CS 477/677 Analysis of Algorithms
Fall 2007 – Dr. George Bebis

Catalog Description: Analysis and design of algorithms on sequences, sets, graphs, and trees. Geometric, algebraic and numeric algorithms, FFTs, reductions. Parallel algorithms.

Prerequisites: CS 365 or EE 291. Also, good working knowledge of data structures such as linked lists, trees, and dynamically allocated structures is required. If you do not meet the prerequisite requirements for this course, you should see me immediately. *Credit hours:* 3.0

Meets: TR 2:30-3:45 PM (SEM 347)

Instructor: Dr.George Bebis

Office: 235 SEM **Phone:** 784-6463

E-mail: bebis@cse.unr.edu

Course Webpage: http://www.cse.unr.edu/~bebis/CS477

Office Hours: MW: 11:00 am - 12:30 pm and by appointment

Required Text:

Introduction to Algorithms by T.H. Cormen, C.E. Leiserson, and R.L. Rivest, 2nd edition, McGraw-Hill, 2001.

Optional Texts:

Fundamentals of Algorithms, by G.Brassard and P.Bratley, Prentice Hall, 1996.
Computer Algorithms, by Horowitz, Sahni, and Rajasekaran, Computer Science Press, 1996.
Algorithms, by Sedgewick, Addison-Wesley.

Objectives

The design and analysis of algorithms is the core subject matter of Computer Science. Given a problem, we want to **(a)** find an algorithm to solve the problem, **(b)** prove that the algorithm solves the problem correctly, **(c)** prove that we cannot solve the problem any faster, and **(d)** implement the algorithm. Designing an algorithm for a computational problem involves knowledge of the problem domain, a thorough knowledge of the data structures that are available and suitable and no small measure of creativity. This course concentrates on the above problems, studying useful

algorithmic design techniques, and methods for analyzing algorithms.

Course Outline (tentative)

- Introduction/Mathematical Foundations (Chapters 1, 3, Appendix A)
- Recurrences (Chapter 4)
- Intro to Sorting Algorithms (Chapter 2)
- Randomized Algorithms (Chapter 5)
- More on Sorting Algorithms (Chapters 6-9)
- Searching Algorithms (Chapters 11-14)
- Dynamic Programming (Chapter 15)
- Greedy Algorithms (Chapters 16)
- Graph Algorithms (Appendix B4, Chapters 22-25)
- NP-Complete Problems (Chapter 34)

Exams and Assignments

Grading will be based on two exams, 6-8 quizzes, 6-8 homework assignments, a course project, and a short presentation. Details are provided below:

- Quizzes will be announced at least one class period in advance.

- Homework problems will be assigned and collected for grading on a regular basis. Each homework assignment will include 5-8 problems. Undergraduates will be required to solve only those problems designated as **"U-required"**. Graduate students will be required to solve all problems assigned. Homework solutions will be made available within a week of the due date for the assignment.

- There will be two exams: a midterm and a final. The material covered in the exams will be drawn from the lectures, the quizzes, and the homework. Undergraduates will be required to solve only those problems designated as **"U-required"**; while graduate students will be required to solve all of them.

- There will be a course project which will be done in groups of two. Specific details and due dates will be announced in class. Graduate students will be required to do some extra work.

- There will be a short presentation (approximately 15-20 minutes) on a contemporary issue related to algorithms. Each presentation will be done in groups of two. Presentation topics will be decided in coordination with the instructor. The presentations should be professional as if it was presented in a formal conference (i.e., slides/projector).

Course Policies

- Lecture slides, homework assignments, and other useful information will be posted on the course web page.

- Regular attendance is highly recommended. If you miss a class, you are responsible for all material covered or assigned in class.

- You should carefully read the section on Academic Dishonesty found in the UNR Student Handbook (copies of this section are available from http://www.unr.edu/stsv/acdispol.html) Your continued enrollment in this course implies that you have read it, and that you subscribe to the principles stated therein.

- Discussion of the assignments is allowed and encouraged between students. However, each student (or group) would be expected to do his/her own work. Assignments which are too similar will receive a zero.

