISC/CMPE 422 will be computed from their grades in the following six deliverables using the indicated weights:

- 4 assignments (35%)
- 1 mid-term exam (15%)
- 1 final exam (50%)

For students in CISC 835, the following scheme will be used:

- 4 assignments (25%)
- 1 mid-term exam (15%)
- 1 final exam (40%)
- course project (20%)
  - proposal (2 pages) and final report (5-10 pages): 6%
  - presentation (20 mins): 7%
  - project evaluation, i.e., overall difficulty and quality of work carried out: 7%

All components of this course will receive numerical percentage marks. The final grade you receive for the course will be derived by converting your numerical course average to a letter grade according to Queen's Official Grade Conversion Scale:

Grade	Numerical Course Average (Range)
A+	90–100
A	85–89
A−	80–84
B+	77–79
B	73–76
B−	70–72
C+	67–69
C	63–66
C−	60–62
D+	57–59
D	53–56
D−	50–52
F	49 and below

8 Data Sheet

All students are allowed to use one data sheet (size: 8.5 by 11 inches) with information on both sides (no flaps, no tricks, etc) for the midterm and the final exam.

9 Calculator Policy

Calculators are not allowed during midterms or exams.

10 Turnitin Statement

This course makes use of Turnitin, a third-party application that helps maintain standards of excellence in academic integrity. Normally, students will be required to submit their course assignments to through onQ to Turnitin. In doing so, students' work will be included as source documents in the Turnitin reference database, where they will be used solely for the purpose of detecting plagiarism. Turnitin is a suite of tools that provide instructors with information about the authenticity of submitted work and facilitates the process of grading. Turnitin compares submitted files against its extensive database of content, and produces a similarity report and a similarity score for each assignment. A similarity score is the percentage of a document that is similar to content held within the database. Turnitin does not determine if an instance of plagiarism has occurred. Instead, it gives instructors the information they need to determine the authenticity of work as a part of a larger process. Please read Turnitin's Privacy Pledge, Privacy Policy, and Terms of Service, which governs users' relationship with Turnitin. Also, please note that Turnitin uses cookies and other tracking technologies; however, in its service contract with Queen's Turnitin has agreed that neither Turnitin nor its third-party partners will use data collected through cookies or other tracking technologies for marketing or advertising purposes. For further information about how you can exercise control over cookies, see Turnitin's Privacy Policy: Turnitin may provide other services that are not connected to the purpose for which Queen's University has engaged Turnitin. Your independent use of Turnitin's other services is subject solely to Turnitin's Terms of Service and Privacy Policy, and Queen's University has no liability for any independent interaction you choose to have with Turnitin.

11 Copyright of Course Materials

Any written or visual material an instructor produces is automatically copyrighted, and an instructor may pursue any violator of that copyright whether or not a notice is placed on the course material. Copyright does not dampen any ordinary use colleagues or students would make of the material.

12 Additional Required Syllabus Information

For important information on

- location and timing of final examinations,
- academic integrity,
- accommodations, and
- academic consideration for students in extenuating circumstances

please consult `http://www.cs.queensu.ca/students/undergraduate/syllabus/year2018-19.php/` the information on that page should be considered part of this syllabus.