CSC 212: Data Structures and Abstractions
Introduction

Marco Alvarez

Department of Computer Science and Statistics
University of Rhode Island

Fall 2020

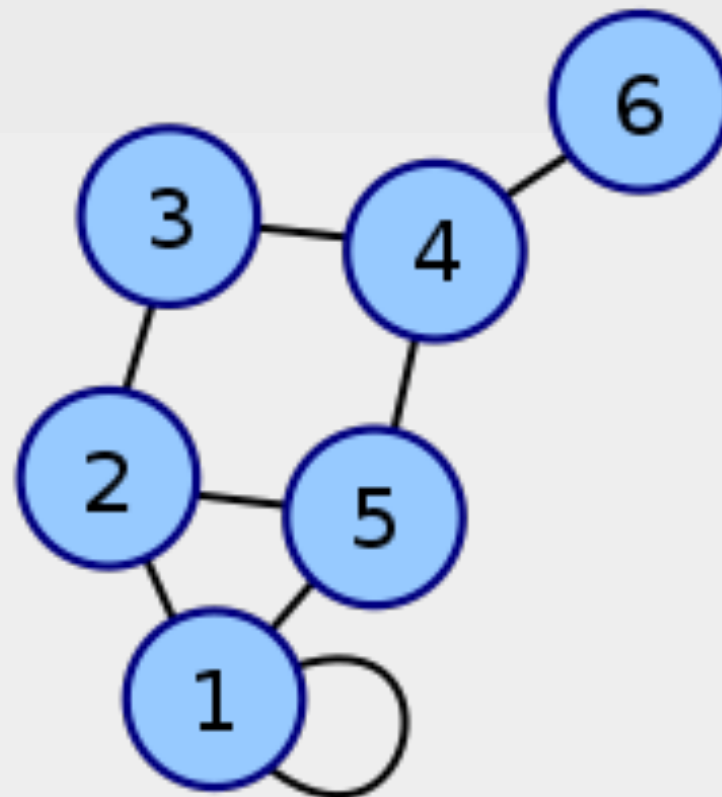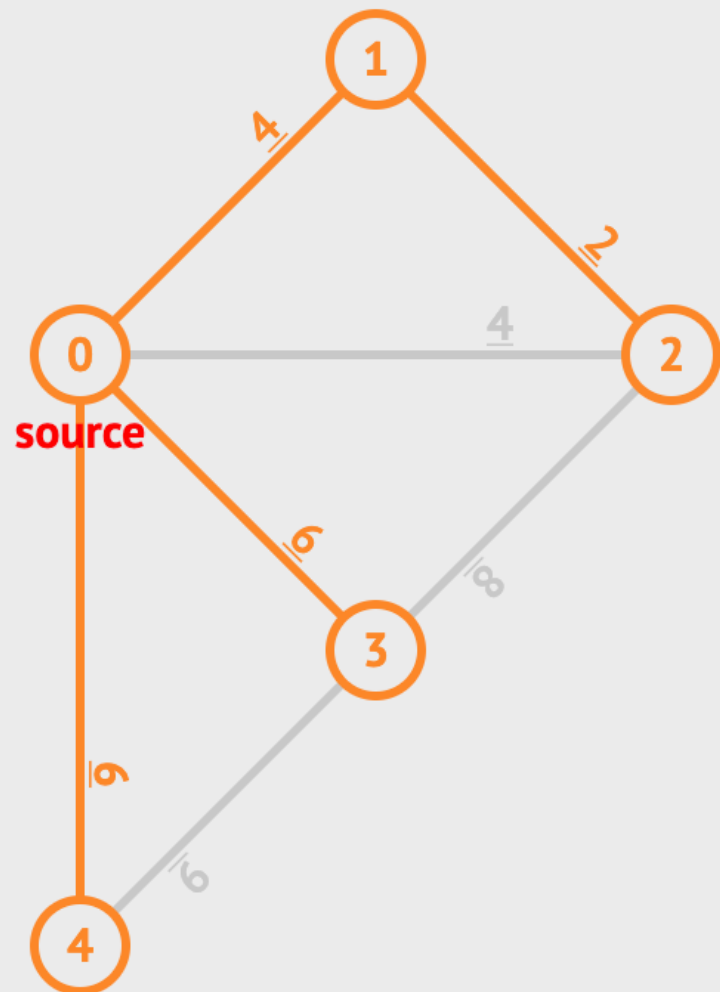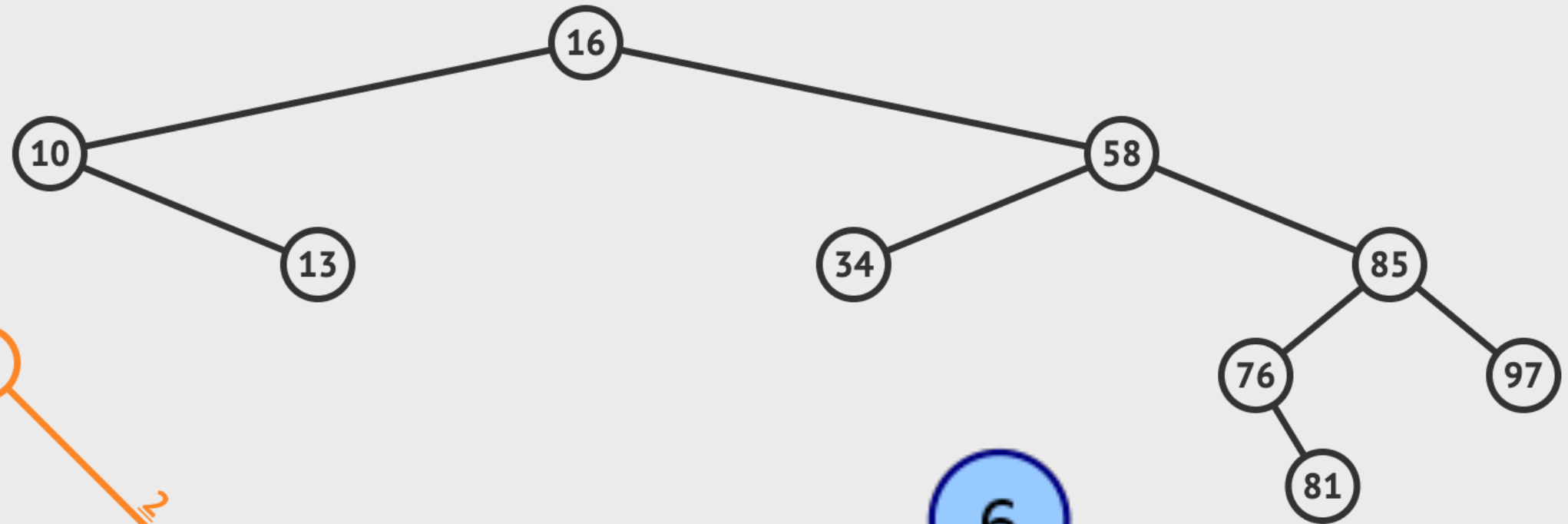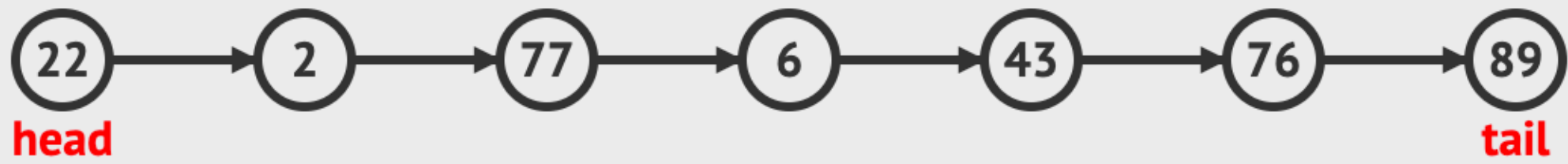THINK BIG WE DO℠

CSC 212?

