CS 472 Computer Architecture, Section SC1

Summer Semester 2020 / Offline via Zoom / Tuesday night 6-9:30 p.m.

Instructor: Dave Hendrickson
   Phone: (978) 659-2107 (leave a message)
   EMail: hendrick@bu.edu
   Office hours: via email

TA/Grader: Karen Palmer (kspalmer@bu.edu) She is amazing. If you have questions, email both of us.

Website: Blackboard (https://learn.bu.edu/) 20sum1metcs472sc1

Text: David A. Patterson and John L. Hennessy
   Computer Organization & Design: The Hardware/Software Interface
   Fifth Edition (ISBN 978-0-12-407726-3)
   Available at the BU bookstore or online. No other edition is acceptable!

Prerequisites: MET CS231 or CS232 **Do not ignore this prerequisite!** You need to be proficient in some language, comfortable handling arrays and classes of data. Some assembly language is helpful but not mandatory.

Description: Computer organization with emphasis on processors, memory, and input/output. Includes pipelining, ALUs, caches, virtual memory, parallelism, measuring performance, and basic operating systems concepts. Discussion of assembly language instruction sets and programming as well as internal representation of instructions. (In short, enable you to understand the internal workings of computers, what makes them fast, and what the primary design challenges are.)

Grades: There will be a midterm (30%) and a final (40%) which will together account for 70% of the grade. There will be also three programming projects (in the language of your choice) for the other 30%. Late submissions will be accepted with a 10% pro-rated penalty for each week late *except for the final project*.

   In general, an "A" will be awarded for work that totals 92-100% of the possible points, "A-" for 90-92%, "B+" for 88-90%, "B" for 82-88% and so on down to F for below 60%. Grades may be scaled upwards based on class scores (though rarely to get an A), but not downwards.

   Grades are YOUR responsibility. If you need a particular grade to get into the M.S. program, receive tuition reimbursement or stay academically eligible, then it is YOUR responsibility to perform at that level. "A" work will get you an "A" and "F" work (or cheating) will get you an "F". I must distinguish between exceptional work and that which falls short of that level. **Grades of D or F are almost always the result of cheating or not doing projects.**

Policies: Attendance is not a part of your grade. However, you are expected to take the exams at the scheduled time. If that is impossible, you must take the exam before the rest of the class, and I must get significant warning. If you are a "no show" you get a zero.

   Exams are online and open book, but consulting with ANYONE is a maximum penalty offense.

Ethics: I try to be friendly and to inject humor into the lectures, but don't mistake that for anything less than a zero-tolerance policy toward cheating. YOU CHEAT, YOU FAIL, subject to procedural review.

   All projects in this course are INDIVIDUAL. Feel free to discuss ideas with your classmates, but **any** shared code will mean an F for the course. **No exceptions!** It is not acceptable for someone else to do your work, whether that be a classmate, a spouse or anyone. Do not share even one line of code. And if you can find a solution online and copy it, your instructor and TA can find it, too.

Post Solutions: You are not allowed to post solutions to projects or exams in public places (e.g. Github) **either during this semester or after**. The University policy on academic integrity states that assisting students in their own acts of academic dishonesty is itself a violation of academic integrity. You also aren't doing other students a favor by posting a temptation to cheat that could have disastrous consequences.

Incompletes: Incompletes will be given only in the case of serious emergencies or other documented extenuating circumstances clearly beyond the student's control. Incompletes will NOT, repeat NOT, be given to students who abandon the class, students not satisfied with their grade, students who are too busy to complete the work on time or students who miss the final exam.

Backups: Buy a USB drive (they are cheap) and **back up your work on projects daily**. There are no project extensions or late penalties waived for a disk crash. (Or just email the code to yourself.)

SCHEDULE

Date	Topics	Related Reading
May 26	Course Overview, Number systems, Signed numbers, Characters	Chap. 1.2, 2.4, 2.5, 2.9, skim rest Chap 1
June 2	MIPS Instructions - Assembly Language and Internal Representation *First project assigned: Due June 23*	Chapter 2 (skip asm programming, focus on internal rep)
June 9	Measuring Computer Performance, CISC/RISC The Memory Hierarchy - Caches (Part 1)	Chapter 1.6 Chap. 5.1, 5.3, 5.4 (ignore math), 5.8
June 16	The Memory Hierarchy - Caches (Part 2), *Second project assigned: Due July 14*	
June 23	The Processor: Datapath and Control	Chapter 4.1 - 4.4
June 30	MIDTERM EXAM	
July 7	Pipelines (Part 1) *Third project assigned: Due Aug. 4*	rest of Chapter 4
July 14	Pipelines (Part 2) Advanced Architectures - Superscalar, Superpiplining	
July 21	Parallelism, Multicores, OS Concepts, Virtual Memory, Start-of-the-Art Cache extensions	Chapter 6, Chap 5.6
July 28	Logic Design, Computer Arithmetic and ALUs	App. B: pp. 1-14
Aug 4	Wrap-up, Final Exam Review	
Aug 11	FINAL EXAM	