



Course Name: Introduction to Software Systems
COMP-206 Fall 2017

Instructor: Joseph Vybihal

Contact Information: Office: ENGMC 323
Office Hour: TBD or by appointment
Email: jvybihal@cs.mcgill.ca

Course Objectives:

- (1) COMP-206 is a 3-credit full semester course in Software Development under the UNIX environment. It is offered in both the fall and winter semesters, and is a required course for students in many of our degree programs. It provides a comprehensive introduction to and overview of the C programming language and how to use it with the UNIX environment to build software.
- (2) This is not only a course on how to program in C and use Unix. This course focuses on systems. This relates to how differing software, languages and environments can be integrated to work together. In this light the course also teaches programming in Bash and systems concepts.
- (3) COMP 206 sets the stage for follow-on courses, COMP-273, COMP-307 and COMP-310, plus others.
- (4) This course also gives the student basic software management skills in the form of the GNU tool set and code management techniques.

Course Description: Comprehensive overview of programming in C, use of system calls and libraries, debugging and testing of code; use of development tools like make and version control systems.

Texts: **Primary Text:**

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

There is a lot of material for this course that can be found on line. In the class slides I will be recommending various online resources that you may find helpful.

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

Evaluation:

Assignments	30%	6 Mini-Assignments
2 class tests	20%	See Course Schedule (10% each – 1 hour long)
Final Exam	50%	Offered during the regular exam period

You may use your final exam to replace you class test grades (70%).
The final exam will cover the entire course.

Tutorials: 1 Session (final exam)

Labs: 4 Sessions (Unix, Bash, C)

Late work: You will be notified in advance of 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. Assignments beyond 2 days late will not be accepted. You may not submit assignments via e-mail without the permission of the instructor.

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: There will be no supplemental exam for this course.

Re-grading: Mistakes can occur when grading. Not surprisingly, requests for re-grading always involve those mistakes in which the student received fewer points than they deserved, rather than more points than they deserved. With that in mind: if you wish me to re-grade a question on an exam or assignment, I will do so. I reserve the right to re-grade other questions as well.

Cheating/Collaboration: Collaboration is encouraged but your discussions should be public in the sense that anyone including the professor should be allowed to listen in. Assignments are original works created by the student alone. You are permitted and encouraged to have conversations with other students concerning the contents of the assignments and how to do them, but your work must be original. If two or more assignments are found to be identical (or portions of assignments) then all parties will lose points. This includes the student who permitted their assignment to be copied. This includes written solutions and software source code.

Grading: All software solutions 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.

Communication

My Courses: All official communication, including announcements, lecture material, assignments, grades will be found on My Courses.

Email: This is best for private communication. I reply to email within 24 hours.

Facebook: This course has a Facebook group. I try to reply promptly, and I am often available at “odd” hours. You can “tag” me on FaceBook.

Appointments: Please feel free to come to my office at any time. If I am busy I will set a time to meet you later. Even better, email or Facebook message me for an appointment.

Office Hours: I have posted office hours. Students can come to those times without appointment.

After lecture: Some optional time will be available just after class to ask questions. I do not guarantee the length of this time since other constraints may interfere.

