Course Information

26th August, 2008

Data Structures and Algorithms
(JDEP 283, Fall 2008)
URL: http://www.cse.unl.edu/~vinod/jdef08/info.html
Handout 1

	Instructor	TA1	TA2
Name	Vinod Variyam	Tyler Lemburg	Raghu Tewari
Office	262, Avery	Kauffman 133	Avery 123C
Phone	(402) 472-5002	(308) 390-2504	(402) 472-4679
E-mail	vinod@cse.unl.edu	tlemburg@cse.unl.edu	rtewari@cse.unl.edu
Off. Hrs	TuTh 12:20 - 1:20pm and by appointment	TuTh 3:30 - 4:30pm	M 11:30am - 12:30pm (A123c) Tu 1:00 - 2:00 pm (A123c) W 11:30am - 12:30pm (A13A)

	Venue	Time
	Kauffman 112	TuTh 1:30pm-3:20pm

Textbook	*Introduction to The Design and Analysis of Algorithms* Second Edition by Anany Levitin, Addison Wesley
Prereq	CSCE 156 or equivalent
Web site	http://www.cse.unl.edu/~vinod/jdef08/info.html

Course Overview

Relevance and Broad description.

An algorithm is a set of instructions that, when followed, solves a specific problem. Programs are implementations of algorithms that are executed by computers. A data structure is a way of storing data in a computer so that it can be retrieved efficiently when required. Algorithms for problems such as sorting and graph reachability are at the heart of many real-life computer applications. The efficiency/correctness of these algorithms directly influence the efficiency/correctness of the applications. Often a well-designed data structure is critical for designing ultra-fast algorithms.

Learning Goals and Methods

The goals of this course are to study classic data structures and algorithms that solve common problems and to learn standard approaches to solving new problems. A rigorous approach to the analysis and comparison of algorithms will be followed that includes asymptotic notation and proofs of correctness. Discrete mathematics, which forms the foundation for rigorous analysis, will be covered as needed. The course will involve substantial programming and written assignments. A broader set of objectives for this course is to teach critical thinking, how to learn, and how to communicate technical concepts. These objectives will be met through lectures, challenging assignments, regular quizzes, and a final examination.

Topics to be Covered

- The basics of algorithm analysis including asymptotic notation, complexity classes, discrete mathematics, recursion and induction (proofs), creating and solving recurrence relations.

- Algorithmic techniques, including, Brute Force, Divide-and-Conquer, Decrease-and-Conquer, Transform-and-Conquer, Space and Time Tradeoffs, Dynamic Programming, Greedy Techniques, Distributed Algorithm.

- Theory of computing, including, finite state machines, the halting problem, tractable and intractable problems, complexity classes like P, NP, and NP-Complete.

If time permits, we will also try to cover some advanced topics including networks flows and linear programming.

Textbook and Course Materials

Most of the lectures will be based on the text book *Introduction to the Design and Analysis of Algorithms*, Second Eidtion, by Ananay Levitin, Addison Wesley. We will also cover essential topics from discrete mathematics as needed.

You will be doing programming in the C++/Java language and working with the Linux/Unix operating system. The book, *Just Enough UNIX*, Paul Anderson, McGraw-Hill, 2003, will be helpful in learning the operating system and the programming environment.

Testing and Grading

This section describes grading guidelines. The instructor retains the authority to make changes if necessary. Your grades will be based on 4 homeworks assignments, several in-class quizzes, a midterm, and a final comprehensive examination. Following table gives a guideline for evaluating your final marks.

Assignments	45%
Midterm	10%
Quizzes	30%
Final Exam	15%

Marks to Letter Grade Conversion

Following table gives the tentative conversion of marks to letter grades. It may be modified if necessary.

A+	≥ 97	B+	≥ 87	C+	≥ 77	D+	≥ 67	F	< 60
A	≥ 93	B	≥ 83	C	≥ 73	D	≥ 63		
A-	≥ 90	B-	≥ 80	C-	≥ 70	D-	≥ 60		

Homework Assignments

Homeworks will have two parts; a programming part and an analytical part (pencil and paper problem solving). There will be 4 homework assignments. You may expect one assignment every two or three weeks. Assignments are due at the beginning of the class on the given due date. Late assignments will lose 25% of total points per day. Hence you will not get any points for an assignment handed in 4 days late. Programming assignments should be submitted via the Web-handin program.

Clarity and legibility of presentation of your submission are as important as your answers to problems. You are very strongly advised to typeset your solution using some document processing system. If the grader cannot easily read your writings, you may not be awarded full points even if you claim your answers are correct.

Examinations

There will be one midterm and a final exam. Tentative schedule for the midterm is **16th October, during 1:30- 3:20pm**. The final examination is scheduled for **Thursday, 18th December, during 1:00pm-3:00pm**. The final examination will be comprehensive.

Quizzes

There will be in-class quizzes approximately every two weeks. In these quizzes you will be asked problems based on the materials covered in the immediate past. The first quiz will also be served as a prerequisite test.

Academic Dishonesty

Students enrolled in any computer science course are bound by the department of Computer Science and Engineering's Academic Integrity Policy. I strongly recommend every student to read and understand it. Any violation of the policy will be dealt with severely. You can read the policy at `http://cse.unl.edu/ugrad/academic_integrity.php`

Special Needs

Any student in this course who has a disability that necessitates accommodation should contact the instructor as soon as possible to discuss the appropriate accommodations necessary to complete the course requirements.