⪯ COMP 220 Computer Organization & Assembly Language ⪰

M-W-F	*Lecture (DC 1315)* – 11:30-12:20
W	*Lab (DC 1315)* – 3:30-5:20

`(Some assembly required)`

Who: Michael Gousie
Where: Discovery Center 1325
When: Mon 1:30-3:00, Tue 11:00-12:30; Fri 2:00-3:00 (unless meeting)
and by appointment
E-mail: mgousie(at)wheatoncollege(dot)edu
Web: http://cs.wheatoncollege.edu/mgousie

Content:

- Are computer processors getting faster?
- If x and y are floating point values, why shouldn't a programmer write `if (x == y)`?
- How does a program written in C++ ultimately run using just zeroes and ones?

These are some of the questions to be answered in this course. We will explore the basic workings of computer hardware and how different architectures affect performance. We will cover how numbers, especially floating point values, are represented in the computer processor (CPU). An important theme of the course is to understand how data flows through the CPU, and how that is connected to the software. In this vein, we will cover basic circuitry using a virtual circuit board and write programs in MIPS, a common assembly language. Weekly labs put lecture concepts to immediate use in designing parts of the hardware (such as an adder) by creating and experimenting with circuits built with a virtual circuit board and writing programs in MIPS. By combining architecture and programming, the relationship between the hardware and software will be made more obvious. This, in turn, will give you an understanding of how the way you write programs in high-level languages affects performance. Thinking like this is another aspect of computer science, or more broadly, Computational Thinking.

Required Text:

Patterson and Hennessy. *Computer Organization and Design: The Hardware/Software Interface; MIPS Edition*, 6th Edition (Morgan Kaufmann, 2021).

Recommended Text:

Matthews, Newhall, and Webb. *Dive into Systems* (Published online, updated 2022). This is available through a link on the course web page.

Requirements:

There will be 2 exams during the semester and a comprehensive final exam. The exams will comprise 50% of your grade. Exams will take place during the scheduled afternoon lab time so as to afford you more time.

We will also cover some assembly language programming, for which you will complete three small projects, the first two worth 6% each, and the third worth 10%. The course text and/or *The Google* is a valuable reference for these programs. Due dates for these projects are shown below.

Five or six written homework assignments, roughly one week in length, will comprise 20% of your grade. Due dates for all homeworks will be announced in class. Note that some labs may have a graded component that will be added to the homework or project that is next due.

The remaining 8% of your grade will be a project comprised of a physical model of some [portion of] computer hardware **or** a working set of virtual circuits that model a part of a simplified CPU. The project also entails creating a poster and giving a presentation during the *2025 Comp Org Crazy Model Expo*. A portion of the grade will be determined by your peers. More details about the project will be given as the due date nears.

Grading:

Grades will be assigned according to the following scale:

A = 93-100, A− = 90-92, B+ = 87-89, B = 83-86, B− = 80-82, C+ = 77-79, etc.

Exam Schedule:

Exam	**Weight**	**Date**	**Time**
Exam 1	15%	February 19	3:30-5:20
Exam 2	15%	April 2	3:30-5:20
Final	20%	Tuesday, May 6	9:00-12:00

Programming Assignments:

Program	**Weight**	**Topic** **(Subject to change)**	**Due Date**
MIPS 1	6%	Conditionals, loops, arrays	February 23
MIPS 2	6%	Integer vs. floating point	April 6
MIPS 3	10%	Subroutines (functions)	April 24

Course Policies:

- You are responsible for all material covered in class, including the reading (shown below).
- You should bring a scientific calculator to class; your phone calculator may or may not have the necessary functionality, or at the very least, may not be very intuitive. Phones will not be permitted during exams, so a calculator is necessary in any case.
- You should bring your heavy book to class, especially when we cover Chapter 4 and/or when instructed to do so.

- If you must miss a quiz or exam for any reason, you must inform me **before** the test. Except in the case of emergency, illness, or you found aliens in Wheaton's original pool (Where is that?), makeup exams will not be given.

- Programs will be written in MIPS, a PC assembly language. The MARS simulator will be used to edit, translate (assemble), and run your MIPS code. MARS is available for free and is written in Java, so it works on all platforms.

- Written homeworks should be neat and done on loose-leaf or plain paper. Do not tear paper out of a notebook. Staple multiple pages together.