Course Schedule

WEEK	TOPIC	READINGS	WORK
Unit 1 – INTRODUCTION			
1	(A) INTRODUCTION Introduction to the course, about systems programming, about software systems, about command-line development	Chapter 1	
Unit 2 - THE UNIX ENVIRONMENT			
1	(B) OPERATING SYSTEMS Why do we need an operating system? The story of Unix and Linux. Architecture of the Unix OS. Getting access to the SOCS Unix servers. SFTP clients and logging in, file transfer, working home/school. (C) THE SHELL The shell environment, the command-line prompt, home vs root directory, Basic commands: ls, cd, mkdir, rmdir, cp, mv, rm, cat, more, man, logout, paths.	Chapter 2	
2	(A) REGULAR EXPRESSION/WILD CARDS Command line commands using wild cards, using regular expressions, using indirection, Grep. (B) SESSIONS Session architecture, ssh, ampersand operator, session memory definition and usage, ps, kill, pwd, the login and logout scripts, importance of passwords. (C) VIM & DEVELOPER TECHNIQUES Unix editors, non-GUI editors (why?), common directory structures and procedures, examples, chmod, tar, backups	Links in slides	- Lab A – Using UNIX - Assignment #1 given out
Unit 3 – BASH PROGRAMMING			
3	(A) INTRO TO BASH SCRIPTING What is Bash programming? When to use Bash programming? Examples of simple Bash programs. Introduction to basic structure and using it with the command-line. (B) EXPRESSIONS Variables, session variables, command-line variables, math, and I/O, examples (C) CONTROL STRUCTURES Conditions, conditional control and Iteration, examples	Links in slides	- Lab B – Using Bash - Assignment #2 given out
4	(A) FUNCTIONS Functions in Bash, scope, return values, examples. (B) BASH DEVELOPER TECHNIQUES Using Bash as an aid to the developer, examples: development environment setup, archives, backups. TAR, ZIP. (C) SYSTEM SCRIPTS Customizing the shell and session. Different system/session scripts.		
5	(A) CLASS TEST		Class Test #1
Unit 4 – C PROGRAMMING			
5	(B) INTRODUCTION TO C The story of C. Why C? Children of C. Hello World example with puts() and getc(), GCC basics, compiling, running, errors, bugs/unix-more cmd	Chapter 3	Lab C – Programming in C

	(C) C CONTROL STRUCTURES If, for, while, do..while, switch, examples, types, variables, mathematical and logical expressions		
6	(A) BASIC STDIOH getchar, putchar, puts, printf, scanf, sprintf, sscanf. Examples. I/O issues and data validation. STDIN, STDOUT and STDERR (B) POINTERS, STRINGS & STRINGH Learn basic string.h functions then see how to make them using pointer referencing. Define pointer referencing and dereferencing. (C) ARRAYS AND STRINGS Mult-D Array syntax & usage, strings, static & invariant data, writable data, array addressing, examples.	Chapter 3	Assignment #3 given out
7	(A) FUNCTIONS & SCOPE Function syntax & usage, scope rules. Examples. Call-by-value and call-by-reference. (B) SEQUENTIAL TEXT FILES The file concept. Fopen(), fclose(), fprintf(), fscanf(), feof(), examples. Text files. (C) STRUCT AND UNION Struct and Union syntax. Examples. Array of struct.	Links in slides	Assignment #4 given out
8	(A) DYNAMIC MEMORY Dynamic arrays, dynamic structs, linked lists.		
Unit 5 – BASIC SOFTWARE ENGINEERING TECHNIQUES			
8	(B) MODULAR PROGRAMMING C object files, compiler performance, team programming basics, about large projects. The extern expression. The Pre-processor (C) GNU TOOLS What is a makefile? What is a profiler? How to use them.	Chapter 4	
9	(A) REPOSITORIES & More GNU Tools What is a repository? Ways to use repositories. Using git. More team programming basics. Branching. About GDB and how to use it. (B) CLASS TEST	Links in slides	Class Test #2
Unit 6 - SYSTEMS PROGRAMMING			
9	(C) INTRODUCTION TO SYSTEMS PROG. Machines, time.h, Bit-wise operations. Bash to C parameter passing.		
10	(A) SEQUENTIAL, BLOCK, RANDOM FILES CSV, sequential Block, Random and Binary file manipulation (fread, fwrite, fseek) INTER PROCESS COMMUNICATION: (B) What is a process. C process creation: system and fork. Producer Consumer problem via fork and via & concurrent programs. (C) Shell memory based communication & concurrent programs.	Links in slides	Assignment #5 given out
11	ADVANCED: (A) Void * and function * (B) Socket communication (C) Other system calls	Links in slides	Assignment #6 given out
Unit 7 – If time permits			
12	(A) Signal handling (B) Assembler with C (C) FINAL EXAM REVIEW		

General Course Information

Course Requirements: The pre-requisite for this course is COMP-202 or COMP-250.

