CSCI 432: Advanced Algorithm Topics

MWF, 15:10-16:00, Roberts Hall 218

Fall 2015

1 Course Instructors and Assistants

Instructor: Brittany Terese Fasy
Office Hours: M 17:00 - 18:15; H 8:45 - 10:00
Office: EPS 363
Email: brittany@cs.montana.edu
Phone: 406.994.4804

GTA: TBA
Office Hours:
Office:
Email:
Phone:

UTA: James Soddy
Office Hours: TBA
Office:
Email:
Phone:

2 Course Description

A rigorous examination of advanced algorithms and data structures. Topics include average case analysis, probabilistic algorithms, advanced graph problems and theory, distributed and parallel programming.

3 Prerequisites

CSCI 246 (Discrete) and CSCI 232 (Data Structures and Algorithms) are a prerequisite for this course. A student enrolled in CSCI 432 should be familiar with big-Oh notation, proof by induction, and Chapter 0 of CLRS.

4 Course Textbook

The required course textbook is Introduction to Algorithms, Third Edition by Cormen, Leiserson, Rivest, and Stein (CLRS).
5 Course Objectives

This course introduces students to the analysis and design of computer algorithms. In this course, students will:

- Understand the elementary concepts and properties of time complexity.
- Analyze asymptotic performance of algorithms.
- Apply important algorithmic design paradigms and methods of analysis.
- Design simple randomized algorithms.
- Design algorithms using dynamic programming.
- Use and analyze major graph algorithms and data structures.
- Design NP completeness reductions.
- Read current research publications in the area of algorithms.

6 Learning Outcomes

Upon completion of this course, students will be able to:

- Argue the correctness of algorithms using inductive proofs and loop invariants.
- Analyze worst-case running times of algorithms using asymptotic analysis.
- Explain the basic properties of randomized algorithms and methods for analyzing them.
- Analyze algorithms using amortized analysis, when appropriate.
- Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it.
- Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it.
- Describe the greedy paradigm and explain when an algorithmic design situation calls for it.
- Explain the major graph algorithms and their analyses.
- Demonstrate a familiarity with applied algorithmic settings - such as computational geometry, security and cryptography, - by reciting several recent algorithmic advances in different fields.
7 Grades

The final grade will be comprised of the following elements:

- Project: 35%
- Individual Homework: 20%
- Group Homework: 15%
- Exams: 30%

Note: Each project/homework assignment and test question will be graded on some number of points, and all assignments are not given equal weight. The point value will be given next to the assignment/question.

8 Homework

Homework must be typset in LaTeX, and submitted as a PDF electronically. Homework is designed to make you think. Working together to solve the homework is highly encouraged (and often necessary). Be sure to properly cite your collaborations though.

Group homework assignments can be done in groups of one to five students. Only one solution set should be handed in per group. These groups can change between assignments.

Homework is due at 23:59 on the due date. Once we start grading, late submissions will not be accepted.

9 Extra Credit

Opportunities to earn extra credit by attending colloquia will be announced in class and posted on the course website. To earn the extra credit (5 points towards homework), you must attend the entire presentation and write a 1-2 page summary and reflection on the presentation(s).

Another option for earning extra credit will be to neatly type (in LaTeX) the lecture notes from one class. You may work alone or with a partner. You may choose one class (or two if you work in partners) for which you wish to be the scribe, and let me know BEFORE the class so that there is at most one scribe/pair of scribes for each class. If you work in partners, you will split the points. The notes will be due one week after the lecture. The notes will be worth 15 points towards homework.

10 Collaboration Policy

Collaboration is encouraged on all aspects of the class, except where explicitly forbidden. Note:

- All collaboration (who and what) must be clearly indicated in writing on anything turned in. This should be stated at the top of any assignment.

- Unless otherwise indicated on an assignment, homework may be solved collaboratively, but solutions must be written up independently. Groups should be small enough that each member plays a significant role.

- For the project, every collaborator must contribute significantly. How the work is divided is at the discretion of the group.
11 Plagiarism

Plagiarism will not be tolerated in this course. According to the Meriam-Webster dictionary, plagiarism is ‘the act of using another person’s words or ideas without giving credit to that person.’ Proper credit means describing all outside resources (conversations, websites, etc.), and explaining the extent to which the resource was used. Penalties for plagiarism at MSU include (but are not limited to) failing the assignment, failing the class, or having your degree revoked. This is serious, so do not plagiarize.

12 Academic Integrity

By participating in this class, you agree to abide by the student conduct code.