CPSC 335 - Algorithm Engineering - Syllabus

Spring 2022 - Prof. Dr. Wortman - Sections 04, 06

Catalog Description

Algorithm design using classical patterns: exhaustive search, divide and conquer, randomization, hashing, reduction, dynamic programming, and the greedy method. Asymptotic and experimental efficiency analysis. *NP*-completeness and decidability. Implementing algorithms to solve practical problems.

Prerequisites

Prerequisites: MATH 270B, CPSC 131; Computer Science or Computer Engineering major or minor; or Computer Science or Computer Engineering graduate standing.

Class Meetings

(see the Remote Delivery section for important temporary adjustments)

- 335-04: Fr 1:00PM - 3:45PM, room CS-408
- 335-06: TuTh 1:00PM - 2:15PM, room RGC-19

Instructor

Name: Dr. Kevin Wortman (he/him/his)

Email: **kwortman@fullerton.edu**

Phone: 657-278-2968

Office: CS-536

In order from fastest to slowest, the best way of reaching me outside of class is

1. Canvas message
2. Email
3. Phone

My work schedule is Monday-Friday, 9am - 6 pm. During these times I will respond to messages as soon as I can, and within two business days. Outside these times I may not be reading messages. Please plan accordingly.

Office Hours

(see the Remote Delivery section for important temporary adjustments)

- Time

    - TuTh 2:30-3:30 PM
    - Fri 11 AM - 12 noon
    - These regular times do not apply during Spring Recess or the Final Examinations period.
    - Other times may be available by request.
- Place
    - During remote delivery: **https://fullerton.zoom.us/j/281636101**
    - Otherwise: room CS-536
- Walk-ins are welcome at regularly-scheduled office hours. You do not need to ask before attending.
- When my office hours get busy, I answer questions on a round-robin basis so everyone gets a turn.

Important Dates

Monday February 21: President's Day, no classes or office hours

Thursday March 24, Friday March 25: Midterm Exam (week 9)

Monday March 28 - Sunday April 3: Spring Recess, no classes or office hours

Final Exams:

- 335-04: 5/20/2022, Friday, 1:00PM - 2:50PM
- 335-06: 5/17/2022, Tuesday, 1:00PM - 2:50PM

Please see **CSUF's official Academic Calendar** for more information, and your Titan Online portal for your full exam schedule.

Textbooks

Required:

- *Algorithm Design in Three Acts,* Kevin Wortman, Beta Edition, available as a free PDF in Titanium.

Optional:

- *Introduction to the Design and Analysis of Algorithms,* Anany Levitin, Addison Wesley, 3rd Ed., 2011.
- *Introduction to Algorithms,* Cormen, Leiserson, Rivest and Stein, 3rd Ed., 2009.
- These books provide an alternative perspective on the material of the course. They are not required, but are suggested to students who feel they would benefit from varied explanations of the material.
- The Cormen et al. book is an excellent reference and is a worthwhile investment for those students who expect to study algorithms further. It is required in CPSC 535 Advanced Algorithms.

G.E. Requirements

This course does not satisfy any CSU General Education requirements.

Weekly Topics

This is a tentative schedule of topics. We may need to adjust the schedule during the semester.

1. Week 1
    1. class **introduction**
    2. **terminology, pseudocode**, and algorithm patterns (ADITA chapters 1-2)

2. Week 2 (ADITA sections 3.1-3.6.3)
    1. **Efficiency analysis,** functions measuring resources, asymptotic notation, experimental analysis
    2. mathematical analysis, the standard model, step counting
3. Week 3
    1. proving efficiency classes, amortized analysis (ADITA sections 3.6.4-3.7)
    2. **review essential data structures** (ADITA chapter 4)
4. Week 4 (ADITA chapter 5)
    1. the **naïve pattern**, sequential search, the sorting problem, one-at-a-time sorting, and selection sort
    2. in-place selection sort
5. Week 5 (ADITA chapter 6)
    1. the **greedy pattern**; the Prim-Jarník algorithm for minimum spanning trees
    2. Dijkstra's algorithm for nonnegative single-source shortest paths
6. Week 6
    1. **exhaustive search** and optimization; generating and verifying candidates (ADITA sections 7.1-7.5)
    2. exhaustive search algorithms for minimum spanning trees, circuit satisfaction, traveling salesperson (ADITA section 7.6-7.9)
7. Week 7
    1. **decrease by half**; recursive algorithms, master theorem, merge sort process (ADITA sections 8.1-8.4)
    2. merge sort analysis, binary search, indivisible problems (ADITA sections 8.5 - 8.6)
8. Week 8
    1. **randomization**, pure quicksort (ADITA sections 9.1 - 9.5.2)
    2. analysis of quicksort, in-place quicksort (ADITA sections 9.5.3-9.5.5)
9. Week 9
    1. **reduction**, reduction to sorting, reduction to hash tables (ADITA sections 10.1-10.3)
    2. **midterm exam** (covers chapters 1-7)
10. Week 10
    1. priority search queues; optimizing Prim-Jarník and Dijkstra (ADITA section 10.4)
    2. **dynamic programming**, Fibonacci numbers, change making (ADITA sections 11.1-11.4)
11. Week 11 (ADITA sections 12.1-12.4)
    1. edit distance
    2. **limitations of algorithms**, **lower bounds**, trivial lower bounds
12. Week 12
    1. sorting lower bound, reduction arguments (ADITA sections 12.5-12.6)
    2. **intractable problems**, complexity classes, class *P,* unsolvable problems and **decidability**, halting problem (ADITA sections 13.1-13.3)
13. Week 13
    1. verifiable problems, *NP, NP*-hard, and ***NP*-complete** (ADITA sections 13.4-13.5)
    2. *NP*-completeness reduction proofs (ADITA section 13.7)
14. Week 14
    1. important NP-complete problems; the *P* vs. *NP* question (ADITA section 13.8)
    2. **overcoming limitations; parallel algorithms**; embarrassingly parallel problems; analyzing parallel algorithms
15. Week 15
    1. parallel sorting; parallel merge sort; samplesort
    2. catch-up or review
16. Final Exam Week: **final exam** (covers chapters 8-13)

