

CS 120 section 2, Winter 2025

Instructor: Michael Kowalczyk

Office: 2222 Jamrich Hall

Office phone: 227-1600

Class meetings: 2:00pm – 3:40am Monday and Wednesday in room 1311 Jamrich Hall

In-person “walk-in” office hours:

Monday and Wednesday 11:00am – 1:50pm

By-appointment online office hours:

Tuesday, Thursday, and Friday 3:00pm – 5:00pm

Reserve a time by going to ESP in MyNMU; reserve your appointment the day before (at latest).

At your appointed time visit <https://nmu.zoom.us/my/drkow> and use passcode: 147147

Email: mkowalcz@nmu.edu

Course Website: We will use <https://codio.com>, but the syllabus and a few links are on <https://educat.nmu.edu>

How to succeed in this course: 1) Show up to class, 2) work ahead, 3) if confused “Summon DrKow” in Codio.

Work at least a few days ahead of the deadlines! Keeping up with lab exercises is especially crucial, since that is where most of the learning happens. Lab exercises are graded in-person only and I will guide you to fix errors or omissions until your program is correct. Outside of class, if you are confused on something or want me to check your work over (lab, homework, worksheet, or even a video), click “Summon DrKow” in Codio, which will signal me to look at that particular exercise. If I’m online then we can video-chat right inside Codio, and I can coach you on your work. Otherwise, after summoning me you can leave a note by clicking the left-margin of your program code, and I can do the same for you when I log in and see that you have a help request. The key is to start early. If you wait until near the deadline before starting on an assignment, then there is less chance to collaborate in this manner, and you may waste lots of time on bugs that I could have explained quickly. Finally, you can click “Mark as completed” in Codio to take an item off your “To Do” list, but that doesn’t signal me to look at it before the deadline.

Overview:

This course is an introduction to writing computer programs using Java. Although I assume you have never used a programming language before, you will probably find the course challenging and interesting even if you have.

Prerequisites:

Mathematics Placement recommendation of CS120 or higher, or B- or better in either MA100, CS101, or CIS110.

Textbook:

None required, but I have links to e-books and other useful resources online.

Equipment:

You will need a laptop computer with a web browser and Internet access. You will also need to visit <https://codio.com/p/signup?courseToken=genesis-student> to sign up for the online development environment (Codio) that we will be using for this course. You can get a Codio license either through the bookstore or the website above (it’s around \$50 directly; a bit more if you go through the bookstore). If for some reason you don’t have a laptop computer, talk to me as soon as you can, since we will be using them often.

Grading:

Grades will be based on exams, homework assignments, labs, and quizzes. Homework assignments are weighted based on their size and complexity.

10% Labs

10% Quizzes

25% Midterm exam

25% Final exam

30% Homework (each worksheet weight=5; assignment weights are from 5 to 40 depending on complexity):

Homework	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Points	5	5	10	5	10	20	20	20	10	25	30	15	15	40	40

Handing in Work and Late Policy:

I allow for partial credit on almost everything, but not on labs.

Programming assignments: You will be using Codio as a platform for working on programming assignments. Your progress is automatically saved, and nothing special needs to be done to hand in an assignment. If you do your work in some other system, be sure to transfer your code well ahead of the deadline, as deadlines are strict.

Quizzes and worksheets: Worksheets are completed online in Codio. Quizzes are taken on paper during class on Fridays, but I will also give short unannounced quizzes at other times.

Exams: Exams are closed-notes and taken in person on paper. There is a 1-hour midterm and a 2-hour final.

Labs: Labs are handed in during class so I can review your work and give you instant feedback. If there is any error, I will require you to fix it before the deadline to get any credit on it, so keep a lab or two ahead of the game. **To double your credit on a lab**, do some challenge that goes above and beyond the basic requirements!

Late policy: Deadlines are strict. Labs are due by the end of class, and all other deadlines are 10:00pm. If for some reason you need to email some work as an attachment, go ahead, but I must receive it **before** the deadline. If some circumstance such as illness occurs which might interfere with getting your work in on time, let me know about it right away. I can usually work something out if your concern is brought up well in advance of the deadline. Treat it like a job. If you wait until days after the fact to say that you were sick, don't expect much slack.

Exam Dates & Schedule Conflicts:

The midterm and final exams are administered *on paper only*; no book, no computer, no notes. The midterm exam will be during our regular class meeting on Monday, March 17. The final exam will be held on Monday, April 28 from 2:00pm – 3:50pm. Let me know as soon as any conflict prevents you from attending (due to religious observances, illness, intercollegiate athletics, etc).

Laptop Use:

Your laptop will be needed nearly every day for labs or other in-class exercises. At times, we will have a lecture or discussion in which I will need your complete attention (laptops closed). You are responsible for keeping your laptop in good working condition. All work is auto-saved in Codio, but if you work offline in a different editor, make frequent backups. The helpdesk does not backup your work if they need to fix your laptop (unless you pay a fee), so make frequent backups to hardware external to your laptop *before* a crisis strikes.

Academic Conduct:

I strive for honesty and integrity and I expect my students to do the same. Every assignment must be written entirely by you. There are precisely two instances where including program source code from elsewhere is acceptable:

- You may include any code that I give out in my lab tutorials and lecture notes, without citation.
- Any other code that you didn't author **must** be accompanied with a full citation (this includes people, websites, books, etc.). **Indicate precisely any lines of code you didn't write**, and note that if you rely on sizable portions of code from elsewhere, that may of course reduce your credit earned for the assignment.

The best way to help others succeed in the course is by explaining concepts and working through examples – not by giving away source code.

Course objectives:

CS 120 is an introductory programming course. It forms the foundation for later CS courses, but it also satisfies the Quantitative Reasoning Requirement. Upon successful completion of this course, a student should be able to do the following in the Java programming language:

- Solve programming problems through the use of conditionals, loops, and nested control structures
- Write an instantiable class from scratch
- Write code to call constructors and invoke methods on existing objects, including correct use of parameters and return values
- Demonstrate an understanding of commonly used operators (logical, arithmetic, and comparison)
- Demonstrate a basic working knowledge of arrays and their syntax

Evaluation of these learning outcomes will be done through written assessments (quizzes and/or exams).

Quantitative Reasoning Requirement:

This course satisfies the Quantitative Reasoning component of your general education requirements.

Disability Services:

If you have a need for disability-related accommodations or services, please inform the Coordinator of Disability Services in the Dean of Students Office at 2001 C. B. Hedgecock Building (227-1737 or disserv@nmu.edu). Reasonable and effective accommodations and services will be provided to students if requests are made in a timely manner, with appropriate documentation, in accordance with federal, state, and University guidelines.