

1 Syllabus

CSc 4520/6520
Fall, 2006

Sushil K. Prasad

Course Syllabus

Course Title: Design and Analysis of Algorithms

Text: Cormen, Leiserson, and Rivest, *Introduction to Algorithms*, MIT Press.

Prerequisite: CSc 3410 – Data Structures

Class Time and Place: 2:45-4:25 TR; 313 ALC

Instructor: Sushil K. Prasad; Room 1453, 34 Peachtree; **Email** sprasad@gsu.edu (do not leave phone messages).

Office Hours: TT 9:00-9:50 a.m. and 4:25-5:15 p.m. Meetings at other hours by appointment only. Office hours may be cancelled occasionally due to meetings.

Content: The purpose is to learn to be able to design various classes of algorithms, and analyze them.

Design Techniques: Divide-and-Conquer, Greedy, Dynamic Programming.

Topics: Analyzing algorithms and problems; Growth rates; Searching; Sorting; Selection; Lower Bounds; Graph Algorithms: minimum spanning tree and shortest path; Introduction to NP-complete Problems; Coloring, Clique, Satisfiability; Cook's Theorem, Approximation Algorithms. Some material will be covered by reading assignments.

Attendance: You may be dropped if you have more than two unauthorized absence. Students are responsible for all the material covered or assigned (whether or not in the text). Nov 21-24 is thanksgiving break. Last class is on Dec 7 (Thu).

Withdrawals: The last date to withdraw is Oct 16 and possibly receive a W.

Grading:

	CSc 4520	CSc 6520	Dates
Home Assignments	30%	30%	
Test 1	20%	20%	Sep 21
Test 2	25%	20%	Oct 26 (or, if Take Home then due date)
Test 3	25%	20%	Dec 5
Term Project	bonus 5%	10% (mandatory)	Due Nov 30

Final grades will be relative to the class performance (to be calculated separately for CSc 4520 and CSc 6520). To ensure a grade, however, 90 and above will result in an 'A,' 80-89 a 'B,' 70-79 a 'C' and 65-69 a 'D.' Relative grading may yield a better grade even with a lower score. There will be no makeup test given except for documented medical emergencies. There will be zero credit for late submissions.

Academic Honesty: Work submitted for grading *must* be student's own. Collaboration for work assigned to be carried out individually is not allowed; in case of any plagiarism in my professional opinion, all involved will receive 'F' for the course, and will risk further action. Refer to the university catalogs for policy on plagiarism.

Disclaimer: The course syllabus provides a general plan for the course; deviations may be necessary.

A teacher can never truly teach unless he is still learning himself. A lamp can never light another lamp unless it continues to burn its own flame. The teacher who has come to the end of his subject, who has no living traffic with his knowledge but merely repeats his lesson to his students, can only load their minds, he cannot quicken them.

Rabindranath Tagore, Indian Poet
Nobel Laureate in Literature, 1913

2 Term Project

CSc 4520/6520 Term Project

1. Select a problem from the following list (get approval for your topic). Identify three distinct algorithms for solving the problem. Use the textbook, other reference books, and research papers for this purpose. Submit the statement of the problem and sketches of the three algorithms in no more than 500 words.
2. Code one of the algorithms. Pay attention to the data structures. Also, ensure that your coding is free of any unnecessary overheads, time- or space-wise.
Run your program on random inputs of varying lengths. Report your program and its run-time performance in 2-3 pages, typed double-spaced. Your report should describe the algorithm, the data structures, the time behavior, and an analytical worst-case time and space complexity analysis. Show the run-time data of your program using a plot with input length on x-axis, and time required on y-axis. Submit this report with a hard copy of your program (commented adequately). The report should be succinct and well-written.
3. Repeat 2 for your second algorithm.
4. Repeat 2 for your third algorithm.
5. Write a final report comparing and contrasting the three algorithms. Which algorithm would you recommend (and under what circumstances)? Substantiate drawing upon your previous reports. Also submit a plot containing data for all three algorithms. Limit the final report to two pages - append all your previous reports to this.

Due Dates: Submit at the beginning of the class:

- Part I: Oct 3
- Part II: Oct 31
- Part III: Nov 14
- Part IV and V: Nov 30

List of Project Topics

1. Priority Queues: heap, binomial heap, and fibonacci heap
2. splay tree, heap, and skewed heap
3. maximum flow in a weighted graph
4. convex hulls
5. traveling salesperson problem
6. graph coloring problem
7. VLSI layout algorithms

Some Reference Books for CSc 4520/6520

Algorithms Mainly dealing with algorithms, but often include complexity analysis

Basse, Computer Algorithms

Even, Graph Algorithms

Greene and Knuth, Mathematics for the Analysis of Algorithms

Horowitz and Sahni, Fundamentals of Computer Algorithms

Knuth, The Art of Computer Programming, 3 volumes

Minieke, Optimization Algorithms for Networks and Graphs

Nijenhuis and Wilf, Combinatorial Algorithms

Reingold, Nievergelt and Deo, Combinatorial Algorithms

Skeina, Algorithm Design Manual

Wells, Elements of Combinatorial Computing

NP-Completeness: Emphasizing Complexity Theory

Aho, Hopcroft and Ullman, The design and analysis of computer algorithms

Davis, Computability and Unsolvability

Garey and Johnson, Computers and Interactability: A guide to the theory of NP-Completeness (the most well-known introductory text; easy to read)

Hartmanis, Feasible computation and provable complexity properties (very difficult)

Hennie, Introduction to computability

Machtey and Young, An introduction to the general theory of algorithms

Savage, The complexity of computing

Parallel Processing:

Quinn, M. J., Parallel Computing: Theory and Practice, McGraw-Hill, NY, 1994. (A good introductory text)

J'aJ'a, J., An Introduction to Parallel Algorithms, Addison-Wesley Pub Co, Reading, MA, 1992.

F. T. Leighton, Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes, Morgan Kaufmann, CA 1992.

J. H. Rief, Synthesis of Parallel Algorithms, Morgan Kaufman, San Mateo, CA, 1993.

3 Sample exam

Answer all questions. Show your work, as partial credit will be given. Write neatly and to the point. For graph problems, assume n and m , respectively, to be the number of nodes and edges.

Questions

1. (a) Formulate the TSP problem as a decision problem, $TSP_{Decision}$ by giving its *Instance* and *Question*. Prove that $TSP_{Decision}$ is in NP .
(b) Define class NP -Complete (NPC). Prove that $TSP_{Decision}$ is in NPC by reducing *Hamiltonian Cycle (HP)* problem to it. Does it make $TSP_{Optimization}$ NP -Hard or NP -Complete?
2. (a) Formulate all-to-all shortest path problem using dynamic programming and derive the Warshal-Floyd algorithm. For this, give the numerical formulation, its recursive solution (with brief justification), its bottom-up calculation algorithm, and time and space complexities.
(b) Workout Warshal-Floyd algorithm on the following graph
3. Workout Kruskal's MST algorithm on the above graph. Show the details of the disjoint set data structure - both its tree and array representations - for each C-Find and W-Union operations.
4. Prove that $n - 1$ iterations of relaxation steps in Bellman-Ford algorithm suffices.

4 Student Information

CSc 4520/6520 Student Information

Name:

Major:

Minor:

Prerequisite CSc 3410 (Data Structure) completed:

Semester:

Instructor:

Grade:

Optional – Company working for (Address): .

.

Optional – Telephone numbers where you can be reached:

Any special topics you would like to see covered in this course: .

.

.

.

.

.

Your expectations for this course: