Data structures with C++

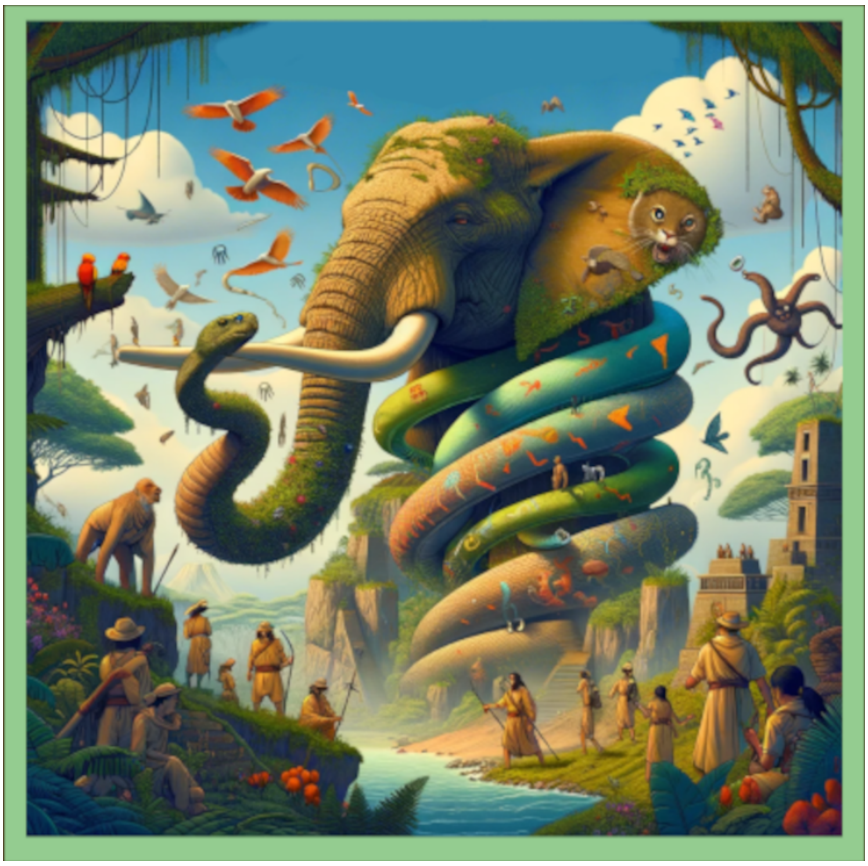
CSC 240 - Data structures with C++ - Syllabus - Fall 2024

Marcus Birkenkrahe

June 29, 2024

Contents

1 General Course Information



- Course title: Data structures with C++
- Course number and section: CSC 240.01
- Meeting Times: Tue-Thu 9:30-10.45 AM
- Meeting place: Derby Science Building computer lab room 239
- Professor: Marcus Birkenkrahe
- Professor's Office: Derby Science Building 210
- Phone: (870) 307-7254 (office) / (501) 422-4725 (private)
- Office hours: by appointment MWF 4pm, Tue 3pm, Thu 11 am & 3 pm

2 Materials and multimedia

- Textbook (optional): Helfrich, C++ Data Structurnes (KendallHunt, 2020). Recommended: Kanetkar Y, Data Structures Through C++ 5th ed (bpb,2024). Malik, C++ Programming - Program Design Including Data Structures (Cengage 2015); La Rocca, Grokking Data Structures (Manning 2023) - with Python; Morin, Open Data Structures (in C++) (OpenText, 2013).
- Videos: The Udemy course "C++ Data Structures & Algorithms + LEETCODE Exercises" is very good (nice animations). I've used this as well as Helfrich's YouTube videos (for his book). Animations can be helpful for this topic to visualize multi-dimensional, dynamic processes. The freeCodeCamp course is more teacher-centric but OK.
- There is a gazillion sources available online and offline on the topics of data structures and algorithms. Many of the books are great for the first 10, good for the next 30, and disappointing for the lats 300 pages. Many of the videos are great for the first 5 and lousy for the last 15-30 minutes.
- I have looked at dozens of books, videos, courses, and tutorials. Ask me if you found a specific source - I may know it, or I may want to learn more about it!

3 Objectives

This course on "Data Structures with C++" offers an in-depth exploration into the theory and application of data structures using the C++ programming language. It aims to equip students with the knowledge to efficiently store, process, and retrieve data using various data structures such as arrays, linked lists, stacks, queues, trees, and graphs. The curriculum emphasizes the importance of algorithmic thinking and optimization techniques, providing a solid foundation for understanding the complexities and capabilities of modern computing systems.

4 Target audience

Ideal participants are those who have a basic understanding of programming concepts and are interested in advancing their knowledge in data structuring to improve software efficiency and performance.