‣ Review of basic principles of analysis of algorithms

‣ Introduction to fundamental data structures and their algorithms

  ✓ arrays, lists, stacks, queues, trees, hash tables, graphs

‣ Survey of classic algorithms for sorting and searching

Prerequisites: CSC 211 (at least C-) and MTH 180

1	3	5	3	2	5	6	7				

	2			3		
		3				
	1			2		
		4		5		
					6	
					20	
	21					

4

CSC 418 — Information and Network Security

CSC 492 — Special Topics in Computer Science

CSC 499 — Project in Computer Science

CSC 491 — Directed Study in Computer Science

CSC 415 — Introduction to Parallel Computing

CSC 406 — Computer Graphics

CSC 450 — Scientific Computing

CSC 412 — Operating Systems and Networks

CSC 402 — Programming Language Implementation

CSC 481 — Artificial Intelligence

MTH 243

MTH 362

CSC 411 — Computer Organization

CSC 436 — Database Management Systems

CSC 445 — Models of Computation

CSC 461 — Machine Learning

MTH 215

CSC 301 — Fundamentals of Programming Languages

CSC 305 — Software Engineering

CSC 440 — Design and Analysis of Algorithms

CSC 392 — Intermediate Topics in Computing

CSC 372 — Dynamic Web Design and Programming

CSC 320 — Social Issues in Computing

CSC 417 — Computer Communications

CSC 310 — Programming for Data Science

CSC 212 — Data Structures and Abstractions

CSC 340 — Applied Combinatorics

CSC 447 — Discrete Mathematical Structures

MTH 451

CSC 271 — Web Design and Programming

CSC 201 — Introduction to Computer Programming

CSC 211 — Object Oriented Programming

MTH 180

CSC 200 — Computer Problem Solving for S&E

CSC 292 — Topics in Computing

CSC 110 — Survey of Computer Science

MTH 141

MTH 131

CSC 106 — The Joy of Programming

CSC 104 — Puzzles and Games

CSC 101 — Computing Concepts

CSC 192 — Introductory Topics in Computing

- - - taken concurrently

"data structures" for technical interviews   ✕ 🎤 🔍

🔍 All   ▶ Videos   📰 News   🖼 Images   🏷 Shopping   ⋮ More   Settings   Tools

About 21,100,000 results (0.57 seconds)

Commonly used Data Structures

- Arrays.
- Stacks.
- Queues.
- Linked List.
- Trees.
- Graphs.
- Tries (They are effectively trees but it's still good to call them out separately).
- Hash Tables.

