CS/SE 2340 Course Syllabus

COURSE INFORMATION:

Course Title: Computer Architecture
Section: 007
Course Term: Spring 2024

INSTRUCTOR CONTACT INFORMATION:

Name: Brian Ricks
Office: ECSS 4.701
Email Address: bwr031000@utdallas.edu
Phone: (972) 883-2674

Communication: For course related questions, please email me from your UTD account with a subject that begins with: *CS2340.007*. There is also an MS Teams channel which can be used for general course questions / discussion. I will give course announcements on eLearning.

Office Hours: Tuesday: 11:00am – 12:30pm
Wednesday: 11:00am – 12:30pm

Meetings outside of office hours can be arranged by appointment.

Do not procrastinate. I will not schedule meetings to assist students with assignments that are due within 24 hours.

TA CONTACT INFORMATION:

Kenneth Imphean - kenneth.imphean@utdallas.edu

COURSE PREREQUISITES AND COREQUISITES:

Prerequisite: CS/CE 1337, and CS/CE 2305
Corequisite: None

Credit cannot be received for both CS/SE 2340 and CE/EE 4304.

COURSE DESCRIPTION:

CS/SE 2340 – Computer Architecture (3 semester credit hours)
This course introduces the concepts of computer architecture by going through multiple levels of abstraction, and the numbering systems and their basic computations. It focuses on the instruction-set architecture of the MIPS machine, including MIPS assembly programming, translation between MIPS and C, and between MIPS and machine code. General topics include performance calculation, processor datapath, pipelining, and memory hierarchy.

Credit cannot be received for both courses, (CS 2340 or SE 2340) and (CE 4304 or EE 4304). Prerequisites: (CE 1337 or CS 1337) with a grade of C or better or equivalent and (CE 2305 or CS 2305) with a grade of C or better. (Same as SE 2340) (3-0) S

STUDENT LEARNING OBJECTIVES/OUTCOMES:

1. Ability to convert numbers between bases 2, 10, and 16 and perform binary 2's complement arithmetic calculations.
2. Ability to create and debug assembly language programs that include (a) integer operations, (b) floating-point operations, (c) control structures, (d) functions and macros, (e) static memory allocation for varied data types, (f) dynamic memory allocation, (g) standard I/O, and (h) memory-mapped I/O.
3. Ability to critique programs for efficient use of data types, addressing modes, and instructions, as well as for readability and maintainability using a provided rubric.
4. Ability to calculate computer performance measures. Students will use performance metrics to evaluate a chosen CPU architecture. Students will define terms used in CPU technical brochures.
5. Ability to differentiate between systems programs such as compilers, assemblers, linkers, and loaders, and explain how they are used by operating systems to convert programs into machine language to be executed.
6. Ability to explain the flow of information through a computer's core components, including the datapath, control, memory, and I/O.
7. Ability to diagram the flow of data, code, and control signals through a pipelined architecture. Students will evaluate the presence or absence of hazards in a pipelined architecture, and explain how the hazards could be avoided.
8. Ability to diagram and explain the memory hierarchy, including caches and virtual memory. Students will explain the different memory technologies used in computers.
9. Ability to contrast the four types of parallelism.

COURSE MODALITY:

This course is scheduled to be taught in the traditional (in-person) mode.

Should it become necessary to meet online, meetings will be held in MS Teams in the appropriate lecture channel. Check the course announcements and your UTD email account for updates.

Online meetings will be recorded and posted in MS Teams.

REQUIRED TEXTBOOKS AND MATERIALS:

Required Textbook:
Computer Organization and Design - The Hardware/Software Interface (MIPS Edition)
Patterson and Hennessey
6th Edition
ISBN: 978-0-12-820109-1

Note: Make sure that you get the correct edition and processor (MIPS).

MIPS Simulator:

This course will involve coding in the MIPS assembly language using MARS (MIPS Assembler and Runtime Simulator). Get it free here: http://courses.missouristate.edu/KenVollmar/mars/

The MARS simulator's help screen is very useful.

CLASS MATERIALS:

Other materials including the syllabus, sample programs, assignments, announcements, etc. will be posted in eLearning.

https://elearning.utdallas.edu

COMPUTER SCIENCE MENTOR CENTER (CSMC):

The Computer Science Mentor Center (CSMC) provides tutoring and assignment / project help. It is located at ECSS 4.415.

https://csmc.utdallas.edu

TENTATIVE COURSE CALENDAR:

