COMP 218: Data Structures and Program Design
Fall 2023

General Course Information

Professor: Noah Aydin **Office:** RBH 319 **Phone:** 5674 **E-mail**: aydinn@kenyon.edu
Course web page: http://www2.kenyon.edu/depts/math/aydin/teach/218
Office Hours: MW: 10:10-12pm; F: 10:10-11am, and by appointment (remote option available).
See my weekly schedule on course website. Zoom link for in-class activities and remote office hours.
Class Meetings: T & R: 2:40-4 pm. Location: PRCL 109

Textbook: Data Structures Using C, by D. S. Malik, Cengage, ISBN 9780324782011 (a pdf is available, see the course website)
IDE: We will be using the free online IDE repl.it. Please create an account there. More options on the course website.

Course Description and Learning Objectives: The main goal of this course is to learn basics of fundamental data structures used in computer science and practice using them to solve problems. These include: stacks, queues, lists, heaps, hash tables, trees, and graphs. We will also examine a number of searching and sorting algorithms. Both array-based and linked implementations are analyzed where appropriate. You will also learn how to use the pre-written classes contained in the Standard Template Library (STL). An introduction to analysis of algorithms and the big-O notation will be given. Analysis of major algorithms will be discussed. Another important learning goal in this course is to practice software engineering principles and write programs with good user interface. Good programming practices are emphasized. We will cover most of the chapters in the textbook. We will be using pair programming so that you practice software development as a team in a collaborative way. This is often a necessary and highly valued skill. Many real life software development projects are collaborative.
Prerequisite: Scmp 118 or equivalent, proficiency in C++.

Grading and Evaluation Criteria:
Final grades will be determined based on the performance in the following components.

Component	**Percentage**
Quizzes	20
Weekly Labs in Pairs	40
Midterm Exam	10
Participation & Oral Presentations	5
Final Exam	25

Quizzes: To encourage regular study, there will be a short quiz (about 10 minutes) every day. Most of the quizzes will be on Moodle, some on paper. You must bring a laptop to the class every day. Of the 20+n quizzes, the lowest n scores will be dropped. No make ups will be given for missed quizzes for any reason, except possibly for long-term special circumstances. The quizzes may cover content from the sections that you are expected to read for that day. Therefore, it is imperative that you do the readings *before* each class. We won't have time in class to go over every detail in the book. You are still responsible for the material. See the course website for the agenda and reading assignment for each day.

Labs: The programming projects are the most important aspect of this course. Consequently, they will have the largest weight in the final course grades. Programming assignments will be assigned weekly and some assignment consist of multiple programs. You will be working in pairs for the programming assignments. See the course website for more info on assignments and pair programming.

Exams: Midterm Exam- Tue Oct 17.
Final Exam- Final exam will have two components: Programming project and test. Due, Tue Dec 12, 6:30 pm.

Participation/Attendance/Engagement: Pedagogically, regular engagement with the course material is essential for deep learning and it is an expectation in this course. Staying healthy is a prerequisite for this kind of engagement. Unless you have a legitimate excuse, you are expected to attend the class meetings. Legitimate excuses include: illness, religious observations, college's official athletic events and similar situations. If you have a situation that prevents you from attending the class, please communicate with me as soon as possible. Timely communication is a key factor here. Math Dept's attendance policy applies to this course.

Much of the class time will be devoted to a discussion of the major concepts from the assigned reading and hands-on activities to practice the concepts. Therefore, attending class regularly and being prepared is essential. After each assignment is due, 2-3 people will be randomly selected to briefly present their program and algorithms. Make sure you can explain the code you submit for each assignment. Everyone is expected to actively participate in class discussions and activities. Your grade on this component will be based on the combination of your attendance, the level of your engagement in class activities and discussions, and how well you explain your code. Contributing to the discussions on Moodle forums will also count.

Program Grading: All programs will be graded according to the following components.

Correctness: Each program should conform to specifications stated in the problem statement. A program should demonstrate correct handling of ordinary input, special cases, and error conditions.

Design: Your programs should be modularized into coherent independent functions or classes with strong cohesion.

User Interface: Writing a reasonable test program with good user interface *is **always a default requirement for all programming assignments in this course*** since this course is also about *program design*. So, this requirement is always part of the assignments. Having solved a problem correctly is not good enough to get full credit. You need to write a good test program and design a good user interface as well. A good test program and a good user interface are not fully prescribed and they may change from program to program. It is something you need to think about for each assignment. An obvious example of a good user interface would be giving the user the chance to repeat a computation before exiting the program (let the user repeat as long as s/he likes). Make sure your program tests all aspects of the assignment. Another point to consider is that asking too much input from the user is not convenient.

