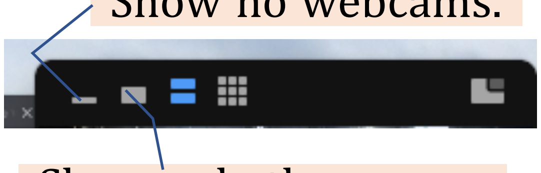
COMP 206
Introduction to Software Systems

Joseph D’Silva
Computer Science
McGill University

Zoom Etiquette

- You are not required to have your webcam on during the lectures if you have privacy concerns or if it affects your bandwidth.
- You can also reduce the clutter on the zoom by minimizing the number of webcam videos showing on your screen.



Show no webcams.

Show only the person who is talking.

- You are muted throughout the duration of lectures to avoid problems from background noise, etc.

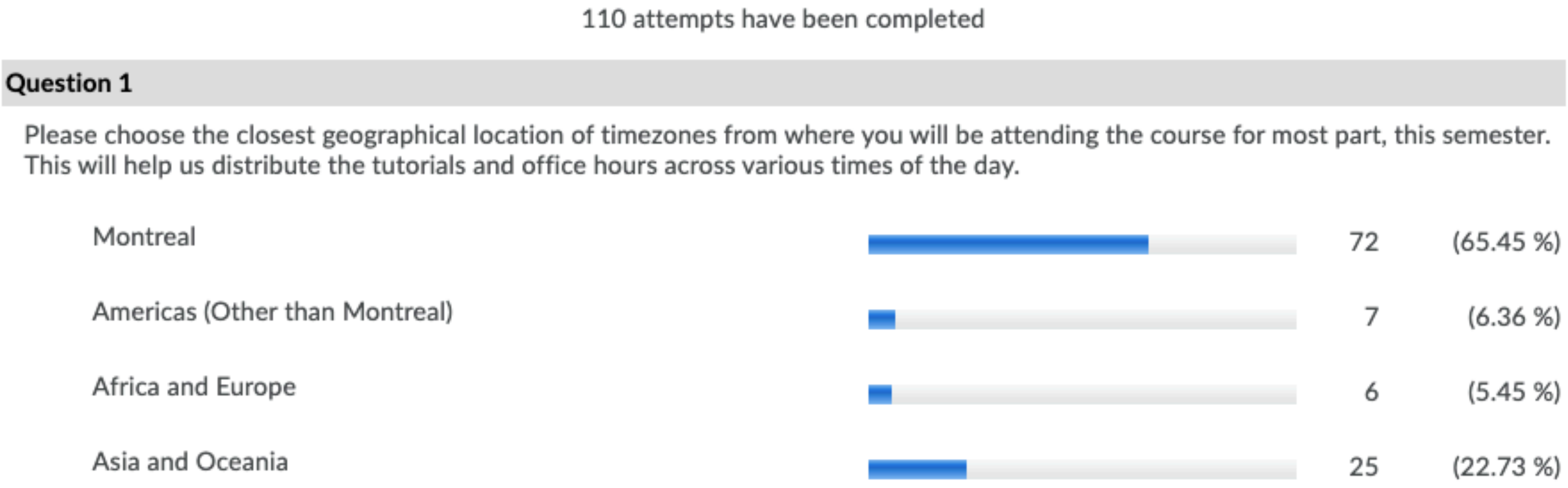
- If you have questions during the lecture, please use the Live Q&A feature of Piazza.
  - Upvote questions instead of repeating them.
  - I will pause and go over the questions occasionally, and also at the end of every lecture.
  - Students can also respond to questions here.

- If your question is not specific to the lecture topic, please save it for the end of the class where you can ask all those questions.
- We are not doing discussions over zoom chat as this leads to a lot of cross talks/tangent conversations and I will loose track of unanswered questions.

