Course Syllabus

CS2200 - Introduction to Systems and Networking

Purpose and Outcomes

Purpose

Provide a broad exposure to computer system structure and networking including software abstractions in operating systems for orchestrating the usage of the computing resources:

- Organization of the processor
- Memory hierarchy
- Storage devices
- Parallel processors
- Networking hardware
- Software abstractions in the operating systems for orchestrating their usage
- Networking protocols to connect the computer system to its environment

Outcomes

- (Competency Knowledge) Understand the difference between RISC and CISC architectures. Be able to identify the strengths and weaknesses of each paradigm.
- (Competency Knowledge) Understand and be able to explain runtime system concepts such as procedure calls and register saving. Be able to write recursive subroutines in assembly.
- (Competency Application) Understand how a processor is controlled. Given a datapath and an instruction set be able to write the finite state machine steps in a high-level meta language.
- (Competency Knowledge) Understand and be able to explain (at a high level) hardware modifications required to implement an interrupt system and to understand the basic concepts required to write an interrupt handler (in assembly language).
- (Competency Knowledge) Understand the basic principles of pipelining:
  - Pipelining registers
  - Potential performance improvements with pipelining
  - Pipelining Hazards: Structural, Data, and Control
- (Competency Knowledge) Understand basic concepts of processor scheduling: Process vs program, PCB, scheduling algorithms (Round Robin, Shortest Job First, First Come First Served, Priority, Multilevel Queues), types of scheduler (short, medium, and long term), and context switching.
- (Competency Comprehension) Given a set of processes with appropriate parameters show scheduling behavior under different scheduling algorithms.
- (Competency Application) Be able to calculate the proper size required for pipeline register and speedups with pipelining.
- (Competency Application) Be able to solve basic word problems involving Amdahl's Law.

- (Competency Knowledge) Be able to identify and explain how to avoid or minimize the effect of the different types of pipelining hazards.
- (Competency Knowledge) Understand the drivers of memory cache designs: Temporal locality, spatial locality, and working set. Be able to match the design with the motivator.
- (Competency Knowledge) Understand the basic operation of virtual memory and typical components: Page table, virtual pages, physical frames, TLB, page/frame offset, page replacement algorithms (LRU, Random, FCFS, Optimum). Be able to describe the basic operation and identify the necessary subsystems.
- (Competency Knowledge) Understand the basic design of typical caches including indexes, tags, dirty and valid bits as well as multi-word blocks, set-associative, and fully associative caches. Given selected design parameters (i.e. word size, memory available for data
- (Competency Knowledge) Understand basic concepts of parallel processing: UMA (SMP) vs NUMA configurations, multiprocessor cache coherency, network interconnection schemes, threads, mutex, condition variables.
- (Accomplishment Application) Be able to write multi-threaded programs using the pthreads package. An example would be a multithreaded producer-consumer application.
- (Competency Knowledge) Understand basic networking concepts: Ethernet (CSMA/CD), Token Ring, Payload vs. header and trailer, checksums, bandwidth, effective bandwidth, latency, MAC addresses, Network (IP) addresses, protocol stacks, TCP/IP, routing, hubs/repeaters, bridges, VLANS, routers.
- (Competency Knowledge) Understand fundamentals of I/O devices such as polling versus interrupts, memory-mapped I/O, device registers (data, control, and status), disk memory concepts (sectors, tracks, platters, cylinders, seek time, rotational latency), disk scheduling algorithms (FCFS, SSTF, scan, c-scan, look, c-look)
- (Accomplishment Synthesis) Write and debug medium-sized C programs that simulate various of the above subsystems (interrupt enabled processor, virtual memory, multi-threaded operating system schedulers, reliable transport layer protocol which will be examples of operating-system-like coding.

Course Organization

This course is being officially delivered in hybrid form and it is an experiment to see if we should do this again in the future. What does that mean to you? Only a little. We ran out of large classrooms for this semester and somebody had to yield. I volunteered us to try it (sorry I didn't get a chance to consult you). Almost everything you'd expect from an in-person course still applies: The course is synchronous, we expect you to attend lectures and labs, and tests will be administered in person. The only difference is that you will attend one in-person lecture and one livestream/recorded lecture each week. Everything else stays the same, so you get no license to procrastinate at will as you might in a differently structured hybrid course. I recommend that you not delay your engagement in the lectures and assignments; in addition to incurring penalties, this material is difficult to catch up, even in the best of circumstances.

Lectures

There will be a single lecture series for the A and B sections. The lectures will be delivered at 2pm on Tuesdays and Thursdays. They also will be live-streamed on Zoom, recorded and the slides will be published immediately before the lecture on Canvas in Files>Lecture Slides. Expect the lecture recordings to become available several hours after the lecture ends.

Section A students are expected to attend in-person on Tuesdays and join the live stream or watch the recording on Thursdays; section B students are expected to attend in-person on Thursdays and join the live stream or watch the recording on Tuesdays. If you want to ask questions, you will need to attend the in-person or the live-streamed session; the recorded lectures don't respond well to ad hoc questions. We will have someone monitoring the Zoom Chat during the lecture to make sure questions get answered.

