CAS 760
Simple Type Theory
Winter 2026

Course Outline

Dr. William M. Farmer
McMaster University

Revised: December 26, 2025

Note: This course outline contains important information that may affect your grade. You should retain it and refer to it throughout the term, as you will be assumed to be familiar with the rules specified in this document.

Instructor

Dr. William M. Farmer
Office: ITB 163
Email: wmfarmer@mcmaster.ca
Web: http://imps.mcmaster.ca/wmfarmer/
Office hours: To see me in person or via a Microsoft Teams call, please send me an email message with some times you are free.

Schedule

Lectures:	Tue	2:30–4:00 PM	ITB 222
	Thu	2:30–4:00 PM	ITB 222

Course Web Site

This course will be administered via *Avenue to Learn* (*Avenue* for short). Go to

http://avenue.mcmaster.ca/.

to access the course's Avenue page. Please send only normal email to the instructional staff; do not send mail via Avenue.

Students should be aware that, when they access the electronic components of this course, private information such as first and last names, user names for the McMaster email accounts, and program affiliation may become apparent to all other students in the same course. The available information is dependent on the technology used. Continuation in this course will be deemed consent to this disclosure. If you have any questions or concerns about such disclosure please discuss this with the Instructor.

It is the student's responsibility to be aware of the information on the course's Avenue pages and to check regularly for announcements.

Calendar Description

"Introduction to simple type theory as a practical logic for expressing and reasoning about mathematical ideas. Syntax and semantics of classical higher-order predicate logic with partial functions and undefined expressions. Proof systems, tools for constructing libraries of formal mathematical knowledge, applications, and software support."

Mission

Simple type theory is a classical version of higher-order logic that extends first-order logic. Alonzo Church introduced in 1940 a very practical form of simple type theory, known as *Church's type theory*, that is tailored for reasoning with functions. This course will study a version of simple type theory named *Alonzo* based on Church's type theory that admits, unlike traditional predicate logics, undefined expressions (resulting from partial functions and definite descriptions). The course is intended as an introduction to simple type theory as well as to the use of logic as a practical tool for expressing and reasoning about mathematical ideas. It is also intended to be a logical foundation for the study of higher-order logic and type theory and the use of programming languages, proof assistants, and other mathematical software systems based on higher-order logic and type theory.

Learning Objectives: Precondition

A *learning objective* for a course is something the student is expected to know and understand or to be able to do. The *precondition* of the course is the set of learning objectives that the student is expected to have achieved before the start of the course. The precondition for this course is that the student is familiar with university-level mathematics, set theory (including ordinals and cardinality), first-order logic, recursive definitions, mathematical proof (including proof by induction), and decidability.

Learning Objectives: Postcondition

The *postcondition* of a course is the set of learning objectives that the student is expected to have achieved by the end of the course. The postcondition for this course is given below.

1. Students should know and understand:

    a. The logical principles underlying predicate logics such as first-order logic and simple type theory.

    b. The general and standard semantics for simple type theory.

c. What mathematical structures are and how they can be specified in a simple type theory like Alonzo.

d. How to reason with undefinedness in a logic like Alonzo that admits undefined expressions.

e. Proof systems for simple type theory.

f. The little theories method for organizing mathematical knowledge.

g. What kinds of software systems support the use of simple type theory.

2. Students should be able to:

   a. Express and reason about mathematical ideas in simple type theory.

   b. Write a theory in simple type theory to specify a single mathematical structure or a collection of similar mathematical structures.

   c. Build a mathematics library as a network of theories connected by theory morphisms.

Resources

1. W. M. Farmer, **Simple Type Theory: A Practical Logic for Expressing and Reasoning About Mathematical Ideas, Second Edition** [abbreviated **STT**], Computer Science Foundations and Applied Logic, Birkhäuser/Springer, 2025. Available at

   https://link.springer.com/book/10.1007/978-3-031-85352-4 .

2. W. M. Farmer, **LaTeX for Alonzo**, 2023 (revised 2024). LaTeX macros and environments for writing Alonzo types, expressions, and mathematical knowledge modules. Available on Avenue under Content / Resources.

3. W. M. Farmer and D. Y. Zvigelsky, **Monoid Theory in Alonzo: A Little Theories Formalization in Simple Type Theory**, Journal of Applied Logics, 12:1853–1939, 2025. Available on Avenue under Content / Resources.

Work Plan

There will be lectures, a bio sheet, M&Ms, a discussion forum, assignments, and two projects.

Two 75-minute lectures will be given in person each week by the Instructor. Slides for the lectures will be posted on Avenue.

