CpSc 8400: Design and Analysis of Algorithms

Instructor: Dr. Brian Dean   Spring 2019
Webpage: `http://www.cs.clemson.edu/~bcdean/`   TTh 12:30-1:45
Handout 1: Course Syllabus   McAdams 114

1   Overview and Course Goals

The study of algorithms is a significant part of the foundation for the discipline of computing. Over the past several decades, research in algorithmic computer science has advanced at a rapid pace its contributions have had a profound impact on almost every area of science and industry. In this graduate-level course, we aim to provide an introduction to the study of algorithms that is both broad and deep. The primary goals of the course are:

- to become proficient in the application of fundamental algorithm design techniques (e.g., divide and conquer, greedy algorithms, dynamic programming),

- to gain familiarity with the main theoretical tools used in the analysis of algorithms (e.g., recurrences, probability theory, etc.)

- to study and analyze different algorithms for many of the most common types of "standard" algorithmic problems (e.g., sorting, searching, graph problems), and

- to introduce students to some of the prominent subfields of algorithmic study (e.g., computational geometry, randomized algorithms, data structures, etc.) in which they may wish to pursue further study.

The final point is particularly worth noting — the instructor intends this course as well as its successor, CpSc 9400, to serve as a "gateway" for new students with interests in theoretical computer science to learn about opportunities for pursuing further research and studies in areas related to algorithms.

This is a *theory* course. We will occasionally discuss implementation-related issues when appropriate, but for the most part all lectures and assignments will involve the analysis of algorithms in a rigorous mathematical fashion. Many of these are quite elegant and draw on a variety of useful techniques that should serve to broaden and strengthen your analytical abilities.

2   Prerequisites

Students should have taken at least one algorithms course at the undergraduate level, and they should have a reasonable amount of mathematical maturity. Familiarity with basic proof techniques (e.g., induction) and fundamental discrete mathematics (e.g., "Big Oh" notation, combinations and

permutations) is assumed. One of the goals of the class is to illustrate the power and simplicity of randomized algorithms, so familiarity with basic probability theory will be helpful; however, we will briefly review most of this material in class.

3 Tentative List of Topics

The field of algorithms includes far too much interesting material than we can hope to cover in just a single course. In order to cover as much ground as possible, we will move at a fairly fast pace through the semester. The following is a preliminary list of topics the instructor plans to cover. It is subject to change, so we may omit some of these topics (or cover them only at a high level) and we may also cover topics not on this list. Fundamental topics often taught in an introductory algorithms course (e.g., basic sorting algorithms and simple data structures) will be covered at a relatively fast pace, since we assume students already have some familiarity with these concepts.

- **Analysis Techniques.** Solving recurrences, expected and "with high probability" analysis of randomized algorithms, amortized analysis of data structures, basic modular arithmetic.

- **Sorting and Selection.** Insertion sort, bubble sort, merge sort, (randomized) quicksort, heap sort, counting sort, radix sort, bucket sort, stable in-place sorting, randomized and deterministic selection in linear time, lower bounds on comparison-based sorting.

- **Data Structures.** Arrays and linked lists, binary heaps, balanced binary search trees, universal hash tables.

- **Divide and Conquer Algorithms.** Merge sort, quicksort, quickselect, repeated squaring, Strassen's fast matrix multiplication algorithm, the Fast Fourier Transform and its applications, in-place matrix reblocking and transposition.

- **Optimization.** Greedy algorithms, dynamic programming, brief introduction to integer and linear programming and convex continuous optimization. Solving linear and nonlinear systems of equations.

- **Graph Algorithms.** Depth-first and breadth-first search, topological sorting, strongly-connected components, shortest paths, minimum spanning trees, possibly flow and bipartite matching algorithms.

- **Computational Geometry.** Convex hulls, half-space intersection, closest pair and diameter of a set of points, sweep line and randomized incremental construction techniques.

- **Hard Problems.** Approximation algorithms for NP-hard problems, heuristics for solving hard problems: branch and bound, neighborhood search, genetic algorithms, simulated annealing.

- **Number Theory and Cryptography.** Euclid's algorithm, primality testing, key exchange, the RSA public-key cryptosystem.

4   Instructor Information

Instructor: Dr. Brian Dean
Office: McAdams 205
Phone: 656-5866
Email: bcdean@cs.clemson.edu
Web: http://cs.clemson.edu/~bcdean/
Office Hours: Tuesday 2:00-3:30am.

