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

Course Outline

Course Name: Introduction to Computer Systems
COMP-273 Winter 2024

Course Time: Tuesday / Thursday 16:00 – 17:30, ADAMS AUD

Instructors: Joseph Vybihal

Office	ENGMC 323
Office Hour	M & Th 10:00 – 11:00
Email	Joseph.vybihal@mcgill.ca
Home page	www.cs.mcgill.ca/~jvybihal
Discussions	ED
Course	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, 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, like the operating system, the machine language, and the underlying physical mechanics. This course peels 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 learn the MIPS assembly language, as a tool to interact with the internal components of the computer. Last, we will look at the engineering of the computer's CPU, RAM and various topics about the system-board. 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 use an assembly language to interact directly with the hardware of a computer.

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.

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: **Primary Texts:**

- COMPUTER ORGANIZATION & DESIGN 6th Edition, Patterson & Hennessy, **zyBook**. Link here:
    - 1. Sign in or create an account at http://learn.zybooks.com
    - 2. Enter zyBook code: **MCGILLCOMP273VybihalSpring2024**
    - 3. Subscribe
    - A subscription is $97. Students may begin subscribing on Dec 21, 2023 and the cutoff to subscribe is Apr 15, 2024. Subscriptions will last until May 14, 2024.

- COURSE NOTES posted on myCourses
  (written by Prof. Michael Langer & Prof. Joseph Vybihal)

Other texts:

- "See MIPS run", by D. Sweetman

Bonus material:

(1) The Man Who Invented the Computer, Double Day, ISBN 978-0-385-52713-2
(2) The Soul of a New Machine, Avon Books, ISBN: 0-380-59931-7

Evaluation:

Mini Assignments	30%	6 Assignments (5% each)
Project	20%	April 14 (23:59)
Class Test 1	20%	February 8 (online, evening)
Class Test 2	20%	March 12 (online, evening)
Class Test 3	10%	April 4 (online, **in person**)

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.

Submission: You will be notified in advance of assignment due dates. All assignments are due on MyCourses at the indicated time and date.

Late work: Late assignments will lose **10%** of its grade per day late. No work will be accepted after **two** days (i.e., a mark of zero will be assigned). Late work will be submitted on MyCourses. Do not submit assignments via e-mail.

Penalty Waiver: The late penalty can be waived once during the term. Please use this for any sickness, time management, or any other personal reason. Simply include a "penalty-free.txt" with your submission, and there is no need to ask 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: Check with the university for this opportunity.

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 the TA to re-grade a question on an exam or assignment, we reserve the right to re-grade other questions as well. **You must ask for a regrade within 7 days from the date the TA returned your grade.**

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 **assignment questions** and how to do them, but your work must be original and **no discussion about your work** can occur with other students or online resources. 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, software source code, and online resources. The students will be reported to the university plagiarism department.

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

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

Course Discussions: The online free tool, ED, is used as our course discussion board. Use this as your primary communication medium, since your questions are public and can help other students.

Private Email: The professor and TA have private email accounts that you may also use, however these communication channels are for personal queries. For example: if you have a problem with your grade then email the TA who graded you directly, do not email the prof and do not use the course email address.

Appointments: Please email me directly to book at appointment outside office hours.

Office Hours: See the hours posted in myCourses/Content/Course Information..

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.

```
CommunicationAlgorithm() :
        if (public) ED(); // all will benefit
        else if (about marks) emailTAPrivately();
        else if (medical or special) emailProfPrivately();
        else privateED();
```

Teaching Assistants

The teaching assistants for this course are divided into two types: tutorial TAs and office hour TAs. A tutorial TA does not have office hours but will hold labs that you can attend. The office hour TAs will be on ED and will have regular office hours (either in-person or online). The Instructor and TA Coordinates is on myCourses/Content/Course Information.

Each student is assigned a single TA who will be “their” grading TA for the entire course. This TA will be responsible for grading your assignments and tests.

Regardless, you can attend any TA office hour for help.

The full details about all the TA s of this course can be found in myCourses.

Tutorials are optional. You can attend any tutorial section.

You will find the list of TAs in our myCourses course page. Go to **Content/Course Information/TA & Prof Coordinates** link. You will also find **grading schedules** and **office hours** information at that same link.

