

COS 221: DATA STRUCTURES IN C++

Sudarshan S. Chawathe

University of Maine

Fall 2020

DATA STRUCTURES ARE SCHEMES THAT ORGANIZE DATA to permit efficient access in certain modes. The desired modes of access (different kinds of look-ups and modifications), and their relative importance in an application, typically guide the choice of existing data structures and the design of new ones. A judicious choice of data structures often results in very significant improvements in the running time of a program. In order to make such decisions, as well as to design new data structures, we need to understand existing data structures, their access modes, and performance characteristics. In this course, we study data structures from several perspectives, including design, analysis, and application. We use C++ as the primary programming language.

News and Reminders:

- **Please read the class newsgroup for timely announcements.**
- Class newsgroup: Local group `umaine.cos221` on NNTP server `creak.um.maine.edu`. A simple Web interface is at <http://chaw.eip10.org/news/>.
- The most recent version of this document may be found at <http://chaw.eip10.org/cos221/>.
- Some sections below point to material in separate documents that are found on the class Web site, linked from the online version of this document.
- Please use the PDF version of this document for printing and reference: `cos221.pdf`

Goals and Outcomes

Goals

- Understand several interesting data structures and their properties.
- Learn how to use data structures and other tools to solve problems in various application areas.
- Gain experience in reading the relevant research literature and other publications used to disseminate knowledge in the field.
- Practice the appropriate and ethical use of existing material of different kinds, such as source code, services, and documentation.
- Gain experience in contributing to the body of knowledge.
- Learn how to analyze of the running times of programs using simple mathematical methods.
- Gain experience in conducting and documenting experimental studies of programs.
- Improve our programming skills, with attention to software engineering principles.
- Improve our communication skills, with particular emphasis on written communication and, further, well-written programs.

Student Learning Outcomes

Upon successful completion of this course, students are able to

- List commonly used data structures, and the advantages and drawbacks of each.
- Determine suitable data structures for solving a given problem.
- Effectively read suitable publications related to the topic.
- Use resources such as others' code and writing in an ethical and professional manner.
- Contribute to the body of knowledge at an undergraduate level.
- Analyze the running times of programs using simple methods.
- Perform simple experimental studies of programs.
- Program with attention to community standards and good practices.
- Communicate their programming work effectively.
- Meet Quantitative Literacy General Education requirements, such as being able to [following text is from U. Maine Gen. Ed. documents]:
 - Translate problems from everyday spoken and written language to appropriate quantitative questions.
 - Interpret quantitative information from formulas, graphs, tables, schematics, simulations, and visualizations, and draw inferences from that information.
 - Solve problems using arithmetical, algebraic, geometrical, statistical, or computational methods.
 - Analyze answers to quantitative problems in order to determine reasonableness. Suggest alternative approaches if necessary.
 - Represent quantitative information symbolically, visually, and numerically.
 - Present quantitative results in context using everyday spoken and written language as well as using formulas, graphs, tables, schematics, simulations, and visualizations.

Contact Information

Class meetings:

Time: Mondays, Wednesdays, and Fridays, 11:00–11:50 a.m.

Location: Online, using the *Zoom* system. (Details announced on the class newsgroup and the class email list.)

Instructor: Sudarshan S. Chawathe

Office: Boardman Hall, Room 329. (For reference only; physical meetings are extremely unlikely in Fall 2020.)

Office hours: (Please check for changes.)

Mondays, Wednesdays, and Fridays; 11:50 a.m.–12:30 p.m.

Phone: +1-207-581-3930.

Please avoid calling except for truly urgent matters.

Email: sudarshan.chawathe@maine.edu

Use email only for messages unsuitable for the newsgroup. (See below.) Please use only this email address and put the string *COS221* near the beginning of the Subject header of the message.

Responses to all other messages may be very significantly delayed.

Web: <http://chaw.eip10.org/>.

Online Resources

Class Web site:

<http://chaw.eip10.org/cos221/>

We will use the class Web site for posting assignments, readings, notes, and other material. Please monitor it.

Class Newsgroup: We will use the local USENET newsgroup `umaine.cos221` on the NNTP (net news) server `creak.um.maine.edu` for electronic discussions. The Web interface at <http://chaw.eip10.org/news/> provides convenient access. Some further, more general, information on USENET appears at <http://en.wikipedia.org/wiki/Usenet>. The newsgroup is the primary forum for electronic announcements and discussions, so please monitor it regularly, and post messages there as well. Unless there is a reason for not sharing a question or comment, please *use the newsgroup, not email*, for questions and comments related to this course.