Zoom Etiquette

- If you are attending lectures live, your bandwidth issues may cause video streaming to “freeze”.
  - If that happens, please do not ask me to repeat the material during the lecture, there are 500+ of you, and we will never finish a topic :)
  - You can certainly ask for clarifications at the end of the class (I promise to stick around till you are done) or re-watch the relevant part from the lecture recording.

Class Distribution



110 attempts have been completed

Question 1

Please choose the closest geographical location of timezones from where you will be attending the course for most part, this semester. This will help us distribute the tutorials and office hours across various times of the day.

Montreal	72	(65.45 %)
Americas (Other than Montreal)	7	(6.36 %)
Africa and Europe	6	(5.45 %)
Asia and Oceania	25	(22.73 %)

Information

The only email that I check frequently.

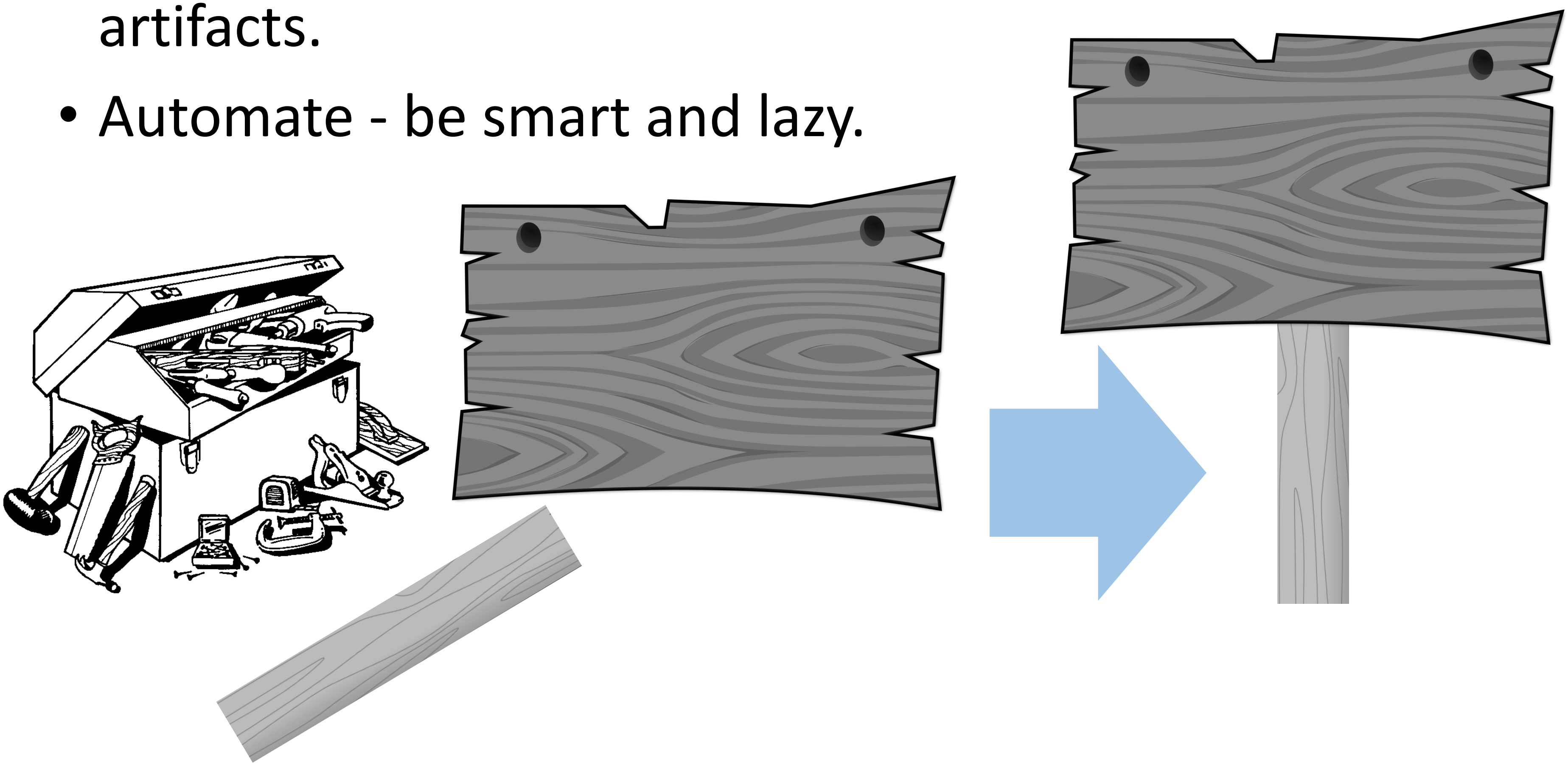
- Email: joseph.dsilva@cs.mcgill.ca
- URL: www.cs.mcgill.ca/~jdsilv2

