McGill School of Computer Science
www.cs.mcgill.ca

Course Outline

Course Name: Introduction to Computer Systems
COMP-273 Winter 2018

Instructor: Joseph Vybihal

Contact Information:
Office: ENGMC 323
Office Hours: TBD in ENGMC 323 or by appointment
Email: jvybihal@cs.mcgill.ca
Website: MyCourses

Course Objectives: This course covers two major topics in computer systems: assembly language programming and the architecture of a CPU. Programmers are normally familiar with the computer only from a very high level. This is understood as manipulating the computer using an object-oriented programming language. They are not aware of the many abstraction layers that hide the underlying structure and functions carried out by a computer. The computer's Operating System and machine language manage the underlying mechanics. This course peals back these layers of abstraction allowing the student to directly manipulate and study the engineering of some of the most important components of a computer. Our method of study will be two pronged: First, we will look at the engineering of the computer's CPU, RAM and topics about the system-board. Last, we will learn the MIPS assembly language, as a tool to interact with the internal components of the computer. We will use MARS as our MIPS emulator and LOGISIM as our circuit emulator.

Primary learning outcomes: to get a clear understanding of how the CPU and RAM are built and function from the microcircuit and functional levels. To then use an actual assembly language that uses the microcircuits discussed in class.

Secondary learning outcome: to demystify the abstraction layers in a computer system and to take advantage of the new-found capabilities. The student will be able to intelligently discuss the internal workings of computers and produce solutions that take advantage of the built-in hardware of a machine. Discussions on robotics and modems may also be addressed.

Course Description: Number representations, combinatorial and sequential digital circuits, MIPS instructions and architecture data-paths and control, caches, virtual memory, interrupts and exceptions, pipelining.

Texts: COMPUTER ORGANIZATION & DESIGN: The Hardware / Software Interface
Author: Patterson & Hennessy, Publisher: Morgan Kaufmann, ISBN: 1558604286

- Hard copy: $ 89.95 US
- Zybooks : $ 72.00 US (cannot save copies on disk, but can screenshot)

Texts are available on two-hour reserve in the Schulich Library. Call Numbers can be found from the McGill library website (see MUSE, Course Reserves).

Optional Texts:

(1) "See MIPS run", by D. Sweetman

(2) The Soul of a New Machine
Author: Tracy Kidder, Publisher: Avon Books, ISBN: 0-380-59931-7

Evaluation:

Assignments	30%	6 Assignments (5% each)
Class Tests 1	10%	TBD
Class Test 2	10%	TBD
Final Exam	50% or 100%	During regular final exam period

You may use your final exam to replace you class test grades (100%).
The final exam will cover the entire course.

Tutorial: 3 Sessions (1 LOGISIM, 1 MARS, 1 final exam)

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.

Co-Requisites

* COMP 250 Introduction to Computer Science (unofficial, but strongly recommended)
* COMP 206 Introduction to Software Systems (official)

If you have not taken 206 or you are not taking it currently, then you should not take COMP 273, unless you have previous experience with C or C++ programming.

Communication

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

Facebook: This course has a Facebook group. It is my preferred way to communicate with the class. I try to reply promptly, and I am often available at "all" hours. I often reply to Facebook group and private messages quickly.

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.