- **No late homework or project report will be accepted.** If you are unable to hand in your homework or project report by the designated deadline, you must notify me *before* the deadline.

- No incomplete grades (INC) will be given in this course and a missed quiz/exam may be made up only if it was missed due to an extreme emergency.

Useful Tips

Since the material in this course is highly integrated, a limited understanding of one topic will have a serious effect on the understanding of subsequent topics. You should expect to spend many hours on this course outside the classroom. Do not expect to fully understand the material covered in this class if you do not study on a regular basis.

Disability Statement

Any student with a disability needing academic accommodations is requested to speak with me or contact the Disability Resource Center (Thompson Building, Suite 101), as soon as possible to arrange for appropriate accommodations.

Grading Scheme

Midterm: 20%

Final: 20%

Quizzes: 20%

Homework: 15%

Course project: 15%

Presentation: 10%

A 90 and above

B 80-89

C 70-79

D 60-69

F<59

Important dates

10/16/2007 - Midterm
10/19/2007 - last day for dropping classes
11/22/2007 - Thanksgiving Day
12/12/1007 – Prep Day
12/13/2007 - Final exam (2:15pm - 4:15 pm)

Outcomes and Objectives

ABET Criterion 3 Outcomes	Course Outcomes	Assessment Methods/Metrics	CS Program Objectives Impacted	CIE Program Objectives Impacted
a.	Students will acquire knowledge and understanding of analyzing the space/time complexity of both recursive and non-recursive algorithms using analytic techniques (involving O-notation, recurrence equations, the Master Theorem, etc.) and high-level abstractions (abstract data types).	Examinations, quizzes, and homework will measure level of knowledge and understanding.	1, 2	1,2
c	Students will develop appreciation of design, analysis and algorithmic performance by working on a programming project.	Testing of project performance. Evaluation of written documentation for the design, implementation and final project report.	3, 4	3, 4
d	Students will acquire an understanding of team dynamics by working in groups on a programming project and a short presentation.	Graded project reports. Evaluate student presentations. Evaluate comments written by students discussing their experiences working in groups.	4	4
g	Students will improve their communication skills by working in groups, writing a project report, and making a short presentation to the rest of the class.	Graded project reports. Evaluate student presentations. Evaluate comments written by the students discussing their experiences working in groups.	4	4
j	Students will acquire knowledge of contemporary issues in the area of algorithms by giving a short presentation on a contemporary issue to the rest of the class.	Evaluate level of understanding during student presentations. Questions in the final will test student knowledge and level of understanding on the contemporary issues discussed in class.	1, 2, 4	4

ABET Criterion 3 Outcomes:

a. an ability to apply knowledge of mathematics, science, and engineering
b. an ability to design and conduct experiments, as well as to analyze and interpret data
c. an ability to design a system, component, or process to meet desired needs
d. an ability to function on multi-disciplinary teams
e. an ability to identify, formulate, and solve engineering problems
f. an understanding of professional and ethical responsibility
g. an ability to communicate effectively
h. the broad education necessary to understand the impact of engineering solutions in a global and societal context
i. a recognition of the need for, and an ability to engage in life-long learning
j. a knowledge of contemporary issues
k. an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice

Computer Science Program Objectives:

Our graduates will have achieved:

1. a broad general education assuring an adequate foundation in science and mathematics relevant to computing.
2. a solid understanding of concepts fundamental to the discipline of computer science.
3. good analytic, design, and implementation skills required to formulate and solve computing problems.
4. the ability to function, communicate, and continue to learn effectively as ethically and socially responsible computer science professionals.

Computer and Information Engineering Program Objectives:

Within 3 to 5 years of graduation our graduates will:

1. be employed as computer engineering professionals beyond entry level positions or be making satisfactory progress in graduate programs.
2. have peer-recognized expertise together with the ability to articulate that expertise as computer engineering professionals.
3. apply good analytic, design, and implementation skills required to formulate and solve computer engineering problems.
4. demonstrate that they can function, communicate, collaborate and continue to learn effectively as ethically and socially responsible computer engineering professionals.