- Lectures:
  - Monday, Wednesday, Friday 3:35pm – 4:25pm
  - Most will be live (see zoom information in MyCourses).
  - Recordings will be available for later.
  - Some lectures might be delivered as pre-recorded (TBD).

- Office hours:
  - (Public) after class, as long as it takes (same zoom session as the lectures).
  - (Private) Friday 9:00 am – 10:00 am https://mcgill.zoom.us/j/93352787766
    - You might have to be in a waiting room, if there are other students.

Part I – Unix : A Tool Chest

- Implement solutions by integrating existing software artifacts.
- Automate - be smart and lazy.



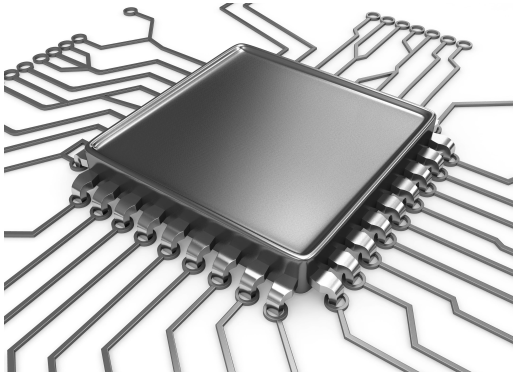
Syllabus

PART I - Unix	
Unit 1 – Course Introduction	
INTRODUCTION TO SOFTWARE SYSTEMS Introduction to the course. What is this course about? Importance of Systems and command-line development.	Course outline Textbook: Chapter 1
Unit 2 – The Unix Environment	
THE UNIX/LINUX OPERATING SYSTEM The story of Unix. The architecture of Unix OS. Getting access to the SOCS Linux servers. SFTP and SSH clients, logging in, file transfer, working from home and school	Textbook: Section 2.0
THE SHELL The OS shell environment. The command-line prompt. Home vs root. Basic commands: ls, cd, mkdir, rmdir, cp, mv, cat, more, man, logout, paths.	Textbook: Sections 2.1-2.2
REGULAR EXPRESSIONS AND WILD CARDS Command-line commands that use wild cards and regular expressions. Using redirection. Using grep.	Textbook: Sections 2.2-2.3
VIM & DEVELOPER TECHNIQUES Non-GUI editors and their importance. Common developer techniques: directory structures, procedures, commands: chmod, tar, zip, backups.	Textbook: Sections 2.2-2.3
ADVANCED UTILITIES **NEW** find, sed, awk. sort	
Unit 3 – Bash Programming	
INTRODUCTION TO BASH SCRIPTING What is BASH programming? When to use BASH programing? Examples of simple Bash programs.	Textbook: Section 2.4 Example 1
BASH EXPRESSIONS Variables (bash, shell, session), math expressions, and I/O.	Textbook: Section 2.4 Example 2
BASH CONTROL STRUCTURES Conditions, iteration, and functions.	Textbook: Section 2.4 Examples 3 – 4
BASH DEVELOPER TECHNIQUES Bash as an aid to developers, development environment initialization, standardizing operations (archiving and backups, etc.) Debugging shell scripts. **NEW**	
SESSIONS AND BASH SCIPTS The Session. Session memory. Customization. System vs session scripts.	
SCHEDULING at, crontab **NEW**	
Midterm #1	Units 1 to 3

