Syllabus

Welcome

In this course, each voice in the classroom has something of value to contribute. Please take care to respect the different experiences, beliefs and values expressed by students and staff involved in this course. My colleagues and I support UMass's commitment to diversity, and welcome individuals regardless of age, background, citizenship, disability, sex, education, ethnicity, family status, gender, gender identity, geographical origin, language, military experience, political views, race, religion, sexual orientation, socioeconomic status, and work experience.

View this syllabus as a guide to the course. It provides important information regarding the course, its assignments, policies, grading, and available university resources.

You should read it once, thoroughly, at the start of the semester. However, this document should be considered a working document. It is possible throughout the semester that a topic may take more time than expected, topics or assignments may change, or some material may be canceled or delayed due to a snow day or another emergency. If that is the case, the syllabus and schedule will be updated and a revised version will be posted on the course Moodle.

Course overview

Description: The course begins with a discussion of data representation in C, and moves up the stack from there to the features of modern architectures, assembly languages, and operating system services such as I/O, process, and synchronization.

Prerequisites: COMPSCI 187 with a grade of 'C' or better. Open to Computer Science majors only. Students may only take one of 220 and 230 in the same semester. The UPC will drop students enrolled in both from one of the two courses. It is recommended that students take COMPSCI 198C before taking COMPSCI 230. CS minors, applicants-on-contract, and others not meeting eligibility, or students needing special permission must request overrides via the on-line form at https://www.cics.umass.edu/overrides.

Objectives: COMPSCI 230 introduces students to the basic principles of modern computer systems. Through a combination of lecture, labs, problem sets, and programming assignments, students expand their ability to understand, reason about, and utilize these principles and related knowledge. Key concepts to be covered include data representation, modern machine architecture, assembly, the memory hierarchy, the process abstraction, threads and synchronization primitives, and systems programming concepts, including network programming.

What, when, where, who

COMPSCI 230: Computer Systems Principles
Tuesdays and Thursdays, 1:00-2:15pm in Hasbrouck Lab Add, room 20
Labs meet Fridays in various locations (see SPIRE for your lab section time)

Instructor: Marc Liberatore (please call me “Marc”)
Email: liberato@cs.umass.edu (though see “Communication Policy” below before emailing me!)
Phone: 413-545-3061 (on campus: 5-3061)
Office: Computer Science, Room 318
Office hours: TBA

Graduate Teaching Assistants (TAs) and Undergraduate Course Assistants (UCAs):
There are so many! Please see Moodle for details on people and office hours and the like.

Supplemental instruction is available from the UMass Library’s Learning Resource Center. Our SI leader is Aadam Lokhandwala alokhandwala@umass.edu. Visit https://www.umass.edu/lrc/si.html for schedule and other information.

Note that regular office hours end when classes end! Unless you hear otherwise, we won’t be holding regularly-scheduled office hours after December 12th.

Required and optional material

An Internet connection is required.

An EdLab account is required, as are the ability to access the EdLab remotely and some familiarity with UNIX terminal usage.

A computer with a modern C development environment, preferable a UNIX-compatible one, is required, as is reasonable proficiency in C programming and debugging. Most Linux and macos systems can be configured to provide an appropriate environment; you may be able to work under Windows, or in other systems. As time and inclination permit, course staff may choose to help with minor technical issues, but we are not IT support staff: We cannot generally solve installation or configuration issues, especially remotely, and we cannot promise to assist with C programming or debugging issues.

For portions of this course, we will use two free online textbooks:

- Dive into Systems
- Operating Systems: Three Easy Pieces

Both have non-free eBook and paper versions available, if you prefer those formats.

