McMaster University
Department of Computing and Software
C01: **Dr. W. Kahl** C02: **M. Al-hassy**

COMPSCI&SFWRENG 2DM3
Outline
2020-09-07

Discrete Mathematics and Applications I

COMPSCI&SFWRENG 2DM3 — Fall 2020

Instructors:
- Sections C01: Dr. Wolfram **Kahl**, Dept. of Computing & Software, kahl@cas.mcmaster.ca
- Sections C02: Musa **Al-hassy**, Dept. of Computing & Software, alhassm@mcmaster.ca

Office Hours: *To be announced*, and by appointment.

Calendar Description:

Functions, relations and sets; the language of predicate logic, propositional logic; proof techniques, counting principles; induction and recursion, discrete probabilities, graphs, and their application to computing.

Goals:

This course will teach the mathematical foundations of computer science.

To a large degree, this can be seen as analogous to acquiring **language skills**, including **knowledge** and **skills** concerning syntax, semantics, pragmatics, and vocabulary of the **language of (discrete) mathematics**.

Conscious and precise use of this language is the foundation for **precise specification and rigorous reasoning**, which take a central place in this course.

Course Page: http://www.cas.mcmaster.ca/~kahl/CS2DM3/2020/

While most of the internal electronic information exchange for this course will be handled via **Avenue**, the course pages will contain useful links to external material, and will also serve as central fallback location for making information and material available outside Avenue, in particular in the case of Avenue accessibility problems. It is the student's responsibility to be aware of the information on the course Avenue site, and, while Avenue is down, on the course web page, and to check regularly for announcements (or RSS subscribe, where possible).

Schedule:

Lectures: Tuesday, Thursday, Friday 11:30–12:20

Tutorials:

Sections C01				Sections C02	
		T6	Fr 10:30–11:20	T1	Fr 11:30–12:20
T3	Th 12:30–13:20	T7	Mo 12:30–13:20	T2	We 13:30–14:20
T4	We 11:30–12:20	T9	We 11:30–12:20	T5	Fr 13:30–14:20
				T8	We 11:30–12:20

Tutorials start on **Wednesday of the first week**, the 9th of September.

At least at the beginning of the term, lectures and tutorials will be conducted in MSTeams, in dedicated "teams" there. For the sake of students in different time zones, lecture recordings will be made available via links in Avenue. **Strive to attend the lectures — if you watch a recording, you are already behind!**

Students are expected to attend all lectures and tutorials.

Textbook

"LADM": David Gries and Fred B. Schneider: **A Logical Approach to Discrete Math**, Springer 1993, ISBN 3-540-94115-0

> "This is a rather extraordinary book, and deserves to be read by everyone involved in computer science and — perhaps more importantly — software engineering. I recommend it highly [. . .] . If the book is taken seriously, the rigor that it unfolds and the clarity of its concepts could have a significant impact on the way in which software is conceived and developed."
>
> — Peter G. Neumann

Additional material will be made available electronically via the course pages.

Outline:

(With the most relevant textbook chapters indicated — not all textbook contents will be covered in detail, and material will be interleaved heavily. Times are rough estimates.)

•	Introduction to Calculational Reasoning	Parts of Chapters 1, 15	
	Boolean Expressions and Propositional Logic	Chapters 1–5	≈ 4 weeks
	Quantification and Predicate Logic	Chapters 8–9	
•	Predicates and Programming	Chapter 10	≈ 1 week
•	Sets	Chapter 11	≈ 1 week
•	Relations and Functions	Chapter 14	≈ 2 weeks
•	Induction and Sequences	Chapters 12–13	≈ 1.5 weeks
•	Graphs, Counting	Chapters 19, 16	≈ 2.5 weeks

Using **tool support** will be part of the expectations; details to be announced.

After notations, presentation rules, and basic definitions, axioma, and proof rules have been introduced in class, **students are expected to know them at all times.**

Exercises and Tutorials:

In most weeks, **Exercises** will be provided, from which the main material for the tutorials will be taken.

Every week (runnig Wednesday to Monday), starting September 9, there will be one-hour tutorial session in eight tutorial groups. The main purpose of the tutorials is to **discuss student work** on exercise problems. Therefore, every student is expected to complete the scheduled work, i.e., exercise problems or necessary reading, **before** the corresponding tutorial session — in particular, solutions and solution attempts to the Exercises of the current week are to be brought to the tutorial.

Grading:

All examinations in this course will be **“Closed Book”**. This means that they are **designed** to be completed without reference to any written or printed material and without use of a calculator or any other electronic aids except for the tools specified for use in each exam, and that reference to such extra material or use of extra tools would only waste time that is then not available anymore for the actual thinking necessary to solve the problems. (In contrast, “open book” examinations are **designed taking the time for consulting extra material into account.**)

All homework, assignment, test, and exam grades will be counted as percentage grades.