Syllabus

PART II - The C Programming Language	
Unit 4 – C Programming	
INTRODUCTION TO THE C LANGUAGE The story of C. Why C? Children of C. Hello World example with puts() and getc(), GCC basics, compiling, running, errors. Bash compiling scripts.	Textbook: Section 3.0 Example 1
DATA AND CONTROL STRUCTURES IN C Types, variables, expressions, conditions and iteration.	Textbook: Section 3.0 Example 2 & 4
STDIO.H and STDLIB.H getchar, putchar, puts, printf, scanf, sprintf, sscanf. I/O issues and data validation. STDIN, STDOUT and STDERR.	Textbook: Section 3.0 Example 3, 3.1
ARRAYS AND STRINGS Array, strings, static & invariant data, writable data, array addressing.	Textbook: Section 3.0 Example 5
POINTERS, STRINGS, AND STRING.H Pointer referencing and de-referencing. Example: make string.h functions using pointer referencing. Understanding pointers using symbol tables **NEW**	Textbook: Section 3.0 Example 6
FUNCTIONS AND SCOPE Function syntax, scope rules. Call-by-value and call-by-reference.	
STRUCT AND UNION Struct and Union syntax. Array of struct.	Textbook: Section 3.0 Example 7
DYNAMIC MEMORY Dynamic arrays, dynamic structs, linked lists.	Textbook: Section 3.0 Example 8
SEQUENCIAL TEXT FILES The file concept. Streams. Text and CSV files.	
Unit 5 – Basic Software Development Techniques	
MODULAR PROGRAMMING C object files, compiler performance, team programming basics, about large projects. The extern expression. The Pre-processor.	Textbook: Section 3.0 Example 9, 4.0
GNU TOOLS The makefile. The profiler. The GDB. call graph visualization – doxygen **NEW**	Textbook: Section 4.1
REPOSITORIES What is a repository? Ways to use repositories. Using git. More team programming basics. Branching.	Textbook: Section 4.1

Part II - C : The Raw Power



Syllabus

PART II - The C Programming Language	
Unit 6 – Systems Programming	
INTRODUCTION TO SYSTEMS time.h, Bit-wise operations. Bash to C parameter passing. Void * referencing. About machines.	Textbook: Section 3.0 Example 6
CONCURRENT PROGRAMS Shell memory-based communication, ampersand operator, ps, kill, and pwd.	Textbook: Section 4.2
INTER PROCESS COMMUNICATION What is a process. C process creation: system and fork. Producer Consumer problem.	Textbook: Section 4.2
BLOCK FILES AND RANDOM FILES Sequential Block, Random and Binary files (fread, fwrite, fseek).	
Time permitting …	
NETWORKS AND WEBSITES Basic network architecture & the SOCS web server.	Textbook: Sections 5.0 – 5.1
INTER PROCESS COMMUNICATION WITH CGI Using CGI to communicate with C using the Internet.	Textbook: Sections 5.2-5.4
MORE CGI and C	Textbook: Sections 5.5-5.7
Unit 7	
VOID * AND FUNCTION *	
SIGNAL HANDLING	
SOCKET COMMUNICATION	
ASSEMBLER WITH C	
Unit 8 – Final Exam Review	
FINAL EXAM REVIEW	FINAL – Units 1 to 6*

* Plus the topics covered in Unit 7, if any were covered during class.

Text Book

Primary Text:

Software Systems ed3; Vybihal & Azar; Kendall/Hunt; ISBN 978-0-7575-9514-1.
You can purchase the textbook from here:
https://he.kendallhunt.com/product/software-systems

We will use the textbook in the course for most of the sections.

