Welcome

CS 2233-01

Data Structures and Algorithms

Instructor: Dr. Qingguo Wang

College of Computing & Technology

January 8, 2018

Let's know each other a bit

◦ Your name
◦ Interest
◦ Major
◦ Expectations
◦ Programming skills, languages, etc.

A brief about myself

Ph.D. in Computer Science, University of Missouri, ~2011

Postdoctoral Fellow, Biomedical Informatics, Vanderbilt University, ~2014

Computational Engineer III, Memorial Sloan Kettering Cancer Center, ~2016

Associate Professor of Data Science, Lipscomb University, 2016~

My contact

Email: qingguo.wang@lipscomb.edu / qwang@lipscomb.edu

Office: Swang 120

Phone: (615) 966-5814
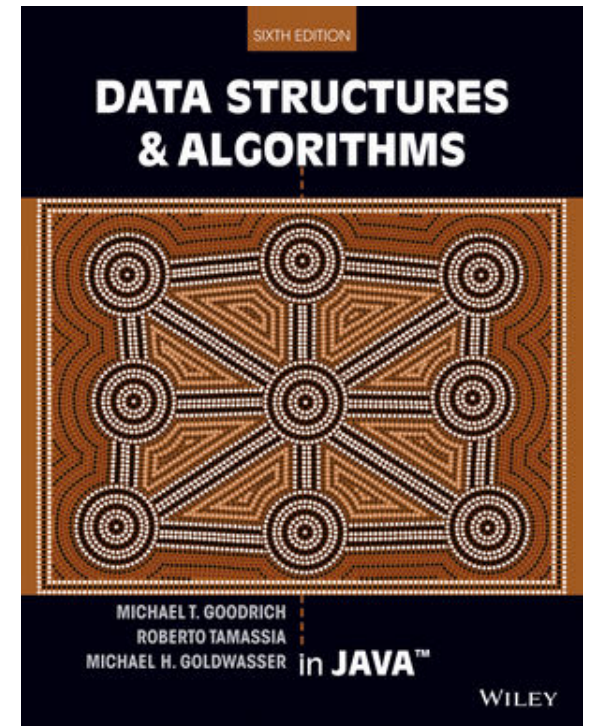
Office Hours: M-F: 9-11am

Prerequisites

Two courses with grades of 'C' or higher (see Syllabus in Blackboard)

- Computer Science 1233 - Object-Oriented System Design and Programming

- Math 2103 - Discrete Mathematics

Reading

Data Structures and Algorithms in Java, 6th edition, Wiley, 2014. By M. T. Goodrich, R. Tamassia, and M. H. Goldwasser, (http://coltech.vnu.edu.vn/~sonpb/DSA/Data%20Structures%20and%20Algorithms%20in%20Java,%206th%20Edition,%202014.pdf).

About textbook

Skip the following two chapters

◦ Chapter 1 Java Primer

◦ Chapter 2 Object-Oriented Design

What are data structures?

In computer science, a **data structure** is a particular way of organizing data in a computer so that it can be used efficiently.

Examples:
- simple variables — primitive types
- objects — collection of data items of various types
- arrays — collection of data items of the same type, stored contiguously
- linked lists — sequence of data items, each one points to the next one

Data structures in programming languages

Many data structures and algorithms have been implemented in modern programming languages

Take Java **Strings** as example
- ➤ How is the text in a String object stored?
- ➤ Implement length() and other methods
- ➤ How does it concatenate strings?

What is an algorithm?

An *algorithm* is a sequence of unambiguous instructions for manipulating certain data structures in order to solve a problem.

Why study data structures?

Algorithms + Data Structures = Programs

Niklaus Emil Wirth, PhD.
Professor of Informatics at ETH Züric
Turing Award recipient (1984)

Example applications of data structures

Stacks
- Undo sequence in a text editor
- Page-visited history in a Web browser
- Evaluating Arithmetic Expressions