Class mailing list: *Please make sure you are on the class mailing list.* The mailing list will use the email address for each student as recorded in the official university records (*Maine Street* system). We will use this mailing list only for urgent messages because all other messages will go on the class newsgroup. I anticipate fewer than a dozen messages on this list over the semester.

Grading Scheme

Likely to change based on class discussions during the first week of classes in Fall 2020.

Grade components: *Students are expected to complete and submit all assigned coursework in good faith; those who fail to do so will earn a **failing grade, regardless of overall numerical score.***

component	% of grade
homeworks	50
final exam	20
newsgroup activities	15
midterm exam	10
class participation	5

Class participation: Students are expected to contribute to learning by asking questions and making relevant comments in class and on the class newsgroup. Quality is more important than quantity. Disruptive activity contributes negatively. See policies below.

Newsgroup activities Discussions on the class newsgroup are an important mode of learning. To encourage everyone to participate actively in these discussions, there is a portion of the grade assigned to how well, and how often, students participate on the newsgroup.

Homeworks: Homeworks include programming and non-programming ones, often mixed. No collaboration is permitted. Everyone is encouraged to discuss the problems and solution strategies *at a high level*, but the final solution and details must be individual work. If the boundary between permissible and non-permissible interactions is unclear, please ask for clarifications.

Exams: *Fall 2020 note:* The exams may be modified to be take-home (or rather, at home, exams) with more flexible times.

All exams are *open book, open notes*. You are free to bring with you any resources that you find useful. However, no communications are permitted other than between students and me. The use of computers during exams is strongly discouraged, but brief use may be permitted provided it does not cause a disturbance, at the discretion of the proctor. You may use the Internet, but only as a library to look up material you may find useful. Ask for clarifications in case of any doubt. The exams are designed to require no equipment other than a pen and paper, along with the textbook and assigned readings.

The midterm exams will be held during a regular class meetings, and will be roughly an hour long. The final exam follows the usual university schedule, and is thus held outside of regular class meeting times, and often at a different location.

Policies

Fall 2020 COVID-19 issues: The university's policy is included by reference: <https://umaine.edu/citl/2020/08/17/suggested-syllabus-language-for-covid19-is-available/>

My plan is to be at *least* as accommodating as the policy specifies and hopefully a lot more. Please do not hesitate to contact me if there is something I can do to assist in this regard.

Due dates: All due dates and times, as announced in class, are strict, to the second. If you believe your work was delayed by truly exceptional circumstances, let me know as soon as those circumstances are known to you and I will try to make a fair allowance. However, *the default is that you get a zero if you don't turn in the work on time*, and fail the class if you don't turn it in at all (cf. Grade Components above).

Attendance: Although I expect students to attend all class meetings, I will not be taking attendance. *If you miss a class meeting, you are responsible for catching up on the lost material, including any important announcements made in class, on your own.* If you have a valid reason for missing a class, let me know early and I will try to help you make up the class. There will be no make-up exams or quizzes. A missed test earns zero credit. If you have a valid reason for missing a test, let me know as early as that reason is known to you and I will make a fair allowance but there will be no make-up tests in any case.

Classroom activities: (This item will be updated for the Fall 2020 Zoom format, but the intent and spirit remains the same.) This course is based on an active learning format, so effective classroom activities are critical to its success. Students are expected to contribute to their own learning and that of their classmates, and to devote 100% of their attention to these activities while in class. On a similar note, all electronic and other distractions (computers, phones, assorted gizmos, etc.) must be completely silenced and put away for the entire duration of the class. (Students who need any such devices for disability accommodations should follow the guidelines outlined below. Others who need any accommodation in this regard due to special circumstances should make advance arrangements with the instructor.) No food or drink is allowed in class, other than water, tea, coffee, and similar, in a spill-proof container. Students who violate these rules or otherwise cause distractions in class will be asked to leave with *no warning*; habitual violators will face disciplinary action.

Office hours: All students are encouraged to make use of office hours to further their learning, obtain assistance on homework assignments, obtain feedback on their class performance, etc. However, office hours are not to be used as a substitute for attending and participating in class meetings (see above). Similarly, assistance with homework assignments will be limited to what is appropriate based on fairness to all; students are expected to demonstrate substantial effort on the assignment before seeking assistance.

Make-up classes: I may have to reschedule a few classes due to my other professional commitments. I will make every attempt to minimize the number of such occurrences and to reschedule for a time that works for most students. Further, I will make sure no student is penalized by such occurrences.

