Introduction and Overview

Department of Electrical and Computer Engineering
McMaster University

January 2020

McMaster University

Comp Eng 2SI4

Data Structures, Algorithms and Discrete Mathematics

Winter 2020

Instructors	Dr. R. Tharmarasa (Thamas) (C01) and Jianfeng Hu (C02)
Office:	ITB-A317
Phone:	(905) 525-9140 ext. 26503
Email:	thamas@mcmaster.ca & huj19@mcmaster.ca
Web:	http://www.ece.mcmaster.ca/~thamas

Teaching Assistants

- Abhijit Chatterjee, chatta4@mcmaster.ca
- Dawei Guo, guod11@mcmaster.ca
- Fatemeh Armanfard, armanfaf@mcmaster.ca
- Ming Chen, chenm143@mcmaster.ca
- Peiyao Chen, chenp36@mcmaster.ca
- Prabhanjan Mannari, mannarip@mcmaster.ca
- Sahand Sephrevand, sepehrvs@mcmaster.ca
- Tongyu Ge, get3@mcmaster.ca
- Weijian Chen, chenw43@mcmaster.ca

- Office Hours, Contact Info and Lab Supervision times to be posted on Avenue to Learn ...

Lectures and Tutorials

McMaster University

Lectures:

C01: Tuesday, Thursday, Friday, 11:30am–12:20pm, ITB AB102

C02: Monday & Wednesday 11:30am–12:20pm, Friday 1:30pm–2:20pm, ABB 102

Tutorials:

T01: Monday 12:30pm–1:20pm, ITB AB102

T02: Tuesday 10:30am–11:20am, ABB 102

Labs

- 3 hours every other week, in ITB AB111

- L01/L02 Monday, 2:30-5:20pm
- L03/L04 Tuesday, 2:30-5:20pm
- L05/L06 Wednesday, 2:30-5:20pm
- L07/L08 Thursday, 2:30-5:20pm
- L09/L10 Friday, 2:30-5:20pm
- L11/L12 Wednesday, 7:00-10:00pm

Course Resources

- Text Books:
  Mark Allen Weiss, *Data Structures & Problem Solving Using JAVA*, 3rd or 4th edition, Addison Wesley (ISBN 0-321-32213-4).
  Kenneth H. Rosen, *Discrete Mathematics and Its Applications*, 7th or 8th edition, McGraw Hill (ISBN 9781307221848)
- Textbook Website:
  `https://users.cs.fiu.edu/~weiss/`
- Course Website:
  `http://avenue.mcmaster.ca`
- Please check the course website regularly for announcements, office hours, lecture notes, tutorial notes, lab questions, homework/practice questions, past exams etc.

Email Instructions

- Please **do not** send emails to the instructors or TAs via Avenue, but only using your and their McMaster email addresses.
  - → E-mail subject line should start with: COE 2SI4, or CompEng 2SI4
  - → Please include in your email your full name and student ID number
- Instructors' email addresses:
  `thamas@mcmaster.ca`
  `huj19@mcmaster.ca`
- TA contact info is posted on the course website. TAs will offer office hours starting this week.

Software

- Download the latest version of `Java` and `NetBeans` IDE at
  - → `http://www.oracle.com/technetwork/java/javase/downloads/index.html`
- You may use the same version of `Java` used for 2SH4 last term.
- The lab demos may be performed on lab computers or on your own computer.

What is this course all about?

- This course is about the design of **efficient** programs.
- An efficient program is one which minimizes the **cost** required in terms of:
  - **Memory** required
  - **Time** required to obtain the solution
- Why do we need efficient programs? Why not just buy a better computer?
  - As computers become more complex so too do the applications the run.
  - Want to maximize the benefit from a given computing device.