Date	Lecture Material	Reading
January 16	Syllabus Review	
January 18	Introduction to Computer Organization	Chapter 1.1 – 1.5
January 23	Introduction to Assembly Language Programming	Appendix A
January 25	Performance evaluation, Amdahl's law	Chapter 1.6, 1.9
January 30	Data Representations: Binary, Hexadecimal, Octal	Chapter 2.3
February 1	Number Representations: signed, floating point	Chapter 2.4
February 6	Instruction Representation	Chapter 2.5
February 8	Assembly operations: Load, Store, Add, Subtract, etc. Addressing modes.	Chapter 2.2
February 13	Comparing, Branching, Looping	Chapter 2.7
February 15	Exam 1 Review	
February 20	Bit and byte manipulation, etc.	Chapter 2.6
February 22 – 23	**Exam 1 – In the testing center, we will *not* have lecture on February 22**	
February 27	Subroutines in Assembly Language	Chapter 2.8, Appendix A.6
February 29	Memory Allocation and the Heap	
March 5	Integer Arithmetic	Chapter 3.1 – 3.4
March 7	Addressing Modes and System Software	Chapter 2.12 – 2.13
March 12	**No class – Spring Break**	
March 14	**No class – Spring Break**	
March 19	Floating Point Arithmetic	Chapter 3.5
March 21	File Input & Output	
March 26	Memory-Mapped I/O	
March 28	Exam 2 Review	
April 2	Interrupts and Exceptions	Chapter 4.9, Appendix A.7
April 4 – 5	**Exam 2 – In the testing center, we will *not* have lecture on April 4**	
April 9	Processor: Datapath & Control	Chapter 4.5
April 11	Processor: Pipelining	Chapter 4.6 – 4.8
April 16	Advanced Instruction Level Parallelism	Chapter 4.11
April 18	Introduction to memory hierarchy	Chapter 5.1 – 5.3
April 23	Virtual memory	Chapter 5.4 – 5.7
April 25	Comparing ISAs	Chapter 2.16 – 2.17
April 30	Exam 3 Review	
May 6 - 8	**Exam 3 – In the testing center**	

This calendar is tentative and may change as the semester progresses. Please check eLearning for announcements regarding changes to the calendar.

INSTRUCTOR POLICIES:

Grading Policy:
Your course average will be calculated as follows:

Exam 1 – 20%
Exam 2 – 20%
Exam 3 – 25%
Programming Assignments (7) – 35%

The instructor intends to assign letter grades as shown below. An average with a fractional portion of five tenths or above will be rounded up to the next whole number for determining the letter grade.

Averages	Letter grade
97+	A+
93-96	A
90-92	A-
87- 89	B+
83-86	B
80-82	B-
77-79	C+
73-76	C
70-72	C-
67-69	D+
63-66	D
60-62	D-
Below 60	F

Grading Concerns:
If you think there is a mistake in the grading of your *assignment*, you must notify *both* the grader and the instructor by email *within two weeks* after the date the grade is posted in the gradebook on eLearning. Keep in mind that a regrade may result in an increase *or* in a reduction of the original grade.

Most deductions are made because students did not fully read the assignment instructions, disregarded the instructions, did not adequately test their programs, or did not follow the style guidelines provided. You may not change the problem to suit your purposes. Most assignments restrict the use of programming constructs and library functions not covered in lecture, while others require that you use particular constructs or functions. To get maximum credit you MUST read the directions carefully and test your programs thoroughly.

If you think there is a mistake in the grading of your *exam* and would like to request that it be regraded, you must notify the instructor of this by email *within two weeks* after the date the grade is posted in the grade book on eLearning. Your request for any regrade must describe in detail what you perceive as the problem with the grading. Keep in mind that a regrade may result in an increase *or* in a reduction of the original grade.

Student Responsibilities:

- You are responsible for all the material in the assigned reading in the required course textbook.
- You are responsible for all the material in the slides.
- You are responsible for all material discussed during course lectures.
- You are responsible for all material supplied on eLearning (including announcements and discussion postings).
- Students are expected to be respectful of each other and of the course instructor. Disruptive behavior will not be tolerated.

Exams:

Course exams will be administered in the testing center. You must register for each exam in advance of the exam date. Exam registration is located here: https://www.registerblast.com/utdallas/Exam/List

We will *not* have lecture on the day of the exams as your exam window may overlap our normal lecture time.

Make-up exams will be administered *only for well-documented emergencies*, and you must make every attempt possible, via email, MS Teams private message, or phone, to notify the instructor *prior* to the scheduled date and time or *immediately thereafter*. If notification is not received in a timely manner, no make-up exam will be given.

Programming Assignments:

All assignments will be announced and submitted using eLearning. You will be given at least one week to complete each programming assignment. Each assignment will include a due date and time. You may submit an assignment up to 24 hours after the due date and time with a 10-point penalty. No excuses will be entertained for late assignments.