Style and Documentation: Your program should be easy to read and understand. This involves program indentation, modular design, variable names, user interface and comments.

Efficiency: Algorithms should be efficient with respect to both time and space. You should spend thinking about designing good algorithms rather than using brute-force. Be prepared to justify your choice of algorithms.

NOTE: If a submitted program **fails to compile** it will be graded out of 50% of the total point value. If a submitted program has a **run-time error**, then it will be graded out of 75% of the original point value.

Late and Make-up Policy. There will be no make-up for daily quizzes For weekly programming assignments, each student will be allowed two "free" 24-hour extensions; no reason needs to be provided. Simply email the professor in advance of the due date (no later than the evening prior to the due date) to say you'd like to use one of your extensions. After the second extension, late lab assignments will not be accepted. **Do not modify your submitted files after the due date**, until graded. In case of pair programming, if at least one of the partners already used their two extensions, then the pair is not eligible for another extension. For exams, a make-up can only be granted with an official notice from one of the deans (the dean of academics or the dean of students).

Academic Honesty: The rules set forth in the 2023-2024 Course Catalog apply to all aspects of this course. In general, any work submitted for credit must result directly from your own understanding, thoughts, and ideas. Presenting the work of others as your own is strictly prohibited. In particular, using chatGPT or other generative AI tools are prohibited for any of the assignments in this course. For the weekly programming assignments, follow the guidelines for pair programming carefully. If a partner does not do their fair share of the work, please let your professor know.

Accessibility and Accommodations: Students who anticipate they may need accommodations in this course because of the impact of a learning, physical, or psychological disability are encouraged to meet with me privately early in the semester to discuss their concerns. In addition, students must contact Student Accessibility and Support Services (SASS) at 740-427-5453 or sass@kenyon.edu, as soon as possible, to verify their eligibility for reasonable academic accommodations. Though I am happy to help you in any way I can, I cannot make any special accommodations without proper authorization from the SASS staff. Except in extraordinary circumstances (and at the very start of the course), accommodations must be certified and discussed with me at least one week before they are to take effect.

Non-Discrimination, Civil Rights and Title IX Compliance

Kenyon College does not discriminate in its educational programs and activities on the basis of race, color, national origin, ancestry, sex, gender, gender identity, gender expression, sexual orientation, disability, age, religion, medical condition, veteran status, marital status, genetic information, or any other characteristic protected by institutional policy or state, local, or federal law. The requirement of non-discrimination in educational programs and activities extends to employment and admission.

As a faculty member, I am deeply invested in the well-being of each student I teach. I am here to assist you with your work in this course. If you come to me with non-course-related concerns, I will do my best to help. However, it is important for you to know that *all faculty, are considered Mandated Reporters* of any incidents of harassment, discrimination, and intimate partner violence and stalking. Meaning, I must report any such discussion to the Civil Rights/Title IX coordinator. I cannot keep information involving sexual harassment, sexual misconduct, interpersonal violence, or any other form of harassment or discrimination based on a protected characteristic, confidential. The Health and Counseling Center, the College chaplains, and the staff at New Directions Domestic Abuse Shelter & Rape Crisis Center are confidential resources. For further information, please refer to the following Kenyon College policies: Discrimination, Sexual Misconduct & Harassment: Title IX, VAWA, Title VII. Civil Rights Policy
ADA & Section 504 Student Grievance Procedures

How to Study for this Class

- Read the textbook before the class (and watch the accompanying video when there is one available). You may not understand everything in the first reading but that’s OK. Do your best. Take notes to ask questions in class.
- Join the class meetings on time and actively participate in class discussions and activities. Do not hesitate to ask and answer questions or contribute to class discussions in other ways. Postings in Moodle forums count as participation.
- Start doing the lab assignments early. You know from your earlier experience that they will often take longer than you think.
- I strongly encourage you to have a partner and follow the guidelines for pair programming for weekly programming assignments.
- If you have an issue about your program that you cannot resolve with your partner, or if you have other questions about the material, join Professor Aydin for a chat during student/office hours or make an appointment.
- You are welcome to chat with Professor Aydin for matters outside the course content as well.