CSC236H1F: Introduction to the Theory of Computation

Instructor: Daniela Rosu
Email: drosu@cs.toronto.edu

Office Hours*:*
TBD

Lectures Time and Location*:*
L0101: Wed/Fr 11am - 12pm, BA1130
L5101: Th 6pm - 8pm, BA1130

Tutorials Time and Location
L0101: Mo 11am - 12pm, TBD
L5101: Th 8pm - 9pm, TBD

<u>Course website</u>*:* Blackboard-portal
All course material, including lecture slides, will be posted on the website.

All announcements will be made through the Blackboard portal and it is your responsibility to check it regularly.

Course Overview

Outline The course will cover the following subjects:

- Proof by Simple, Complete, and Structural Induction, Well-ordering Principle.
- Time Complexity of Recursive Algorithms, Recurrence Relations.
- Divide-and-Conquer Algorithms.
- Algorithm Correctness
- Regular Languages, Finite-State Automata, Regular Expression

Prerequisite CSC148, CSC165

Tutorials During each tutorial session, students will work on a set of exercises. Exercises for each session will be posted on the course web page a few days before the session. Students are expected to work on the exercises *before* the tutorial and be prepared to correct and/or complete their solutions with the help of the TA. Tutorial rooms will be announced once the room assignments are finalized, in the second week of classes. There are no tutorials in the first week of classes.

Textbook Course Notes for CSC B36/236/240, Copyright 1998, 2007 by Vassos Hadzilacos (Free PDF available on Blackboard portal). Note that we will *not* follow these notes to the letter.

Additional Reference K.H. Rosen: *Discrete Mathematics and Its Applications*, 7th Ed. (2011), McGraw-Hill Science.

How to do well in this course The key to mastering any subject, especially in theoretical subjects, is to *comprehend* the concepts of the subject, and *practice* applying the concepts.

Throughout the course, we provide as much practice as possible. However, depending on your mathematical background, you might need to do extra exercises that are not part of the course work. The following are hence necessary for doing well in the course, but might not be sufficient:

- Attend the *lectures* and *tutorials*, *ask* questions, *participate* in class discussions, and go to *officer hours*.
- Work on the given exercises *before* the tutorials, show your solutions to the TA, and ask for *feedback*.
- Spend 8-10 hours/week:
  - 2 hours in lectures
  - 1 hour in tutorial
  - 5-7 hours reviewing slides and course notes, working on exercises and assignments.
- Check the course web page and emails *regularly*, pay attention to the course *instructions*, *policies*, *announcements* and *deadlines*.

Evaluation

Summary The following table summarizes the course-work percentages and due dates

Item	***Tentative Due Dates***	***Weight***
Assignment #1 Assignment #2 Assignment #3	February 9 , 10pm March 9, 10pm April 4, 10pm	15% 15% 15%
Term Test	TBD	15%
Final Exam	TBD	40%

Details

- ***3 Assignments***: worth *45%* in total.
  Assignments are to be completed in groups of no more than **three** students.
  Assignments will be posted on Blackboard, *at least two weeks* before the due dates. Start working on them *early*, so that you have an estimate of how much time you need to complete them, and to identify the parts that you need clarification and/or help with.
  Assignment solutions will *not* be discussed during the lectures and tutorials, but will be posted on the course web page. See the "Policies and Other Instructions" section for information about assignment submission, late submission policy, and remark requests.

- ***Mid-Term Test***: The term test takes 1 *hour*. You will be allowed to bring one handwritten 8.5"x11" aid sheet.

- ***Final Exam***: The final exam is *3 hours* and will cover *all* the topics discussed in the course. You will be allowed to bring one handwritten 8.5"x11" aid sheet.

Policies and Other Instructions

Assignments Submission Submissions must be *typed* and submitted as PDF files on MarkUs.

Re-marking Requests If you feel a piece of your work has been graded unfairly, please submit a remarking request within *one week* of receiving the work back. Explain your request clearly and briefly. Remark request for assignments must be submitted through MarkUs.

Late Work Late assignments will not be accepted except for valid and *documented* reasons. Documents for justifying late or missed work must be submitted to the *instructor* as soon as possible.

Discussion Board *General* questions about the course organization, material, and assignments should be posted on the discussion board (the link to the discussion board will be announced on the course webpage on the Blackboard Portal). The discussion board will be monitored by the instructor and TAs, but can also be used for discussion among students. You may NOT discuss the assignment solutions on the board **until *1 week*** after the due dates.

Email Policy Please use your *university email address* and put the *course code* in the subject line of your emails, e.g., ***CSC236 request for appointment.*** Use email only for *personal issues* such as requesting special considerations.

Academic Integrity Academic integrity is a fundamental principle in higher education. Any breach of academic honesty is a serious academic offence which eventually can affect one's professional life dramatically. Suspected cases of academic dishonesty will be investigated based on the University's Integrity Policies, with no exception.

When discussing assignment problems with other groups, do NOT take any notes (paper or electronic) from the discussions. Your submissions must be developed and written solely based on *your own interpretation* of group discussions, otherwise it will be considered as plagiarism. For details on the meaning of plagiarism and how it can be avoided read this document.

Tentative Schedule

Week	***Topic***	***Readings from the Text Book***	***Important Dates***
1	Introduction, Simple & Complete Induction	Sections 1.1, 1.2, 1.3	
2	Complete Induction, Well-ordering, Recursively Defined Sets	Sections 1.3, 4.1	
3	Structural Induction, Recursively Defined Function	Chapter 4, Section 3.1	
4	Complexity, Master Theorem	Sections 3.1, 3.2	
5	Complexity, Divide-and-Conquer Algorithms	Section 3.3	
6	Program Correctness	Sections 2.1, 2.2	
7	Program Correctness (recursive programs)	Sections 2.7, 2.8	
8	READING WEEK		
9	Program Correctness (iterative programs)	Sections 2.3, 2.4, 2.5, 2.6	
10	Regular Languages, Regular Expressions	Sections 7.1, 7.2	
11	Deterministic Finite State Automata (DFSA)	Sections 7.3,7.5	
12	Equivalence of regular expressions and DFSA, Nondeterministic Finite State Automata	Sections 7.4, 7.6	
13	Review		