Right to submit in English or French written work that is to be graded

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.

Classroom Rules: All electronic devices (cell phones and beepers) must be turned off or left on silent mode during class time.

Assignments Pickup: All assignments are submitted to and picked-up from My Courses.

Computing Resources: Trottier 3rd floor.

Examinations and Grading:

Students are responsible for all materials for the tests and exams, whether it is covered in class. Exams will be a combination of all types of questions based on all sources, and students may be required to integrate theoretical concepts from the text to substantiate their arguments.

No make-up tests or make-up assignments are allowed in this course. A supplemental exam is possible for 50% of the grade (to replace your final exam).

If you are not satisfied with the grading of an assignment or mid-term test, you may request a review within 7 days of return. Indicate in writing or during a meeting with the instructor where and why you feel the marks are unjustified and give it back to your instructor for re-grading. Note that the entire assignment or mid-term test will be re-graded and your grade can go up or down (or stay the same) accordingly.

Calculators

Only non-programmable, no-tape, noiseless calculators are permitted. Calculators capable of storing text are not permitted in tests and examinations.

Dictionaries

Dictionaries are not permitted, but translation dictionaries are.

Handheld Devices

Handheld devices capable of storing text and having calculator functionality (e.g. Palm, etc.) are not permitted.

Additional Information: The course slides are not meant as a complete set of notes or a substitute for a textbook, but simply constitute the focus of the lecture. Important gaps are left in the slides that are filled in during class, thus lecture attendance should be considered essential.

The material covered in the classroom will be used to supplement textbook readings.

Academic Integrity: *Code of Student Conduct*

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

L'université McGill attache une haute importance à l'honnêteté académique. Il incombe par conséquent à tous les étudiants de comprendre ce que l'on entend par tricherie, plagiat et autres infractions académiques, ainsi que les conséquences que peuvent avoir de telles actions, selon le Code de conduite de l'étudiant et des procédures disciplinaires (pour de plus amples renseignements, veuillez consulter le site www.mcgill.ca/integrity).

Final Exam Policy:

Regulations

Students should not make other commitments during the final exam period. Vacation plans do not constitute valid grounds for the deferral or the rescheduling of examinations. See the Centre Calendar for the regulations governing Examinations:

<http://www.mcgill.ca/student-records/exams/regulations/>

Students are required to present their I.D. Card (with photo) for entrance to their examination.

Conflicts

If you are unable to write your final examination due to scheduling conflicts, you must submit a Final Exam Conflict Form with supporting documentation at least **one month** before the start of the final examination period. Late submissions will not be accepted. For details, see

<http://www.mcgill.ca/student-records/exams/conflicts/>

Exam Timetable

Examination schedules are posted at the Centre and on the following page approximately 6-8 weeks before the examination period commences

<http://www.mcgill.ca/student-records/exams/>

The Centre cannot provide examination dates over the telephone.

Email Policy:

E-mail is one of the official means of communication between McGill University and its students. As with all official University communications, it is the student's responsibility to ensure that time-critical e-mail is accessed, read, and acted upon in a timely fashion. If a student chooses to forward University e-mail to another e-mail mailbox, it is that student's responsibility to ensure that the alternate account is viable.

Please note that to protect the privacy of the students, the University will only reply to the students on their McGill e-mail account.

Students Rights and Responsibilities:

Regulations and policies governing students at McGill University can be downloaded from the website:

<http://www.mcgill.ca/deanofstudents/rights/>

Students Services and Resources:

Various services and resources, such as email access, walksafe, library access, etc., are available to students:

<http://www.mcgill.ca/student-records>

Minerva for Students: <http://www.mcgill.ca/minerva-students/>

Note: In the event of extraordinary circumstances beyond the University's control, the content and/or evaluation scheme in this course is subject to change.