

CS3432 Computer Architecture I Syllabus

Contents

- [1 Important: How to Participate](#)
- [2 Course Info and Lab Notes](#)
- [3 Homework, Tests, and Labs](#)
- [4 Readings](#)
- [5 Resources](#)
- [6 Disabilities and Accommodations](#)
- [7 Grading & Academic Honesty](#)

Important: How to Participate

- We'll be meeting using Microsoft teams.
 - Join the course team named s21-arch1 using code **zm0rwni**
 - Install the teams app or browse to teams.microsoft.com (the app is better)
 - Login with your UTEP credentials, join the team.
 - Lectures: MW 9AM on Microsoft teams.
 - Announcements: will be made in the General channel
 - Student questions (and everybody's answers): in the "Questions and Answers" channel
- Recommended: Install this microsoft software
 - onenote
 - whiteboard

Course Info and Lab Notes

- **This course will likely be all-online via teams**
 - This course is synchronous.
 - Medically necessary
 - We've gotten good at this
 - If physical comradery is good for you, can attend in this room at this time.
 - Attend MW 9-10:20 with cameras on
 - Send mail if this is a problem for you.
 - Recordings will be available upon request; send an email.
- **Course homepage:**
 - <https://sites.google.com/site/arch1utep/home>. Or shortened: <https://goo.gl/ejdsQL>
 - Bookmark it.
- **Communication:**
 - If you're asking questions about class topics, please use
 - Post to the course's Microsoft team (I and the team will post there)
 - For more private matters:
 - send mail to the entire instructional team: su20-arch1-teaching-team@googlegroups.com,
 - or send email to instructor or a TA

▪ Teaching team

- Instructors: [Dr. Eric Freudenthal](#) (see homepage for office hours)
- Teaching Assistants:
 - TAs will be present in the Knowledge Base room (CCSB 1.0706) and online/Teams for their office hours.
 - IAs will be present in the Dungeon (CCSB G.0512) and online for their office hours.
 - If you need to meet a TA/IA at another time, send them email.

Team schedule	Eric Freudenthal	Ricardo Alvarea		
Email	efreudenthal	ralvarezlo		
Office hours	after class			

▪ Prerequisites:

- C or better in Digital Design, Discrete Math & Data Structures/CS3.
- Alternative: B or better in Digital Design, Discrete Math & Elementary Data Structures/CS2.

▪ Labs

- Students **must** enroll in a lab section.
- **Participation is mandatory.** It is extremely easy to fall behind and imperative that you make arrangements with the instructors or TAs to make-up the missed lessons and work. Lab sessions and assignments managed by TA. Students at risk of failing due to not engaging with the lab course may be dropped.
- There may be scheduled and unscheduled quizzes in lab course
- Assignments and due dates: Click on [Lab Projects](#) navigation tab.

▪ Texts

- Online: this web site (see *Course Outline & Timetable* navigation tab).
- Absolutely required:
 - Kerningham, Brian W & Ritchie, Dennis M. "The C Programming Language, Second edition," Prentice Hall, ISBN: 0-13-115817-1.
- Recommended by students (and not required):
 - Android app "Programmer Mental Math" by Joel Jurix.
 - "Head First C" by David & Dawn Griffiths
- Also handy if you're interested:
 - MSPGCC cross-tools manual (55 pages): [pdf](#), [web](#)

▪ Grading and Learning Outcomes

- Learning outcomes are described on the [Grading and Course Learning Outcomes](#) navigation tab.
- Your grades will be based upon observations of your knowledge during coaching sessions.
- We will distribute itemized assessments of your observed skills regularly throughout the term.
- Students can request an alternate evaluation modality via email to the instructor if they believe those itemized assessments is incorrect.