For every student, the course grade is calculated as a weighted average of homework, assignment, test, and exam grades as specified below.

Homework: In addition, there will be graded **Homework Questions** that need to be answered via CalcCheck essentially from one lecture to the next, normally only

- posted Friday afternoon by 5 p.m., and due Tuesday morning at 11 a.m., and
- posted Tuesday afternoon by 5 p.m., and due Thursday morning at 11 a.m..

The detailed arrangements may be changed during the term.

Homework questions will typically be graded only automatically, and possibly more summarily than assignments. There is no credit for submissions after the deadline.

For k being the number of **lectures with homework assigned** this course will have had, **10% of the course weight** are given to your $\lceil \frac{7}{8} \cdot k \rceil - 1$ best homework submissions. This allows you to miss probably at least 3 homework submissions — there will be **no further accommodations for missed homework**. (***In particular, MSAFs for homework will not be processed.***)

Assignments: There will be graded **Assignments** essentially on a biweekly basis. (**“Exercises”** are ungraded.)

Assignments may be graded only automatically and/or summarily; evaluation will be conducted mostly via the midterm tests and the final.

It is essential that you meet the deadlines for the assignments; there is no credit for material submitted after the deadline.

For n being the number of **assignments** this course will have had, **10% of the course weight** are given to your $(n - 1)$ best assignments. This allows you to miss one assignment — there will be **no further accommodations for missed assignments**. (***In particular, MSAFs for assignments will not be processed.***)

Some assignments may contain **bonus questions**. All bonus marks will be added to the course grade *only for those who have passed the course otherwise.*

Midterms: There will be **five midterm tests**. These will be 50-minute closed-book tests written **on-line using CalcCheck tool support**. The midterm tests will be written within the following time slots (all in Toronto time):

Midterm test 1:	Tuesday, 22nd September	11:30 to 12:20
Midterm test 2:	Tuesday, 6th October	11:30 to 12:20
Midterm test 3:	Tuesday, 27th October	11:30 to 12:20
Midterm test 4:	Tuesday, 10th November	11:30 to 12:20
Midterm test 5:	Tuesday, 24th November	11:30 to 12:20

Those **midterms** where your result is better than your result in the final count 12% each, and those midterms that are not better than the final count 10% each.

Accomodations for missed midterms will only be deferred midterms (possibly oral). No weight will be transferred to the final exam.

Final Exam: The **remaining course weight (between 20% and 30%)** is given to the **final exam**.

The final exam will be scheduled by the Registrar’s Office in the usual way. It will be a closed-book examination of 2.5 hours (150 minutes) duration and cover the material of **all** lectures, handouts, tutorials, exercises, homeworks, and assignments. The current hope is that the final exam will be written online using CalcCheck tool support, but it may turn out that this will not be possible due to organisational reasons beyond the control of the instructor.

The instructor reserves the right to conduct any deferred midterm or final exams orally.

The current plan is to have all midterms and the final written on computers using CalcCheck tool support.
However, technical or other issues may prevent availability of CalcCheck during tests and/or exam to individual students, or to all students. In case of such issues, tests and exams may still need to be (partially) written by hand.

Therefore, you need to be both **fluent in producing formalisations and proofs using the CalcCheck tool *and* fluent in writing syntactically correct formalisations and proofs by hand**.

Academic Integrity (see also page 4) — Course-Specific Notes

Academic credentials you earn are rooted in principles of honesty and academic integrity.

In the context of COMPSCI&SFWRENG 2DM3, in particular the following behaviours constitute academic dishonesty:

1. *Plagiarism*, i.e., **the submission of work that is not one’s own** or for which other credit has been obtained.
2. **Collaboration where individual work is expected.**

   You have to produce your submissions for homework and assignment questions yourself, and without collaboration.

   For each assignment question there will normally be exercise questions similar to it — you **are allowed** to collaborate on these **exercise questions**. (The tutorials are typically not expected to cover all exercise questions.)
3. **Copying or using unauthorised aids in tests and examinations.**
4. **Accessing another students’ Avenue or other relevant online account, or providing others access to your accounts.**
5. **Accessing or attempting to access midterm or exam material outside the individually assigned writing time and space.**

Note:

If you cheat, you are cheating yourself.

Later in the course, we intend to have individually-generated assignments and tests and so collaboration or cheating early on in the course will result in hardship during time-constrained midterms with individualised assignments where collaboration is no longer feasible and each person must use the allotted time to solve their individual problems.

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 the Department's Associate Chair for Undergraduate Studies, the Department Chair, the Sexual Harassment Office or the Human Rights Consultant, as soon as possible.

