CS4150
Algorithms
Spring Semester 2021

Instructor: Dr. Blair D. Sullivan
Teaching Assistants:
Madi Cooley, Erdi Fan, Andrew Fraser, Matthew Hooper

Class Times: Tues/Thurs, 12:25 - 1:45PM (Zoom meeting; link on Canvas)

Contact Information: All course communication will be handed through Piazza.

Office Hours: Schedule to be posted to Canvas/Piazza as soon as it is available. We will be using the TA Queue and Zoom breakout rooms.

Textbook: *Algorithms* by Jeff Erickson (required)
Freely available for download at https://jeffe.cs.illinois.edu/teaching/algorithms/.

Final Exam: Monday, May 3, 10:30 am – 12:30 pm
The exam is optional, but not reschedulable; see Assignments subsection for details.

Assignments: Weekly Canvas quizzes will be released by Friday and due the following Tuesday by 11:59 pm.

Problem Sets/Programming Assignments will be released by Wednesday and due the following Friday by 11:59 pm via Gradescope/Kattis.

Grading:		
	Problem Sets	50%
	Programming Assignments	30%
	Quizzes	20%

Academic Integrity: Although we encourage collaboration for learning, there are important limits on what is allowable in this course and serious consequences for violating these rules. Please read our policies (bottom of page 3) carefully, and ask if you have any questions.

Prerequisites

1. **Basic algorithms and data structures.** As this course has a prerequisite of CS 2420, we will not start from scratch with algorithms and data structures. We expect you to be familiar with basic

    - algorithms for sorting (selection sort, insertion sort, quicksort, mergesort),
    - algorithms for searching lists (linear search, binary search),
    - data structures for representing lists & sets (array lists, linked lists, binary search trees, hash tables),
    - abstractions of data structures (lists, queues, stacks, sets, maps), and
    - asymptotic complexity analysis (big-O bounds).

2. **C# or Java.** We will assume that you know how to program in either C# or Java (CS 2420 & CS 3500 are prereqs). You will be implementing algorithmic solutions regularly and need to understand how the concepts in this course can be realized as programs.

3. **Discrete math.** Many discrete math concepts underlie complexity analysis, algorithm design, and correctness reasoning. We expect you to be familiar with the topics from CS 2100, including

    - Big-O notation.
    - Exponents, logarithms, summations, and products.
    - Proof techniques, including proof by contradiction and mathematical induction.
    - Sets, relations, functions, trees, and graphs.

If you do not have this background and are not a competent programmer, you should not take this course.

Required Materials

The course will use the textbook *Algorithms* by Jeff Erickson,
downloadable for free online at https://jeffe.cs.illinois.edu/teaching/algorithms/.

This term we will be using Piazza for class discussion. Students will be also required to use Gradescope and Kattis to submit assignments. Additional instructions will be provided through Canvas/Piazza.

Course Description

Study of algorithms, data structures, and complexity analysis beyond the introductory treatment from CS 2420. Balanced trees, heaps, hash tables, string matching, graph algorithms, external sorting and searching. Dynamic programming, exhaustive search. Space and time complexity, derivation and solution of recurrence relations, complexity hierarchies, reducibility, NP completeness.

Learning Outcomes

We will study the algorithms, the ideas behind the algorithms, and the limitations of the algorithms that are used to solve a variety of programming problems that span the field of computer science. You will learn how to determine, both analytically and experimentally, the time and space complexity of algorithms, and you will learn the practical implications of complexity results. You will learn how to apply classical algorithm design techniques to obtain new algorithms, and how to reason about the correctness of algorithms. You'll learn about a variety of problems for which efficient algorithms exist, learn about a variety of problems for which no efficient algorithms have been discovered, and learn why most computer scientists believe efficient algorithms may not exist for those problems.

When you finish the course, you will be familiar with a broad range of algorithms, you will be equipped to read and exploit the extensive literature on algorithms, you will have the mathematical background to understand and reason about algorithms, and you will know how to analyze algorithms.