Programming assignments are individual endeavors and students are not to work in groups on any assignment. Students are permitted (and I openly encourage students) to share ideas on the program logic, but DO NOT SHOW YOUR CODE TO OTHER STUDENTS. When discussing logic, try to keep it general. If you give out every little piece of logic you have, there is a good chance the person you are helping will have very similar code as yours and may be flagged for being too similar. Be careful of posting your code online. Another student could use your code without your knowledge and could involve you in a code plagiarism referral.

Students should avoid using web sites like GitHub, Chegg, and ChatGPT for help on assignments. Copying code from a web site is considered plagiarism and will be treated as such. If you find code on a web site, it is highly likely another student will find it as well which may cause both submissions to be flagged for similarity. Submitting assignment details to web sites for outside help is also considered academic dishonesty by UTD.

Programming assignments will be graded on a 100-point basis. You are responsible for testing your program thoroughly before submission. I will not give you the exact test cases that will be used for grading before the assignment is due. As a computer scientist, you must be able to identify all possible input/edge cases and make sure that your code produces proper output/does not crash**.**

What you need to do to be successful in this course:

- Attend every course lecture and pay close attention.
- Read the appropriate textbook chapter sections *before* the respective lecture.

- Dedicate 9-10 hours per week outside of course lectures for reading, practicing writing code, completing assignments, and studying for exams.
- Do all your work and do it individually. Students who get too much help from others: the instructor, mentors, peers, and the internet may not acquire the knowledge they need to successfully complete the exams.
- Do not procrastinate! Start programming assignments early so that you have time to try alternate approaches, ask questions, and test your program.
- Do not wait till the end of the semester to seek help. If you wait until late in the semester, it may be difficult to catch up, as the course is constantly moving forward.
- If you do not understand a concept: reread the text, slides, and other materials. Arrange a meeting with your instructor and come to the meeting with a list of specific questions. Visit the Computer Science Mentor Center (CSMC), prepared with specific questions.
- Write more programs than are required. The more programs you practice with, the better you will do in this course. You learn to program by doing – coding, testing, and fixing (debugging). This course is like a mathematics course – you need to work *many* problems.

MORE FROM UTD:

Class Materials

The instructor may provide class materials that will be made available to all students registered for this class as they are intended to supplement the classroom experience. These materials may be downloaded during the course, however, these materials are for registered students' use only. Classroom materials may not be reproduced or shared with those not in class, or uploaded to other online environments except to implement an approved Office of Student AccessAbility accommodation. Failure to comply with these University requirements is a violation of the Student Code of Conduct.

Classroom Conduct Requirements Related to Public Health Measures

UT Dallas will follow the public health and safety guidelines put forth by the Centers for Disease Control and Prevention (CDC), the Texas Department of State Health Services (DSHS), and local public health agencies that are in effect at that time during the Fall 2021 semester.

Class Attendance

The University's attendance policy requirement is that individual faculty set their course attendance requirements. Regular and punctual class attendance is expected. Students who fail to attend class regularly are inviting scholastic difficulty. In some courses, instructors may have special attendance requirements; these should be made known to students during the first week of classes.

Class Participation

Regular class participation is expected. Students who fail to participate in class regularly are inviting scholastic difficulty. A portion of the grade for this course is directly tied to your participation in this class. It also includes engaging in group or other activities during class that solicit your feedback on homework assignments, readings, or materials covered in the lectures (and/or labs). Class participation is documented by faculty. Successful participation is defined as consistently adhering to University requirements, as presented in this syllabus. Failure to comply with these University requirements is a violation of the Student Code of Conduct.

Class Recordings

Students are expected to follow appropriate University policies and maintain the security of passwords used to access recorded lectures. Unless the Office of Student AccessAbility has approved the student to record the instruction, students are expressly prohibited from recording any part of this course. Recordings may not be published, reproduced, or shared with those not in the class, or uploaded to other online environments except to implement an approved Office of Student AccessAbility accommodation. Failure to comply with these University requirements is a violation of the Student Code of Conduct.

The instructor may record meetings of this course. These recordings will be made available to all students registered for this class if the intent is to supplement the classroom experience. If the instructor or a UTD school/department/office plans any other uses for the recordings, consent of the students identifiable in the recordings is required prior to such use unless an exception is allowed by law.

Comet Creed

This creed was voted on by the UT Dallas student body in 2014. It is a standard that Comets choose to live by and encourage others to do the same:

"As a Comet, I pledge honesty, integrity, and service in all that I do."

Academic Support Resources

The information contained in the following link lists the University's academic support resources for all students.

Please see http://go.utdallas.edu/academic-support-resources.

UT Dallas Syllabus Policies and Procedures

The information contained in the following link constitutes the University's policies and procedures segment of the course syllabus. Please review the catalog sections regarding the credit/no credit or pass/fail grading option and withdrawal from class.

Please go to http://go.utdallas.edu/syllabus-policies for these policies.

The descriptions and timelines contained in this syllabus are subject to change at the discretion of the Professor.