University of Maine administrative policy statements: [Verbatim, standard wording from <https://umaine.edu/citl/teaching-resources-2/required-syllabus-information/>. Please refer to that site for further details.]

Academic Honesty Statement Academic honesty is very important. It is dishonest to cheat on exams, to copy term papers, to submit papers written by another person, to fake experimental results, or to copy or reword parts of books or articles into your own papers without appropriately citing the source. Students committing or aiding in any of these violations may be given failing grades for an assignment or for an entire course, at the discretion of the instructor. In addition to any academic action taken by an instructor, these violations are also subject to action under the University of Maine Student Conduct Code. The maximum possible sanction under the student conduct code is dismissal from the University.

Students Accessibility Services Statement If you have a disability for which you may be requesting an accommodation, please contact Student Accessibility Services, 121 East Annex, 581.2319, as early as possible in the term. Students who have already been approved for accommodations by SAS and have a current accommodation letter should meet with me (the instructor of the course) privately as soon as possible.

Course Schedule Disclaimer (Disruption Clause) In the event of an extended disruption of normal classroom activities, the format for this course may be modified to enable its completion within its programmed time frame. In that event, you will be provided an addendum to the syllabus that will supersede this version.

Observance of Religious Holidays/Events The University of Maine recognizes that when students are observing significant religious holidays, some may be unable to attend classes or labs, study, take tests, or work on other assignments. If they provide adequate notice (at least one week and longer if at all possible), these students are allowed to make up course requirements as long as this effort does not create an unreasonable burden upon the instructor, department or University. At the discretion of the instructor, such coursework could be due before or after the examination or assignment. No adverse or prejudicial effects shall result to a student's grade for the examination, study, or course requirement on the day of religious observance. The student shall not be marked absent from the class due to observing a significant religious holiday. In the case of an internship or clinical, students should refer to the applicable policy in place by the employer or site.

Sexual Violence Policy Sexual Discrimination Reporting

The University of Maine is committed to making campus a safe place for students. Because of this commitment, if you tell a teacher about an experience of sexual assault, sexual harassment, stalking, relationship abuse (dating violence and domestic violence), sexual misconduct or any form of gender discrimination involving members of the campus, your teacher is required to report this information to the campus Office of Sexual Assault & Violence Prevention or the Office of Equal Opportunity.

If you want to talk in confidence to someone about an experience of sexual discrimination, please contact these resources:

For confidential resources on campus: Counseling Center: 207-581-1392 or Cutler Health Center: at 207-581-4000.

For confidential resources off campus: Rape Response Services: 1-800-310-0000 or Partners for Peace: 1-800-863-9909.

Other resources: The resources listed below can offer support but may have to report the incident to others who can help:

For support services on campus: Office of Sexual Assault & Violence Prevention: 207-581-1406, Office of Community Standards: 207-581-1409, University of Maine Police: 207-581-4040 or 911. Or see the OSAVP website for a complete list of services at <http://www.umaine.edu/osavp/>

Programming

The focus of this course is on data structures, algorithms, algorithm analysis, and problem solving techniques in Computer Science, and not on programming, much less programming in a particular language. Programming is, however, a valuable part of the course as it helps us solidify the abstract concepts we study. We will use C++ as the primary programming language. Submissions will be in the form of packaged, well documented, C++ *source* files. Proper documentation and packaging of source code and other material is a crucial component of assigned work and submissions failing in this regard will receive no credit.

Programming Environment and Tools: You are free to choose details such as operating system, development environment, and editor based on your preferences. However, no matter what you use, the submission should be a *source*-code package that works in a standard C++ environment. Further details on the C++ environment, packaging, submission, and testing procedure will be provided in class and on the newsgroup.

Literate Programming: All submitted work must use a *literate programming style*: Your programs must be designed with a human as the intended reader, although they must also compile and run correctly. *Programs that do not meet this requirement are likely to receive a zero score with no further consideration.* Details will be discussed in class. The use of any specific literate-programming or documentation tool is neither necessary nor sufficient for this requirement.

Class Accounts: Although the use of official class accounts, on department computers, is not strictly required, it is a good idea for everyone to have accounts on both our main Unix host (**aturing**) and the cluster of PCs. Among other uses, these accounts will permit testing that code submissions work correctly in a reference environment. Class accounts will be generated based on enrollment information soon after the first class meeting. You should be able to access your **aturing** account from anywhere on the Internet, including the labs in Neville Hall and elsewhere on campus, by using *ssh* to connect to **aturing.umcs.maine.edu**. On most Unix hosts, the command `ssh -l username aturing.umcs.maine.edu` should suffice. For Windows hosts, the freely available *PuTTY* program works well: <http://www.chiark.greenend.org.uk/~sgtatham/putty/>.