You may also find the following resources helpful:

- https://www.cs.rochester.edu/u/ferguson/csc/c/c-for-java-programmers.pdf
- Modern C by Jens Gusedt; you can download a PDF here: https://gustedt.gitlabpages.inria.fr/modern-c/

Code of conduct

- The course staff are committed to providing a friendly, safe and welcoming environment for all, regardless of level of experience, gender identity and expression, sexual orientation, disability, personal appearance, body size, race, ethnicity, age, religion, nationality, or other similar characteristic.
- Please be kind and courteous. There's no need to be mean or rude.
- Respect that people have differences of opinion and that differing approaches to problems in this course may each carry a trade-off and numerous costs. There isn't always a single right answer to complicated questions.
- Please keep unstructured critique to a minimum. Criticism should be constructive.
- We will informally warn you, once, if you insult, demean or harass anyone. That is not welcome behavior. After that we will report your behavior to the Dean of Students office. We interpret the term "harassment" as including the definition in the Citizen Code of Conduct under "Unacceptable Behavior"; if you have any lack of clarity about what might be included in that concept, please read their definition and then ask us for clarification. In particular, we don't tolerate behavior that excludes people in socially marginalized groups.
- Private harassment is also unacceptable. No matter who you are, if you feel you have been or are being harassed or made uncomfortable by a member of this class, please contact a member of the course staff immediately (or if you do not feel safe doing so, you should contact the Chair of the Faculty of CICS, currently Prof. Erik Learned-Miller, elm@cs.umass.edu, or the Dean of Students office). Whether you've been at UMass for years or are a newcomer, we care about making this course a safe place for you and we've got your back.
- Likewise any spamming, trolling, flaming, baiting or other attention-stealing behavior is not welcome.

Communication policy

Per the University Email Policy, you are expected to check your University email regularly – at least once a day. I will use your UMass email address as your primary point of contact in all online tools we use (Moodle, Campuswire, and Gradescope) and as my primary means to contact you individually outside of class. Group announcement will be posted to Campuswire, which can be configured to send you via email whenever an instructor makes a post.

Content-related questions

For course-content related questions, especially questions that other students might benefit from seeing the answers to, please use Campuswire. Course staff typically respond to questions within one to two business days, but I (Marc) do not typically respond to communications after about 5pm or on weekends. Course staff tend to get a higher volume of messages when a deadline is approaching. If you contact the course staff through the above channels at least two full business days *before* a deadline, you are guaranteed a reply before the deadline. Otherwise we'll do our best, but no guarantees.

Other questions

There is a course contact form, linked to on the Moodle page, where all unusual requests should go. This course typically has an enrollment of around 300 students and almost 20 course staff. We cannot handle the logistics of emails, DMs, phone calls and the like going to different course staff haphazardly – we need these unusual requests to be centralized.

By “unusual requests,” we mean questions about missed deadlines or exams, notifications of illness or injuries, or the like, containing personal information. “Usual” questions about course content or simple logistics should go through Campuswire.

These unusual questions will be reviewed by graduate teaching assistants or the instructor, not undergraduates (though note that all course staff are bound by FERPA) and will be responded to in a timely fashion. If the inquiry is too personal to submit through this form, please submit a note through the form indicating that fact, and contact the instructor directly by email.

In particular, *do not* use Campuswire Direct Messages (DMs) to contact me (Marc) – I do not check them regularly.

Campuswire

Campuswire is a online discussion management system. It will be used as the main hub for questions and answers in this course. Campuswire is a great tool but it can be misused. Please follow these guidelines in your use of Campuswire:

- You should use Campuswire to ask questions and get advice on assignments. But you should not use Campuswire to step through each and every problem you encounter in an assignment.
- You may not post assignment solutions to Campuswire, either in questions or answers to others’ questions.
- If you must post code you are working on, you should do so only through private posts (visible only to the course staff).
- You should not post code without a thoughtful and articulate question. Do not post code and ask only, “what is wrong with my code?” See, for example, http://stackoverflow.com/help/how-to-ask or https://jvns.ca/blog/good-questions/ for constructive advice on asking questions.
- You are encouraged to help other students by answering questions!

The course staff will monitor Campuswire and answer your questions in a timely manner (generally within a business day). But do not expect us to provide real-time answers on Campuswire, especially in the last few hours before an assignment is due!

