COMP 335
Introduction to Theoretical Computer Science
Department of Computer Science and Software Engineering

Fall 2023

Course Instructors:
Ali Jannatpour – Sections F and H
ali.jannatpour@concordia.ca

Denis Pankratov – Section G
denis.pankratov@concordia.ca

Nematollaah Shiri (coordinator) – Section E
nemat.shiri@concordia.ca

Office Hours:
TBA on Moodle

Tutorials: Please see your class schedule for details
Tutors: *TBA on Moodle*

Course Calendar Description:
Finite state automata and regular languages. Push-down automata and context-free languages. Pumping lemmas. Applications to parsing. Turing machines. Undecidability and decidability. Lectures: three hours per week. Tutorial: one hour per week.

Prerequisites: *COMP 232 (or COEN 231) and COMP 249 (or COEN 244).*
Co-requisites: *N/A*

Specific Knowledge and Skills Needed for this Course:
Students taking this course are expected to have sufficient knowledge of the following topics. Should you have difficulties in any of these topics, you are strongly encouraged to review them before the DNE deadline.

This course requires a good understanding of concepts and techniques in discrete mathematics, as studied in COMP 232. We recommend, in particular, that you refresh your knowledge of the following topics: sets, functions, relations, graphs, and proof techniques. The videos L0 in multiple parts provided in the course Moodle review this material; please be sure to watch these videos if you feel you need a refresher.

Course materials
Required Textbook: *An Introduction to Formal Languages and Automata, Peter Linz. and Susan Rodger, 7th Ed., Jones and Bartlett Publishers, March 2022.*

Grading Scheme

Course Component	*% of Total Course Weight*
Assignments (6)	*15*
Quizzes (best 10 out of 11)	*10*
Midterm Exam	*25*
Final Exam	*50*

The default grade is 0 for any missing course component, i.e., assignment, quiz, exam. There is no a priori, fixed relationship between the total percentage obtained and the final letter grade assigned, except that a higher percentage will not be assigned a lower letter grade.

To pass the course, students must obtain a passing grade in the final exam and the total, which is normally 50%.

Tentative Course Schedule

Week Number	*Textbook Sections*	*Topics*
1	*1*	*Introduction to theory of computation*
2	*2*	*Finite State Automata*
3	*3* *3.1 & 3.2* *3.3*	*Regular Languages* *Regular Expressions* *Regular Grammars*
4	*4.1 & 4.2*	*Closure Properties of Regular Languages*
5	*4.3*	*Pumping Lemma for Regular Languages*
6	*N/A*	*Mid-term break*
7	*5*	*Context-Free (CF) Languages and Grammars (MFG)* ***Midterm exam on Saturday, October 21, 2023 at 10:30AM***
8	*6*	*Simplification of CFGs and Normal Forms*
9	*7* *7.1* *7.3* *8*	*Push-Down Automata (PDA)* *Non-deterministic PDA* *Deterministic PDA* *Closure Properties of CFLs*
10	*8.1*	*Pumping Lemma for CFLs*
11	*9*	*Turing Machines (TM)*
12	*9.3 & 10*	*Combining TMs, Turing's Thesis*
13	*10.4* *12.1*	*Universal TM, Decidability & Undecidability* *The Halting Problem*

Engineering Tools

JFLAP – link is provided on Moodle

Details on assessment tools:

Assignments

There will be 6 assignments each worth 2.5%. Solution to assignments must be submitted through Moodle. While we encourage collaborative learning in this course, in particular, you are encouraged to discuss the assignments, however each student should write his/her solutions independently.

Quizzes

There will be 11 quizzes. Best 10 out of 11 will count towards the overall course weight. Each of the 10 counted quizzes is worth 1%. Further instructions will be posted on Moodle.

Exams

The midterm will be held on ***Saturday, 21 October 2023 at 10:30 am****. The final exam will be scheduled by the Examinations Office during the final examination period. At this time, both examinations are planned to be in-person.*

Other information

All individuals participating in courses are expected to be professional and constructive throughout the course, including in their communications. See Code of Rights and Responsibilities.

The most common offense under the Academic Code of Conduct is plagiarism, which the Code defines as "the presentation of the work of another person as one's own or without proper acknowledgement." In simple words, do not copy, paraphrase or translate anything from anywhere without saying where you obtained it. (Source: The Academic Integrity Website)

In the event of extraordinary circumstances and pursuant to the Academic Regulations, the University may modify the delivery, content, structure, forum, location and/or evaluation scheme. In the event of such extraordinary circumstances, students will be informed of the changes.

Content belonging to instructors shared in courses, including, but not limited to, online lectures, course notes, and video recordings of classes remain the intellectual property of the faculty member. It may not be distributed, published or broadcast, in whole or in part, without the express permission of the faculty member. Students are also forbidden to use their own means of recording any elements of an online class or lecture without express permission of the instructor. Any unauthorized sharing of course content may constitute a breach of the Academic Code of Conduct and/or the Code of Rights and Responsibilities. As specified in the Policy on Intellectual Property, the University does not claim any ownership of or interest in any student IP. All university members retain copyright over their work.

Instructor will strive to make learning experience as accessible and inclusive as possible. If you have accessibility needs that require academic accommodations, please meet with an advisor from the Access Centre for Students with Disabilities (ACSD) as soon as possible to set up an accommodation plan.

Graduate Attributes:
The following is the list of graduate attributes (skills) that students use, learn and/or apply throughout the term.

Problem Analysis [PA]:

- [PA-2] Modelling (Intermediate*): Use of graphs to represent abstract devices that model computers. Knowledge of finite state automata and regular languages, pushdown automata, context-free grammars, context-free languages, pumping lemmas, applications to parsing, Turing machines, what can and cannot be computed by a computer.*
- [PA-3] Problem Solving (Intermediate): *Ability to use the knowledge and skills obtained in this course to identify, and analyze problems related to computer and software systems, and solve them using appropriate formalisms.*

Course Learning Outcomes (CLOs):
By the end of this semester, students are expected to master the following concepts.

The learning objectives of this course are to:

- *introduce students to the foundations of computation including automata theory, the theory of formal languages and grammars, the notions of decidability and computability;*
- *develop abilities and skills of students to build and analyze abstract entities and write rigorous arguments and proofs.*

After successfully completing this course, students will be able to:

- *design finite state automata and regular grammars for regular languages;*
- *design context-free grammars and push-down automata for context-free languages;*
- *design Turing machines;*
- *identify the language class of a given language;*
- *prove closure properties of different language classes;*
- *understand the relationship between different language classes;*
- *appreciate the applications of this theory to lexical analysis and parsing.*

Health and Safety Guidelines
General health and safety instructions and available health and safety trainings can be found at: Safety Programs - Concordia University (https://www.concordia.ca/campus-life/safety/general-safety.html)

On Campus Resources
Please visit Student services at Concordia University for the services, available to Gina Cody School students.