Assignments and Grading

Grade Weightage

- Administrative Tasks: 5%, drop 1 lowest score
- Reading Quizzes: 15%, drop 3 lowest scores
- Homework: 0%
- Projects: 20%
- Midterm Exam: 30%
- Final Exam: 30%

Overall course grades are computed by first finding the average score in each category in the list above. Scores are normalized to a scale of 0 to 100 before being averaged. The average score for each category is then used to compute a weighted average, using the weights above.

Due to the logistical challenges imposed by the COVID19 pandemic, Canvas will automatically drop your lowest Administrative Task score and lowest 3 Reading Quizzes scores.

Letter Grades

Plus and minus grading is not used when determining overall course grades:

- "A" = 90 - 100%
- "B" = 80 - 89.99%
- "C" = 70 - 79.99%
- "D" = 60 - 69.99%
- "F" = 0 - 59.99%

Administrative Tasks

Administrative Tasks are things that you need to do in order to participate. These are official graded assignments to be transparent about the fact that you need to complete them by set deadlines, and to give you credit for this work. Examples:

- Introduce yourself in the Flipgrid icebreaker
- Agree to the course honor code
- Practice submitting a PDF, including a hand-drawn sketch, to GradeScope
- Practice requesting a regrade in GradeScope
- Set up a Tuffix programming environment
- Set up Proctorio (only if circumstances dictate, see below)

There may be additional Administrative Tasks.

Administrative Tasks are completed individually, and are graded on a Complete/Incomplete basis: 100% credit if completed on time, 0% otherwise.

Administrative Tasks are due Tuesdays 11:59pm.

Reading Quizzes

There is a reading assignment and Reading Quiz due before each class meeting (except the first meeting and midterm days). Reading Quizzes are an accountability measure to help you stay on track with the reading assignments.

Each Reading Quiz is an individually-completed multiple-choice quiz using the Canvas Quiz interface. The questions are rudimentary if you have completed the reading. You may attempt a quiz an unlimited number of times before the deadline.

Quiz deadlines are the start of lecture. Sections that meet twice per week (MoWe or TuTh) have one quiz due per lecture; sections that meet once per week (Fridays) have two quizzes due per lecture.

Reading Quizzes are graded automatically. Solutions are visible after the deadline.

With adequate time management, every student can earn 100% on the Reading Quizzes.

Homework

Homework is an opportunity to practice the design, analysis, execution, and communication of algorithms. It is excellent preparation for our exams and job interviews, which evaluate these skills.

Each homework has about three textbook questions. You may work in groups of up to three on homework. If you choose to work in a group, make one submission for the entire group.

You submit Homework to GradeScope as a digital PDF. You may create a digital document, or scan handwritten pages.

Homework assignments are due Thursdays 11:59pm. A Homework assignment is due every week, except for the first week, midterm week, and last week, so there are 12 assignments.

Homework submissions are be graded with constructive feedback and a numerical score on the same scale as exam questions. Detailed feedback and rubric scores are stored in GradeScope. Canvas only stores total scores. Report grading concerns as a Regrade Request in GradeScope. Solutions are not available.

Since the Homework category is weighted 0%, these scores will not impact your class grade. However, you are expected to complete every Homework in order to practice, and prepare for exams and interviews. If you want further practice, complete the textbook problems that are not assigned, or problems from the optional textbooks.