Course Policies

Attendance & Participation: You are expected to watch any pre-recorded lecture videos and take the weekly quiz prior to the corresponding synchronous class-time. Participation during the synchronous course components is expected, including asking questions and participating in real-time (anonymous) polls on material being presented.

***Communication*:** Use and regular monitoring of **Piazza is mandatory;** it will be used for all class communication (and replaces the Canvas discussion & announcement systems). **Please do not email the teaching staff directly;** instead, post all questions on Piazza. You can post anonymously if you choose, and you can elect to post to everyone, to the teaching staff, or to any individual. Piazza can be accessed through Canvas, or independently through the Piazza website or app.

Academic Integrity: Students are required to comply with the university policy on academic integrity found in the Code of Student Conduct found at regulations.utah.edu/academics/6-400.php. Please review the detailed definitions of academic misconduct.

Further, please be aware of the School of Computing's internal academic integrity policy - if you get caught cheating twice in any SoC classes, you will be unable to take any future SoC courses. See https://handbook.cs.utah.edu/2019-2020/Academics/policies.php for details.

Discussing the course's ideas, material, and assignments with others is a good way to learn, and we encourage it. However, there **are limits to the degree of cooperation that we will permit**. When taking an *online quiz or exam, you must work completely independently* of everyone else. For programs and problem sets, you must *limit your discussions with others to a high-level discussion of solution strategies*. If you do collaborate with others in this way, the solution that you submit *must identify the individuals involved and describe the nature of the collaboration*. You are not allowed to create a group solution, and all work that you hand in **must be written in your own words**. *If you base your solution on any other written solution, regardless of the source, you are cheating.* For programming assignments, this means you should not look at any code that solves the assigned problem.

Further, ***no course materials from this class may be shared or posted online.*** This includes lecture videos or slides posted to Canvas, problems from problem sets, quiz questions, and final exam problems (this includes posting assignments on sites like Chegg). Further, you may not post solutions – including your own – for any assigned work in this course (including code for Kattis problems on github, etc).

Your submission of any test or assignment indicates "I have neither given nor received unauthorized aid on this test or assignment." Please note that Kattis automatically performs plagiarism checking against all prior submissions (including those from others in the course), and we may also manually check any assignment (code or otherwise) for evidence of cheating.

I**f you cheat, you will be given an E in the course, and your case will be handled as described in Section V here.** *If you have any questions about what constitutes academic dishonesty, please ask - we are more than happy to clarify expectations.*

Assignments

This course will have *weekly quizzes, approximately one major assignment (programming assignment or problem set) due each week, and a final exam.* See the Course Schedule for assignment details broken out by week.

Multiple choice and short answer quizzes are used to assess your understanding of particular algorithms or to lead you through the analysis of algorithms. You will take these quizzes inside of Canvas, and they will be graded automatically. One re-take attempt will be allowed on quizzes to enable you to ask questions during the scheduled lecture time and/or office hours to clarify material.

Programming assignments will be submitted to the online Kattis system, which will automatically grade them. Your grade will be based on whether your program is correct and acceptably efficient. The system will accept programs written in a large variety of languages. Details on using Kattis will be posted on Piazza.

Problem sets must be typeset (using LaTeX or a word processor of your choice), and will be submitted as PDF through Gradescope. Figures must be professionally presented. No scans of hand-drawn figures or handwritten solutions will be accepted.

The **final exam** for this course will consist of five sections – one corresponding to each of Problem Sets 1-4, and one covering material from the final lectures of the course (on material beyond NP-hardness). **The final is *optional and will be during the period published on the university exam schedule*.** *You may choose to take any zero, one, or two of the first four sections of the exam*; for each section that you attempt, if your score on the final is better than that on the corresponding problem set, **it will replace 90% of that assignment in your average**. For example, if you earned a 50 on PS1, then got 100 on section 1 of the final, a 95 would be used for the first problem set in your average. If you submit solutions for more than two sections, we will only grade the first two. Regardless of what other sections you attempt, you may take the fifth section, where points earned will count as extra credit.

