1 Overview and Course Goals

The study of algorithms is a significant part of the foundation for the discipline of computer science. 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. Moreover, the advent of the World Wide Web and increasingly widespread high-performance computing environments have ushered in a new era of “data-driven” applications accompanied by a host of new algorithmic needs and challenges.

This graduate-level course aims to satisfy three main goals:

- To help students become more proficient at reading, writing, and presenting advanced algorithmic material.
- To undertake a case study of prominent algorithmic techniques that have had the greatest impact in practice, so students will understand the deep mathematical foundations behind some of history’s greatest algorithmic success stories.
- To study a collection of elegant, modern algorithmic ideas and innovations that are becoming increasingly relevant, and that will most likely play a role in tomorrow’s greatest algorithmic success stories.

This course is designed to appeal to a wide range of students. It is perhaps best described as an “applied theory” course, since it contains a fair amount of deep mathematical content, but with all topics being motivated entirely by modern applications in practice.

2 Prerequisites

The official prerequisite for this class is prior coursework in algorithms and/or data structures at the graduate level. Although many of the topics considered in this course do not depend much on specific prior algorithmic knowledge, we will assume familiarity with common mathematical and algorithmic analysis techniques, as well as a number of fundamental algorithmic results (e.g., sorting, common graph algorithms). If you do not satisfy the prerequisite, you must obtain permission of the instructor to take this course.
We will be using a wide range of mathematical tools in this course, including those from algebra and linear algebra, probability and statistics, and some calculus. Familiarity with these areas will certainly be helpful, but this is not necessarily required, as part of the goal of this class is to teach computer science students how to effectively use these mathematical tools.

3 Tentative List of Topics

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.

- **Network and Data Analysis.** Clustering, classification, regression, density estimation, ranking, embedding, Pagerank/HITS and other eigenvector-based tools, recommendation systems.

- **Cryptography.** Key exchange, secret sharing, public-key cryptosystems, digital signatures, voting systems, private information retrieval and other applications in secure multi-party communication.

- **Distributed Data Management.** Consistent hashing, distributed hash tables.

- **Optimization and Heuristics.** Overview of optimization models and algorithms, heuristics for discrete and continuous optimization.

- **Modern Applications of Classical Algorithms.** Matchings and auctions applied to ad placement, point-to-point shortest path heuristics for mobile GPS units.

- **Robust Management of Massive Data Sets.** Compression (Huffman, JPEG, Lempel-Ziv), error-correcting codes, searching and mining large data sets (inverted indices, BLAST, nearest neighbor search).

4 Instructor Information

**Instructor:** Dr. Brian Dean  
**Office:** McAdams 205  
**Phone:** 656-5866  
**Email:** bcd@cs.clemson.edu  
**Web:** http://cs.clemson.edu/~bcd/  
**Office Hours:** You may stop by my office any time.

5 Grading

There are not quizzes or exams in this course. The following items contribute to your grade:

- **Scribing (50%).** In lieu of traditional homework assignments, each student will be responsible for scribing one or more lectures (the total number depends on the enrollment). This
involves taking careful notes during lecture and producing a polished, typeset copy of your lecture notes (typesetting must be done with LaTeX, using a template provided with the instructor). Each set of scribe notes should go above and beyond the material covered directly in lecture by correcting any necessary mistakes, filling in any missing details with full mathematical precision, and adding an appropriate set of references to related background material in the literature (along with a brief literature review summarizing these references). Final copies of all scribe notes will be assembled into a collection of polished lecture notes available to all students. Grades for scribe notes will depend on the extent to which the instructor needs to assist with editing, combined with the quality and depth of the final notes.

- **Presentation (50%).** Starting in late September, each Friday class meeting will be reserved for a pair of 25-minute student presentations. Each student will select (with help of the instructor) one or more papers from the modern research literature on an advanced algorithmic topic to read and present individually in class. Presentations each take 25 minutes (20 minutes plus 5 minutes of questions+answers), and are expected to resemble conference talks of high quality. Slides are required. By class on the Wednesday prior to each presentation, presenting students must submit a short (1..2 page) abstract summarizing the key results they will be presenting, in order to give other students a good idea what to anticipate. Grades will be determined by the quality of presentation and slides, and the extent to which each student manages to convey the most important key ideas from their respective papers within a limited amount of time (e.g., a student who attempts to present their topic in too much detail may actually receive a lower score than one who gives a more appropriate clear, concise summary as well as key intuition behind the main concepts, abbreviating unnecessary details).

Final letter grades are not assigned according to an absolute scale. Rather, the instructor will determine letter grades cutoffs as deemed appropriate.

6 Course Materials

There is no required text for this class, since course content is drawn from a number of disparate sources. The instructor will be posting a large number of original research papers on the course website for students to read.

7 Course Conduct

- **Attendance.** Although attendance is not mandatory, you are generally expected to arrive on-time for every class meeting (and this is especially important on student presentation days). 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.
• **Collaboration and Citation.** Collaboration on all class exercises is highly encouraged. Feel free to have your friends review your scribe notes for you to assist with editing. Modulo editing feedback, however, it is expected that each student is ultimately the sole author of their scribe notes. If you do use some material from external sources like papers, books or web pages, you must include appropriate citations for your sources.

• **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.

• **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.

## 8 Algorithms Seminar

Students are encouraged to attend the ongoing algorithms seminar in the Clemson School of Computing that meets every Friday at 3:30. Since the seminar covers a broad range of advanced topics from many areas of algorithmic study, this is an excellent resource for students interested in further algorithmic research. Please email the instructor if you wish to be added to the mailing list for seminar announcements.