Jul 12, 2018

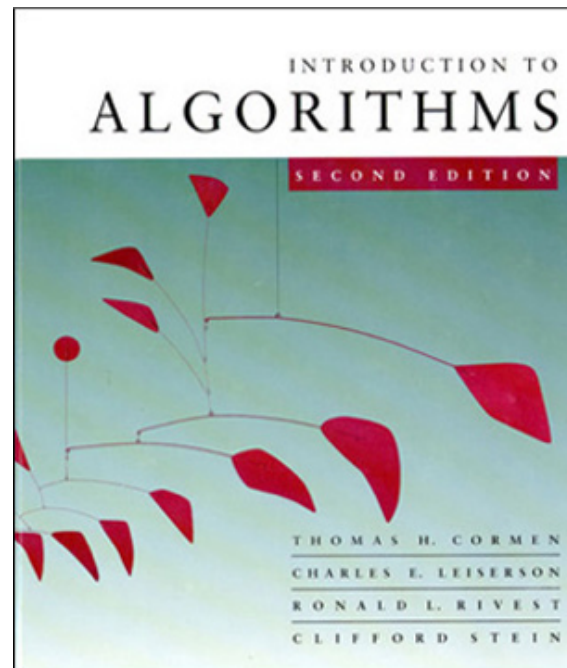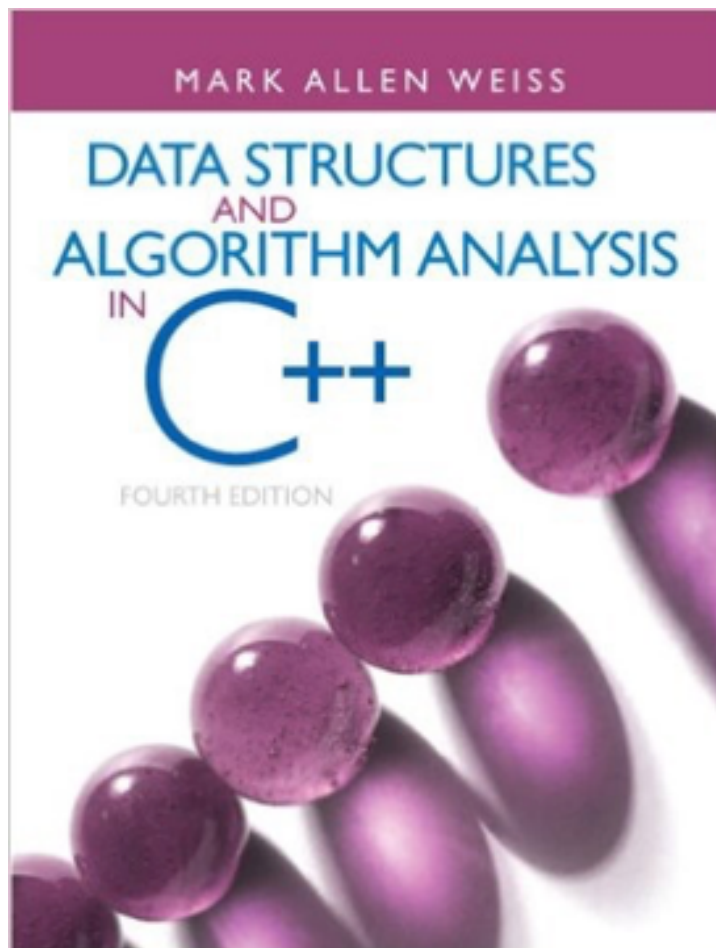dev.to › fahimulhaq › top-8-data-structures-for-coding-in...

Top 8 Data Structures for Coding Interviews and practice ...

❓ About Featured Snippets   🏴 Feedback

6

Recommended Textbooks

Need a refresher on C++ programming?

‣ Read a book

‣ Enroll in a MOOC (massive open online course)

‣ Solve Challenges

Kattis

CS50 IDE

https://ide.cs50.io

Grading (subject to change)

‣ Lab attendance

  ✓ synchronous labs (5%)

‣ Assignments

  ✓ 5 assignments (3 prog. 2 PS) (30%)

  ✓ 1 final project (35%)

‣ Exams

  ✓ 1 midterm exam (15%)

  ✓ 1 final exam (15%)

gradescope

All exams are based on lecture materials and assignments

Homework Assignments

‣ Discussions and collaboration are allowed, however you **must** write your own code and solutions

‣ All assignments are to be turned in on **Gradescope** by the due date

   ✓ late submissions are **NOT** accepted 🚫

Plagiarism?

   ✓ just **don't do it**

   ✓ if you get caught (chances are very high), your name(s) will be immediately reported for further sanctions

What is expected from you?

‣ **Attend** **synchronous** lectures/labs

  ✓ I do not spend time taking attendance … but … students skipping lectures will (**very**) likely **fail** this class

‣ **Organize** your time

  ✓ lectures, labs, homework assignments, project, exams

‣ **Participate** and think critically

  ✓ ask questions (lectures, labs, office hours, Piazza, …)

‣ **Start** working on assignments **early**

  ✓ avoid copying/pasting or google'ing answers

Need help?

‣ Post questions on **EdStem**

  ✓ answer questions, share information

‣ Contact your TAs

‣ Come to **Office Hours**

Seriously, seek help!

‣ Last Semester's Stats

  ✓ 66/105 passed (64%)

  ✓ 39/105 failed (36%)

Programming Assignment 1

Warming up

‣ Adjacent elements sum

  ˅ find the **maximum sum** of any pair of adjacent elements in an array of integers

1	3	5	3	2	5	6	7	9	2	13	1

• • •