CS325: Analysis of Algorithms

Prof. Glencora Borradaile

Fall 2013

Course content

In the first half of this course, we will start **analyzing algorithms** for correctness and running time. We will design **divide and conquer** and use **recurrence relations** to analyze **recursive algorithms**. The first test will cover this material. Projects 1 and 2 will involve implementing and analyzing iterative and recursive algorithms.

The second half the course is all about solving much more difficult problems. We will study **dynamic programming** and **linear programming**, with Projects 3 and 4 looking at these methods in more depth. We will characterize the difficulty of problems by way of **NP completeness**. Throughout the last half of the course, we will develop **heuristics** for solving the travelling salesperson problem. Groups will present their ideas mid-way through the project. The final exam will focus mostly on material from this second half of the course.

Learning resources

All learning materials will be made freely available on the course website.

Evaluation of student performance

Grades will be posted to the course website via an anonymizing identifier.

Participation You are expected to be an active participant in class. Before in-class tutorials, you are expected to try the practice problems. Before lectures, you are expected to watch or read any required material.

Projects: 50% of your grade. Projects are worked on in teams of 2-3 that will be formed in the first class. Projects involve implementing and analyzing algorithms (experimentally and mathematically). Written reports will be submitted. On the day of the submission, there will be a short quiz designed, if you participated fully in the project, to be easy and reflective of your performance on the project. Each project quiz will be worth 5% of your final grade and each project report will be worth 5% of your final grade.

All projects are due at the start of class on the date listed. Projects submitted within 24 hours of the deadline will not be penalized, but projects handed in more than 24 hours late will not be graded without prior arrangements.

Exams: 50% of your grade. There are two tests (midterm and final), each worth 25% of your final grade. These tests are non-cumulative. For example, the second test will explicitly ask questions on topics covered after the first test; however, mastery of those topics covered by the first test will be implicitly tested by these questions.

Practice problems will be provided on the course website. These problems will not be graded. Solutions to these problems will be taken up by TAs during in-class tutorials. Trying these questions before the tutorial and studying them after is the recommended way to prepare for the mid-quarter and final exam.

Measurable Student Learning Outcomes

At the completion of the course, students will be able to

1. Use O, Ω, Θ and simple recurrences to analyze the time complexity of iterative and recursive algorithms.

   This will be tested in project 1 and the midterm exam.

2. Prove the correctness of algorithms.

   This will be tested in the midterm (for recursive algorithms) and final exams (for dynamic programming algorithms).

3. Implement recursive, iterative and heuristic algorithms.

   Projects 1,2 and 4, respectively, will test this.

4. Prove that a problem is NP-complete using reductions.

   Project 3 and the final exam will test this.

Students with Disabilities

Accommodations are collaborative efforts between students, faculty and Disability Access Services (DAS). Students with accommodations approved through DAS are responsible for contacting the faculty member in charge of the course prior to or during the first week of the term to discuss accommodations. Students who believe they are eligible for accommodations but who have not yet obtained approval through DAS should contact DAS immediately at 737-4098.