If a question has already been answered in a previous post we may not respond to you, instead directing you to the previous answer. If a question does not follow the guidelines above we may not answer it. If a question is out of scope for this course (for example, related to the basics of C programming or debugging) we may not answer it. If we find that a private question is relevant to a larger audience, we may make mark it public to help others in the course.

Time management and what to expect

As a general guideline, the university suggests that students spend three to four hours of time on a class per credit hour. This is a four-credit course, therefore you should plan to spend twelve to sixteen hours a week on this class.

In a typical week, you will:

- do the assigned reading
- attend lecture, participate in any in-class exercises, and read the associated lecture notes as needed
- attend lab where there will be an exercise/worksheet to complete (and/or occasionally you will take a project quiz)
- complete the lesson quiz (demonstrating mastery with unlimited retries) and then the timed weekly quiz
- complete one larger programming assignment (about every other week)
- optionally, attend office hours

Attendance

I expect you to attend lectures and lab, and to complete all assignments by their due date. Generally, submitting clear documentation about the circumstances for any absence makes excusal, makeups, or permission to submit assignments late more likely to be approved. Makeups and excusals are only guaranteed as required by the Academic Rules and Regulations.

- If you will be absent or miss deadlines due to religious reasons, you must provide me with a written list of such dates within one week of your enrollment in the course.
- If you will be absent or miss deadlines for a University-related event, such as an athletic event, field trip, or performance, you must notify me as soon as possible.
- If you are absent or miss deadlines for health reasons, I expect you to notify me as soon as possible – generally before, not after, the missed course element.
  If you miss an assignment deadline for health reasons, I may excuse you from the assignment or grant you permission to submit late if that is your request. Everyone gets sick from time to time and I will not generally require documentation for one or two missed deadlines. More frequent missing of deadlines for health reasons implies a multi-week illness or chronic condition, for which I will require documentation. Note that if you are in isolation due to COVID, the UMass contact tracers will provide you documentation on request.
- If you are absent or miss deadlines for other extenuating non-academic reasons, such as a military obligation, family illness, jury duty, automobile collision, etc., I expect you to notify me as soon as possible.

If you must miss an exam for an excusable reason, I will work with you to find an acceptable time for you to take a makeup. If you miss an exam without prior notice, I will require an explanation and clear written documentation in order to judge whether the absence is excusable.

If you add the class late, I will excuse you from missed work, but you are responsible for both notifying me in a timely fashion when you add the course, and for completing the work on your own.

Incompletes

Incompletes will be granted only in exceptional cases, and only if you have completed at least half the course, with a passing grade. Prior to that, withdrawal is the recommended course of action.

Schedule

The course Moodle contains a week-by-week schedule. Approximately, it is as follows:

Week 1: Introduction and Systems Overview
Week 2: Data Representation
Week 3: Systems Fundamentals
Week 4: Machine Organization
Week 5: Machine-Level Representation of Programs
Week 6: Memory Hierarchy
Week 7: Virtual Memory
Week 8: Processes
Week 9: Threads and Synchronization
Week 10: Signals and Pipes
Week 11: Network Programming
Week 12: The Rust Programming Language
Week 13: Course Summary

Grading

I urge you not to overly focus on numeric scores and grades. Most students pass this course! So instead, focus on where assessments show that you can or need to improve. A decade from now the grade you got in this course will be irrelevant, but what you learned about computer systems might be crucial.

Nonetheless, in the midst of a busy semester we all end up with moments of triage in which we need to understand where to concentrate our efforts. So to give you a sense of the relative importance of each form of assessment, we expect the breakdown for the final course grade to be as follows:

10% lesson quizzes
10% lab exercise/worksheet
10% weekly quizzes
35% exams
35% projects

The numerical cutoff for final course letter grade assignment will be made after all grading is completed. Expect to require at least a 93 to get an A, a 90 to get an A-, an 87 to get a B+, an 83 to get a B, an 80 to get a B-, and so on.

Individual grade items are not usually curved, so you should not get stressed about means, standard deviations, etc. related to particular scores you receive. What matters is your weighted average; we do not give favorable (or unfair) treatment by raising or lowering individual students' letter grades.