Supplementary Texts:

GNU Software; Louksides & Oram; O'Reilly; ISBN 1565921127 (free on web)
Just Enough Unix; P.K. Anderson; McGraw Hill; ISBN 0697131726
C Programming Language; Kernighan & Ritchie; Prentice-Hall; ISBN 0131101633

Unix Accounts

- Required for your labs and doing assignments (and most of your programming oriented COMP courses henceforth).
- Physical Location (before the apocalypse)
  - Trottier 3rd floor
  - Assistance:
    - CS help desk - help@cs.mcgill.ca
    - Office: McConnell 209N
    - Phone: 514-398-7087
- Accounts
  - **DO NOT** use your McGill account! first.last@mail.mcgill.ca
    - These files are automatically deleted when you logout of a computer.
  - You must use a SOCS account
    - https://newuser.cs.mcgill.ca
    - You must be on McGill Wifi to access the above website or on a McGill VPN
      - https://mcgill.service-now.com/itportal?id=kb_article&sysparm_article=KB0010687
      - Contact McGill IT if you have issues accessing VPN (not the CS help desk)
    - Forgot your Username and/or Password?
      - Reset it at https://newpassword.cs.mcgill.ca/

Information

Know Your TA

TA who will grade you submissions, email for help with issues.

	Student Last Name	TA Name	TA Email
1	A - Bl	Harmanpreet Singh Grover	harmanpreet.grover@mail.mcgill.ca
2	Bo - Cho	Anirudha Jitani	anirudha.jitani@mail.mcgill.ca
3	Chu - Fa	Anirudha Jitani	anirudha.jitani@mail.mcgill.ca
4	Fe - Hao	Safa Alver	safa.alver@mail.mcgill.ca
5	Har - J	Ayrin Ahia-Tabibi	airin.ahia-tabibi@mail.mcgill.ca
6	K - Le	Ayrin Ahia-Tabibi	airin.ahia-tabibi@mail.mcgill.ca
7	Li - Mac	Ridwan Kurmally	ridwan.kurmally@mail.mcgill.ca
8	Mag - Pan	Stephanie Xie	xinyi.xie@mail.mcgill.ca
9	Pap - Seg	Tzu-Yang (Ben) Yu	tzu-yang.yu@mail.mcgill.ca
10	Sel - Te	Tzu-Yang (Ben) Yu	tzu-yang.yu@mail.mcgill.ca
11	Ti - W	Jingyi He	jingyi.he@mail.mcgill.ca
12	X - Z	Abraham Yesgat	abraham.yesgat@mail.mcgill.ca

Information

- TA office hours (private, waiting room)
  - Open to all students in the course
  - Through zoom
  - Lecture topics, assignment questions, etc.
  - Have your question ready when you are meeting the TA.
  - Do not “start working on the assignment” during the office hour. There are others who need to meet the TA for clarifications. You can always join again if you have new questions.

TA Hours and Zoom will be soon posted in MyCourses

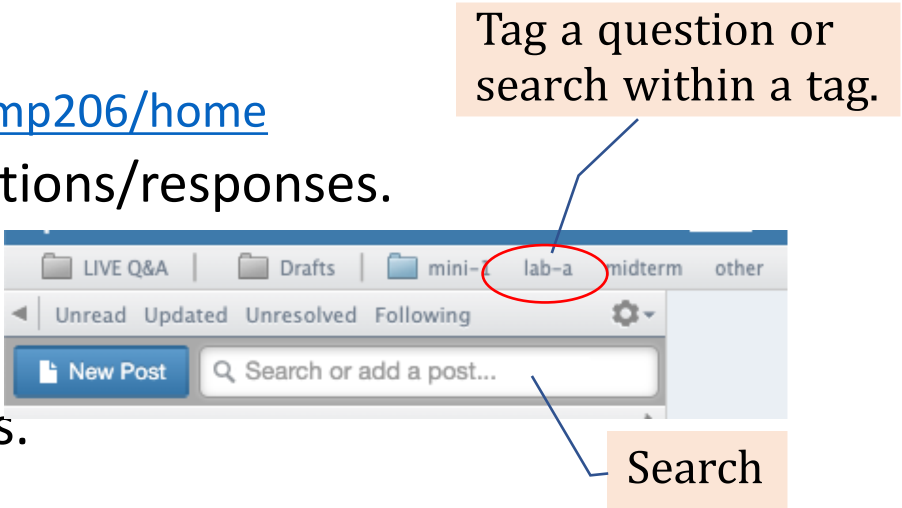
TA Name	TA Email
Abraham Yesgat	abraham.yesgat@mail.mcgill.ca
Harmanpreet Singh Grover	harmanpreet.grover@mail.mcgill.ca
Safa Alver	safa.alver@mail.mcgill.ca
Jingyi He	jingyi.he@mail.mcgill.ca
Ridwan Kurmally	ridwan.kurmally@mail.mcgill.ca
Stephanie Xie	xinyi.xie@mail.mcgill.ca