Queues
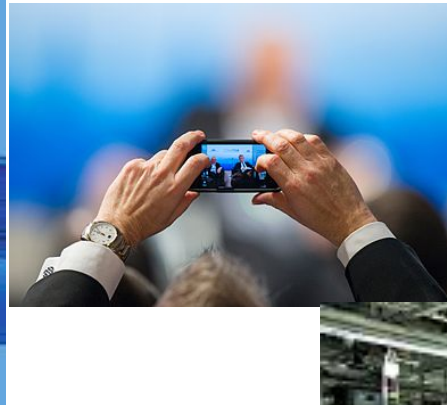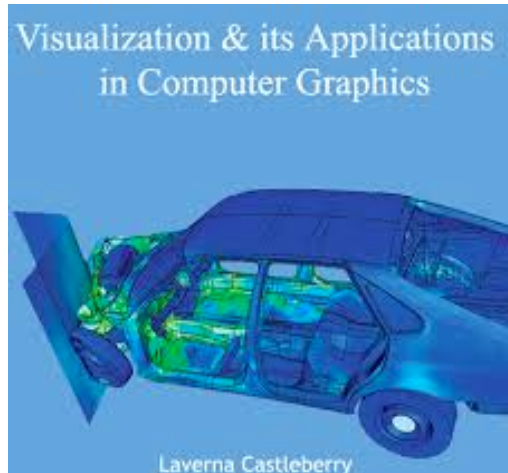- Waiting lists
- Printer

Hashing
- Database indexing
- Google, yahoo search services

Impact of data structures and algorithms

Impact of improved data structures and algorithms
- On me
- On organization
- On society

Why study data structures?

Example Google Interview Questions on Data Structure and Algorithm

Find out the <u>fastest</u> way to locate the largest element in a circular sorted array ?

You have a binary search tree and integer n, find out the most <u>efficient</u> way to locate two nodes of the three whose summation is equaled to "n" ?

How do you convert a max heap to min heap ?

You have given two lists L1 and L2 write an algorithm to see if both the lists are equivalent or not?

How do you figure out time and space complexity of recursive function ?

how will you implement three stacks with one array?

Topics

Data structures: arrays, stacks, linked lists, queues, trees, sets, and graphs

Programming techniques for processing data structures: sorting, searching, hashing, storage management

Design and analysis of algorithms

Course Goals

By the end of this semester, you will have

- ➤ Knowledge of common data structures and algorithms
- ➤ Skills to design useful and efficient data structures
- ➤ Ability to analyze data structures or algorithms
- ➤ Capability of designing efficient algorithms

Grading (See Syllabus in Blackboard)

◦ Submit assignment to the blackboard (in pdf, word, etc.) that I can read directly in Blackboard. Don't submit ZIP file

◦ Homework   40 points
◦ Midterm Exam 1 15 points
◦ Midterm Exam 2 15 points
◦ Final Exam   20 points
◦ Quizzes   10 points

◦ **A** = 90+ points; **B** = 80 ~ 89; **C** = 70 ~ 79; **D** = 60 ~ 69; **F** = below 60 points

Policies

Late homework policy:
- assignment submitted after the due date without prior authorization will receive deduction of 10% of the potential points for that assignment for each day (or portion thereof) that the assignment is past due.
- Assignment turned in more than 7 days from the due date will not be accepted (you will receive a zero on the assignment).
- **No assignment is accepted during and after the week of final exam, i.e. after the study day on April 26.**

Exam policy:
- Make-up exams will be considered for only two reasons, sickness and unforeseen tragedy.

No makeup quizzes will be given.

Academic Integrity

◦ Do your own work on all tests and assignments unless I indicate that collaboration is allowed on a specific assignment

◦ Do not use external entities, e.g. friends, relatives, to obtain solutions to the homework and/or programming projects

◦ A student who cheats receive a grade lower than a student who did not

Other Issues

Student needing accommodations

Snow schedule

Read Syllabus (in Blackboard)

Class Expectations

Attend class, ask questions

Start homework early, submit to Blackboard timely

Write well and clearly

Get help when you need it

Other Expectations

Student Expectations

CCT's expectations for student learning align with its mission statement. CCT is here to assist its students to reach their academic, professional, and personal goals and to instill hearts of service and compassion in them. As the student, you will be expected to:

- Refer to your syllabus throughout the semester;

- Learn and think critically and take time to ask questions when you do not understand something;

- Build problem-solving skills by attempting to solve the problem(s) for yourself first;

- Maintain relationships with your professors (and advisors) and report to them any degree of difficulty that you may experience with your courses;

- Come to every advising meeting prepared with questions, materials and / or topics to be discussed;

- Accept full responsibility for your academic success and acknowledge that academic advisors are only one of the many resources for achieving success;

- Be a self-advocate and learn how to locate and utilize available campus resources and student support services;

- Embrace service activities to prepare for your future career opportunities;

- Make connections between classroom computing and the real-world and get involved beyond the classroom;

- Demonstrate respectful, courteous, and professional behavior, all-around, and when communicating with faculty in person or through e-mail; and

- Display an attitude toward work that includes integrity, responsibility and a desire for excellence

lipscomb.edu/technology

Homework

Read Chapter 3 Fundamental Data Structures: Array and Linked lists