There are no unannounced opportunities for extra credit in this course; please do not ask.

Also: It's 2022. Storage and bandwidth are virtually free, and backup software comes as standard on all commodity operating systems. Back your work up, store it in the cloud, whatever. "My computer crashed" won't be acceptable as an excuse in this class.

I will retain all graded materials for this course until the end of next semester. If you wish to review them, please come to see me during office hours (or make an appointment).

You are responsible for monitoring your grades. The course gradebook will be available through Moodle (though note that some grades will be available in Gradescope first). You should check your grades regularly and review any provided feedback. If you encounter any issues with your grades, you will have one week past the *first* posting of a particular assignment's grade to contact the course staff so that we can investigate. Please contact us via the regrade request system in Gradescope. We will not generally accept questions about an individual assignment's grade beyond this one week, so you must be prompt.

Lesson quizzes

A lesson quiz is a short 5–10 question true/false or multiple choice style quiz, typically given through Moodle. They are based on the lecture and assigned readings and are typically due weekly. Within a lesson quiz, you may not proceed to the next activity until you get a 100% on the current one. You are allowed to take these quizzes as many times until you get 100%. Collaboration on the lesson quizzes is allowed, but you should be sure *you* understand the material!

Lab exercise / worksheet

Most weeks in lab, you will complete an activity that will focus on preparing you for weekly topics. They are intentionally designed to be short and provide guidance to enhance your understanding of content for that week. Collaboration in labs is allowed.

Weekly quizzes

Weekly quizzes are short, timed evaluations given through Moodle. These **must be completed on your own**, without collaborating with your peers. They are typically based on the course material for the last week or two: lectures, lab exercises, and readings. You will have 50 minutes to complete these quizzes. They will be available at the end of the week after you complete all the activities for that week. You must complete the associated lesson quizzes with a 100% to attempt the weekly quiz.

Exams

There are two mid-term exams and a final exam. Each is cumulative.

These **must be completed on your own**, without collaborating with your peers or performing Internet searches or the like (they are closed-book, closed-note, closed-computer, etc.). It's critical that you (and we) be able to evaluate your *own* learning progress.

The mid-terms will be given at a University-scheduled time outside of class. They are currently scheduled for Tuesday, October 11th and Tuesday, November 8th, both at 6pm, in our regular classroom (HASA 20).

The final exam will be given on our scheduled exam day, currently listed as 10:30am on Wednesday, December 14th. **You must achieve a passing grade on the final exam to pass the class.**

Please note (from the Academic Rules and Regulations):

> …it is University policy not to require students to take more than two final examinations in one day of the final examination period. If any student is scheduled to take three examinations on the same day, the faculty member running the chronologically middle examination is required to offer a make-up examination if the student notifies the instructor of the conflict at least two weeks prior to the time the examination is scheduled. The student must provide proof of the conflict. This may be obtained from the Registrar's Office, 213 Whitmore.

You are responsible for clearing your schedule at the beginning of the semester to take exams. The Registrar announces the final exam period before the beginning of classes, and I will announce scheduled mid-term exams as soon as the Registrar provides me with that information. In particular, if you cannot commit to taking the final exam during the exam period, you should drop this class immediately. Makeup exams will be offered only in those cases where required by university policy.

Projects

There are six assigned projects; most require significant use of C and its associated tooling (debuggers, etc.). You are allowed to collaborate with a single partner on each project, sharing code freely with them. If you choose to do so, only one of you should make a single submission into Gradescope, and you should indicate it is a group submission (and identify your partner!) using the interface Gradescope provides. You do not need to work with the same partner on every project.

For some of the projects, you will be provided a “starter” project, which will require additions, modifications, or completion of various tasks. Most projects come with some automated tests to guide you. You will submit your projects to Gradescope which will run our autograder and apply additional private tests; some projects will also have a manually graded component, called “project quiz.”

Project quizzes

Some projects have an associated project quiz, which will probe your understanding of that project. These **must be completed on your own**, without collaborating with your peers. These will be timed evaluations given after a project is due. Understanding the projects will allow you to be successful in a project quiz.