Learning Objectives

Course Precondition

Students are expected to have achieved the following learning objectives before taking this course:

1. Students should know and understand
   a) Basic number types, such as integers, rationals, reals
   b) General principles of mathematical notation
   c) General principles of calculation in mathematics
   d) Basic imperative programming
2. Students should be able to
   a) Perform calculations and solve equations in numerical domains
   b) Write and "mentally execute" simple imperative programs

Course Postcondition

Students are expected to achieve the following learning objectives **as a result of taking this course**:

1. Students should know and understand
   a) Syntax of propositional logic
   b) Syntax of predicate logic, including variable binding issues
   c) Principles of typed expressions, and the types of the operators they are using
   d) Basic semantics of expressions and formulae (truth tables, validity, . . .)
   e) Principles of calculational proofs
   f) Basic and derived proof rules of propositional logic
   g) Basics of (typed) set theory
   h) Basic concepts and theorems about functions and (especially binary) relations
   i) Basics of program specification using pre- and postconditions
   j) Basic theory of integers and counting
   k) Basic graph theory
2. Students should be able to
   a) Extract the propositional-logic structure from English sentences, and translate propositional expressions back into English
   b) Translate English statements of moderate complexity into predicate-logic specification
   c) Translate simple mathematical prose into predicate-logic definitions
   d) Produce calculational proofs in propositional logic
   e) Explain and prove basic properties using induction
   f) Produce correctness proofs for simple imperative programs.
   g) Produce calculational proofs in applications to set and relation theory, integers and counting, etc.
   h) Explain and prove basic graph properties.

Coverage of Graduate Attribute Indicators by Learning Outcomes (Preliminary)

For each indicator, the learning outcome topics (defined in the rubrics below) used to measure it are indicated:

1 (A knowledge base for engineering)

1.1 (Competence in Mathematics)

- Logical Formalism
- Calculational Proofs in Propositional Logic
- Calculational Proofs in Predicate Logic Applications
- Induction
- Program correctness proofs
- Formalisation
- Discrete Structures: Sets, Functions, Relations, Graphs, Counting

Learning Outcomes Rubric (Preliminary)

Topic	**Below**	**Marginal**	**Meets**	**Exceeds**
Logical Formalism	when writing formulae, frequently introduces syntax or type errors	writes mostly syntactically correct formulae with occasional type errors	writes syntactically and type-correct formulae, and correctly renames variables when substituting	can explain typing and variable binding issues with good understanding
Calculational Proofs in Propositional Logic	cannot produce correct calculational proofs even for simple theorems	typically makes about one mistake every three proof steps	normally produces correct proof calculations, but may not succeed with more complex tasks, or take unnecessary detours	confidently produces even more complex propositional-logic proofs
Calculational Proofs in Predicate Logic Applications	cannot produce correct calculational proof steps using predicate-logic proof rules even in simple contexts	shows some ability to apply predicate-logic proof rules in simple contexts	correctly applies predicate-logic proof rules in simple contexts, and can at least attempt to tackle more complex settings	correctly applies predicate-logic proof rules also in more complex settings
Induction	cannot handle even simple induction proofs	can mostly produce simple routine induction proofs	confidently performs simple and nested induction proofs	handles even complex induction settings confidently
Program correctness proofs	cannot handle even simple program correctness proofs; does not propose useful invariants	can mostly produce simple routine program correctness proofs and proposes mostly-appropriate invariants	confidently produces program correctness proofs for simple programs and identifies sufficient invariants	handles even nested program structures confidently, and identifies appropriate invariants even in settings that require more understanding
Formalisation	cannot handle even simple formalisation tasks	can produce simple formalisations that mostly capture the natural-language meaning	confidently performs simple formalisation tasks, but may not be able to cope with more complex tasks	produces reasonable formalisations even for moderately more complex tasks

Discrete Structures: Sets, Functions, Relations, Graphs, Counting	demonstrates hardly any understanding of basic concepts of discrete structures	knows and can apply some basic concepts of discrete structures	knows and can apply the majority of the concepts of discrete structures taught in class	demonstrates a solid understanding of the concepts of discrete structures taught in class

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/university-policies-procedures-guidelines/

The following illustrates only three forms of academic dishonesty:

- plagiarism, e.g. the submission of work that is not one's own or for which other credit has been obtained.
- improper collaboration in group work.
- copying or using unauthorized aids in tests and examinations.

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., on-line search, other software, etc.). For more details about McMaster's use of Turnitin.com please go to www.mcmaster.ca/academicintegrity.

COURSES WITH AN ON-LINE ELEMENT

Some courses may use on-line elements (e.g. e-mail, 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 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 a course that uses on-line 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 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* (the "Code"). 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 OF 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.

REQUESTS FOR RELIEF FOR MISSED ACADEMIC TERM WORK

McMaster Student Absence Form (MSAF): In the event of an absence for medical or other reasons, students should review and follow the Academic Regulation in the Undergraduate Calendar "Requests for Relief for Missed Academic Term Work".

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.