

CS3331 - Foundations of Computer Science

Course description

We live during the computer revolution, which is changing fast everything around us. While programming languages change fast, the basic underlying theory does not. This course covers the basic concepts of the theory of computation. To study computation thoroughly, we need models. Ideally, we would like simple models to solve our problems. This is what the theory of computation is about: computational models and their power, with an impressive array of applications. That includes finite state machines, regular expressions, push-down automata, and context-free grammars. A crucial aspect is studying the limits of computations, which involves investigating all powerful models, such as Turing machines. Some problems are intractable, that is, it takes ages to solve them, others are provably impossible to solve even on an infinitely powerful computer. Computability theory sheds light on these issues of fundamental importance to anyone attempting to understand what computers can do for us.

Topics

- Regular Languages
 - Finite State Machines (FSM)
 - Deterministic (DFSM)
 - Nondeterministic (NDFSM)
 - Regular Expressions
 - Regular Grammars
 - Minimizing DFSM
 - Conversions between DFSM, NDFSM, Reg. Exp., and Reg. Grammars
 - Proving regularity
 - Closure properties
 - Proving nonregularity
 - Pumping theorem
 - Using closure properties
 - Decision Problems
 - Membership, Emptiness, Totality, Finiteness, Equivalence, Minimality
- Context-free Languages
 - Pushdown Automata (PDA)
 - Context-free Grammars (CFG)
 - Conversions
 - $PDA \leftrightarrow CFG$
 - $CFG \rightarrow$ Chomsky Normal Form
 - Ambiguity
 - Proving context-freeness
 - Closure properties
 - Proving noncontext-freeness
 - Pumping theorem
 - Using closure properties
 - Decision Problems

- Membership, Emptiness, Finiteness
- Undecidability
 - Turing Machines (TM)
 - Deterministic TM
 - Decidable languages (D)
 - Semidecidable languages (SD)
 - Multi tape TM
 - Nondeterministic TM
 - Universal TM
 - Halting Problem
 - D and SD
 - Enumeration
 - Reduction
 - Using reduction to prove undecidability
 - Rice's Theorem
 - Non-SD languages
 - Unrestricted Grammars
 - Non-TM Problems
 - Post Correspondence Problem (PCP)
 - Context-free language problems

Prerequisites

Computer Science 2214: Discrete Structures for Computing or Mathematics 2155: Discrete Structures I or registration in the third or fourth year of an honors program that combines Computer Science and other mathematical science or SE 2251A/B (251A/B) and registration in the third year of the BESC program in Software Engineering.

Unless you have either the prerequisites for this course or written special permission from your Dean to enrol in it, you will be removed from this course and it will be deleted from your record.

This decision may not be appealed. You will receive no adjustment to your fees in the event that you are dropped from a course for failing to have the necessary prerequisites.

Instructor

- Prof. [Lucian Ilie](#), MC368, e-mail: ilie@uwo.ca
-

Textbook (required)

- Elaine Rich, *Automata, Computability, and Complexity. Theory and Applications*, © Person Prentice Hall (2008), ISBN 978-0-13-228806-4. ([errata](#))
-

Tools

- [Jflap](#) (You are allowed to use jflap for assignments.)
-

Evaluation (tentative due dates) -- Assignments will be available on [OWL](#)

- Assignment 1 (10%) -- due Oct.17
- Assignment 2 (10%) -- due Oct.24
- Assignment 3 (10%) -- due Nov.30

- Assignment 4 (10%) -- due Dec.5
 - Midterm Exam (30%) -- (covers Regular and Context-free Languages) -- **Tuesday, Oct. 31, in class (2 hours); see lecture notes for sample midterm**
 - Review sessions by TAs (in additions to in class):
 - TBA
 - Final Exam (30%) -- (covers Turing Machines and Undecidability) -- TBA
-

Class time

- Tuesdays, 3:30 - 5:30pm, NCB-113
 - Wednesdays, 9:30 - 10:30am, NS-7
-

TAs

- Yiwei Li (yli922@uwo.ca)
 - Nick DelBen (ndelben@uwo.ca)
 - Andy Yu (ayu82@uwo.ca)
-

Office hours

- Instructor: Wednesdays, 3:00 - 4:00pm, MC368
 - Yiwei: TBA
 - Nick: TBA
 - Andy: TBA
-

Assignments

- The assignments will consist of a set of exercises related to the material covered in class. The solutions for the exercises should be neatly written or typed.
 - All assignments will be made available on the course web site. The availability of assignments will be announced on class and/or via e-mail. Students are responsible for checking their e-mail on a regular basis.
-

Appeals of Assignment Marks

- Appeals of assignment marks should be addressed to the T.A. first. If you and the T.A. cannot agree, then the T.A. will discuss the situation with the lecturer.
 - Appeals must occur within 1 week from the first day that the marked assignments were made available to students. After that 1 week period has gone by, no more appeals will be considered.
-

Computing Facilities

Each student will be given an account on the Computer Science Department senior undergraduate computing facility, GAUL. In accepting the GAUL account, a student agrees to abide by the department's; Rules of Ethical Conduct.

Adherence to Deadlines

There is no penalty for late submissions up to three days. After that the late work is no longer accepted.

Academic Accommodation for Medical Illness

If you are unable to meet a course requirement due to illness or other serious circumstances, you must provide valid medical or other supporting documentation to your Dean's office as soon as possible and contact your instructor immediately. It is the student's responsibility to make alternative arrangements with their instructor once the accommodation has been approved and the instructor has been informed. In the event of a missed final exam, a "Recommendation of Special Examination" form must be obtained from the Dean's Office immediately.

For further information please see: <http://www.uwo.ca/univsec/handbook/appeals/medical.pdf>

A student requiring academic accommodation due to illness should use the Student Medical Certificate when visiting an off-campus medical facility or request a Record's Release Form (located in the Dean's Office) for visits to Student Health Services. The form can be found here:

https://studentservices.uwo.ca/secure/medical_document.pdf

Students who are in emotional/mental distress should refer to Mental Health@Western

<http://www.uwo.ca/uwocom/mentalhealth/> for a complete list of options about how to obtain help.

Accessibility Statement

Please contact the course instructor if you require material in an alternate format or if you require any other arrangements to make this course more accessible to you. You may also wish to contact [Services for Students with Disabilities \(SSD\)](#) at 661-2111 x 82147 for any specific question regarding an accommodation.

Ethical Conduct

Scholastic offences are taken seriously and students are directed to read the appropriate policy, specifically, the definition of what constitutes a Scholastic Offence, at the following Web site:

<http://www.uwo.ca/univsec/handbook/appeals/scholoff.pdf>.

Plagiarism: Students must write their essays and assignments in their own words. Whenever students take an idea, or a passage from another author, they must acknowledge their debt both by using quotation marks where appropriate and by proper referencing such as footnotes or citations. Plagiarism is a major academic offence.

If assignments are to be individual assignments: All assignments are individual assignments. You may discuss approaches to problems among yourselves; however, the actual details of the work (assignment coding, answers to concept questions, etc.) must be an individual effort.

The standard departmental penalty for assignments that are judged to be the result of academic dishonesty is, for the student's first offence, a mark of zero for the assignment, with an additional penalty equal to the weight of the assignment also being applied. You are responsible for reading and respecting the Computer Science Department's policy on Scholastic Offences and Rules of Ethical Conduct.

The University of Western Ontario uses software for plagiarism checking. Students may be required to submit their written work and programs in electronic form for plagiarism checking.