Information

Discussions – Piazza

https://piazza.com/mcgill.ca/fall2020/comp206/home

- Search for previously posted questions/responses.
  - Search within a tag.



Tag a question or search within a tag.

Search

- Ask questions about lecture topics.
  - Tag your questions appropriately.
- Ask clarifications on assignment / lab questions (also tag them to appropriate topics).
- Students can answer questions posted by other students.
  - Instructors/TAs do check for the validity of these answers.
- Learn from the collective knowledge of the “hive”.
- **DO NOT** Post your entire assignment code here asking for “help” to debug. You are essentially sharing your solutions – repeat offenders can potentially start losing assignment points.

Information

Whom do I…?

- Public questions (lectures, clarifications on lab/assignment question) – use Piazza – One of the many TAs or fellow students would have a quick answer. I go over them regularly. Everyone who reads it also gets to benefit – contribute to the collective knowledge of the hive.
- Assignment grades - Email the TA assigned to grading your submissions.
- Having problems with assignment code.
  - Remember that TAs will only offer you suggestions on how to debug, or approach the problem at a high level. They are not responsible to make your code work. It is YOUR assignment and it should demonstrate that you are able to master the content and build a software artifact with minimal help from others.
  - Go to a TA office hour (best and productive approach for immediate results).
  - Email the TA that is assigned to you (not the most efficient way to address the issue – your TA maybe having a busy day to respond immediately – especially requests can pile up closer to assignment deadlines).
  - Do not wait till last minute to start working on assignments – you might not be able to get immediate help – there is only a limited number of TA resources.
- Other concerns (approvals, etc.)
  - Email me.
- Questions about applying/learning some of the lecture topics outside the scope of the course.
  - Definitely email me!
- Please use your McGill email for all official communications. We do not respond to other emails (security reasons).

Code Etiquettes

For both Piazza and Emails to TAs.

- Please do not post/email “screenshots” (image format) of source code.
  - Somebody else need to type out your code again (can be long and boring, they can make mistakes, etc.)
  - Less inclined to help you.
  - Include your source code (if you can, try to format it so that it is readable) in the actual text (for reasonably small code snippets), or attach the files (when emailing).
  - DO NOT Post your assignment code on Piazza!
- Include the actual code snippet / command that is causing the error message along with the error.
  - An error could be due to various reasons.
  - Knowing the actual code triggering the error helps someone reproduce it quickly and give you suggestions.
- Make it easy for someone to help you.

Labs

- ~ 10 Labs
  - Hands-on practice of topics covered in lectures.
  - Not graded, these are meant to give some practical exposure before students start working on individual assignments.
- 6 slots (1 hour each)
  - Through zoom (not recorded).
  - Attendance not mandatory, you can also do it on your own.
  - TAs may not entertain “assignment problems” during Lab sessions.