Tentative Course Outline

Tentative Schedule			
WEEK	**TOPIC**	**READINGS**	**WORK HANDED OUT**
Unit 1 – BASICS			
1	(A) INTRODUCTION Introduction to the course, about CPUs and assembler development, performance issues, the tools we will use in this course, famous names (B) THE SYSTEM BOARD The computer, the system board and its parts, the CPU fetch cycle, binary/bytes		
2	(A) NUMBER SYSTEMS & MEMORY FORMATS Binary, hex, octal, math operations, conversions, basic data storage, basic data types, strings & arrays, IEEE floating point		Ass#1 – Numbers & Memory
Unit 2 – CIRCUITS			
2	(B) INTRO TO CIRCUITS AND FLOW Gates, wires, truth tables, truth tables to circuits, combinatorial circuits, ROM, Clock		
3	(A) RAM High level definition, low level definition, the flip-flop, addressing, bytes, example data storage, 2-nibble RAM example with full bus, D-flip-flop (B) REGISTERS High level definition, low level definition, gate control, RAM/Register/Bus example, external gate control example (using RAM/Register/Bus)		LOGISIM tutorial Ass#2 – Basic Circuits
4	(A) ALU High level definition, low level definition, adder circuitry, subtraction circuitry, status register, multiplication and division (B) CLASS TEST		Class Test W1-4
Unit 3 – MICRO INSTRUCTIONS			
5	(A) WRITING MICRO INSTRUCTIONS Macro instructions are made from multiple micro instruction steps, writing syntax, examples with LW, SW, ADD, CMP, BEQ, J (B) WRITING MICRO CIRCUITRY Designing circuits that implement LW, SW, ADD, CMP, BEQ, J		Ass#3 – Circuits & Micro instructions 1
6	(A) THE CONTROL UNIT High level definition, low level definition, controlling the micro instructions, examples connecting to previous lectures, the IR and instruction formats		
Unit 4 – CPU & PERFORMANCE			
6	(B) CLASSICAL CPU High level definition, instruction execution flow, performance calculations		
7	(A) PIPELINE CPU High level definition, instruction execution flow, performance calculations, faults		Ass#4 – Performance & Micro instructions 2
Unit 5 – MIPS ASSEMBLER PROGRAMMING			
7	(B) THE MIPS CPU Registers, addressing modes, instruction syntax, writing		

	a program, compiling a program, example hello world program, OS Interrupts introduction, co-processors		
8	(A) CLASS TEST (B) DATA & MATHEMATICS PROGRAMMING Defining simple data, data accessing instructions, basic mathematical instructions, simple I/O, example programs		Class Test: W5-7 MARS tutorial
9	(A) CONDITIONS & ITERATION How to express a condition, a simple if-statement example, how to express a loop, a simple while-loop example, complex array conditions, example printing characters from a string (B) SUBROUTINES & FUNCTIONS Definition of a subroutine, definition of a function, examples, MIPS and ANSI parameter passing, the run-time stack, saving variables		Ass#5 – Basic MIPS programming
10	(A) More in class programming examples and problems		
Unit 6 – ADVANCED TOPICS			
10	(B) CACHE Basic architecture, modulo addressing, how it is used, performance, hit and miss, associative caches		
11	(A) EXTERNAL DEVICE I/O Polling/Interrupts/DMI: High level definitions, performance, some example MIPS code (B) VIRTUAL MEMORY Virtual vs real memory, page tables, OS run-time loop		Ass#6 – MIPS programming & performance
Unit 7 – If time permits			
12	(A) MULTI-CORE CPU Strategies, control flow, queues, the OS involvement, multi-threading, the programmer involvement (B) FINAL EXAM REVEW Talk about the final exam and what will be on it. Undergraduate research opportunities & about grad school		Tutorial for Final exam

General Course Information

Examination and Grading:

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.

Students are responsible for all materials for the tests and exams, whether or not 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.

Course Requirements: COMP-206 is a co-requisite for this course.

If you already know how to program in C or C++ or received a very good grade in COMP 250 then you can probably take this course without the course requisite.

Additional Information: COMP-308 Computer Systems Lab in a continuation course to COMP-273. It is a hands-on assembler and C lab course that will teach students how to interface with actual computer hardware using low-level programming techniques. This may be of interest to you if you would like to continue learning how to interact at low levels.

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 **l**ecture 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 reseignements, veuillez consulter le site *www.mcgill.ca/integrity*).

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.

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.

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.

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.

Computing Resources: Trottier 3rd floor.

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/stundet-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.