- A representation for information within a computer and the routines for manipulating the data.
- There are often many ways to represent a given data set each with their own strengths and weaknesses
  - → **Time** required to process data
  - → **Space** required to store the information and any overhead structures
  - → **Coding Complexity** to implement the data structure.
- Often their is a trade-off between space efficiency, time-efficiency and complexity of the data structure.
  - → Typically we want the "simplest" data structure which meets the performance requirements.
- This course builds on list and stack data structures covered in 2SH4.
  - → queues, heaps, binary search trees
  - → definition, efficient implementations, applications

- An algorithm is a procedure or method to solve a problem (like a "recipe"). It has the following features:
  - It must be **effective**, i.e., it must solve the problem.
  - It consists of a series of **unambiguous** instructions.
  - It must process one or more **inputs** and produce at least one **output**.
  - It must be **finite**, i.e., terminates in after a set number of steps for any input.
- A computer **program** is a particular implementation of an algorithm.
- Example: We will study a number of algorithms for searching and sorting in some data structures and quantify their efficiency.

- What is discrete mathematics?
  - → study of the mathematics concerning discrete or distinct objects.
- Many problems in computer science and engineering fall into this category
  - → computer network planning
  - → design of logic circuits
  - → complexity analysis
  - → counting problems
- We have an introduction into basic notions in set theory, functions and relations, elements of graph theory and modular arithmetic.

Course Objectives

- To learn how to analyze and design **efficient** programs
- To gain exposure to popular data structures such as stacks, queues, trees, graphs, heaps.
- To gain familiarity with important searching, sorting and graph algorithms.
- To be able to analyze the space and time complexity of algorithms
- To have an introduction into discrete mathematics
- To gain practical programming experience
  - → This is **not** a course in `Java` or `C` programming!

Course Outline (approximate)

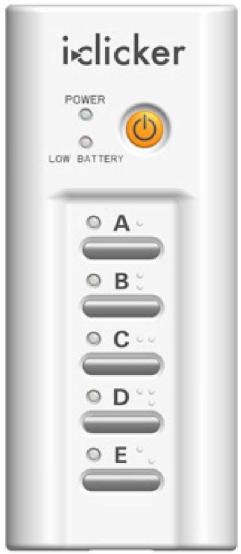
- Algorithm Analysis (analysis of time and space complexity of algorithms)
- Stacks and Queues
- Recursion
- Trees. Binary Search Trees
- Hash Tables
- Algorithms for sorting
- Heaps. Heapsort
- Sets, functions and relations
- Elements of graph theory. Graph representation, paths, spanning trees, Dijkstra's algorithm, Prim's algorithm.
- Modular arithmetic

Assessment

Labs	20%	Five **individual** labs
max(Quizzes, Final Exam)	10%	More than 15 pop quizzes
max(Midterm Test, Final Exam)	25%	Midterm schedule TBD
Final Exam	45%	To be scheduled by registrar

iClicker

- Each student is expected to have his/her own i>clicker
- Register it
- Bring it to every lecture and tutorial
- Ensure that the batteries are in working order
- Frequency code: AA



Assessment: Quizzes

- Multiple choice questions using i>clicker
- Best 60% of the quizzes
  - Example 1: participate in 12 quizzes out of 15 quizzes
    - 60% of 15 is 9
    - If the marks for 12 quizzes are (10, 8, 6, 10, 4, 8, 6, 10, 10, 8, 6, 10), the final marks for the quizzes is

$$\frac{10+10+10+10+10+8+8+8+6}{9} = 8.89 \text{ (out of 10)} \quad (1)$$

  - Example 2: participate in 6 quizzes out of 15 quizzes
    - 60% of 15 is 9
    - If the marks for 6 quizzes are (10, 10, 10, 10, 10, 2), the final marks for the quizzes is

$$\frac{10+10+10+10+10+2}{9} = 5.78 \text{ (out of 10)} \quad (2)$$

- Marks for a quiz:
  - 10 marks for each quiz
  - 2 marks for answering at least 50% of the questions
  - 8 marks for answering all the questions correctly
  - Example: 4 questions => 2 marks for answering at least two questions, 2 marks for each correct answer
    - Student 1: Answers to all the questions and 3 of out of 4 answers are correct.
      Marks: $2 + 3 \times 2 = 8$ (out of 10)
    - Student 2: Answers only one question and that answer is correct
      Marks: $0 + 1 \times 2 = 2$ (out of 10)

