CSCI 232: Data Structures and Algorithms

Course information
Fall 2018
Meeting time/location:

Lecture: M/W 1:00-1:50PM Social Sciences 356
Labs:

W	3:00-4:50	Liberal Arts 103B
Th	11:00-12:50	Social Sciences 344
F	1:00-2:50	Social Sciences 344

Course material/submissions/grades are in Moodle (http://umonline.umt.edu)

Instructor information
Instructor: Travis Wheeler
Office: Social Science 420
E-mail: travis.wheeler@umontana.edu
Phone: 406-243-6219
Office Hours:
Mon 2:00-4:00
Wed 10:30-12:00
Or by appointment (http://wheelerlab.org/schedule)

Teaching Assistants:

David Rich
Office: Social Science 421
E-mail: david.rich@umconnect.umt.edu
Office Hours: Tues 2:00-3:30 - meet in SS 402

Jesse Davidson
Office: Social Science 422
E-mail: jesse.davidson@umconnect.umt.edu
Office Hours: Wed 2:00-4:30 - meet in SS 402

Course Objectives
The purpose of this course is to introduce you to essential data structures and algorithms that will serve as valuable building blocks for the remainder of your career as a computer scientist. In this class, we emphasize fundamental understanding and implementation. You will:

- Become familiar with fundamental data structures like stacks, queues, priority queues, associative arrays / hashes, and graphs (e.g. search trees and perhaps tries)
- Become familiar with fundamental algorithms based on these data structures, including sorting, clustering, graph search, and string search

- Improve your facility with software development, by implementing these data structures and algorithms in Java
- Become familiar with the basic notion of run time and space analysis, as applied to algorithm development

Other courses required

Prerequisite: CSCI 136

Corequisite: M 225 or M 307 (in prep for Analysis of Algorithms, CSCI 332)

Required textbook

Algorithms Fourth Edition
By Robert Sedgewick and Kevin Wayne
Booksite: http://algs4.cs.princeton.edu

Schedule

Below is an ordered set of topics I expect to cover. It is subject to change. Please consult moodle for up-to-date schedule and reading assignments. Lectures will cover the reading material as comprehensively as possible. Students are expected to supplement lectures with a careful study of the relevant sections of the textbook.

- Fundamentals (Objects, data types, APIs, Analysis, Stacks and Queues)
- Sorting (Elementary, Mergesort, Quicksort, Priority Queues)
- Searching (Symbol Tables, Search trees, Hash tables)
- Graphs (Directed and undirected, Spanning trees, Shortest paths)
- Strings (Tries, Substring search)

Assignments will include both problem sets (e.g. questions from the text) and programming assignments.

Grading

Programming:	35%
Homework:	25%
Exams:	30%
Quizzes:	10%

The "curve": you may have heard that the grades assigned on my exams and homework are often quite low. I account for this, and set grade cutoffs accordingly. Cutoffs are usually lower than the typical 90/80/70 splits. I will provide an update with approximate cutoffs as the semester progresses.

Late policy

Submissions for programming and homework assignments are due at the beginning of class. Late submissions will not be accepted. Every student will get one free extension on an assignment (programming or homework) for up to a week. You do not have to ask for this – just write that you are using your free extension when you turn it in. Don't waste this extension or feel obligated to use it; another extension will be given only in exceptional circumstances.

Flipped classroom

In a flipped classroom, you watch video lectures online before class and the class time is spent on short presentations, discussions, individual and group problem solving. The purpose of this approach is to turn the "talking-head" part of instruction (i.e. the lectures) into something you can do at home, and make our time in class into a more engaged experience.

This approach places a clear burden on you:

- Before class, watch the recorded lectures and skim the assigned pages in the book. I will assume that you have done so. It will be apparent if you arrive in class without having watched the assigned lectures. Why? Because class time will be spent with activities such as:
    - Talking with classmates (and instructors) about the day's material
    - Working on problems in small groups
    - Presenting the day's material to others in the class
    - Answering questions from, and building on concepts presented in, the lectures
- After class
    - Review any of the lecture videos that remain confusing
    - Read the text in detail

Sedgewick lectures, part 1: http://bit.ly/2xJSbVm (youtube)
Sedgewick lectures, part 2: http://bit.ly/2vZFkMy (youtube)

Cheating

Academic dishonesty (including plagiarism and cheating) will not be tolerated. (Un-caught) cheating hurts all involved:

- It devalues the grades earned by others in the class, and the degree from our program
- It leaves you without the skills you've asked (and paid) me to help you gain
- (Actually, it doesn't hurt me)

Consult the university's student conduct code for more details. I will follow the guidelines given there. I will seek out the maximum allowable penalty for any academic dishonesty that occurs in this course. If you have questions about what constitutes acceptable use of resources, please feel free to reach out to me – I'll respect your attempts to understand the ethics of using others work.

Specifically, do not search for answers online. I'm not naïve enough to think these don't exist, and I have caught >20 people plagiarizing from the web in the past three years. Most have received a failing grade.

This is not an idle threat: a graduate student working with me has developed software

for identifying online plagiarism, and we will apply that software to your submissions. I retain the right to question you about the material turned in. If it is evident that you don't understand what you turned in, I may view your submission as an instance of cheating.

Working in groups (homework)
You are encouraged to work together in small groups - the best way to understand the subtleties of the homework problems is to argue about the answers. Each of you should look at (and work on) all the problems independently, and not just sub-divide the questions among group member. After discussing problems and coming up with solutions, you will each write up a separate submission. Do not write your solutions up then share them with someone else. Though the ideas behind your solutions may be similar, the text should be your own – demonstrate your command of the problem with a personalized solution.

(Don't be a leech and let your partner do all the work. Unless you learn how to solve problems, you will get burned on the exams and thus for your final grade. You also won't learn the material, which will harm you in future classes and employment)

Working in groups (programming assignments)
I encourage discussion with others regarding programming assignments, as well. As with homework, these should be high-level discussions. Code should be written independently. If I suspect copying or plagiarism, I will ask you to explain each piece of the code to me, possibly resulting in a reduced grade or removal from class.

Attendance
Attendance is required. You are responsible for all material presented in class; some of that material is not covered in the textbook.

Computers
You may develop your programs on any machine that you like: we encourage you to use your own equipment. In the first lab, we will provide instructions for setting up a Java programming environment under Windows, Mac OS X, and Linux. Laptops will be available during lab sections.

Disabilities
Students with disabilities are encouraged to meet with me to discuss *any* accommodations they require.

Electronic devices
Turn off your cellphone, or set it to vibrate during class. Take calls outside the classroom. Students texting during class will be asked to leave.

Personal contact

I hope to establish as much personal contact with each of you as is possible in a class this size. Don't be afraid to visit my office hours, or stop by my office to ask questions or say hello. In fact: I require that you do so at least once. Details will be given in class.