There will be lecture attendance quizzes for each lecture after the first week. These will start out on Canvas, but we may be able to use a different tool to make them a bit more interactive. To receive credit, these quizzes will have to be completed within 24 hours of the lecture; the necessary answers will be discussed during the lecture and can be obtained from the in-person lecture, the live stream, the recorded lecture, or the posted slides. This is a modest incentive for everyone to stay on the same schedule.

Labs

The supervised laboratory meetings are mandatory and in-person. Attendance will be recorded. Labs will consist of group and individual exercises and problem-solving relevant to the lecture material along with review and questions/answers as needed by the attendees. Tests will be given during the lab meetings and can't be missed.

TA Office Hours

We will post a schedule of hours when TAs will be available for in-person consultation in Room 206. Please try to come with concise questions or requests to briefly review part of a topic. Note that days just before assignments are due become *very* busy and in sharing the TAs, you may not get all the help you want. Start early so that doesn't happen to you.

Textbooks and Software

Textbook

Ramachandran & Leahy
Computer Systems: An Integrated Approach to Architecture and Operating Systems
Publisher: PEARSON
ISBN: 9780321486134

Software

CircuitSim

https://ra4king.github.io/CircuitSim (https://ra4king.github.io/CircuitSim)

CircuitSim is a tool for modeling digital circuits that will be used for the first few projects. You may download it from the CircuitSim website, or from the Files tab on Canvas.

Course Rules

CS2200 Rules and Regulations

1. For this semester, we have a large wait list. If you are planning to drop the course (not that we're encouraging you), please do it during this week so that your seat can be used by someone on the wait list. If you drop after Friday, your seat simply goes empty for the semester.
2. You are responsible for turning in assignments on time. This includes allowing for unforeseen circumstances. You are also responsible for ensuring that what you turned in is what you meant to turn in. Both Canvas and Gradescope include a feature to re-download assignment submissions: This allows you to retrieve exactly what you submitted and ensure that it works. Take advantage of this feature.
3. In general, programming assignments should be turned in with a Makefile and all files needed to compile and run the program. The TA grading your submission should be able to make and run your program without adding files, repairing things, etc. This supersedes all instructions in the project or homework files.
4. Demos will be required for most projects in the semester but may be omitted later on. Demo times are to be signed up with TAs via Canvas. If you cannot attend one of the predetermined slots, e-mail the Head TA to set up another time slot. If you sign up for a demo and miss it without a valid excuse you will receive a penalty starting at 25% off, which doubles for each successive missed demo in the semester. Speeding tickets, sleeping late, etc are NOT valid excuses. If you know you are going to miss a demo ahead of time, remove your name from the list and email the TA you are supposed to demo with about why you are missing the time. Regardless of why you miss your demo, you must sign up for another slot and still demo your project. An unexcused failure to demo before the demo period for the project ends counts as a missed demo.
5. Tests and examinations must be taken at the scheduled date and time. Please do not ask for special treatment because you (or your parents) have purchased non-refundable airline tickets. The safe time to travel is at the end of or after finals week. The finals schedule published at the beginning of the semester is TENTATIVE. The official schedule gets published very late in the semester.
6. The deadline for re-grades is 2 weeks after an assignment grade is posted or returned to you. This deadline also applies to picking up items that are returned in class. After this deadline, no grade changes will be made.
7. The Canvas announcements should be read every day, so be sure you have the announcement forwarding set up. Official announcements about course matters, including changes to the syllabus, will be posted there. In other words, a policy change that doesn't show up in Canvas announcements is just a nasty rumor. The Ed Discussion course forum is for posting technical

questions about assignments, tests, etc. Complaints, questions about your personal problems, etc. should be discussed with your instructor in person or via email.

8. If you have any major personal problems (family/illness/etc.) please go to the Dean of Student's office located in the Student Services Building (Flag Building) next to the Student Center or make a request through https://studentlife.gatech.edu. They are equipped and authorized to verify the problems and the office will issue a note to all your instructors making them aware of the problem and requesting whatever extension, etc. is necessary. Note: We will excuse missing lecture attendance quizzes if you have an institute authorized absence from class.
9. If you need a certain grade in order to stay in school, maintain a scholarship, etc. the time to worry about this is right from the beginning of the course not during the week before finals. Grades are based on demonstrated performance, not individual needs based on factors external to the course. Please do not request special consideration based on this type of situation.
10. If you find yourself slipping behind or face other barriers to your success in this course, please talk with your TA, see the Head TA, or go see the instructor as soon as you can. We can't help if we don't know and no matter what, we can't turn back the calendar.
11. Final grades will be available from Banner normally on the day after grades are due. You may review your final and discuss your grades during the following semester in which you are attending GT. Grades will not be discussed over break.
12. Out of consideration to your fellow students please turn off cell phones, beepers, wristwatch alarms, etc. Also, make every effort to be on time for class. If you are unavoidably late, please sit near the back and try to avoid as much disruption to the class as possible.
13. If you are graduating and need this course to do so please inform your instructor as soon as possible.
14. Complaints about TAs should be directed to the course instructors during office hours or via email.
15. Please be courteous in your correspondence with TAs and instructor.

