

COSC 112: Introduction to Computer Science II

Scott Alfeld

Spring 2022

Contact

Instructor

Scott Alfeld

salfeld@amherst.edu (Include '[COSC112S22]' as a prefix in the subject)

SCCE C218

Office hours: Will be posted on Moodle.

Also by appointment (just email me and we'll find a time to meet).

TA

Primary TA

TBD

Office hours: Will be announced on Moodle.

Evening Help Sessions

Posted on Moodle.

Textbook

None required.

Grading

Labs and Projects: 50 %

Exams: 50 %

There will be weekly labs and three projects plus a team-based final project. There will be an early exam, a midterm and a final exam.

Labs

This is a code-heavy class.

Every lab presented on a Monday will be due the following Monday (before the next lab). There are three possible grades for each lab: 0, 0.5, and 1.0. You'll receive a 0 if you turn in nothing, or if what you do turn in is close to nothing (e.g., you add `System.out.println('hello');` to the code you were provided). You'll receive 0.5 if you turn in something with serious flaws (e.g., it crashes, or has serious bugs). You'll receive 1.0 if your program works and fulfills the requirements laid out in the lab.

In addition, you are **highly encouraged** to play around. In this course you should spend a lot of time writing code and the labs are meant to serve as starting places, not finished products. Want to know if algorithm A is faster than B? Test them. Wondering if organizing your code differently would improve readability? (Make a backup copy and) try it. Always wanted a rock-paper-scissors training partner? Write one. In addition to developing your skills and cementing what you learn, this will help you prepare for the final project (see below).

I reserve the right to reward any lab submission with extra credit should you go above and beyond what's required for the lab.

Projects

There will be three projects you will do as an individual, and one project as part of a team.

The team project will be largely open ended, where you design and implement a program of your choice. More details to come.

Tentative Schedule

We will spend 1-2 weeks on each of the following topics, in roughly the following order:

- Java Review
- User Defined Classes
- Inheritance, Polymorphism
- Abstract Classes/Methods
- Data Structures (Lists, Stacks, Queues)
- Basic Probability, Randomized Algorithms
- Bonus/Advanced Topics (based on class interest)

Policies

Email

The best way to get a hold of me is via email.

All emails to me relating to this class should:

1. Be addressed to salfeld@amherst.edu
2. Contain [COSC112S22] as the first 12 characters of the subject unless they are a reply.
Replies may contain Re: as a prefix.
3. Have an informative subject (that is, dont just look for the last email you sent me and hit reply).
4. Contain [COSC112S22] as the first 12 characters of the subject unless they are a reply.
Replies may contain Re: as a prefix.
I have filters. This is important.
5. Adhere to rules 2 and 4, above.

If you send an email without the correct prefix, and then realize this, please do send a new email with the correct prefix.

Note: If you have not received a reply from me in 16 business hours, feel free to re-ping me about it.

Citing Help

On every assignment, you will be asked to include a list of what resources you used. This can range from the textbook, wikipedia articles, papers, other textbooks, stackoverflow answers, other students, etc..

In general: Working with other students is okay, within limits. You are allowed (and encouraged) to work with others in the class by discussing high-level concepts. Use your best judgement, but a good rule of thumb is that you should never show anybody your code, or give anybody direct answers to homework questions. If you do something that I believe is too far (e.g., sharing code with another student), you will get in trouble. However, if in your acknowledges section of the assignment (see above) you say what you did, you will be in substantially less trouble than if I catch you.

Working with People Outside the Class

Please refrain from working with anybody outside of the class. Your fellow students (in this class), the TAs, me, and official tutors and the like are fine, but avoid getting help from outside.

There are several reasons for this policy. First, there is an issue of fairness. You may have friends/family who are professional coders, superstars in the field, or AI agents themselves, but many students do not. Second, being able to learn from your peers is an important skill.

Late Submissions

Labs: No labs will be accepted late.

Projects may be submitted up to 24 hour late, at a one-third penalty (you get $\approx 67\%$ of the grade you would have gotten had you submitted it on time). Submitting programming projects more than 24 but less than 48 hours late will yield a two-thirds penalty. No credit will be given for programming projects submitted more than 48 hours late.

Note: 1 second late is late.

Early Submissions

Submitting a programming project early will yield 1% extra credit for every 24 hour period (rounded down) it is early, up to 5% extra credit for that project (if submitted more than 120 hours early).

This does not apply to Labs.

Cheating

Don't.

Reading or copying any solution from any source will be considered cheating. If you end up on a webpage which has a solution, leave the page immediately without glancing at the code.

Cheating on any written assignment or exam will not be tolerated.

There are three main reasons you should not cheat.

1. By cheating, you deprive yourself the education that you are here to get. You will learn far more by completing the assignment yourself (with appropriate help from others) than you would by copying a (partial) solution.
2. By cheating, you adversely affect the morale and potentially the grades of your fellow (honest) students.
3. If I catch you cheating, you will fail the class and be reported to the College.

Please note that I am good at catching cheaters. I use sophisticated algorithms and code to automatically analyze your submissions and compare them with other students work as well as online solutions. You may very well be clever enough to copy a solution and then obfuscate the code to the point where my systems

do not catch you. However, in this class, completing the assignment honestly will be a substantially easier and less risky task than fooling my systems.

More generally, there is a somewhat fuzzy line between working together and cheating. It's in your best interest to be honest if there's anything that's borderline. If you fixed a bug in your code by looking at a comment on stackoverflow, provide a link to it when you turn in your assignment. If you had a long conversation about the ins and outs of an assigned problem with a classmate that got a little too specific, make a note of that in your assignment. If you goofed up and in a medically-induced stooper accidentally copied code from online or asked Knuth how to do problem 3, tell me immediately.

As a general rule: If you do something that you're not allowed to and tell me about it, we'll work something out (e.g., you'll demonstrate to me that you understand the material in some other way). If you do something that you're not allowed to and you don't tell me, then bad things happen if I catch you.