Syllabus

CS2110 Computer Organization and Programming

Sections B/C

Fall 2021

Introduction

This is not a class where we are going to "learn you" about how computers work and how to program in C. This is a class where we are going to teach and you are going to do the learning. What does this mean? It means that you must become highly engaged in active learning. Read the textbooks (more than once is allowed), search the web, and above all work problems and CODE. After every lecture and lab go home and try stuff out. Verify that what we told you really works. Test your knowledge by extending what we tell you. (i.e., *if I understand this then I should be able to make the following happen*.) Imagine trying to learn to play tennis by listening to someone describe the theory and then watching another person play. The absolute key to success in this course is to *practice* diligently. Good luck.

Class

- You are required to have a reliable personal computer running Windows, Mac OS, or Linux.
- You are strongly encouraged to attend the lecture and required lab, and to participate. You are responsible for keeping up with all class material, as it is presented. There is credit given for lecture and lab attendance.
- Any changes to this syllabus will be announced using Canvas Announcements.

Purpose

1. To understand the structure and operation of a modern computer from the ground up.
2. Understand basic hardware concepts: digital circuits, gates, bits, bytes, number representation
3. Understand the Von Neumann model and the structure and operation of a basic data path
4. Understand the structure and function of machine language instructions
5. Understand the structure and function of a symbolic assembly language
6. Understand basic concepts of computer systems such as the runtime stack, simple I/O devices
7. Introduce the C language with particular emphasis on the underlying assembly and machine language as well as interaction with hardware.

Instructor

Caleb Southern
Office: CCB 220
Office Hours: TBA (see Canvas), and by appointment (email me)
Email: caleb.southern@gatech.edu (include 2110 in the subject line)

Teaching Assistants

(See the Canvas course website for TA office hour schedule)

Shawn Wahi (Head TA) shawn.wahi@gatech.edu

See Canvas course page for other TAs and office hours

Outcomes

- (Competency Knowledge) Be able to identify and/or construct basic digital structures such as MOS FET logic gates, decoders, multiplexors, adders, memory.
- (Competency Application) Be able to construct a state machine diagram and then implement it as a finite state machine circuit.
- (Competency Comprehension) Understand data representation. Be able to convert numbers between various representations: Binary, octal, decimal, hexadecimal, and IEEE Floating Point.
- (Competency Knowledge) Be able to identify the component parts of the Von Neumann Model of computer and be able to explain the purpose of each component.
- (Competency Synthesis) Be able to write, debug and run assembly language programs including recursive subroutines, traps, basic input/output.
- (Accomplishment Synthesis) Be able to write, debug and run multi-file C programs several hundred lines long using "make" to compile and execute said programs.
- (Competency Synthesis) Be able to utilize (in C programs) proper typing and casting constructs, structs, pointers and arrays, functions, function pointers, dynamic memory allocation and variables of different storage classes (auto, static, volatile, etc.)

Course Topics

1. Course introduction and overview
2. Bits, Data Types, and Operations
3. Digital Logic Structures
4. The Von Neumann Model
5. Introduction to a simple microprocessor, the LC-3
6. Programming
7. Assembly Language
8. I/O
9. TRAP Routines and Subroutines
10. Stacks, programming examples.
11. Introduction to Programming in C
12. Variables and Operators
13. Control Structures
14. Functions
15. Debugging
16. Recursion

17. Pointers and Arrays
18. I/O in C
19. Data Structures in C

Textbooks

Required

Introduction to Computing Systems, 3rd Edition
Yale N. Patt, Sanjay J. Patel
McGraw-Hill
ISBN: 9781260150537

The C Programming Language, 2nd Edition
Brian Kernighan, Dennis Ritchie
Prentice Hall
ISBN: 9780131103627

Recommended

This is a recommended book if you are new to Linux. Buy it if you are the type of person that likes to have a reference book when learning something new, but know that this content is available from various places on the internet. There will be no required readings or assignments from this book.

