Programming HW/SW System - ECE-2035-R

Welcome to ECE2035!

COURSE DESCRIPTION: This course presents execution and storage mechanisms used to support high level programming languages and operating systems. This design-oriented course describes how complex mechanisms are created using operations and storage defined in an instruction set architecture. Assembly language examples illustrate key course concepts. Lecture material is reinforced by design projects that require C programming and MIPS assembly language programming, focusing on performance and storage resource requirements on hardware platforms.

Course Delivery: This course is delivered in person. Only courses listed for video instruction will be delivered that way. ECE 2035 has no video section.

Instructor: Jeff Hurley (jeffery.hurley@gtri.gatech.edu)

Class Time: TBD (GT Europe)

Dr. Hurley Office Hours: TBD.

Course Delivery:

This course is delivered in-person. Only courses listed for video instruction will be delivered that way. ECE 2035 has no video section.

Lab Section: **There is a self-directed lab section, so there is no physical lab, nor any required attendance, outside of lecture.**

In addition to in class and office hours, feel free to ask (and answer) questions on piazza course page.

DESCRIPTION: This course presents execution and storage mechanisms used to support high level programming languages and operating systems. This design-oriented course describes how complex mechanisms are created using operations and storage defined in an instruction set architecture. Assembly language examples illustrate key course concepts. Lecture material is reinforced by design projects that require C programming and MIPS assembly language programming, focusing on performance and storage resource requirements on hardware platforms.

COURSE GOALS: The learning objectives of this course are to:

- Understand how execution and storage constructs in high-level programming languages and operating systems are implemented on a hardware platform.
- Learn design principles for sequential, procedural programming, the C programming language and MIPS assembly language, and the use of common tools for software development and performance evaluation.
- Build experience in systems-oriented design, focusing on performance and storage requirements of the target application and hardware platform.

COURSE OUTCOMES: Upon completion of this course, you should be able to:

- Develop a software design using appropriate data and procedural abstractions given an engineering problem specification.
- Implement high-level programming language storage, control, and procedural constructs in the assembly language of a hardware platform.
- Use commonly available tools for software development, system management, debugging, performance monitoring, and optimization.
- Develop a software system that uses multiple files and libraries.
- Develop, test, and deploy reliable software using appropriate operating system capabilities (e.g., file and directory access, I/O interface).

PREREQUISITES: ECE 2020. It is imperative that you be very familiar with

- the concept of a datapath,
- the major components of a datapath,
- memory, and especially RAM, and
- the specifics of a single-cycle datapath, such as that of the MIPS processor.

That last point is particularly relevant to transfer students or anyone substituting a different prerequisite, since ECE 2020 specifically covered the single-cycle datapath of the MIPS processor. We will immediately begin with review of the MIPS datapath and go into more detail about the MIPS instruction set architecture (ISA).

GRADING: The final grades are determined based on totals earned during the course.

Assessment	Percentage of total grade:
Exams (2)	30 %
Homeworks (3)	10 %
Projects (2)	30 %
Final Exam (1)	25 %
Participation	5 %

Grades are assigned as follows (assuming a passing project average): [90-100] = A, [80-90) = B, [70-80) = C, [60-70) = D, [0-60) = F.

Grading disputes should first be directed at the person grading the assignment (e.g., the GTA or UTA for homework & project grades, Prof. Collins for exams, participation, etc.), but any dispute can be escalated to Prof. Collins as needed. Also, it is expected that any grading concerns be raised within one week of the grade being posted, because it isn't possible to address a large number of issues in the last weeks of the semester.

CLASS PARTICIPATION: Attend and participate to classes

MIDTERM EXAMS: There are two midterm exams, held in the classroom. Each exam focuses on the material covered since the last exam. However, due to the cumulative nature of the material, all exams are comprehensive (i.e., they may draw on all material covered in the class thus far.) Each midterm exam contributes 15% of the overall grade.

FINAL EXAM: The final exam covers all material covered in the class. It is closed-book, closed-note and calculators are not allowed. The final exam contributes 25% of the overall grade.

ONE EXAM REPLACEMENT POLICY: If your score on the final exam is greater than your lowest midterm exam grade, it will be used to replace your lowest midterm exam score.