Labs

- Each Tutorial / lab slot is limited in capacity.
  - First – come – first serve.
  - Tutorials/labs are not mandatory attend (but highly recommended).
- Register by joining a tutorial group in MyCourses under Groups -> Tutorials/Labs slots.
  - You will not be able to switch groups once you join a group.
  - If due to unforeseen circumstances you have to move to a new tutorial group, email the TA responsible for the new group. They will let you in if they still have the capacity.

TAs will be sending the zoom info to respective tutorial groups before the beginning of the tutorials.

	Day	Time	TA Name	TA email
1	Monday	5:00-6:00pm	Anirudha Jitani	anirudha.jitani@mail.mcgill.ca
2	Tuesday	5:00-6:00pm	Anirudha Jitani	anirudha.jitani@mail.mcgill.ca
3	Wednesday	10:00-11:00am	Ayrin Ahia-Tabibi	airin.ahia-tabibi@mail.mcgill.ca
4	Wednesday	9:00-10:00pm	Ayrin Ahia-Tabibi	airin.ahia-tabibi@mail.mcgill.ca
5	Thursday	4:30-5:30pm	Ben Yu	tzu-yang.yu@mail.mcgill.ca
6	Friday	4:30-5:30pm	Ben Yu	tzu-yang.yu@mail.mcgill.ca

Grading Scheme

- Assignments – 32%
  - 6-7 Assignments
  - Couple of assignments will have 2x weight compared to others.
    - Will be conveyed in advance.
  - Last one or two assignments might be in pairs (TBD).

Grading: All software solutions for assignments must compile with zero errors and must run to be graded. It does not need to run correctly for grading, but it must run. If your program compiles with errors or does not run at all then you will receive zero points. The grader will not fix your code or look at the source code to give you partial grades.

Grading Scheme

- Assignments – 32%
- Midterm – 15%
  - 90 minutes timed exam (MyCourses Quiz). – might make it shorter.
  - 5 hours to finish from whenever you start the exam (or slightly less if less than 90 minutes duration).
  - To be taken during a 24 hours window.
- Final Exam – 53%
  - 3 hours timed exam (MyCourses)
  - 9.5 hours to finish from whenever you start the exam.
  - To be taken during a 72 hours window.

Compliant with Faculty of Science regulations

OSD allowance and exam conflicts are already included in the exam duration

Grading Scheme

- Exams can cover topics from lecture slides, labs, assignments (make sure to check the solutions), and concepts and techniques that I discussed and demonstrated in the class.
  - Do not skip lecture recordings.
  - Write your own notes, especially those “tips & tricks” I talk/demo in class.
- Some assignments can contain an small “exploratory” element to encourage students to learn new material on your own.

Changes to Grading Scheme

In the event of extraordinary circumstances beyond the University’s control, the evaluation scheme in a Course is subject to change, provided that there be timely communications to the students regarding the change.

Re-grading

- Mistakes can occur when grading. You are responsible to check your assignment feedback and get back to your TA in a timely fashion. Be very specific about your concern, not the “can you check my assignment all over again” type of requests.
- Keep in mind that the TA / instructor reserve the right to regrade the entire submission and not just a specific question.

Additional Work

Students with grades of D, F or J will not be given the opportunity to complete additional work to upgrade their grade.

Supplemental Exam

- When available, covers only the Midterm + Final Exam Weight (68 %)
  - Points lost in assignment are lost for good. They are testing your practical skills.
  - Do not skip assignments !

Submission & Late Policy

- All Date / Time indicated in the course deliverables are as per EST (Montreal).
- You will be notified in advance of the assignment due dates. All assignments are due on My Courses at the indicated time and date.
- Late assignments will lose 5% of its grade per day late (rounded up).
- Assignments beyond 2 days late will not be accepted. You may not submit assignments via e-mail without the prior permission from the instructor.
- Do not wait till last minute for turning in the submission – Internet slow, laptop slow, etc., are not valid excuses.
- If your assignment description requires you to turn in actual source code (not screenshots), no points will be given to screenshots of code. Learn how to move code from the SOCS servers to your laptop and vice versa as part of your initial labs / assignment.