- Assignment due dates are **firm**.
  - All programming projects must be submitted electronically by 11:59:59 PM on the due date unless otherwise noted. Projects submitted on the following day will receive a 15% penalty. Anything turned in later will receive a 0. Hard copy must be submitted the following day or as indicated in the program specifications.
  - Written homework must be handed in at the **start** of class on the due date. There are **no** provisions for late homework.
  - Although deadlines are firm, please contact me **beforehand** if there are extraordinary circumstances. Life *can* get complicated.
  - Homework and projects may be turned in early! You can also resubmit projects any number of times before the deadline if you find an error in an earlier submission.
  - There will not be any individual "extra credit" work. If you did not have time to do a good job on the original assignment, how will you have time to do *additional* work?

- You are expected to adhere to the Honor Code.
  - Although *discussion* of projects or homework is encouraged, the final *implementation* of programs should be the result of your own work. Any copying of programs or homework is prohibited.
  - AI can help you speed up the programming process by having it do some of the more menial tasks. However, your program should still be **your own work**. Copy/paste is not the way to learn how to program, whether you are doing this from another person or from an AI application.

    If you are unsure where the line is between collaborating with AI and copying from AI, we recommend the following heuristics:
    - Never hit *Copy* within your conversation with an AI assistant (and then *Paste* into your code). You can copy your own work into your conversation, but do not copy anything from the conversation back into your assignment. Instead, use your interaction with the AI assistant as a learning experience, then let your assignment reflect your improved understanding.
    - Do not have your assignment and the AI agent itself open on your device at the same time. Similar to above, use your conversation with the AI as a learning experience, then close the interaction down, open your assignment, and let your assignment

reflect your revised knowledge. This heuristic includes avoiding using AI assistants that are directly integrated into your composition environment: just as you should not let a classmate write content or code directly into your submission, so also you should avoid using tools that directly add content to your submission.

  - If a program (or homework) looks suspicious, I may ask you to explain the purpose, function, and details of your code; if you can't, it will be considered plagiarized.
  - Collaboration on exams is prohibited.
  - You will be required to write and **sign** the pledge on all work turned in:

    I have abided by the Wheaton Honor Code in this work.
  - Any violation of the above guidelines will result in a 0 for the assignment/exam and/or a failing grade for the course.

- **The use of a laptop or other computer/pad which requires typing is not allowed during lecture.** Special arrangements can be made if necessary.
- The use of cell phones, iPods, iPads, iPhones, iPlops, iFlops, and other personal electronic devices is prohibited during class, lab, and exams.
- Please do not leave once class has begun.
- Accommodations for disabilities:

> *Wheaton College is committed to providing equitable access and supportive services for all students to fully access and thrive in the academic, residential and social aspects of student life. Affirmatively, Wheaton provides appropriate accommodations for eligible students with documented disabilities to afford equal access to educational programs and services. Individuals with disabilities and other access concerns requiring accommodations or information on accessibility should reach out to Accessibility Services at the Filene Center:*
>
> ∼ *accessibility@wheatoncollege.edu or (508) 286-3794* ∼

Course Schedule:

Wk #	Week **Begin**	Topic	Reading*	Lab
	January			
1	19	Introduction	Sections 1.1–1.5	TBD
2	26	Performance	Sections 1.6–1.11	Performance
	February			
3	2	Low level instructions, MIPS	Chapter 2	MARS and MIPS
4	9	MIPS	Appendix A	MIPS
5	16	Numbers, digital logic	Sections 2.4–2.7, Appendix B	Exam 1 (Weds)
6	23	Integer arithmetic	Sections 3.1–3.4	Digital logic
	March			
7	2	Floating point arithmetic	Sections 3.5–3.10	Circuit simulator
8	9	*SPRING BREAK*		Powder skiing
9	16	More MIPS	Appendix A	MIPS
	⇒3/19⇐	**No class! MAP Day!**		
10	23	The CPU	Sections 4.1–4.2	K&S Computer
11	30	Building a datapath	Sections 4.3–4.5	Exam 2 (Weds)
	April			
12	6	Pipelining	Sections 4.6–4.16	Circuit simulator
13	13	Cache memory	Sections 5.1–5.5	Circuit simulator
14	20	Virtual memory	Sections 5.6–5.16	Hardware
15	27	Parallelism	Selections, Ch. 6	Crazy Model Expo
	May			
16	4	Final Exam, May 6 @ 9 AM	Happy Summer!	

* Readings are from Patterson & Hennessy.