5   Grading

The following items contribute to your grade in this course:

- **Homework (35%).** Assignments will be issued weekly. Homework problems range from simple exercises to more challenging questions asking you to design and analyze new data structures. Each homework problem is worth a certain number of points, indicated on the problem. Homework solutions must be *typeset*, and turned in electronically as a single PDF file via handin.cs.clemson.edu prior to the start of class on the day they are due. Extensions to homework assignments are only generally only available if a student is sick or has some other reasonable excuse.

- **2 Quizzes (20% each).** Two quizzes will be given, one in the middle of February and the other near the end of March.

- **Final Exam (25%).** The date of the final exam will be announced as soon as possible. The final exam is cumulative, covering all material from the entire course.

Final letter grades are not assigned according to an absolute scale. Based on final numeric scores, the instructor will draw cutoffs between letter grades as deemed appropriate.

6   Course Materials

There is no required text for this course. The instructor may post excerpts from a draft of an algorithms textbook he is currently writing (not to be redistributed outside this course). This book is a "dual-media" textbook, presenting high-level concepts in printed text and explanations of technical low-level details using animated multimedia "whiteboard lectures". Lecture slides and videos will also be posted on the course website.

For additional reference material, students may want to consult other prominent algorithms texts (e.g., the book of Cormen, Leiserson, Rivest, and Stein, or the book by Kleinberg and Tardos). One can also find a great deal of useful material on the web, typically in the form of slides and course notes from other universities.

7   Course Conduct

- **Attendance.** Although attendance is not mandatory, you are generally expected to arrive

on-time for every class meeting. Class attendance and participation will be considered as a "tie breaker" for those with borderline grades. Students may leave class after 15 minutes if the instructor does not arrive by that time. If a student cannot attend class, it is his or her responsibility to obtain notes and other information on the material that was missed.

- **Academic Integrity.** Cheating and plagiarism have no place in a respected academic institution such as Clemson; such activities will not be tolerated and will be reported to the department for further disciplinary action. Please ask the instructor if you have any questions on what constitutes plagiarism or on academic integrity in general.

- **Use of Previous Course Material.** Since some problems are re-used from year to year in this course, it is strictly forbidden to consult any homework solutions from previous years (written by the instructor or by students who have taken the course in previous years). Do not ask students having previously taken the course for specific details on individual homework problems.

- **Use of Web Content.** Please feel free to consult the web for *general* information (e.g., useful mathematical formulae, definitions, supplemental reading on course material). However, you are not to go looking on the web for information *specifically* related to any particular homework problems. The purpose of the homework is to help you develop your skills at problem-solving, not web searching. Use your best judgment as to what is acceptable and what is not, or if there is any question, you can always ask the instructor.

- **Collaboration.** Collaboration on homework is encouraged to the extent that it is carried out appropriately — with all students involved contributing equally to a discussion of how a concept works or how a problem should be solved, then working independently on writing up solutions. Solution write-ups and code must always be your own individual work. If you do use some material from external sources like books or web pages, you must include appropriate citations for your sources. In addition, on every assignment please list any students with whom you have collaborated.

- **Disabilities.** It is University policy to provide, on a flexible and individualized basis, reasonable accommodations to students who have disabilities. Students are encouraged to contact Student Disability Services to discuss their individual needs for accommodation.

- **Title IX Statement.** Clemson University is committed to a policy of equal opportunity for all persons and does not discriminate on the bais of race, color, religion, sex, sexual orientation, gender, pregnancy, national origin, age, disability, veteran's status, genetic information, or protected activity (e.g., opposition to prohibited discrimination or participation in any complaint process) in employment, educational programs and activities, admissions and financial aid. This includes a prohibition against sexual harassment and sexual violence as mandated by Title IX of the Education Ammendments of 1972. The policy is located at http://www.clemson.edu/campus-life/campus-services/access/non-discrimination-policy.html. Alesia Smith serves as Clemson's Title IX Coordinator and may be reached at knight1@clemson.edu or 656-3181.

- **Feedback.** Please feel welcome to ask for feedback on your performance in the class at any time. In addition, please feel welcome to send constructive feedback to your instructor at any time — your instructor places a great deal of value on the ability to present challenging material in manner that is clear, well motivated, and fun, and he appreciates any feedback you can offer on how to improve the quality of the course.