Computing and Software 701

Logic and Discrete Mathematics

In Software Engineering

Fall 2008

Course Outline

Revised: 1 September 2008

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

Instructor

Dr. William M. Farmer
Office: ITB 163
Extension: 27039
E-mail: `wmfarmer@mcmaster.ca`
Web: `http://imps.mcmaster.ca/wmfarmer/`
Office hours: by appointment

Teaching Assistants

Pablo Castro (`castropf@mcmaster.ca`)

Course Web Site

`http://imps.mcmaster.ca/courses/CAS-701-08/`

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 e-mail 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 course instructor.

Schedule

Lectures: TR 14:30–16:00 ITB 222

Calendar Description

"Mathematical objects and logical concepts used in Software Engineering. Higher-order logic. Partial functions and undefined terms. Practical application of the axiomatic method. Recursive definition and inductive proof.

Survey of common mathematical structures and axiomatic theories used in Software Engineering. Effective use of mechanized mathematics systems."

Mission

The mission of this course is to give students a graduate-level understanding of the logic and discrete mathematics that is needed for Software Engineering. The course will focus on the creation, use, and understanding of mathematical models. By the end of the course the student should:

1. Understand what the mathematics process is and how it is employed in software engineering.
2. Understand the logic principles underlying the mathematics process and embodied in first-order logic and simple type theory.
3. Understand the relationship between an axiomatic theory and its set of models.
4. Be able to read and write mathematical proofs.
5. Have a working knowledge of the discrete mathematics that is commonly used in software engineering.

Students taking this course are expected to be familiar with propositional and first-order logic and numbers, sets, functions, and relations.

Recommended Text

K. H. Rosen, *Discrete Mathematics and Its Applications*, 6th Edition, McGraw-Hill, 2006. ISBN-13: 978-0073229720.

Work Plan

There will be lectures, exercises, a presentation, a midterm test, and a final exam. Two 75-minute lectures per week will given by the instructor. The lectures will present the course topics and will illustrate their use with examples. Students will be expected to attend the lectures, complete the exercises, and give one 10-minute presentation to the class. Approximately five exercises will be assigned. The midterm test will be held on Thursday, October 30, 2008 during the regular lecture time at 14:30–16:00. The final exam will be scheduled during the final examination period at the end of the term.

Academic Dishonesty

You are expected to exhibit honesty and use ethical behavior in all aspects of the learning process. Academic credentials you earn are rooted in principles of honesty and academic integrity.

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 behavior 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.

It is your responsibility to understand what constitutes academic dishonesty. For information on the various types of academic dishonesty please refer to the Academic Integrity Policy, located at

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

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.

Students may be asked to defend their written work orally.

Other Policy Statements

1. Significant study and reading outside of class is required.
2. The student is expected to ask questions during class.
3. A student may use calculators and his or her texts and notes during the midterm test and final exam.
4. Exercises may not be submitted late and the midterm test and final exam may not be taken later without *prior* approval from the instructor.
5. The instructor reserves the right to require a deferred final exam to be oral.
6. 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.
7. 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 exercises, presentation, midterm test, and final exam as follows:

Exercises (5)	25%
Presentation	10%
Midterm test	25%
Final exam	40%
Total	**100%**

Syllabus

00 Preliminaries

01 Introduction to Mathematics and Logic

02 Propositional Logic

03 Numbers, Sets, Functions, and Relations

04 Orders and Lattices

05 First-Order Logic

06 Equational Logic and Algebraic Reasoning

07 Recursion and Induction

08 Simple Type Theory

09 Temporal Logic and Model Checking (optional)