Mastering Modern Linux, 2nd Edition
Paul S. Wang
Taylor & Francis Group
ISBN: 9780815380986

Course Structure

- The course will consist of lectures and labs.
- There will be lots of homework assignments.
- Homework assignments will be mostly based on material already covered in lecture.
- One of the purposes of the lab is to give students help with homework assignments.
- Labs will be a mix of instructional labs (e.g. might have a lab on “makefiles”) and assessments (i.e., tests: *did students learn what they were supposed to from the homework?*)
- Homework must your individual work and *not substantially identical* to anyone else’s submission.
- In the schedule, there are reading assignments and suggested exercises that are to be completed **before** the lecture on which they are listed. There is no explicit credit for completing these exercises. They are for your practice and learning.
- There will be some quizzes in lab. Some will be timed coding exercises; some will be paper and pencil quizzes. Or they might be online.
- There will be a final examination

Course Evaluation

The number of and value of each assignment type is show below (subject to change):

Item	**Number (approx.)**	**Totals**
Homework	10	30%
Quizzes	4	20%
Timed Labs	4	20%
Lecture Attendance		2%
Lab Attendance		3%
Final Exam	1	25%
TOTAL		**100%**

Schedule

The initial course schedule is published to give students a view of the plan for the semester. As the semester progresses this schedule may change. Changes will be announced on Canvas. **Students should consult the living schedule on the Canvas course website to see the official due date for assignments.**

Final Exam Procedures

The last two weeks of the semester are divided among Final Instruction Days, Reading Periods and Final Exam Days. Note the homework due on the final instructional days.

- The final homework will be due on December 6, during the final instructional days.
- The final exam will be given per the Georgia Tech official schedule; see the Living Schedule link on the Canvas course page for the specific period for your section's final exam.

Academic Misconduct

- Academic misconduct is taken very seriously in this class and it greatly upsets your instructors because of the danger to you and your future. While you may think it is "just a homework", what we call academic misconduct can end a career instantly in the professional world. Let's just not encourage such a habit.
- If you get behind in your assignments, turn in what you've got and talk with your instructor, not necessarily in that order. You'll ultimately come out better than if you turn in someone else's work.
- Quizzes, timed labs and the final examination are individual work. Period.
- Homework assignments are collaborative at a high level – but you must turn in your own original work. In addition, many homework assignments will be evaluated via demo or code review. During this evaluation, you will be expected to be able to explain every aspect of your submission. Homework assignments will also be examined using electronic computer programs to find evidence of unauthorized collaboration.

- Studying is collaborative. You are encouraged to study old tests, do suggested problems, etc. together. Explaining an answer is one of the best ways to learn it.
- What is unauthorized collaboration? You must code each individual programming assignment by yourself. You may work with others at a high level, but each student should be turning in their own version of the assignment. You may teach each other, in the same manner that an Instructor or TA would teach you. If your collaboration and teaching is sufficiently detailed such that your submission appears substantially identical to that of another student, that is considered unauthorized collaboration. Submissions that are substantially identical will receive a zero and will be sent to the Dean of Students' Office of Student Integrity. Submissions that are copies that have been superficially modified to conceal that they are copies are also considered unauthorized collaboration.
- You are expressly forbidden to supply a copy of your homework to another student via electronic means. If you supply an electronic copy of your homework to another student and they are charged with copying you will also be charged. This includes *storing your code on any site* which would allow other parties to obtain your code, including (but not limited to) public repositories, etc.

Rules and Regulations

1. **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 and that you have followed the specific submission instructions for that assignment. Each assignment will have an official due date and time. For homework assignments only, a 24-hour grace period is automatically available for a 25% penalty. After the grace period no credit will be given for the assignment. It is your responsibility to plan and ensure that you have backups, early safety submissions, etc.**

2. It is your responsibility to schedule and attend makeups for any evaluations you miss during an excused absence. You must have an official excuse from the Dean of Students or the Registrar for excused absences or make-up work. Contact the Head TA and your instructor to make arrangements You are responsible for following up with your instructor promptly to make sure your make-up work is received and your correct grades are shown in Canvas.

3. All grades as entered in Canvas on the final day of class will be considered your final grades for each assignment at that time (with the exception of any final homework, quizzes, or timed labs that have not yet been graded). While we do all we can to avoid errors in grade recording, It is ultimately your responsibility to verify your correct grades are entered in Canvas and to notify us of any clerical errors.

4. 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.

5. Quizzes 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**.**

6. 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. Get help early! Grades are based on demonstrated performance and not individual need-based factors external to the course. Please do not request special consideration based on this type of situation. There is no "extra credit" given in this course.

7. If you find yourself earning grades that disappoint you, do not delay in seeking help. Piazza, your fellow students, the TAs, and your instructor are all available to you.