Assessment: Max

- Example 1:
  - Labs: 18 (out of 20), quizzes: 9 (out of 10), midterm test: 20 (out of 25), final exam: 30 (out of 45)
  - Final marks

$$
\begin{aligned}
&= 18 + \max(9, 30/45 * 10) + \max(20, 30/45 * 25) + 30 && (3)\\
&= 18 + \max(9, 6.67) + \max(20, 16.67) + 30 && (4)\\
&= 18 + 9 + 20 + 30 && (5)\\
&= 77 && (6)
\end{aligned}
$$

- Example 2:
  - Labs: 18 (out of 20), quizzes: 0 (out of 10), midterm test: 0 (out of 25), final exam: 30 (out of 45)
  - Final marks

$$
\begin{aligned}
&= 18 + \max(0, 30/45 * 10) + \max(0, 30/45 * 25) + 30 && (7)\\
&= 18 + \max(0, 6.67) + \max(0, 16.67) + 30 && (8)\\
&= 18 + 6.67 + 16.67 + 30 && (9)\\
&= 71.3 && (10)
\end{aligned}
$$

Assessment: Max (2)

- Example 3:
  - Labs: 18 (out of 20), quizzes: 9 (out of 10), midterm test: 20 (out of 25), final exam: 20 (out of 45)
  - Final marks

$$
\begin{aligned}
&= 18 + \max(9, 20/45 * 10) + \max(20, 20/45 * 25) + 20 && (11)\\
&= 18 + \max(9, 4.44) + \max(20, 11.11) + 20 && (12)\\
&= 18 + 9 + 20 + 20 && (13)\\
&= 67 && (14)
\end{aligned}
$$

- Example 4:
  - Labs: 18 (out of 20), quizzes: 0 (out of 10), midterm test: 0 (out of 25), final exam: 20 (out of 45)
  - Final marks

$$
\begin{aligned}
&= 18 + \max(0, 30/45 * 10) + \max(0, 30/45 * 25) + 20 && (15)\\
&= 18 + \max(0, 4.44) + \max(0, 11.11) + 20 && (16)\\
&= 18 + 4.44 + 11.11 + 20 && (17)\\
&= 53.55 && (18)
\end{aligned}
$$

Assessment: Max (3)

- Example 5:
  - Labs: 15 (out of 20), quizzes: 8 (out of 10), midterm test: 12 (out of 25), final exam: 15 (out of 45)
  - Final marks

$$
\begin{aligned}
&= 15 + \max(8, 15/45 * 10) + \max(12, 15/45 * 25) + 15 && (19)\\
&= 15 + \max(8, 3.33) + \max(12, 8.33) + 15 && (20)\\
&= 15 + 8 + 12 + 15 && (21)\\
&= 50 && (22)
\end{aligned}
$$

- Example 6:
  - Labs: 15 (out of 20), quizzes: 0 (out of 10), midterm test: 0 (out of 25), final exam: 15 (out of 45)
  - Final marks

$$
\begin{aligned}
&= 15 + \max(0, 15/45 * 10) + \max(0, 15/45 * 25) + 15 && (23)\\
&= 15 + \max(0, 3.33) + \max(0, 8.33) + 15 && (24)\\
&= 15 + 3.33 + 8.33 + 15 && (25)\\
&= 41.66 && (26)
\end{aligned}
$$

Laboratories

- Labs start the week of January 13th
- There are 5 labs in all, worth 20 % of the course mark.
  - Lab 1: Jan. 13-24; Lab 2: Jan. 27-Feb. 7; Lab 3: Feb. 10-28; Lab 4: Mar. 2-13; Lab 5: Mar. 16-27; Make-up labs: Mar. 30, 31, Apr. 1,2,3, 2020.
- Each lab will be posted online at least 2 weeks before the lab starts, except lab 1.
- You must demonstrate the solution in front of a TA by the end of your lab session to get the assigned mark. The source code and the report have to be submitted electronically on Avenue by the end of your designated lab session or as specified in the lab instructions.
- Students must attend the lab from the beginning of the lab session.
- You should start working (or complete) the lab exercises well before your lab session in order to complete everything on time.

