

# Scmp 218: Data Structures and Program Design Fall 2021

## 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>

**Student Hours:** MW: 10:10-12; F: 10:10-11am, and by appointment. In-person or remotely.

See my weekly schedule on course website. Zoom link for remote office hours <https://zoom.us/j/986340648>

**Class Meetings:** T & R: 1:10-2:30 pm . Location: Hayes 311

**Textbook:** Data Structures Using C, by D. S. Malik, Cengage, ISBN 9780324782011

**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. Some 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 19.

**Final Exam-** Final exam will have two components: Test and project. Due, Thu Dec 16, 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 Policy:** *No work will be accepted late, unless permission is granted by the instructor in advance. Do not modify your submitted files after the due date*, until graded.

**Academic Honesty:** The rules set forth in the 2021-2022 Course Catalog apply to all aspects of this course.

<http://www.kenyon.edu/directories/offices-services/registrar/course-catalog-2/administrative-matters/academic-integrity-and-questions-of-plagiarism/>  
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. 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.

**Disabilities:** If you have a disability which requires an accommodations in this class, please feel free to discuss your concern with me, but you should also consult Ms. Erin Salva, the coordinator of student access and support services ([salvae@kenyon.edu](mailto:salvae@kenyon.edu), x5453). It is Ms. Salva who has the authority and expertise to decide on the accommodations that are proper for your disability. Though I am happy to help you in any way I can, I cannot grant any accommodations without a notification from Ms. Salva.

### **Statement on Civil Rights and Title IX**

Kenyon College seeks to provide an environment that is free of bias, discrimination, and harassment. If you have experienced any form of harassment/misconduct/assault, interpersonal violence, or stalking we encourage you to report it. If you report the incident to a faculty member, they must notify Kenyon's Civil Rights & Title IX Coordinator and share the information you provided. More information can be found on the following College web pages:

Sexual Harassment & Misconduct Policy: [www.kenyon.edu/directories/offices-services/title-ix/policy/](http://www.kenyon.edu/directories/offices-services/title-ix/policy/)

Discrimination & Discriminatory Harassment Policy: <https://www.kenyon.edu/directories/offices-services/ocr/discrimination/>

ADA/504 Student Grievance Procedures:

[www.kenyon.edu/directories/offices-services/ocr/discrimination/504-ada-grievance/student-grievance-procedure-resolving-complaints-under-ada-section-504/](http://www.kenyon.edu/directories/offices-services/ocr/discrimination/504-ada-grievance/student-grievance-procedure-resolving-complaints-under-ada-section-504/)

### **COVID-19 Class Policies**

For the health and safety of all on campus, students studying in-person are required to know and follow Kenyon's COVID-19 policies and the local public health rules. Kenyon's policies include daily health screening reports, physical distancing, proper wearing of masks or facial coverings, COVID-19 testing and regular hand washing. If a student fails to follow any of the college COVID-19 policies, they will not be permitted in the classroom nor allowed to engage in class activities, including office hours. Violations will be reported to college officials. If a student is unable to attend class due to illness, quarantine or other circumstances related to COVID-19, please contact the appropriate college officials. They will notify your instructors of any necessary accommodations. It is also very important to let the instructor know of your absence or need for accommodations as soon as possible; however, you are not obligated to disclose any specific health or private information. The faculty instructor will work with the student to implement the accommodations requested by college officials.

**PANDEMIC CONTINGENCY PLAN:** Students who are impacted by the pandemic during the semester should contact the professor as soon as possible. The professor will make appropriate arrangements on a case-by-case basis. If you have to miss classes due to quarantine or isolation, please let me know. I will make materials available to you electronically.

### **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 drop-in hours or make an appointment.
- You are welcome to chat with Professor Aydin for matters outside the course content as well.