Grading Policy (Evaluation Methods & Criteria)

- 50% of your grade will be determined by problem sets.
- 20% of your grade will be determined by quizzes.
- 30% of your grade will be determined by programming assignments.

The lowest quiz score will be dropped. Problem set scores may be replaced by higher scores on the corresponding section of the final exam, as described in Assignments.

Grade Scale: (∞, 92] A, (92, 90] A-, (90, 88] B+, (88, 82] B, (82, 80] B-,
(80, 78] C+, (78, 72] C, (72, 70] C-, (70, 68] D+, (68, 62] D, (62, 60] D-, (60, 0] E

Late Policy: Quiz due dates are strict, and no late submissions will be accepted. Programming assignments and problem sets will be accepted up to 48 hours late; there is a penalty of 10% of the potential score on late assignments. (e.g. 3 points on a 30 point assignment) Please note there is no guarantee of support from the teaching staff (office hours, Piazza answers, etc) during the late submission period.

Regret Policy: In support of this course's Academic Integrity Policy, students may retract their submission for a problem set or programming assignment with no questions asked, up to 48 hours after submission. TAs will not grade these assignments; no credit may be earned for an assignment where a student has exercised the regret policy. You may replace such a zero on a Problem Set by taking the corresponding section of the final (see Final Exam policy). If you have questions concerning what constitutes academic misconduct or how to cite collaboration, *you must ask them prior to submission*.

CR/NC Option*:* Please note that although you may elect CR/NC through April 9, 2021, a CR grade cannot be used for required classes in the School of Computing (including 4150).

Note: This syllabus is meant to serve as an outline and guide for our course. Please note that I may modify it with reasonable notice to you. I may also modify the Tentative Course Schedule to accommodate the needs of our class; changes will be posted in the Canvas modules where appropriate, and announced on Piazza.

Tentative Course Schedule

Weeks 1-2 (1/18, 1/25): Review of Data Structures, Sorting, Big-O; Kattis Tutorial (Chapter 0)
Problem Set 0; Canvas Quizzes 1-2; Kattis Problems 1-2

Week 3 (02/01): Divide & Conquer, Solving Recurrences (Chapter 1)
Canvas Quiz 3; Problem Set 1

Week 4 (02/08): Backtracking (Chapter 2)
Canvas Quiz 4; Kattis Problem 3

Week 5 (02/15): Dynamic Programming (Chapter 3)
Canvas Quiz 5; Kattis Problem 4

Week 6 (02/22): Dynamic Programming, Greedy Algorithms (Chapters 3/4)
Canvas Quiz 6; Problem Set 2

Week 7 (03/01): Greedy Algorithms, Linear Programming (Chapter 4, Chapter H*)
Canvas Quiz 7; Kattis Problem 5

Week 8 (03/08): Intro to Graphs (Chapter 5); **No class 03/09 ("spring mini-break")**
Withdraw deadline 03/12
No Assignments Due

Week 9 (03/15): Graph Searching (Chapters 5/6)
Canvas Quiz 8; Kattis Problem 6

Week 10 (03/22): DP in DAGs, Strong Components (Chapter 6)
Canvas Quiz 9; Problem Set 3

Week 11 (03/29): Minimum Spanning Trees, Shortest Paths (Chapter 7/8)
Canvas Quiz 10; Kattis Problem 7

Week 12 (04/05): Bellman-Ford, All-Pairs Shortest Paths (Chapters 8/9)
Canvas Quiz 11; Kattis Problem 8

Week 13 (04/12):, P vs NP, NP-Hardness reductions (Chapter 12)
Canvas Quiz 12

Week 14 (04/19): More NP-Hardness, Approximation Algorithms* (Chapter 12, Chapter J*)*;*
Canvas Quiz 13; Problem Set 4

Week 15 (04/26): Randomized Algorithms*
Canvas Quiz 14

Week 16 (05/03): Final Exam on Monday, May 3 from 10:30 am to 12:30 pm.

