

## ECE 606, Fall 2020, Syllabus, Logistics and Schedule

Algorithms are at the very foundations of computing. It is important that one understands how to design them, and analyze them for correctness and efficiency. It is important also that one recognizes whether a problem is intractable so one does not naively seek an efficient algorithm when none may exist. The intent of this course is to provide students with fundamental training in these aspects.

Prerequisite	ECE 250 or an equivalent <a href="http://www.ucalendar.uwaterloo.ca/1920/COURSE/course-ECE.html#ECE250">http://www.ucalendar.uwaterloo.ca/1920/COURSE/course-ECE.html#ECE250</a>
Lectures	Online. Videos will be posted weekly by midnight Monday.
Instructor	Mahesh Tripunitara, <a href="mailto:tripunit@uwaterloo.ca">tripunit@uwaterloo.ca</a> “Mahesh” “Dr. T” “Prof. T” Office hours: <div style="background-color: black; width: 200px; height: 15px; margin: 5px 0;"></div> Or by appointment
TAs <i>For issues regarding assignments only</i>	Kittiphon Phalakarn, <a href="mailto:kphalakarn@uwaterloo.ca">kphalakarn@uwaterloo.ca</a> Indrani Ray, <a href="mailto:indrani.ray@uwaterloo.ca">indrani.ray@uwaterloo.ca</a>
Course materials	<a href="http://learn.uwaterloo.ca">learn.uwaterloo.ca</a>
“Textbook”	Tripunitara, “ECE 606 – Algorithms” Available on Learn. May be posted in stages only.
Discussions	Self signup at <a href="http://piazza.com/uwaterloo.ca/fall12020/ece606">piazza.com/uwaterloo.ca/fall12020/ece606</a>
Marking	Weekly assignments 50% Final exam 50%
Audit	All deliverables (assignments + final exam) must be met. A mark of 50 on the course must be achieved.
If you are waiting to get into the course	There is usually a precipitous drop in enrollment in the first couple of weeks of term. Simply pretend as though you are in the course. Submit assignments via email. Keep checking for open spots to sign up.
AccessAbility	<a href="http://uwaterloo.ca/accessability-services">uwaterloo.ca/accessability-services</a>
Academic Integrity	<a href="http://uwaterloo.ca/academic-integrity">uwaterloo.ca/academic-integrity</a> → Students

# Content Schedule

<i>Week of</i>	<i>Topics</i>
(1) Sep 08	Intro to the course; Discrete math review; intro to Python
(2) Sep 15	Expressing algorithms; Data structures review
(3) Sep 22	Properties of algorithms: existence, correctness, efficiency
(4) Sep 29	Design strategy I: incremental
(5) Oct 06	Design strategy II: divide-n-conquer
Oct 13, reading week	
(6) Oct 20	Design strategy III: greedy
(7) Oct 27	Design strategy IV: dynamic programming
(8) Nov 03	Randomization, Probabilistic and approximation algorithms
(9) Nov 10	Non-determinism; computational complexity; the class <b>NP</b> ; other complexity classes
(10) Nov 17	Cook- and Karp-reductions; hardness and completeness for a complexity class
(11) Nov 24	<b>NP</b> -complete problems and reductions between them
(12) Dec 01	Reconciling intractability; common mistakes
Dec 14 – 23, final exam	

# Assignments

There will be weekly assignments, for a total of 12 assignments across the course. Each comprises a few problems that the TAs will mark. There may be problems in the assignment that involve programming in Python 3; each such problem will be annotated with “[python3].” Assignments will be published by midnight every Tuesday. Each is due in a week, i.e., by midnight the following Tuesday. Written solutions must be typeset, or written legibly and scanned, and uploaded as PDF.

Lateness policy: no late submissions accepted.

Collaboration policy: you may collaborate with your colleagues when working on your assignments in that you can discuss ideas with one another. However, your final submission must be your own. That is, when you sit down to write your solutions down, you should do so on your own. Any sources you use, whether they are your colleagues, books, papers or online resources, should be appropriately credited in your submission. There is no penalty for utilizing such (re)sources, provided they are credited explicitly. Otherwise, it is regarded as plagiarism, and is an academic offence.

Originality detection: We will be using the following software in this course to check that your submissions are indeed original, and not plagiarized.

**Turnitin.com**: Text matching software (Turnitin<sup>®</sup>) may be used to screen assignments in this course. Turnitin<sup>®</sup> is used to verify that all materials and sources in assignments are documented. Students submissions are stored on a U.S. server, therefore students must be given an alternative (e.g., scaffolded assignment or annotated bibliography), if they are concerned about their privacy and/or security. Students will be given due notice, in the first week of the term and/or at the time assignment details are provided, about arrangements and alternatives for the use of Turnitin<sup>®</sup> in this course.

It is the responsibility of the student to notify the instructor if they, in the first week of term or at the time assignment details are provided, wish to submit the alternate assignment.

# Final Exam

The final exam will be published similarly as the assignments. However, you will have 24 hours only to turn in your solutions. Your submission is expected to be your own; you are not to collaborate with anyone, in the course nor outside. However, you're free to use any resources, e.g., books/papers/online resources, that you like. You should credit any (re)source you use in your solutions.