5 Student Learning Outcomes

Students who complete this course will be able to:

- Demonstrate a thorough understanding of major data structures and their applications.
- Implement various data structures in C++, including arrays, linked lists, stacks, queues, trees, and graphs.
- Analyze the efficiency of data structures and algorithms in terms of time and space complexity.
- Apply algorithmic thinking to solve complex problems using appropriate data structures.
- Design and develop efficient software solutions for real-world applications.
- Critically evaluate the choice of data structures in software development projects.
- Employ advanced C++ features and object-oriented programming techniques in the context of data structures.
- Enhance problem-solving skills through the design and analysis of algorithms for data manipulation.
- Prepare to contribute effectively to technology-driven environments with a strong foundation in data structuring.

Students, who complete CSC 240 will have fulfilled the prerequisites for CSC 265 Algorithms.

6 Course requirements

Formal prerequisites: Introduction to Programming (either CSC100 or CSC115 or CSC109, and MTH101 (College Algebra).

Course requirements include a foundational knowledge of programming principles and familiarity with any programming language. Students are expected to have completed introductory courses in computer science or possess equivalent practical experience. A willingness to engage in complex problem-solving and the ability to think critically about algorithm design and data manipulation are essential for success in this course.

7 Grading

WHEN	DESCRIPTION	IMPACT
Weekly	Programming assignments	25%
Monthly	Sprint reviews	25%
Weekly	Tests	25%
TBD	Final exam (optional)	25%

- Sprint reviews are monthly project progress reports
- Tests are open-book multiple choice exams for home
- The final exam is optional if you want to improve your grade

8 Rubric

Component	Weight	Excellent	Good	Satisfactory	Needs Improvement	Unsatisfactory
Participation and Attendance	0%	Consistently attends and actively participates in all classes.	Attends most classes and participates in discussions.	Attends classes but participation is minimal.	Frequently absent and rarely participates.	Rarely attends classes and does not participate.
Programming Assignments	25%	Completes all assignments on time with high accuracy (90-100%).	Completes most assignments on time with good accuracy (80-89%).	Completes assignments but with some inaccuracies or delays (70-79%).	Frequently late or incomplete assignments with several inaccuracies (60-69%).	Rarely completes assignments and shows minimal understanding (0-59%).
Project Sprint Reviews	25%	Consistently demonstrates significant progress, excellent teamwork, and high-quality work (90-100%).	Shows good progress, effective teamwork, and good-quality work (80-89%).	Adequate progress, teamwork, and satisfactory work quality (70-79%).	Minimal progress, poor teamwork, and below-average work quality (60-69%).	Little to no progress, ineffective teamwork, and poor-quality work (0-59%).
Tests	25%	Demonstrates thorough understanding and application of concepts (90-100%).	Shows good understanding with minor errors (80-89%).	Displays basic understanding with some errors (70-79%).	Limited understanding with several errors (60-69%).	Minimal understanding and many errors (0-59%).
Final Exam (Optional)	25%	Demonstrates comprehensive understanding and application of course concepts (90-100%).	Shows strong understanding with minor errors (80-89%).	Displays adequate understanding with some errors (70-79%).	Limited understanding with several errors (60-69%).	Minimal understanding and many errors (0-59%).

9 Learning management system

- We use Lyon's Canvas installation for this course.
- The home page contains: assignments, grades, pages, people, syllabus, quizzes, Google Drive, Course evaluation and Zoom.
- The Zoom page includes cloud recordings of all past sessions.

- Recorded sessions will be deleted after the last class.

10 GitHub

All course materials are available in a public GitHub repository (github.com/birkenkrahe/csc240). Registration for students includes a free subscription to GitHub codespaces with the AI coding assistant Copilot. GitHub is the worldwide largest online platform for software development.

11 Lyon College Standard Policies

Online: `https://tinyurl.com/LyonPolicyOnline`, see also Class Attendance

12 Dates and class schedule

"Data structures" and "algorithms" are a little hard to separate:

- "Data structures" is concerned with storing and organizing data.
- "Algorithms" is concerned with using data to solve problems.

Both topics can be taught language-agnostic (without referencing a particular language) but your understanding will benefit greatly from examples and programming assignments.

Week	Topic	Pg	Project
1	Boolean	3	
2	Integer	20	
3	Real Number	51	
4	Linked List	81	1st sprint review
5	Binary Tree	109	
6	Binary Search Tree	138	
7	Graph	179	
8	Vector	212	2nd sprint review
9	List	245	
10	Deque	275	
11	Stack	320	
12	Queue	353	3rd sprint review
13	Priority Queue	382	
14	Hash	412	
15	Set	442	
16	Map	478	4th sprint review

Pg refers to pages in Helfrich, C++ Data Structures (2e), Kendall Hunt Publishing 2020. The book also contains 10 appendices with an overview of specific topics, like UML modeling, pseudocode keywords, operator overloading, recursion, etc., and a glossary.