8. The TAs that are assigned to your lab section are responsible for conducting the labs and grading your assignments.

9. Final grades will be available from OSCAR normally the Tuesday following the end of term. Once you have taken the final exam you should direct any and all questions to your instructor, not your TAs. You may review your final and discuss your grades during the **following semester** in which you are attending Ga Tech. Grades will not be discussed during the break except in emergency cases.

10. There are significant resources available to support you. Free 1-to-1 tutoring is available for most 1000 and 2000 level courses. Students can book up to two appointments per week for each course in which they are currently enrolled. Tutoring appointments can be scheduled up to two weeks in advance, but no sooner than 24 hours before the appointment start time. For more details, visit https://tutoring.gatech.edu/tutoring

11. If you encounter a major personal problem (family/illness/etc.), please contact the Dean of Student's office, located in the Student Services Building (Flag Building) next to the Student Center or via https://studentlife.gatech.edu. The Dean's staff and associated resources are equipped to help you and are authorized to verify the problems and issue notifications to all your instructors making them aware you are having difficulty (without disclosing details) and requesting whatever extensions, etc. that may be necessary. If you have issues of any kind that affect your performance in the course: Personal problems, illness, accidents, etc., please seek help from the Dean of Students' office, your instructor, or another trusted individual.

12. The official announcements and any email from the class should be checked and read every day. Our official course site is at http://canvas.gatech.edu. Make sure you log in at least once to *check that your Notification preferences for Announcements* do not delay them. Announcements about course matters will be posted to Canvas. Piazza 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.

13. Out of consideration to your fellow students, please stay muted when you are not waiting to speak turn off cell phone alerts, beepers, wristwatch alarms, etc. Also, make every effort to be on time for class in person classes. Important announcements are often made at the very beginning of the class period. If you are unavoidably late to an in-person class, please sit near the door and try to avoid as much disruption to the class as possible.

14. If you are graduating and need this course to do so, please inform your instructor as soon as possible.

15. Complaints about TA's should be directed to the course instructors during office hours or via email as soon as possible.

16. 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 and tests not picked up will be destroyed.

17. Students participating in approved university activities or religious observances MAY be given extensions, be allowed to take tests at different times, etc. It is important to request such items well in advance and to supply supporting documentation.

18. Please remember that your TA's are also students. They have very full schedules and they are TAing out of a genuine desire to help you learn the material.

19. In order to help you in your programming assignments, we may provide you with "autograder" test cases that will exercise your code. It is certain that these test cases will not test every feature in your code, so thorough testing is still **your** responsibility. In addition, there may be unintentional errors in these test cases that cause the autograder to give null or wrong answers. The programming assignment is always the authoritative document that describes the correct results. Please plan accordingly.

20. You are responsible for backing up your computer. We highly recommend some sort of automatic off-site backup. If you have a catastrophic computer failure we will work with you while you get your hardware fixed but we cannot accept having no backup as a reason for an extension.

21. Note that in unusual circumstances, grades may need to be changed to correct grading errors after they are released. You will be notified if changes occur and will have the opportunity to request reconsideration.

Lectures

Lectures are held on Tuesdays and Thursdays. Attendance in class is strongly encouraged.

There may be an "Attendance Quiz" for each lecture in Canvas Quizzes which must be completed within 24 hours of the end of lecture time with no grace period. The credit for each quiz is minimal, but keeping up with the Attendance Quizzes helps keep us all on the same schedule. Officially excused absences will be accommodated.

Labs

While lectures are held on Tuesday and Thursday, labs are held during the blocks on Monday and Wednesday. Participation in and attendance at lab are required. All students have signed up for those blocks, so there should not be any time conflicts with other courses. Attendance counts in labs.

Labs are designed as follows:

- Attendance is taken at Labs; more than 3 officially unexcused absences will result in a deduction of 0.5% from the 3% lab attendance grade for each additional unexcused absence (resulting in a loss of up to 3% of your final grade).
- The goal of Lab is not for TAs to stand in front of the room and lecture on the same material that was already covered in class.