MISSED EXAM POLICY: Exams are taken at the scheduled class time or at the scheduled final period. A missed exam will be recorded as a zero. Family emergencies and extreme medical emergencies are handled specially.

EXAM SCHEDULE:

TBD

Check at information provided by Mrs Corinne Guyot

ASSIGNMENTS: Multiple homeworks and projects are assigned throughout the semester. Homeworks are assigned more frequently, while projects have longer timeframes and are weighted more heavily. All homework assignments and projects are to be completed and submitted individually.

LATE POLICY: The Canvas page for each homework and project assignment specifies the late policy. In general, for all assignments, *except parts of* Project 2, the assignment may be submitted up to 5 days after the posted due date, with a 10% per day late penalty. For your benefit, that penalty is applied per hour (0.4167% per hour), so that you do not get hit with a 10% penalty for missing a deadline by a few minutes, or even a few hours.

Do not hesitate to contact your instructor if extenuating circumstances arise. Staying in communication is critical. If you are struggling or falling behind make an appointment to discuss how to proceed.

ASSIGNMENT INFRASTRUCTURE: To perform the assignments, you need the following:

- **Misasim** simulator for the MIPS assembly language. (FREE)
- **Linux**: MacOS (terminal) or Windows (WSL). (FREE)
- **Mbed classroom kit is purchased by GT Europe**
    - Code examples and tutorials are available here and in the Experiment Guide.
    -

BACKING UP WORK: It is each student's responsibility to create back-ups of work performed in this class. Lost work or time due to computer/disk/web server failures is not a valid excuse for late submissions.

TEXTBOOK: Patt and Patel, *Introduction to Computing Systems*, 3rd edition, 2019. (2nd edition is fine, too.) Yes, there are required readings in the textbook.

ACADEMIC HONESTY: Although students are encouraged to work together to learn the course material, graded class work must be completed individually. Specifically, while you are permitted to discuss the homework and project assignments and algorithms with other students in the class, you must design, write, and debug your solutions individually. **You should not share any code, homework solution or any graded work before or after the due date.**

Additionally, all code and course materials provided in ECE2035 are copyrighted. They are for the use of the students currently enrolled in the course. Copyrighted course materials may not be further disseminated. You may not, nor may you knowingly allow others to reproduce or distribute code or other course materials publicly. This includes providing materials to commercial course material suppliers such as CourseHero, chegg, and other similar services, or posting your project code on github. Students who publicly distribute or display or help others publicly distribute or display copies or modified copies of ECE2035's course materials are in violation of Georgia Tech's Honor Code.

All exams are to be completed individually with no collaboration or interaction with anyone else. You may neither give nor receive unauthorized assistance on any exam. You may not work with others on the exam and you many not share questions or answers with anyone else, including looking for or posting questions/answers on any website.

All conduct in this course will be governed by the Georgia Tech honor code. Additionally, it is expected that students will respect their peers and the instructor such that no one takes unfair advantage of anyone else associated with the course. Any suspected cases of academic dishonesty will be reported to the Office of Student Integrity for further action.

STUDENT WELL-BEING: Dr Voss and his team will help you if you need

ACCOMMODATIONS FOR STUDENTS WITH DISABILITIES: If you are a student with learning needs that require special accommodation, contact me and Mrs Corinne Guyot, as soon as possible with your accommodations letter.

STUDENT-FACULTY EXPECTATIONS AGREEMENT: At Georgia Tech we believe that it is important to strive for an atmosphere of mutual respect, acknowledgement, and responsibility between faculty members and the student body. See this catalog page for an articulation of some basic expectation that you can have of me and that I have of you. In the end, simple respect for knowledge, hard work, and cordial interactions will help build the environment we seek. Therefore, I encourage you to remain committed to the ideals of Georgia Tech while in this class.

INFORMATION RELATED TO COVID-19:

Students are expected to be familiar with and abide by the Institute guidelines, information, and updates related to Covid-19. Find campus operational updates, Frequently Asked Questions, and details on campus surveillance testing and vaccine appointments on the **Tech Moving Forward site**.