CSCI 232: Data Structures and Algorithms

Course information
Fall 2017
Meeting time/location:

Lecture: M/W 10:00-10:50AM Health Sciences 207
Labs = one of:
W 1:00-2:50 Fine Arts 210
F 1:00-2:50 Fine Arts 210
F 3:00-4:50 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:
Tue 1:00 – 2:30
Th 10:00 – 11:15
Or by appointment (see schedule at http://wheelerlab.org/calendar)

Teaching Assistant: James Stauder
Office: Social Science 423
E-mail: james.stauder@umconnect.umt.edu
Office Hours: Mon 11:00-12:30, Th 12:30-2:00

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. tries and search trees)
- Become familiar with fundamental algorithms based on these data structures, including sorting, clustering, 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 analysis, as applied to algorithm development

High-level student learning goals

As part of our accreditation process, we actively evaluate the ways in which our courses achieve student learning outcomes. At a high level, this course is designed to impart:

a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
c. An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
d. An ability to use current techniques, skills, and tools necessary for computing practice.

During the course of the semester, I will use homework assignments and exam questions to assess the extent to which these goals are being met.

Other courses required

Prerequisite: CSCI 136
Corequisite: M 225 or M 307

Required textbook

Algorithms Fourth Edition
By Robert Sedgewick and Kevin Wayne

Booksite: http://algs4.cs.princeton.edu

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
  - Presenting today'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)

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.

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, Regular Expressions)

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

Grading

Programming:	35%
Homework:	15%
Exams:	30%
Quizzes:	10%
In class:	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 a good deal 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.

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 all the problems independently, and not just divide the list in two parts each time. 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 not be identical – 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.

Cheating
Academic dishonesty (including plagiarism and cheating) will not be tolerated. 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 which behaviors are acceptable, please ask me.

Specifically, do not search for answers online. I'm not naïve enough to think these don't exist, and I have caught nearly a dozen people plagiarizing from the web in the past two 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.

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. To facilitate interaction, every few weeks I plan to have a 'Pizza with the Prof' (pizza lunch, free for you). Await further details.