Submission of graded items

You are responsible for completing and uploading your submissions before the regular deadline. The deadline for an assignment is not the time by which you must finish writing a solution; rather, the deadline is the time by which you must successfully upload your solution files and confirm the system has recorded the correct versions of those files. We recommend that you upload your files at least one hour before the deadline, in case Gradescope or Moodle happens to lag or go down near the deadline. File system timestamps on your local hard drive or in a Github repository or the like are never acceptable as evidence of existence of a file prior to the deadline, because you are under complete control of that timestamp. Requests to submit after the late submission deadline will be handled as described in "Extensions and late policy." Email submissions, whether late or on time, will usually not be accepted, regardless of whether the autograder or Moodle appeared to be online or not. If either did truly go down for an extended period, I will find out by the next workday, and the entire class will get an extension.

You are responsible for verifying that you have submitted the intended versions of your files. Similarly, you are responsible for ensuring that the final submission you make in either Gradescope or Moodle is the one you wish to have graded. Requests to substitute another version may be granted or not, entirely at the discretion of the course staff.

You are responsible for submitting code that compiles and runs on the autograder; if you submit code that does not compile or that gets stuck in an infinite loop, you will receive no credit. You should expect to receive the grade that Gradescope produces on the autograded portions; in particular, "But it runs fine on my computer" will not serve as the basis for an appeal. If you are seeing a discrepancy, you must ask us about it *before* the assignment deadline to receive a chance of consideration!

Attempts to manipulate, game, or otherwise incorrectly use the autograder will be treated as academic dishonesty, and will be penalized with a failure of the course.

Extensions and late policy

Assignments are due at the time specified in Moodle and Gradescope, and we strongly suggest that you always submit before that deadline, even if your solution is incomplete. That at least shows us that you are making progress.

Be aware that after Moodle or Gradescope decide that the late submission deadline has passed – even one second past it – no further submissions will be accepted without course staff intervention.

We will grant exceptional extensions in truly exceptional circumstances. Contact the course staff as described in the Communication Policy to request such an exceptional extension.

A message from Student Success

Your success in this course is important. Your instructor has partnered with Student Success and your academic advisors to assist you in better understanding course material which can aid you on your path to success. Resources available to students include: **Supplemental Instruction, ExSEL Group Tutoring, and 1:1 Tutoring**. Visit the Learning Resource Center online at http://www.umass.edu/lrc.

Throughout the semester, your instructor will communicate with Student Success & academic advisors regarding your progress in the course. If you are contacted, please consider scheduling appointments such as tutoring or academic advising and talk with your professor. Referrals are not punitive and are meant to assist you in connecting with resources at UMass. Please email academicalert@umass.edu if you have any questions or need assistance connecting with resources.

Academic honesty

General academic honesty statement

Since the integrity of the academic enterprise of any institution of higher education requires honesty in scholarship and research, academic honesty is required of all students at the University of Massachusetts Amherst. Academic dishonesty is prohibited in all programs of the University. Academic dishonesty includes but is not limited to: cheating, fabrication, plagiarism, and facilitating dishonesty. Appropriate sanctions may be imposed on any student who has committed an act of academic dishonesty. Instructors should take reasonable steps to address academic misconduct. Any person who has reason to believe that a student has committed academic dishonesty should bring such information to the attention of the appropriate course instructor as soon as possible. Instances of academic dishonesty not related to a specific course should be brought to the attention of the appropriate department Head or Chair. Since students are expected to be familiar with this policy and the commonly accepted standards of academic integrity, ignorance of such standards is not normally sufficient evidence of lack of intent.

In addition, you should read the UMass Academic Honesty Policy (ignorance of the policy is no excuse).

Course-specific academic honesty policy

Academic dishonesty is usually the result of other problems in school. Please come see me or the other course staff if you are unable to keep up with the work for any reason and we will do our best to work something out. The course staff want to see you succeed, but we cannot and will not tolerate academic dishonesty.