- Lab is designed to have students actively working on sample problems and assignments that deal with the material that was previously taught.
- The TAs cover any overarching announcements for the course and then let the students work on their assignment for the day.
- Students can still come to the TAs with any questions about the course or the assignments.
- Depending on the day, students can expect to either work on a regular lab, take a Quiz, or take a Timed Lab.
- On regular days, a lab assignment is given to the students to be completed.
    - This is an assignment that reinforces the concepts that were taught in lecture.
    - Generally, these assignments also deal with similar concepts that the current homework covers.
    - Students are encouraged to talk and to help each other as they work through the lab.
    - The TAs will address any questions on individual basis, or to the class as a whole if those questions persist.
    - The students are required to submit their lab assignments for each lab session.
- On Quiz days, a quiz will be given to the students for a specified time.
    - Collaboration on the quizzes is not allowed.
    - Use of any outside resources is prohibited, unless explicitly allowed in the Quiz instructions.
    - The quizzes will generally take all of the lab time for that day.
    - If the quiz is not scheduled to take the entire lab period, there may be a regular lab assignment to be completed on that quiz day.
- On Timed Lab days, a “Timed Lab” assignment will be given to the students.
    - A Timed Lab resembles a regular lab or a homework problem.
    - Collaboration on the Timed Labs is not allowed.
    - Students may use any previous labs or homework assignment that they have worked on as reference for the Timed Lab. No other material may be used, unless explicitly allowed in the Timed Lab instructions.
    - Students have a specified period in the Lab period to work on the Timed Lab, and they must submit it before leaving.

Homework

- There will be approximately 10 homework assignments which should be submitted by the published due date (see the living schedule).
- Homework assignments will be largely based on material covered in lecture; lab evaluations, and the final exam may be based on the homework materials.
- Late policy
    - All **homework** deliverables are due at the times shown in the Course Schedule. These times are subject to change so please check the living schedule and announcements on Canvas.
    - Your deliverable may be submitted (and resubmitted) up to 24 hours after the official deadline **with a 25% penalty**; Canvas or Gradescope will mark your submission as “late.”

- Submit early and often. **We will only grade your most recent submission.** Don't worry about the version numbers Canvas assigns**.**
- Any deliverable submitted after the grace period will get **zero** credit.
- We will **not** consider late submission of any missing parts of a deliverable. To make sure you have submitted everything, download your submitted files to double check. If your submitting large files, you are responsible for making sure they get uploaded to the system in time. You have 48 hours to verify your submissions!
- No penalties will be assessed for excused medical reasons or emergencies. Should one arise, you should contact the Dean of Students office to arrange for an excused absence, and notify your instructor that you have done so (in advance if possible).

Demos

1. You can sign up with any TA for any time slot, but it is your responsibility to do so at least 24 hours in advance. Sign up for whatever time is convenient for you; if you cannot make any of the available times, you must notify the Head TA immediately or risk not being able to demo.
2. If you miss your demo without a valid excuse, you will get a zero and you may not sign up for another demo slot. If something comes up after your demo is locked in that might make you miss your demo, tell us immediately. We can work with you or possibly allow you to change your demo time if you let us know BEFORE your demo.
3. Your demo time is locked in 24 hours before your demo. You cannot change your demo time after that time and we assume that you will be there. See the previous item. If you try to change your demo time without talking to us, we will know it and you will get a zero.
4. **Each student will be allowed to re-demo one assignment per semester with a 50% penalty**

Requirements for C Programs

Your programs will be tested and graded on Ubuntu 18.04 LTS 64-bit. If you are using something else and it works on your computer but not on ours, you risk getting a zero. So our advice is to run one of these two (even if you have a Mac). For this purpose, we have created a Docker image for you to use. If this doesn't work for your computer, you may use VMWare Player or dual boot. We will provide information about setup and installation.

All programming assignments must:

- build cleanly (i.e. no warnings or errors) on a linux box with:

```
gcc -std=c99 -Wall -pedantic -Wextra \
     -Werror -O2 -Wstrict-prototypes -Wold-style-definition
```

- exit gracefully (no segfaults, bus errors, etc) if appropriate
- produce useful output where applicable (i.e. error messages)
- not leak memory if applicable (we use valgrind to check this, you should too!)

- Not produce spurious output (no debugging output, extra messages to the user, etc. - this breaks the grader and gives you a zero)

Capital crimes (== automatic 0):

- non-building (how can we grade it if it doesn't build?)
- core dumps, including segmentation faults (or any other ungraceful exit)
- infinite loops (makes grading quite difficult, don't do it!)
- spurious output (see note above)