Schedule

At the beginning and end of each class, I typically announce the topics and textbook sections covered in that class and those due at the next class. It is important that students read the material *before* the class in which it is discussed and, in general, keep up with readings and studies.

Textbook and Readings

Textbook: Mark Allen Weiss. *Data Structures and Problem Solving in C++*. Pearson, 2nd edition, 1999. This book is a *required textbook* for this course. Detailed coverage information will be announced as we progress in the semester.

Assigned Readings: Material will be added here as we progress through the semester.

Other Readings: All the following are recommended, but not all are required. Further details and additional readings will be announced in class and may appear here as well. Some may move to the *assigned* category as we progress.

1. Arne Andersson. Balanced search trees made simple. In *Proceedings of the Workshop on Algorithms and Data Structures*, pages 60–71, Montreal, Canada, August 1993.
This paper introduces AA-trees and includes very nice examples and figures.
2. Samuel W. Reynolds. A generalized polyphase merge algorithm. *Communications of the ACM*, 4(8):347–349, 1961.
This paper provides a succinct and readable description of polyphase merging. It is a very useful supplement to the description in the textbook, which is missing many important details.
3. Sanjeev Saxena. Dominance made simple. *Information Processing Letters*, 109(9):419–421, April 2009.
This short paper is a good example of how some of the basic concepts studied in this course may be used as building blocks to solve more complex problems.
4. Derrick Coetsee. An efficient implementation of Blum, Floyd, Pratt, Rivest, and Tarjan’s worst-case linear selection algorithm. <http://moonflare.com/>, January 2004.
5. Jon Bentley and Don Knuth. Programming pearls: Literate programming. *Communications of the ACM*, 29(5):364–369, May 1986.
6. Paul E. Black. Dictionary of algorithms and data structures. <http://www.nist.gov/dads/>, September 1998.
7. Lloyd Allison. Suffix trees. <http://www.allisons.org/ll/AlgDS/Tree/Suffix/>, 2008.

Exercises, Homeworks, Tests, and Notes

Material will appear here as we move along the semester.

- Homework assignments:
 - Homework 1: [hwq/hw01.pdf](#).
 - Homework 2: [hwq/hw02.pdf](#).
 - Homework 3: [hwq/hw03.pdf](#).

- Exams:
 - Midterm Exam 1: `hwq/mt01.pdf`.
 - Midterm Exam 2: `hwq/mt02.pdf`.
 - Final Exam: `hwq/fin.pdf`. (Available after exam start time.)

Homework and Project Submissions

Answers to programming problems should be submitted electronically, using the packaging and submission procedure that will be described in class and on the class newsgroup.

*All electronic submissions must be made using the upload interface at <http://chaw.eip10.org/u/>. Electronic submissions in **all other forms**, such as email or physical media, will be **discarded and receive no credit**.*

If your upload is successful, you will be presented with a confirmation Web page similar to the following sample. You should record the reported MD5 checksum and timestamp.

```
SUCCESS: Please note the following for your records.
```

```
Successfully saved cos221-hw01-aardvark-alice-4233.tgz.  
MD5 checksum: 09ee098b83d94c7c046d6b55ebe84ae1  
Timestamp: 2020-09-10 03:31:24
```

If you do not see something very similar then your submission is unsuccessful.

Answers to non-programming problems should also be submitted electronically as above in PDF format. The PDF file may be either a scanned copy of handwritten work or generated directly from a typed electronic version (your choice). However, it is critically important that your submission have good legibility regardless of how it is produced. Please be very mindful of file sizes especially if you use scanned PDFs. Unless you are adept at a program that typesets math /well/, well written and well scanned hand-written versions are likely to be better than clumsily typed versions. *Illegible, hard to read, or otherwise messy submissions, whether handwritten on typed, are likely to be returned without grading, for zero credit.*

Fallback procedure If (and only if) there are unexpected problems and you are unable to submit your work as above, then you should save your file on your own computer (with some backups), compute its MD5 checksum using the `md5sum` utility on Unix-like systems (or other similar tools), and submit the file name, time stamp, and MD5 checksum (only, not the file itself) by email with a suitable Subject header.