

CS 112: Course Policies

CS 112 uses the C++ programming language to introduce students to the elementary data structures (vectors, lists, stacks, queues, binary search trees, ...), recursion, big-oh notation, and other intermediate-level programming topics.

Instructors: [Professor Victor Norman](#)

[Professor Eric Araujo](#)

Office Hours: click the above link to find our office hours.

Grading

Your grade in this course will be based on:

- Daily Quizzes: 5%
 - The lowest 3 quizzes will be dropped.
- Labs: 20%
- Homework projects: 25%
- Tests (5, including the final test during the Exam time): 50%
 - The lowest test score will be dropped.

Student Learning Outcomes

Upon successful completion of this course students will be able to:

- Use indirection (pointers) and run-time memory allocation to implement dynamic, generic container classes (e.g., vectors, lists, stacks, queues, trees, etc.).
- Analyze, categorize, and compare the complexity of data structure operations using "big-oh" notation.
- Design and implement recursive data structure operations.
- Design, create, and use unit-tests to test the correctness of data structure operations.
- For select algorithms, use simple parallel computing techniques to improve algorithm performance and measure the improvement.

Course Organization

The course is organized around these components:

1. The *lectures* that will introduce data structures topics at the conceptual level;
2. The *labs* in which students will (in pairs) practice using and building data structures; and
3. The *projects* in which students will (individually) practice using and building data structures.

Each week's lectures, labs and projects will cover the same *concepts*, but will do so in different ways.

The first 4 weeks will be dedicated to learning C++. After that, we will begin studying data structures.

Grading and Late Work Policies

1. Each exercise and project will have a **Due date** associated with it. This is the date by which you should submit your work in order to avoid falling behind in the course.
2. Each day that your submission is late, you will be assessed a penalty of 10% (unless you use grace days -- see below).
3. Periodically during the course, I will announce (in class or through email) that assignments have been graded.
 - a. **You will have 7 days from that announcement to check your grades** and make appeals for changes.
 - b. // Appeals are made through the GradeBoost Appeal app: <https://gradeboost.cs.calvin.edu>.
 - c. You **may also resubmit** your assignment and appeal to have it regraded. Again, use the GradeBoost Appeal app.
 - d. After the 7 days have expired, you cannot ask to have your assignment graded or regraded.
4. Assignments that are turned in very late (i.e., so late that at 10% per day, the assignment would receive a 0) will not be graded, but will be given a maximum score of 30%.

Grace Days

You have 8 Grace Days available to you during the semester. These grace days may be used to submit an assignment late without penalty. **Grace Days Remaining** will be seen in the GradeBook in Moodle.

Here are some circumstances in which you might use grace days:

1. You are an athlete and you have a very busy week. Use a few grace days to catch up the next week.
2. You become ill (physically or mentally) and cannot do the work on time.
3. You need to go home for an emergency.

Here is the procedure for using grace days:

- DO NOT email me to tell me you are going to use grace days. I don't need to know this -- there is no action I need to take if you tell me this.
- When your assignment grade comes back and you have lost points due to lateness, then use the Grade Appeal app to ask for the points to be restored and tell me how many grace days you want to use for that.

Extenuating Circumstances

If you fall ill (physically or mentally) for an extended period of time, AND you send me documentation from Student Life, Student Health Center, or the Center for Health and Wellness, then, and only then, I will consider allowing you to make up late assignments.

If you begin to skip class repeatedly due to mental health problems, go to the Center for Health and Wellness! Don't delay!

Incompletes

An Incomplete (I) grade will be granted only in unusual circumstances, and only if those circumstances have been verified by the *Student Life Office*. Procrastination does not qualify as an unusual circumstance.

No work will be accepted after the last day of classes.

Honesty.

Laboratory exercises are to be done in pairs, using pair programming. One copy of the assignment will be submitted for grading.

Unless specified otherwise, exams, homework assignments, and projects are to be done *individually*, and must consist of your own original work. Submitting another student's work as your own (even with their knowledge) is **academic fraud**; and any detected duplication of work will be dealt with in accordance with Calvin's policies on Academic Honesty. Likewise, giving another person your work to submit as their own constitutes academic fraud and may result in a failing grade (F) for both parties.

Feel free to discuss ideas with us or with your classmates, but don't copy code (i.e., plagiarize). Here are examples of what plagiarism looks like:

- You copy a program from an online source, change some small things (e.g., variable names, output messages), and submit the code without attribution.
- Your roommate writes some code, which you add to your program. You add documentation that shows you understand their code, but never indicate the source of the code.
- You find an online tutorial that walks you through the construction of a system that you then submit without attribution.
- You use a generative model (e.g., ChatGPT) to produce your code, review it, perhaps edit it a bit, and then submit it without attribution.

Consider these rules of thumb:

- If you found it efficient to use copy/paste or use a generative language model to create more than one or two lines of your application, you must document the original source of the code.

- If the moment you figure out how to do something occurs while you are looking at a website or at the output of a generative language model, you should document that website.

Note that these rules of thumb apply to the code supplied in this course's materials as well.

Our graders will be using a tool called the Measure of Software Similarity (MOSS) to check all submissions against all other submissions. If two submissions are found to be significantly similar, we will assume cheating has happened. Note that this software is pretty smart. It knows what computer language it is inspecting and can determine if two submissions are the same except for differences in variable names, e.g.

Note that if you and someone else both independently ask ChatGPT or Copilot (or some other LLM) to write code for you, and you both submit it, MOSS will detect it as identical and we will have to assume you cheated.

Classroom Policy

No phones, no laptops, unless I tell you you can.

This is a difficult class, and you don't need to be distracted by goofing around on your phone or laptop. And, I don't want you to distract others.

Daily Quizzes will be administered in the first 5 minutes of class, on Moodle. Do not show up late to class.

Attendance

Class attendance is optional.

Also, lab attendance **is required!** Lab is not optional. Skipping lab may result in you doing poorly in class.

Missing class due to illness: if you are going to miss class due to (physical or mental) illness, please send me an email **before class**.

Disabilities

Calvin University is committed to providing access to all students. If you are a student with a documented disability, please notify a Disability Coordinator in Student Success (located in Hiemenga Hall 227) to discuss necessary accommodations. If you have an accommodation memo, please come talk to me in the first two weeks of class. If you are a student needing reasonable modifications for pregnancy and related conditions, please contact the Title IX Coordinator in Student Life (located in Spoelhof University Center 364).

Statement of Diversity and Inclusion

Inside and outside of the classroom, I work to treat every person with equal respect, regardless of racial identity, gender identity, physicality, brain chemistry, nationality, political leanings, and religious background.

Whoever you are, you are fearfully and wonderfully made. God delights in diversity and I do, too. We humans need diverse viewpoints because every viewpoint, on its own, has blind spots. That has been proven to be true in technology-related fields, where under-represented minorities are significantly under-represented in the working world. When companies work hard to rectify this, they find their products and services improve. We have found this to be true in Computer Science classrooms as well.

If you or someone else in this class is hurt by something I say or do in class, I would like to know about it so that we can work toward a remedy. This has happened in the past in my classroom -- usually when I'm trying to be "cool" or be funny. If this happens this semester, please know that it was unintentional. Because this kind of feedback can be uncomfortable for all involved, I'll take it however it comes: in public or in private, on the spot or days later, directly from you or indirectly through another student, another professor, or through my department chair Keith VanderLinden (kvlinden@calvin.edu).

Also, please note that I am half deaf (at least). I am completely deaf on my left side. This results in my not being able to hear students that speak quietly, and not being able to tell which direction sounds are coming from. Please speak loudly! Thanks.