Each student is required to submit a bio sheet about themself using the Instructor's bio sheet as a model (which is posted on Avenue under

Content / Course Information).[1] Bio sheets are submitted as an assignment on Avenue under Assessments / Assignments / Bio Sheets.

At the end of each week, each student is required to submit a short paragraph (2–4 sentences) that (1) describes something from the week's lectures or reading that was especially meaningful or memorable to the student, (2) identifies something that was not understood by the student, or (3) comments on how the student thinks the course is going. These *meaningfuls and memorables (M&Ms)* are intended to help the students to reflect on what they are learning and to give the Instructor feedback on what the students are experiencing in the course. The M&Ms are submitted to a discussion topic on Avenue under Communication / Discussions.

The discussion forum is located on Avenue under Communication / Discussions. Students are encouraged to use the forum to discuss the course material and to ask and answer questions. The Instructor will monitor the forum and participate in the discussion as needed.

Four assignments will be posted on Avenue over the course of the term under Content / Assignments. They will involve solving problems; proving conjectures; expressing and reasoning about mathematical ideas in Alonzo; and formulating theories, developments, and morphisms in Alonzo.

There will be two projects. The first will be a two-part *development project*. In the first part, the student will specify a single mathematical structure or a collection of similar mathematical structures as a theory development in Alonzo. In the second part, the student will develop a chosen mathematical model as a theory development graph in Alonzo. The second project will be a *software system project* in which the student will study a chosen software system that supports simple type theory. Each project will be done in groups of two or three. The deliverables for the development project are a paper and an oral presentation for each of the two parts. The deliverables for the software system project are a paper and a software system demonstration. Further details on the project will be presented in two documents entitled "Development Project" and "Software System Project" that will be posted on Avenue under Content / Projects.

There will be no midterm tests or final exam.

Midterm Course Review

In the middle of the term, the students will have the opportunity to complete an anonymous survey on Avenue about how the course is going. In addition, each student will be invited to attend an online course review session in which the student can ask the Instructor questions about the course and give feedback directly to the Instructor. Attendance is optional. The feedback that is received from the course survey and course review sessions may be used to modify how the course is working.

[1] I would like to thank Dr. Lydell Wiebe for communicating the bio sheet idea to me.

End-of-Term Course Evaluation

Near the end of the term, each student will have the opportunity to evaluate the effectiveness of this course. The feedback that is received from the course evaluation is very valuable to the Instructor and will be used to improve the course in subsequent years.

Discrimination

The Faculty of Engineering is concerned with ensuring an environment that is free of all adverse discrimination. If there is a problem, that cannot be resolved by discussion among the persons concerned, individuals are reminded that they should contact their Department Chair and the *Human Rights and Equity Services (HRES)* office as soon as possible.

Academic Integrity

You are expected to exhibit honesty and use ethical behaviour in all aspects of the learning process. Academic credentials you earn are rooted in principles of honesty and academic integrity. It is your responsibility to understand what constitutes academic dishonesty.

Academic dishonesty is to knowingly act or fail to act in a way that results or could result in unearned academic credit or advantage. This behaviour can result in serious consequences, e.g., the grade of zero on an assignment, loss of credit with a notation on the transcript (notation reads: "Grade of F assigned for academic dishonesty"), and/or suspension or expulsion from the university.

For information on the various types of academic dishonesty please refer to the *Academic Integrity Policy*, located at

`https://secretariat.mcmaster.ca/app/uploads/Academic-Integrity-Policy-1-1.pdf`.

The following illustrates only three forms of academic dishonesty:

1. Plagiarism, e.g., the submission of work that is not one's own or for which other credit has been obtained.
2. Improper collaboration in group work.
3. Copying or using unauthorized aids in tests and examinations.

Your work must be your own. Plagiarism and copying will not be tolerated! If it is discovered that you plagiarized or copied, it will be considered as academic dishonesty. In particular, submitting work produced by ChatGPT and other such apps will be considered plagiarism.

Students may be asked to defend their written work orally.

Authenticity/Plagiarism Detection

Some courses may use a web-based service (Turnitin.com) to reveal authenticity and ownership of student submitted work. For courses using such software, students will be expected to submit their work electronically either directly to Turnitin.com or via an online learning platform (e.g., A2L, etc.) using plagiarism detection (a service supported by Turnitin.com) so it can be checked for academic dishonesty.

Students who do not wish their work to be submitted through the plagiarism detection software must inform the Instructor before the assignment is due. No penalty will be assigned to a student who does not submit work to the plagiarism detection software. All submitted work is subject to normal verification that standards of academic integrity have been upheld (e.g., online search, other software, etc.). For more details about McMaster's use of Turnitin.com please go to

http://www.mcmaster.ca/academicintegrity/.