Investigating academic dishonesty is an unpleasant experience for both the instructor and the student. Please help me by avoiding any questionable behavior.

Be aware that if something looks like academic dishonesty to us, we will treat it as such, unless you can provide strong evidence to the contrary. When in doubt, it is your responsibility to contact the course staff about whether a potential action would be considered academic dishonesty.

What is permitted and what is not?

You may discuss material with others, but when collaboration is forbidden (especially but not limited to exams), your submission (code and prose) must be entirely your own. The exams are entirely closed-book.

More generally, you may not get help with your work from anyone who is not a current member of COMPSCI 230, either as a student or as course staff, or a specific University resource, such as a CICS tutor or LRC SI leader. “Help” includes designing algorithms, writing code, debugging, developing test cases, and so on.

You may not provide your solutions to others, either directly or via some sort of public or private posting, except when collaboration is explicitly permitted (as it is on problem sets and the programming assignments) and when both you and the other person(s) are currently enrolled in this course.

You may not copy code from online sources – except from the current semester website / Moodle site. Copying and pasting code from another student (whom you are not submitting with) or from a third party is a violation of academic honesty, and we will endeavor to detect this by any means available to us, including automated similarity analysis of submitted assignments.

You may not use third-party online forums such as StackOverflow to ask for specific help on assignments, nor third-party course “notes” sites that traffic in solutions to assignments, nor may you search for solutions online.

When you ask for help, either in person or on Campuswire, it’s good practice to ask your question by describing the problem you’re having, or using a small synthetic example that illustrates your difficulty. If you must include a large chunk of your code to ask your question on Campuswire, mark it as a “private” question, and only the course staff will be able to see it.

Other academic regulations

The Office of the Registrar publishes Academic Regulations yearly. You should be familiar with them. Particularly relevant are the policies on attendance, absences due to religious observance, and examinations.

A word about putting your solutions on GitHub, GitLab, BitBucket, etc.

Per the course-specific academic honesty policy, you are not permitted to make your solutions to the assignments in this class available to others. This includes reposting them to public GitHub repositories (or other service where another student might plausibly see them).

A word about copyrights

Some of the material (lecture notes, lectures, assignments, and so on) in this course is original work created by current and previous instructors (Tim Richards, Joe Chiu, and Marc Liberatore); exceptions are clearly noted. These works are protected by U.S. copyright laws and by university policy. The instructors are the exclusive owner of the copyright in materials they create.

You may take notes and make copies of course materials for your own use in this class. You may also share those materials with another student who is registered and enrolled in this course.

You may NOT reproduce, distribute, upload, or display any lecture notes or recordings or course materials in any other way – whether or not a fee is charged – without my express written consent. If you do so, you may be subject to disciplinary action under the UMass Code of Student Conduct.

While you are welcome to use the material for your own personal and educational use, you may not redistribute them to others outside the class. In particular, selling or otherwise redistributing your notes (or mine!), making or selling audio, video, or still recordings of course material, is not allowed without express written permission from me.

As Farshid Hajir (Senior Vice Provost and Dean of Undergraduate Education) noted at the start of the Fall 2022 semester, “students can only use class notes/recordings for their own personal use, and may not share (or sell) these notes via an outside vendor or entity without the instructor’s permission. Usage of the notes or in-class recordings in this way without the faculty member’s permission is a violation of the instructor’s copyright protection.” (This does not pertain to accommodations under the Americans with Disabilities Act (ADA), although recordings or sharing of Notes for ADA accommodations should not pertain to distribution beyond the students in the class receiving the accommodations.)

Accommodation statement

The University of Massachusetts Amherst is committed to providing an equal educational opportunity for all students. If you have a documented physical, psychological, or learning disability on file with Disability Services (DS), you may be eligible for reasonable academic accommodations to help you succeed in this course. If you have a documented disability that requires an accommodation, please notify me within the first two weeks of the semester so that we may make appropriate arrangements.

Acknowledgments

Some material taken from the Rust Code of Conduct.

Some material taken from the Cornell CS 3110 syllabus and related policies.