▪ Computers

- You will need a portable computer capable of running virtualization software and Microsoft OneNote to participate in class activities. All required software is available to students at no cost and are available for windows, linux, and macOs. Should you not have access to an appropriate machine, both the department and instructor have a limited number of portable computers that can be lent out to students.
- **Attendance**
 - Mandatory at all all sessions of lecture & lab course unless special circumstances arranged ahead of time with instructor

Homework, Tests, and Labs

- **Homework**
 - Students are expected to review techniques taught in class and be able to demonstrate those skills during the next class session. Example problems with solutions are published on the course web site.
- **Exams/quizzes:**
 - Quizzes
 - Quizzes assess individual students' abilities to demonstrate knowledge, to design solutions to realistic problems, and to present these solutions in a clear and professional fashion. Quizzes will be graded "by skill" (see below), can cover any concept or skill previously taught in the course, are generally offered at the rate of once per week, and unannounced (so students must be continuously prepared).
 - Final Exam
 - The final exam date is scheduled by the university based upon lecture time. Like quizzes, the final exam will be graded "by skill" (see below) and can cover any concept or skill previously taught in the course.
 - Just in case you thought otherwise, the following are prohibited during quizzes and exams unless unambiguous and explicit permission are provided by the instructor:
 - Collaboration, communication with assistance from (or to), or just peeking at other's exam answers while preparing your own.
 - ~~Participation without being physically present in the classroom.~~
- **Labs**
 - Intention:
 - Labs are intended to provide an opportunity for students to practice and explore the use of tools and apply concepts presented in class within the context of programming projects.
 - The lab course will principally be organized around a few projects, many are open-ended.
 - Projects will be graded during in-person demonstrations with individual students.
 - Demonstrations must occur prior to the project due date. Exceptions must be justified and pre-arranged with the instructor.
 - Students may demonstrate labs multiple times (for updated grades) prior to the project due date.
 - Demonstrations may require students to modify their programs and demonstrate competency with development tools.
 - Students are expected to act professionally
 - By reading whatever resources they find relevant
 - By attributing credit to any person or reference materials that substantively contributed to their solutions

- By only submitting solutions they fully understand.
- Professionalism includes honesty, clarity, and accuracy.
- Students are encouraged to help each other develop and tune their projects. There is no penalty for collaboration as long as it's documented.
- Requirements
 - **Functional:** Assignments will either require students to create complete programs or modify programs provided by the instructor.
 - **Documentation:** Submissions should include documentation that facilitates the grader's determination of
 - How to compile, use, and test
 - Principles of operation (e.g. comments & other descriptive prose)
 - **The project must reflect Individual mastery.** We don't care where the code came from, only that students possess the capability to explain their solution or apply the same techniques towards solving a similar problem.
 - **Completeness:** Students who labs that do not substantially satisfy functional and documentation requirements will receive no credit.
- Homework:
 - Most class sessions will conclude with an assignment due at the beginning of the next class session (unless otherwise indicated).
 - While most assignments are neither collected nor graded, student mastery of relevant skills will be tested within quizzes and tests.
- Students may contact the instructor to request a non-verbal assessment of their skills.
- [More notes on grading policy](#)

Readings

- Students should read chapters 1-6 of K&R (Kernigham & Ritchie's "The C Programming Language" each of the first 6 weeks of class.
- Rationale: These chapters of K&R introduce concepts in the context of C that the course examines at a lower level.

Resources

- [Obtaining and installing the Arch1 Virtual Machine](#)
- [MSP-430](#) (the embedded controller we will study)
- [Getting started in Linux/Unix and SVN](#)

Disabilities and Accommodations

- If you have a disability and need classroom accommodations, please contact The Center for Accommodations and Support Services (CASS) at 747-5148, or by email to cass@utep.edu, or visit their office located in UTEP Union East, Room 106. For additional information, please visit the CASS website at www.sa.utep.edu/cass.
- Students are expected to conduct themselves in a professional and courteous manner, as prescribed by the Standards of Conduct: http://hoop.utep.edu/Student_Affairs_Chapter_One-HOP.htm
- Professors are required to report academic dishonesty and any other violation of the Standards of Conduct to the Dean of Students

Grading & Academic Honesty

- Permitted collaboration: Students are strongly encouraged to collaborate and utilize online resources during all assignments and learning activities **except during explicit summative assessment activities and 1-on-1 coaching sessions.**
 - You can incorporate solutions you obtain from others or published sources.
- Unless otherwise noted:
 - You may collaborate with others while preparing solutions to problems posted in projects, quizzes
 - **You may not submit any solution or answer that you cannot independently synthesize.**
 - **This includes justifying the appropriateness of the techniques you use.**
 - This ability to independently solve those (and related problems) will be evaluated by observation and Socratic conversation.
- If academic dishonesty is suspected: The [student conduct office](#) will be contacted for adjudication. A temporary "incomplete" grade will be issued if their investigation extends beyond the grading period.