Courses with an Online Element

Some courses may use online elements (e.g., email, Avenue to Learn (A2L), LearnLink, web pages, capa, Moodle, ThinkingCap, etc.). Students should be aware that, when they access the electronic components of a course using these elements, private information such as first and last names, user names for the McMaster email accounts, and program affiliation may become apparent to all other students in the same course. The available information is dependent on the technology used. Continuation in a course that uses online elements will be deemed consent to this disclosure. If you have any questions or concerns about such disclosure please discuss this with the course instructor.

Online Proctoring

Some courses may use online proctoring software for tests and exams. This software may require students to turn on their video camera, present identification, monitor and record their computer activities, and/or lock/restrict their browser or other applications/software during tests or exams. This software may be required to be installed before the test/exam begins.

Conduct Expectations

As a McMaster graduate student, you have the right to experience, and the responsibility to demonstrate, respectful and dignified interactions within all of our living, learning, and working communities. These expectations are described in the *Code of Student Rights & Responsibilities* (*Code* for short), located at

https://secretariat.mcmaster.ca/app/uploads/Code-of-Student-Rights-and-Responsibilities.pdf.

All students share the responsibility of maintaining a positive environment for the academic and personal growth of all McMaster community members, whether in person or online.

It is essential that students be mindful of their interactions online, as the Code remains in effect in virtual learning environments. The Code applies to any interactions that adversely affect, disrupt, or interfere with reasonable participation in University activities. Student disruptions or behaviours that interfere with university functions on online platforms (e.g., use of Avenue 2 Learn, WebEx, or Zoom for delivery), will be taken very seriously and will be investigated. Outcomes may include restriction or removal of the involved students' access to these platforms.

Academic Accommodation for Students with Disabilities

Students with disabilities who require academic accommodation must contact *Student Accessibility Services (SAS)* at 905-525-9140 ext. 28652 or `sas@mcmaster.ca` to make arrangements with a Program Coordinator. For further information, consult McMaster University's *Academic Accommodation of Students with Disabilities* policy.

Academic Accommodation for Religious, Indigenous, or Spiritual Observances (RISO)

Students requiring academic accommodation based on religious, indigenous, or spiritual observances should follow the procedures set out in the RISO policy. Students should submit their request to their Faculty Office normally within 10 working days of the beginning of term in which they anticipate a need for accommodation or to the Registrar's Office prior to their examinations. Students should also contact their instructors as soon as possible to make alternative arrangements for classes, assignments, and tests.

Copyright and Recording

Students are advised that lectures, demonstrations, performances, and any other course material provided by an instructor include copyright protected works. The Copyright Act and copyright law protect every original literary, dramatic, musical, and artistic work, including lectures by University instructors.

The recording of lectures, tutorials, or other methods of instruction may occur during a course. Recording may be done by either the Instructor for the purpose of authorized distribution, or by a student for the purpose of personal study. Students should be aware that their voice and/or image may be recorded by others during the class. Please speak with the instructor if this is a concern for you.

Extreme Circumstances

The University reserves the right to change the dates and deadlines for any or all courses in extreme circumstances (e.g., severe weather, labour disruptions, etc.). Changes will be communicated through regular McMaster communication channels, such as McMaster Daily News, A2L, and/or McMaster email.

Other Policy Statements

1. Significant study and reading outside of class is required.
2. The student is expected to ask questions and discuss the course material during the lectures.
3. Assignments and projects may not be submitted late without *prior* approval from the Instructor.
4. Suggestions on how to improve the course and the Instructor's teaching methods are always welcomed.

Marking Scheme

The course grade will be based on the student's performance on the bio sheet, M&Ms, assignments, development project, and software project as follows:

Bio sheet	2%
M&Ms	8%
Assignments (4)	40%
Development project	30%
Software project	20%
Total	**100%**

Notes:

1. The *bio sheet mark* is 0 if it is not submitted; 1 if submitted after February 8; and 2 if submitted before or on February 8.
2. The *M&Ms mark* is x with $0 \leq x \leq 8$ where x is the percentage of weekly M&Ms that the student submitted multiplied by 8.
3. There is no mark for participation in the discussion forum.

Syllabus

0 Course overview.

1 Review of first-order logic.

2 Introduction to simple type theory (Chapters 1–2 of STT).

3 Preliminary concepts (Chapter 3).

4 Alonzo: syntax and semantics (Chapters 4–7).

5 Alonzo: proof system (Chapter 8, Appendices A–C).

6 Theories (Chapter 9).

7 Inductive sets and types (Chapter 10),

8 Sequences (Chapter 11)

9 Developments (Chapters 12–13).

10 Morphisms (Chapter 14).

11 Alonzo variants (Chapters 15).

12 Software support (Chapter 16).

13 Project presentations