Laboratories (2)

- Demonstrations may be performed on the computers in the lab or on students' computers.
- When you demonstrate your code you will be asked to do some or all of the following
  - → To run the code on some specified inputs.
  - → To describe your algorithm without looking at the code.
  - → To explain portions of your code or answer related questions.
- **You are expected to be able to explain all portions of your code.** Any instances of academic dishonesty will be reported to the University Office of Academic Integrity. Please check the policy on Academic Integrity to find out the possible consequences.

Make-up Labs

- Make-up labs (all in ITB AB111):
  - (L01, L02) March 30 from 2:30-5:20pm
  - (L03,L04) March 31 from 2:30-5:20pm
  - (L05,L06) April 1 from 2:30-5:20pm
  - (L07,L08,L11) April 2 from 2:30-5:20pm
  - (L09,L10, L12) April 3 from 2:30-5:20pm
- Labs under an MSAF are to be demonstrated and submitted during the make-up lab.
- When submitting an MSAF, confirm with me that I have received it.
- Missed labs for which an MSAF was not submitted may also be demonstrated during the make-up lab, with a 50% penalty to overall mark.

Homework/Practice Questions

- Homework/Practice questions will be assigned weekly (mainly from the textbook) and posted on the website.
- These questions are not marked, however, students are encouraged to solve them to gain valuable problem solving experience.
- Solutions to some homework problems will be discussed at the tutorial.

Tests

- Midterm Test and Final Exam will cover
  - → Theory: definitions, description of algorithms, Java code given in class, material from the textbook.
  - → Exercises: part of them similar to homework exercises, lab assignments or to examples discussed in class, part of them new; some of them will require writing Java code.

How To Succeed?

- STUDY! STUDY! STUDY!
- Attend lectures and pay attention in class.
  - → Lecture notes posted on the web are not complete! Be prepared to take additional notes in class;
- Review lecture notes at home.
- Read the material assigned from the textbook.
- Solve homework exercises; attend tutorials.
- Complete the lab assignments.
- Solve additional problems.
- If you do not understand, do not wait!!! Attend tutorial and office hours if you need help.

Office Hours

- Instructor's office hours:
  - → Every week at least one hour (see Avenue to Learn)
  - → By appointment
  - → More office hours will be scheduled before the midterm
- TAs will offer office hours starting this week; to be posted soon on Avenue.

Academic Dishonesty

- Academic dishonesty is taken **extremely** seriously.
- All cases of academic dishonesty will be reported to the Office of Academic Integrity.
- On the first offence, the standard penalty is a zero on the work in question.
- Subsequent offences are much more serious: the student is typically assigned an F in the course, with a transcript notation indicating the F is for academic dishonesty.
- For more information, please visit:

`http://www.mcmaster.ca/policy/ac_ethics.htm`
`http://www.mcmaster.ca/academicintegrity/students.cfm`

Academic Dishonesty Involving Copying in Lab Assignments

- A student who submits an assignment containing portions which are not their own work will receive a mark of 0 for the whole assignment.
- A student who provides a portion of his/her assignment to another student to copy, commits an act of academic dishonesty, and will receive a mark of 0 for the whole assignment.
- All incidents will be reported to the Office of Academic Integrity.
- Electronic screening of all submitted code will be performed.

Next Topic ...

How do we determine if an algorithm is “efficient”? How do we measure its cost in terms of

- Memory use
- Time use