Projects

There will be two hands-on projects involving implementing and empirically analyzing algorithms. The topics for these projects are:

1. Empirical analysis and comparing efficiency classes
2. Exhaustive search; exponential versus polynomial time

Each project is divided into two parts: a programming part involving implementing algorithms in C++; and a written report where you present your empirical data and analysis. The two parts are separate assignment objects in Canvas.

The programming part uses GitHub Classroom to distribute starter code and store your completed code. You are given instructor-created unit tests to check the correctness of your code. The report involves creating line graphs and answering questions. You may work in groups of up to three on homework. If you choose to work in a group, make one submission for the entire group. Your group must be the same for both parts of a project.

Code is submitted to GradeScope. You may make unlimited submissions before the deadline. GradeScope will automatically grade your submissions according to instructor-created correctness tests. Unacceptable submissions will automatically receive an "F" grade. A submission is unacceptable if it exhibits any of the following:

- no author names;
- no work added to the starter code;
- cannot compile cleanly in the grading environment;
- causes the unit tests to crash (e.g. segmentation fault, invalid instruction);

- tampers with the autograding system; or
- is plagiarized.

This system works smoothly, but in rare cases students experience minor technical issues (e.g. illegal filenames) while making a submission. Plan to complete and submit your code at least 24 hours before the deadline to leave time to troubleshoot technical issues. We will not grant extensions or grade exceptions for submission technical issues.

Projects are due Tuesdays 11:59pm. The projects will be spaced so that you have at least three weeks to complete each project.

Code rubric scores are stored in GradeScope. Canvas only stores total scores. Your score on a programming assignment is precisely the score computed by GradeScope. Report grading concerns as a Regrade Request in GradeScope. Solutions are not available. Be advised that we may use the plagiarism detection feature of GradeScope.

Reports are submitted as a PDF file upload in Canvas. Report grading concerns by messaging your instructor. Be advised that we may use TurnItIn plagiarism detection.

Examinations

(see the Remote Delivery section for important temporary adjustments)

The class includes a midterm exam and one final exam. The final exam is not cumulative.

Exams are written, closed-book, closed-note, individually-completed, timed, tests. Each exam is a paper packet with a mixture of multiple choice, short answer, and long answer questions. The questions are similar in nature to Reading Quiz and Homework exercises. Engaging with all the exercises is good preparation for the exams. A study guide listing fair-game material is posted at least one week prior to each exam.

The midterm happens during the second class meeting of week 9. The final exam happens at the official time specified by the CSUF Scheduling Office and found in Titan Online. You must take each exam at the scheduled time. There is no rescheduling of exams, or makeup exams (except for the emergency Make Up Policy described below).

Exams are scanned and graded in GradeScope. Feedback and rubric scores are stored in GradeScope. Canvas only stores total scores. Report grading concerns as a Regrade Request in GradeScope. Solutions are not available.

Each exam may be curved. At the instructor's discretion, your curved score will be a linear function of your raw score that maps the highest raw score to 100% and the lowest acceptable performance to 60%.

Remote Delivery

Due to the ongoing COVID19 pandemic, **we will begin our spring semester by delivering our courses remotely for the first two weeks of the semester (January 22-February 6)** . It is possible that this period of remote delivery will be extended.

Remote delivery means the following adjustments:

- **Lectures** meet in Zoom. Use the Zoom tab on the left.
- **Office hours** meet in Zoom at **https://fullerton.zoom.us/j/281636101** .
- Lectures are recorded to the cloud. Office hours are not recorded.
- The office hours meeting has a **Zoom waiting room** and I will admit you as soon as possible, usually immediately. If you find yourself in the waiting room, that means that I'm already discussing

a private matter with a student. Please wait a few minutes and I will admit you as soon as possible.

Exams: In the unlikely event that the remote delivery period is extended to include the midterm exam (week 9) or final exam (examination week), the format of the affected exam(s) will change from a paper packet to a Canvas Quiz with Proctorio automated proctoring. Setting up Proctorio will be assigned as an Administrative Task due before the exam. If you are unwilling or unable to install Proctorio on a personal device, you can get a loaner laptop to use on exams through the **Dean of Students COVID-19 Related Device Request** . On exam days, the zoom meeting will only be used for emergency troubleshooting. So do not join the zoom meeting, and instead go directly to the exam in Canvas. If you have trouble starting the exam, join the zoom meeting for help.

Course Policies

University Policies

We are using Gradescope to collect and grade homeworks and projects. You will submit these assignments at **gradescope.com** (not Canvas).

Canvas will only store numerical scores for these assignments. After your submission is graded, detailed feedback will be at **gradescope.com** (not Canvas). If you believe your assignment was graded in error, use the regrade request feature in Gradescope.

The Gradescope interface is intuitive, but if you need help, please refer to the **Gradescope student workflow documentation** , in particular

- **Submitting an assignment**
- **Adding group members**
- **Viewing your submission**
- **Submitting a regrade request**