Course Intro

COMPSCI 3331

Fall 2022

Overview

- Course Format
- Course Evaluation.
- What is this course about?

Course Website

- Everything for the course is on `owl.uwo.ca`
- Course ID: COMPSCI 3331A 001 FW22
- What's on the website?
  - Videos
  - Course notes.
  - Forum
  - Syllabus
  - All Assignment Submission (gradescope)

Me

- Mike Domaratzki - `mdomarat@uwo.ca`
- Office Hours (MC 355 E) - Monday 2:30-3:30 and Thursday, 2:30-4:30.
- I may have to change office hours on short notice - I will let you know on **owl** when this happens.
- Directions to MC 355 E are on the website.

Course Times

- Tuesday - 3:30-5:30 - SSC 2050
- Wednesday - 2:30-3:30 - MC 110

Textbook

- No required textbook for the course.
- Potential textbooks for those who would like one:
  - Rich, Automata, Computatibility and Complexity. (free PDF available here.)
  - Sipser, Introduction to the Theory of Computation.
  - Martin, Introduction to Languages and the Theory of Computation.
  - Hopcroft, Motwani and Ullman, Introduction to Automata Theory, Languages and Computation.
- Differences in notation may exist between different textbooks and what we use in class.

Course Evaluation

- Assignments (4) – 32 %
- In-class Quizzes (6) – 12 %
- Midterm (in class, Oct 25) – 20 %
- Final – 35 %

+ 1% free

Assignments

- All assignments are submitted on Gradescope.
- All assignments equally weighted.
- Assignments are individual.
- No programming in this course.

Assignment Due Dates

- Assignment 1 – October 11
- Assignment 2 – October 26
- Assignment 3 – November 22
- Assignment 4 – December 7

Assignment Deadlines

- Late penalty for assignments - 1 % of the assignment value per hour, up to 48 hours.
- No assignments accepted after 48 hours.
- If you have a serious medical or compassionate reason, talk to your Academic Counsellor.
- Beyond the above, students can get **ONE** extension for an assignment. The extension is for 3 days (72 hours), no questions asked. See more details in the course outline.

Quizzes

- Quizzes are held in class on Wednesday, for the first 15 minutes of class.
- Quizzes are open book.
- 8 quizzes, top 6 will be used to calculate your grade (2 % per quiz).
- If you miss a quiz for a documented reason, markes will be transferred to other quizzes.
- There are no make-up quizzes.

Quiz Schedule

- Q1 - Sept 28
- Q2 - Oct 5
- Q3 - Oct 12
- Q4 - Oct 19
- Q5 - Nov 9
- Q6 - Nov 16
- Q7 - Nov 23
- Q8 - Nov 30

Midterm and Final

- Closed book
- Midterm Exam is Oct 26 in class.
- No make-up midterm - marks will be transferred to final for documented reasons.
- Final exam will be scheduled by the registrar during the exam period.

Midterm Format

- Midterm will be a **two-stage exam**
- Contains both an individual and a group portion.
  - You write the individual portion, then hand it in.
  - You get into groups of 3-4 (assigned before exam).
  - You solve a subset of very similar problems with your group and hand in **one copy.**
- More details will come, but **you don't need to know anyone in the course.**
- Vast majority of marks will be for individual portion (TBA, but $\geq 85\%$ of midterm grade)
- Your mark will **never go down** because of the group portion.

Course Format

- Pre-class videos:
  - posted on owl,
  - should be reviewed before class and the quiz.
- In-class: examples, proofs and more!
- Wednesday quizzes.
- Mentimeter - for questions

Why is the class structured like this?

- Improved learning and retention.
- I know it may not be what you are used to.
- Philosophies:
  - Frequent Evaluation: You should know how you are doing in the class.
  - Separation of background and in-class material: Class time is to see course concepts in action, more detailed examples.
  - Two-stage exam helps ~~retenion~~ retention - discussion of exam material.

Mentimeter

- Go to menti.com.
- Use the code 4660 1263.
- Questions are not marked - they are for you only.
- All questions are anonymous.

Scholastic Offenses

- Scholastic offenses are taken seriously in Computer Science.
- Do not share your solutions to any evalution with anyone, including posting in a public location.

What is this course about?

- “Languages as sets of strings over an alphabet; operations on languages; finite automata, regular expressions; language hierarchy; Turing machines; models of computation.”
- This course looks at the underlying theory behind computation.
- By looking at restricted models, we focus on the fundamentals.
- Tools in this course are found in foundational ways in many areas of CS.
- Can ask questions like “What does it mean to be computable?” and “Are there problems that are not computable?”

Major concepts in the course

- Finite automata and regular expressions
- Grammars: context-free grammars and parsing.
- Turing Machines: abstract models of computation.

Canola Plants through Grammar Systems

Image from M. Cieslak *et al.*, L-system models for image-based phenomics: case studies of maize and canola, in silico Plants, Volume 4, Issue 1, 2022. (Creative Commons license)

ANTLR

- ANTLR defines grammars for programming languages.
- Allows us to efficently and automatically build compilers for these languages.
- Relies on the concepts of regular and context-free languages that we will learn in the course.

Prokaryotic Gene Prediction

- Genomes are the collection of DNA in an organism. Not all the DNA is a gene.
- Given a genome (large sequence of nucleotides), where are the genes?
- Complex problem, even for simple organisms, like prokaryotes, like bacteria.
- Use HMMs (a type of automata) to predict where genes occur.

How hard are problems?

- Turing Machines are the basis for examining how hard problems are.
- Can show that there are some problems that cannot be solved by any algorithm.
- Implications for tasks like program verification.