Algorithms — Spring, 2011
CS 5800 — Professor G. Cooperman

Course Description:

Niklaus Wirth famously wrote "programs = algorithms + data structures". This course is about developing algorithms and data structures. It is distinguished from programming in that we will work primarily with pseudo-code.

Note that progress in computer science would be greatly accelerated if we were able to completely eliminate the implementation stage. This would allow us to spend more time on improving the algorithms, while eliminating all the drudgery of the implementation stage: no debugging, no object-oriented programming, no software engineering. This course explores the benefits of such a utopia.

The course presents a tour of interesting algorithms that continue to have a large impact. Examples include: search (breadth-first and depth-first search); data compression (Huffman encoding, which is the second stage of the Deflate algorithm used by zip and gzip); dynamic programming; and automation of logic (formal verification). There will be a special emphasis on graph algorithms and search, leading up to an NP-complete algorithm: BDD (binary decision diagrams). BDDs provide an approach to formal verification that has been broadly successful in silicon circuit verification. Formal verification is now increasingly used for proving software correctness.

Faculty Information:

Professor G. Cooperman
Office: 336 West Village Hall
Phone: 617 373-8686
e-mail: gene@ccs.neu.edu
Office Hours: Mondays, 5:00 - 6:00; Mondays, after class;
and either by appointment or walk-in (open door policy).

Textbook:

The textbook for the course is:
Algorithms, by Sanjoy Dasgupta, Christos Papadimitriou, and Umesh Vazirani; McGraw-Hill, 2006.

Exams and Grades:

The course grade will be based on homeworks, a mid-term, and a final exam. The midterm and final exam will contribute 30% and 40%, respectively, to your course grade. The remainder of the grade will be based on written homework assignments (30%). There will be a homework assignment most weeks when there is no exam.

Syllabus:

Week	*Topics*	*Reading*
Sept. 9	Math Preliminaries; Hashing, Divide-and-Conquer	Prologue; Chap. 1, 2
Jan. 10	Graph (BFS, DFS)	Chapters 3 – 4
Jan. 24	Graph (Dijkstra, priority queue)	Chapter 4
Jan. 31	Greedy Algorithms: Huffman enc., Horn formulas	Chapter 5
Feb. 7	Dynamic Programming (Part 1))	Chapter 6
Feb. 14	Midterm; More Dynamic Programming	Chapter 6
Mar. 7	Dynamic Programming (part 2)	Chapter 6
Mar. 14	NP-Completeness	Chapters 8 – 9
Mar. 21	Non-polynomial time algos.	Chapters 8 – 9
Mar. 28	Non-poly. (BDD: formal verification)	Handouts
Apr. 4	More formal verification	Handouts
Apr. 11	Topics of interest to class, Last Day of Classes	
Apr. 25	Final Exam	