Tentative Course Schedule

Lecture	TOPIC	READINGS	WORK HANDED OUT
Unit 1 – BASICS		zyBook Ch 1 & 2	
1 **Jan 4**	INTRODUCTION TO COMPUTER SYSTEMS Introduction to the course. Overview of the computer and programs. Why is this course important? Where do we use assembler? Where is this useful?		
2 **Jan 9**	NUMBER REPRESENTATION How information is represented in a machine. Bits/bytes, binary, hex, octal, base conversion, signed numbers. What is RAM?	binary-representations.pdf + twos-complement.pdf	
3 **Jan 11**	FLOATING POINT and STRING vs ARRAYS Strings & arrays, IEEE floating point. What is a CPU? How is this connected to the CPU?	floating-points-representations.pdf	Ass 1 – Numbers Representations, base conversions, data representation
Unit 2 – MIPS ASSEMBLER PROGRAMMING		zyBook Ch 3 - 6	
4 **Jan 16**	INTRODUCTION to MIPS Instruction execution flow, registers, addressing modes, instruction syntax, writing a program, compiling a program, I/O, example hello world program, defining simple data, basic mathematical instructions.	intro-to-mips.pdf	Tutorial A – Introduction to MARS with "hello world"
5 **Jan 18**	CONDITIONS & ITERATION if-else statements, loops, equality and inequalities, example printing characters from a string. Arrays.	mips-branch.pdf mips-arrays.pdf	Ass 1 – Due Ass 2 – Basic MIPS programming
6 **Jan 23**	SUBROUTINES & FUNCTIONS Definition of a subroutine, definition of a function, examples, the run-time stack, saving variables.	mips-functions.pdf	Tutorial B – Introduction to MIPS programming
7 **Jan 25**	LOGIC BIT OPERATORS Bitwise logical operation, masks, shift operators. Why is this important? Controller card architecture.	mips-branch.pdf	Ass 2 – Due Ass 3 – Advanced MIPS programming
8 **Jan 30**	EXTERNAL DEVICE IO-OPERATIONS Polling I/O in MIPS, examples. Polling vs Interrupts vs DMI: High level definitions, performance.	IO.pdf	Tutorial C – Introduction to MIPS functions
Unit 3 – CIRCUITS and SYSTEM BOARD		zyBook Ch 7 & 8	
9 **Feb 1**	EXECUTION ENVIRONMENT Compiler, Assembler, Linker, Loader, OS interface, memory protection, the system board, CPU, RAM (primary storage) vs Drive (secondary storage). Machine language & wires.	Environment.pdf	Ass 3 – Due Ass 4 – MIPS IO & Boolean algebra
10 **Feb 6**	INTRO TO BOOLEAN ALGEBRA & CIRCUITS Truth tables, Boolean Algebra, ~~K-map~~, Don't care condition. Basic Circuit elements. Introduction to Logisim.	Truth-tables.pdf	

11 **Feb 8**	COMBINATORIAL CIRCUITS Combinatorial circuits. Common circuits. Examples.	Circuits.pdf	**CLASS TEST #1 lectures 1-9 (evening)**
12 **Feb 13**	HOW TO BUILD CIRCUITS More examples. Using Logisim. Bus. Indexing. Error codes. Routing. Built-in Logisim objects (input pin, output pin).		Tutorial D - Logisim
Unit 4 – THE CLASSICAL CPU			
13 **Feb 15**	THE CLASSICAL CPU & REGISERTS CPU Types, Classical CPU architecture, register architecture, gate control (addressing (register box) vs triggers (register bus)). Registers in Logisim.	Classical.pdf	Ass 4 – Due Ass 5 – RAM
14 **Feb 20**	SYSTEM BOARD RAM & U-BUS Architecture of RAM and U-Bus. RAM/Register/Bus example. Performance calculations. Faults (part 1).	SystemRAM.pdf	Tutorial E – Registers, RAM and U-Bus
15 **Feb 22**	ALU Architecture, signed numbers, adder circuitry, subtraction circuitry (complement), status register (logical operations), multiplication and division as a form of adding and subtracting loop.	ALU.pdf	Ass 5 – Due Ass 6 – ALU
16 **Feb 27**	THE CONTROL UNIT + MICRO INSTRUCTIONS CU architecture, IR architecture and instruction formats. Macro instructions are made from multiple micro instruction steps, microinstruction notation, example notation with LW, SW, ADD, CMP, BEQ, J..	CU.pdf instructions-representations.pdf	Tutorial F – Introduction to MICRO circuitry writing.
17 **Feb 29**	BUILDING MICRO CIRCUITRY + COMPLETE CPU Designing circuits that implement LW, SW, ADD, CMP, BEQ, J. The complete Classical CPU architecture review		
Mar 4 – **Mar 8**	**READING BREAK**		
18 **Mar 12**	PERFORMANCE & FAULTS Exploring the computer environment through math and logic problems.	Performance.pdf	**CLASS TEST #2 lectures 10-17 (evening)**
Unit 5 – THE PIPELINE CPU		zyBook Ch 9-11	
19 **Mar 14**	THE PIPELINE CPU Pipeline architecture, instruction execution flow, performance calculations. Faults (part 2).	Pipeline.pdf	Ass 6 - Due **Project** – Building a Classic CPU
20 **Mar 19**	PIPELINE CACHE Why cache? Basic architecture, modulo addressing, performance, hit and miss, associative caches. How is the cache loaded (dumb vs predictive)?	Cache.pdf	Tutorial G – Introduction to building control units.

21 **Mar 21**	PIPELINE MICRO CIRCUITRY & TIMING Designing circuits that implement LW, SW, ADD, CMP, BEQ, J in a pipeline CPU. The complete Pipeline CPU architecture review.	Timing.pdf	Tutorial H: CPU faults and performance
Unit 6 – ADVANCED TOPICS		zyBook Ch 12 & 13.9	
22 **Mar 26**	VIRTUAL MEMORY Virtual vs real memory, page tables, OS run-time loop, TLB Cache.	VM.pdf	
23 **Mar 28**	MULTI-CORE CPU Strategies, control flow, OS queues, the OS involvement, multi-threading, the programmer involvement.	Cores.pdf	
24 **Apr 2**	MULTI-CORE CPU MEMORY Cache issues. Accessing RAM. Cache architectures.	CoreMemory.pdf	
25 **Apr 4**	(Lecture 25) **CLASS TEST #3 – Pipeline & Advanced Topics**		**CLASS TEST #3 lectures 18-24**
26 **Apr 14**	(Lecture 26) Project Submission		**Project Due**

General Course Information

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.

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.

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

Examination 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.

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: 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.

Other related courses are:

- COMP 310 Operating Systems: control and scheduling of large information processing systems.
- COMP-520 Compiler Design: structure of the compiler and how high-level languages are translated into assembly.
- COMP-596 Principles of Computer Systems: Topic course on computer systems
- COMP-764 High-level Synthesis of Digital Systems: research topics on computer systems

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 reseignements, 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/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.