Late Policy

- Exemptions are made only under extreme circumstances.
  - Illness, Family emergency etc.
- The following are some example of “not emergency”.
  - Laptop crashed (keep regular backups on McGill OneDrive, google drive, etc.) – besides most of your code should be in McGill SOCS servers – a requirement for assignments to work.
  - Other exams, family events, interviews, extra-curricular events – These are planned events – plan your work accordingly.
- In some valid cases, instructor may chose at their discretion to shift the weight of your missed deliverable to something comparable that does not compromise the evaluation objectives of the course.

English / French

In accord with McGill University’s Charter of Students’ Rights, students in this course have the right to submit in English or in French any written work that is to be graded.

How do I perform well in this course?

- Attend lectures / watch recordings diligently.
- Try out the ”hands-on” activities from lectures by yourself. DO NOT study by “reading slides” alone.
  - Try variations of what was “demoed” in class/labs.
- Try to do lab exercises on your own first. Attend tutorials – maybe your TA has a better approach.
- Browse through piazza – read questions and solutions.
  - Can you try finding solutions? Sometimes the best way to learn is by finding solutions to others’ problems.
- Start working on assignments early enough so that you have time to think about the solutions, debug them, talk to TAs, etc.
- Do not skip assignments – In many cases you will have to rely on the skills learned from the previous assignments to effectively tackle the next assignments (can sometimes reuse code snippets).
- Try exploring the recommended topics listed for advanced students.
- See how you can adopt some of the concepts you are learning into your day-to-day programming life (other courses, internships, etc.) – especially things like automation, code-reuse, version controls, code repository to work in a group, debugging skills, etc.

Also

- Please check the course announcements and McGill emails regularly.
- Instructor and TAs usually send out a lot of useful and important information through those medium regularly.
  - You are responsible to keep track of the information being sent to you – Responses to individual queries over information already disseminated may not be fast.

Student Interactions

- Primary source for discussions (monitored by the instructor and TAs)
  - Piazza https://piazza.com/mcgill.ca/fall2020/comp206/home

- Student run initiatives ?
  - Facebook groups
  - Slack
  - Discord
  - …
  - If you start any of these, post your initiatives in piazza.
  - Form study groups with your classmates:
    - Find other students from your locality / time zone.
    - Keep it under 4 to avoid chaos and to be productive.
    - Use technologies such as zoom to interact.

- Student created question banks?

  Create questions and solutions – practice lecture topics with each other.

Academic Integrity

- McGill University values academic integrity. Therefore all students must understand the meaning and consequences of cheating, plagiarism, and other academic offenses under the Code of Student Conduct and Disciplinary Procedures.
- See www.mcgill.ca/integrity/ for more information.

Plagiarism

- Discussion of concepts and approaches between students is acceptable.
- Sharing of code between students is forbidden (other than when team work is explicitly mentioned).
- Reusing the entire code from the public domain is not acceptable. Whenever parts of code is borrowed from the public domain, include the source (e.g. URL) in the comments of the code (citation).
- You must be able to explain what your code is to the instructor / TA if we ask you to. We reserve the right to do so if we suspect plagiarism from the public domain.
- Software mechanisms maybe used to detect plagiarism cases.
- If the instructor suspects plagiarism after the manual verification of students deliverables, it will be reported to the disciplinary officer of the student's Faculty.
  - Your final course grades may be delayed till the officer makes a decision.
- Your are not allowed to share your assignment code with other students or put them in public domain.
- You may reuse the code provided to you by the instructor and TAs in lecture slides, assignment solutions (including your own), labs, text book, etc.

Copyright

- All the materials provided to you for lectures, assignments, labs, etc. are copyright of the respective person who publishes it.
- Students are not allowed to pass these materials to anyone outside of the class, including posting to third party websites.
- Students are allowed to retain such materials for their personal use, including in their cloud storage, private repositories, etc.
- Violators can be subject to legal and University disciplinary procedures.