Project Grading Criteria

1. Student submits project as his/her own (this is what is expected of the students):
   - We will grade the project (out of 100%) as their own submission
   - Use 0 to 100% multiplier from the demo to assess the final grade
     Note: discussion among the students on how to do the project through Ed Discussion and other means is encouraged. HOWEVER, THE CODING OF THE PROJECT HAS TO BE DONE INDIVIDUALLY FOR FULL CREDIT. ZERO TOLERANCE FOR PLAGIARISM (SEE next item).
2. Student submits project as his/her own:
   - The TAs determine the project is plagiarized. TRUST US, IT IS REALLY EASY TO DETERMINE THIS….
   - The projects get a ZERO and all the colluders are reported directly to the Office of Student Integrity
3. Student does not submit his/her own project but says at the time of the oral interview that he/she obtained the project from a peer just to learn, but he/she did not do any coding themselves
   - The TAs will grade their knowledge of the project through the oral interview on a scale of 0 to 25% (MAX)

- If a student's overall project grade is below a 25% and attends the demo their overall project grade will be the maximum of their demo score on a scale of 0 to 25% or their overall project grade.

Assignment Submission Instructions

The following list is designed to help you submit assignments that can be effectively graded giving you the best possible results.

1. ALWAYS retrieve your files from Canvas and check that what you turned in is what you intended to turn in. NO EXCEPTIONS.
2. If you have any problems submitting, email your lab TA(s) what you would have submitted BEFORE the deadline for the assignment(s).
3. Homework and Project Assignments may be worked on collaboratively. What that means is you can work with fellow students to learn how to do the project, post project and homework-related questions to Ed Discussion, and seek help from fellow students and TAs. But what you turn in should be YOUR OWN work.
4. YOU CANNOT COPY CODE OR HOMEWORK SOLUTIONS FROM THE INTERNET AND SUBMIT THEM AS YOUR OWN. THAT IS PLAGIARISM.
5. If you are not sure about the collaboration policy in this class please ask a TA or stop by the instructor's office during office hours.
6. Tests and Project demos are individual efforts.
7. Submit text as plain ASCII text. Do NOT submit word/wordstar/excel/123/dbase2/etc. files.
8. For written (as opposed to code) submissions: Only submit the answers (properly numbered). Do not submit the questions.
9. Wrap text to 80 characters or less.
10. Put your name and GT username at the top of each file (if code, as a comment).
11. Coding guidelines
    1. You must provide a Makefile that compiles and links your code by default.
    2. Your code must compile with gcc on Linux. If your code does not compile you will get a zero for the assignment.
    3. You will be penalized if your code produces warnings when compiled with the following flags: -Wall -pedantic -O2
    4. You must turn in ALL files that are used to build your executable, including the files which we provide you - no matter what makes submit returns or what the assignment says.
    5. Code should be well commented and use a clean, consistent (readable) style i.e. proper indenting, etc.
    6. You must use the provided script/Makefile to package your submission if one is given. Much of the grading is automated with scripts, and failing to submit the proper format may delay receiving a grade on your submission.

Please be careful when submitting assignments, and double-check each one! Ask a TA if you are unsure of the procedure.

Late Policy

Students are given **ONE** use of one-time-forgiveness throughout the semester. When used, the project/homework may be turned in up to 3 days late without penalty. The usage is all or nothing, so you cannot use one day for 3 projects. We will assume the first late assignment you turn in uses this one-time forgiveness unless otherwise specified. The one-time forgiveness applies **only** to projects and homeworks, it doesn't apply to missed demos or failure to register for demos.

If you have already used your one-time-forgiveness, you will receive -25% for each day it is late. For example, a project that is two days late gets -50% so an 80 would be a 40. If it is later than 3 days, then it is a 0.

Prerequisites

CS2110
or
ECE2020 and ECE2035

Examinations

1. As a rule, makeup exams will be given ONLY for absences authorized by the institute and/or medical conditions (for either of them providing documentary evidence is the ONUS of the student).
2. No exceptions will be made for family-related personal travel (exceptions will be made if the travel is necessitated due to the health condition of a family member; the onus is on the student to provide documentary evidence).
3. Exceptions for personal travel that is for an interview will be made on a case-by-case basis. It is the responsibility of the student to make every attempt to ensure that the interview does not conflict with a posted exam date. For example, even if the company proposes a date that conflicts with an exam date, the student should let the company know of the conflict. Nine out of 10 times, the company will respect such conflicts and give a different date. There could be some extenuating circumstance due to which the company may not be able to offer a different date. The student should provide convincing evidence of such a circumstance to get an exception for a makeup exam. Any exception should be obtained PRIOR TO THE EXAM DATE.

Participation Grade

- Attendance quizzes account for 2%
- Lab attendance quizzes account for 2%

Grading Breakdown

Participation (Attendance Quiz Response + Lab Attendance)	4%
Projects (5 of them weighted equally)	25%

Homework	11%
Tests (5 of them weighted equally)	60%
Total	**100%**