University of Waterloo
Faculty of Engineering
Department of Electrical and Computer Engineering

ECE 208, Discrete Mathematics and Logic 2
Fall 2021

J.G. Thistle
jthistle@uwaterloo.ca

Calendar description

Formal logics, methods, and associated tools, and their uses in specifying, synthesizing, and verifying computing systems. Predicate logic. Temporal logic. Relational logic. Set theory. Proof theory. Model theory. Graph theory. Formal models of computation. Applications in computer and software engineering.

Prerequisites: ECE108; level at least 2B Computer Engineering or Electrical Engineering.

Themes and objectives

This course is an introduction to "mathematical" logic, but with an emphasis on its applications in engineering and computer science.

ECE 124 introduced Boolean algebra, of which the unique 2-element instance – switching algebra – forms the foundation of digital systems. ECE 108 touched on the related subject of mathematical logic, which was developed independently of Boolean algebra. Invented as a means of helping mathematicians write rigorous proofs, today mathematical logic is employed in databases, in artificial intelligence, in the design and implementation of programming languages, in the formal verification of both hardware and software, and in many other applications.

Mathematical logic was designed to capture the idea of a *formal proof*. Its proofs are defined by the structure and rules of *deductive systems*, and are 'formal' in the sense that it is possible to check them without considering their meaning – only their form. In principle, that allows computers to verify proofs, and to search for them. In practice, programmers and engineers often specify key steps of a proof, leaving details to be fleshed out by automatic *theorem provers*.

At one time, mathematicians envisioned using theorem provers to resolve all mathematical questions automatically, but that program ran into serious roadblocks. Not only did set-theoretic paradoxes show that care has to be taken in formulating such questions, but Kurt Gödel proved that any logic that allows the expression of sufficient properties of numbers, coupled with a consistent proof system for which there is an algorithm that lists all theorems, also allows the expression of true statements about numbers that are not theorems. Alonzo Church and Alan Turing then showed that some important questions about mathematics – and specifically, about program correctness – are *undecidable*: they cannot be answered by means of any mechanical algorithm. Moreover, among questions that *are* decidable, logical issues such as that of the satisfiability of propositional formulas (SAT) are canonical examples of problems that are computationally hard.

But despite these fundamental limitations, mathematical logic is an increasingly important and useful tool for engineers and computer scientists. This course will cover propositional logic and associated deductive systems, and introduce "SAT-solvers" that are used to search for solutions of problems that were once called computationally intractable. While propositional logic is essentially the same as switching algebra, first-order logic adds expressiveness by employing existential and universal quantifiers: this raises issues of undecidability, and (in)completeness, but allows the specification and the proof of important program-correctness properties.

More exotic formalisms such as modal logics (especially temporal logics) are suited to verifying properties of concurrent hardware and software systems: *model-checking* does so by semantic, rather than proof-theoretic means, and is widely used in industry.

This course will necessarily emphasize fundamentals, but, as time permits, will touch on applications that will be covered more fully in subsequent courses such as ECE 327, 451, 453, 457A and 457B.

Teaching assistants

Alexander Botros – in-person tutorials

Hamid Aghamohammadi – remote support

Lukas Fridolin-Schmidt – remote support

References

M. Ben-Ari, *Mathematical Logic for Computer Science*, third edition, Springer-Verlag, London 2012; `ISBN 978-1-4471-4128-0` .

An electronic version of the above reference is available (free of charge) from the UW Library.

Our treatment will differ somewhat from Ben-Ari's. The complete set of lecture notes will be posted on Learn; this will constitute the definitive text for the course.

Syllabus (timing subject to fine-tuning)

Note that throughout this term, academic weeks begin on Wednesdays and end on Tuesdays. Reading week falls between weeks 5 & 6.

Week	Topic		Refs.	Assgt/Quiz due
1	Introduction & overview		Ch. 1	
	Propositional logic:			
2		syntax & semantics (not tableaux)	Ch. 2	
3		deductive systems	Ch. 3	A1: on wks 1-2
4		resolution	Ch. 4	A2: on wk 3
5		binary decision diagrams	Ch. 5	
6		SAT solvers	Ch. 6	Q1: on wks 1-3
	First-order logic:			
7		syntax & semantics (not tableaux)	Ch. 7	A3: on wks 4-6
8		deductive systems	Ch. 8	Q2: on wks 4-6
9		terms & normal forms	Ch. 9	
10		undecidability & models	Ch. 12	A4: on wk 7
11	(buffer)			A5: on wk 8
12	(review)			Q3: on wks 7-8

Assignment due dates and quiz dates will be Thursdays. Assignments will be submitted electronically, and quizzes and the final exam will be held online.

Any cancellation of in-person instructional activities will affect only the in-person tutorials (sections 101 - 106), which will then be held exclusively online.

Students who are enrolled in in-person tutorial sections must attend only with the section in which they are enrolled. Students must not attend in person if they are experiencing influenza-like symptoms, if they have been in close contact with someone who is ill, or if they have travelled outside of Canada within the previous 14 days. Masks are to be worn in all common areas on campus, including all instructional spaces; no food is to be consumed in instructional space. Students are expected to wash their hands with soap and water or to use hand sanitizer frequently, and in particular, immediately before entering an instructional space.

Evaluation

- Late assignment submissions will not be accepted, but the lowest assignment mark will not be counted. The remaining assignment marks will each count for 10% of the final grade.
- The first three quizzes will be time-limited, administered through Learn or Mobius. Each will be worth 10% of the final grade.
- The final examination will cover the whole course, and will count for 30%.

Websites

Lecture notes, assignments, solutions and announcements will be posted on the course's Learn site. Old assignments and solutions will also be posted, for practice purposes.

Piazza will be used for discussion. It is strongly recommended that students use Piazza rather than email, for which the response time may be considerably longer. Sign up for Piazza at

`piazza.com/uwaterloo.ca/fall2021/ece208`

University information

COVID-19: For special considerations applying to this exceptional term, see `https://uwaterloo.ca/coronavirus/academic-information` .

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 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. 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 his/her actions. [Check the Office of 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. For typical penalties, check Guidelines for the Assessment of Penalties.

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.

Note for students with disabilities: AccessAbility 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.