** These topics are not directly mappable to Erickson; pointers to resources will be provided on Piazza.*

University Policies

1. ***COVID-19.*** Based on CDC guidelines, the University requires everyone to wear face coverings in shared public spaces on campus. Some students may qualify for accommodations through the Americans with Disabilities Act (ADA). If you think you meet these criteria and desire an exception to the face covering policy, contact the Center for Disability and Access (CDA) Further, all students *must self-report positive tests* for COVID-19 at https://coronavirus.utah.edu.

2. ***The Americans with Disabilities Act***. The University of Utah seeks to provide equal access to its programs, services, and activities for people with disabilities. If you will need accommodations in this class, reasonable prior notice needs to be given to the Center for Disability Services, 162 Olpin Union Building, (801) 581-5020. CDS will work with you and the instructor to make arrangements for accommodations. All written information in this course can be made available in an alternative format with prior notification to the Center for Disability Services.

3. ***University Safety Statement.*** The University of Utah values the safety of all campus community members. To report suspicious activity or to request a courtesy escort, call campus police at 801-585-COPS (801-585-2677). You will receive important emergency alerts and safety messages regarding campus safety via text message. For more information regarding safety and to view available training resources, including helpful videos, visit http://safeu.utah.edu.

4. ***Addressing Sexual Misconduct***. Title IX makes it clear that violence and harassment based on sex and gender (which Includes sexual orientation and gender identity/expression) is a civil rights offense subject to the same kinds of accountability and the same kinds of support applied to offenses against other protected categories such as race, national origin, color, religion, age, status as a person with a disability, veteran's status or genetic information. If you or someone you know has been harassed or assaulted, you are encouraged to report it to the Title IX Coordinator in the Office of Equal Opportunity and Affirmative Action, 135 Park Building, 801-581-8365, or the Office of the Dean of Students, 270 Union Building, 801-581-7066. For support and confidential consultation, contact the Center for Student Wellness, 426 SSB, 801-581-7776. To report to the police, contact the Department of Public Safety, 801-585-2677(COPS).

5. ***Drop/Withdrawal***. The last day to drop classes is **Friday, January 29, 2021**; the last day to withdraw from this class is **Friday, March 12, 2021**. Withdrawing from a course and other matters of registration are the student's responsibility. Please see https://www.coe.utah.edu/students/current/semester-guidelines/ for more information.

6. Other important information to consider includes:
    a. Student Code: http://regulations.utah.edu/academics/6-400.php

    b. Accommodation Policy (see Section Q): http://regulations.utah.edu/academics/6-100.php

7. ***Wellness Statement****.* Personal concerns such as stress, anxiety, relationship difficulties, depression, cultural differences, etc., can interfere with a student's ability to succeed and thrive at the University of Utah. For helpful resources contact the Center for Student Wellness at www.wellness.utah.edu or 801-581-7776.

8. ***Veterans Center****.* If you are a student veteran, the U of Utah has a Veterans Support Center located in Room 161 in the Olpin Union Building. Please visit their website for more information about what support they offer, a list of ongoing events and links to outside resources: http://veteranscenter.utah.edu/. Please also let me know if you need any additional support in this class for any reason.

9. ***Names and Personal Pronouns****.* Class rosters are provided to the instructor with the student's legal name as well as "Preferred first name" (if previously entered by you in the Student Profile section of your CIS account, which managed can be managed at any time). While CIS refers to this as merely a preference, I will honor you by referring to you with the name and pronoun that feels best for you in class or on assignments. Please advise me of any name or pronoun changes so I can help create a learning environment in which you, your name, and your pronoun are respected. If you need any assistance or support, please reach out to the LGBT Resource Center. https://lgbt.utah.edu/campus/faculty_resources.php

College & Department Policies

For information on withdrawing from courses, appealing grades, and more, see the College of Engineering guidelines at https://www.coe.utah.edu/semester-guidelines

Additional School of Computing policies, including Academic Misconduct, can be found in the undergraduate handbook:
https://handbook.cs.utah.edu/2019-2020/Academics/policies.php.