Michael Goldwasser
Princeton University

Basic Course Information

Theory of Algorithms

Theory of Algorithm is an intermediate course on the design and analysis of algorithms. For the first half of the course, we will focus both on presenting general techniques for designing efficient algorithms, as well as on models and methods for comparing different algorithms by formally analyzing their complexity (e.g. running time, use of space, quality of output).

Later in the course, we will examine several other models for computation, such as when we allow our algorithms the use of randomization or parallelism, or when we restrict the information given to our algorithms. We will also ask the question of whether or not there exist problems for which there are no efficient techniques. This will lead us to the notion of NP-Completeness, and we will see examples of many such problems which are believed to be impossible to solve efficiently. For several of these problems, however, we will show that there are efficient algorithms which are able to find a solution which has cost reasonably close to the optimal.

Prerequisites

The official prerequisite for this course is cos226 (Algorithms and Data Structures) as well as cos341 (Discrete Mathematics). During this course, we will be designing many advanced data structures and algorithms, and the background in algorithms and data structures will be necessary for two reasons. Many of our algorithms will need to use more basic data structures as building blocks, and although we will try when possible to review the earlier material, we expect that students will have seen many of these concepts previously. Even for entirely new material, we expect students to be able to think about how to describe an algorithm, an argue about why it works, and how much time it takes.

In addition, a solid background in mathematics will prove invaluable for this course, in that we will argue rigorously about the correctness and performance of almost every algorithm and data structure we introduce. The ability to write clear and formal proofs will be relied upon heavily throughout the course.

The people

- **Instructor: Michael Goldwasser**
  e-mail: `wass@cs.princeton.edu`
  office: Room 204 (CS Building)
  telephone: 258-1749
  office hours: Tu/Th 10:30–11:30am (or by appointment)

- **Teaching Assistant: Amit Chakrabarti**
  e-mail: `amitc@cs.princeton.edu`
  office: Room 313 (CS Building)
  telephone: 258-6126
  office hours: tba

- **Teaching Assistant: Kostas Tsioutsiouliklis**
  e-mail: `kt@cs.princeton.edu`
  office: Room 317 (CS Building)
  telephone: 258-1796
  office hours: tba

- **Secretary: Sandy Barbu**
  e-mail: `barbu@cs.princeton.edu`
  office: Room 323 (CS Building)
  telephone: 258-4562

The lectures

The material will be presented in two weekly lectures. Class participation is encouraged during class. In addition, in order to allow even more participation with smaller groups, precept sessions will be held each week by the TAs, for which attendance is recommended.

- **Class Lectures: Prof. Goldwasser**
  time: Tuesday/Thursday, 9:00-10:20am
  place: Room 105 (CS Building)

- **Precept Section: Amit Chakrabarti**
  time: tba
  place: tba

Online Information

Web Page:	`http://www.cs.princeton.edu/courses/cs423`
Newsgroup:	`pu.cs.423`

We will attempt to keep all online information related to the course up-to-date. There will be a class web page, to provide current information about office hours, lecture topics, due dates, and the web page will contain electronic copies of all handouts.

Announcements regarding homework problems, common questions, etc will be posted to the newsgroup as necessary. Late breaking news will be posted to this newsgroups, as well as placed on the main course web page.

Collaboration Policy

Each problem throughout the course will be designated with one of two instructions, "work entirely on your own" or "you may collaborate with others." When asked to work entirely on your own, you should not discuss the problem with anyone other than the instructor or the TAs.

For problems on which we allow collaboration, you should follow the following guidelines strictly. **The write-up of your solutions must still be done entirely on your own.** That is, you may feel free to discuss with classmates general ideas, approaches, examples, or stumbling blocks while trying to understand a homework problem, however when it comes time to write up your solution, you may not discuss this with others, nor may you use anyone else's written solution as a guide. Our goal is that you will eventually be able to understand the problem well enough, that you could lock yourself in a room with a blank piece of paper, and still be able to write down your solution.

When you do collaborate with classmates, please work in groups of at most five students, and write the names of your collaborators at the beginning of each such solution. On all problems, we expect that you not search for direct solutions using any other outside reference materials. **Violations of any of the above policies will be dealt with strongly.**

Grading Standards

- **Homework (56%)** There will be nine weekly homework assignments, on which students are to work alone unless otherwise specified. Assignments will generally be handed out on Tuesday in class, and will be due the following Tuesday, at the *beginning* of class at 9:00am. Late homeworks may be turned in between 9:00am–5:00pm on the due date, however with a 20% penalty. After 5:00pm, no late homeworks will be accepted. As a new homework will go out each Tuesday, it is important to stay up-to-date. To make up for the strict late policy, we will throw out the lowest of your nine homeworks before calculating your overall grade. From time to time extenuating cirumstances do arise and in such cases extensions will be granted; requests for extensions should be mailed to the instructor.

  Each homework will consists of some suggested practice problems, and then four or five required problems which you will turn in. Throughout this course, problems will require a formal proof by default. Try to be both clear and precise. The organization of your solutions is equally as important as the correctness.

- **Midterm (14%)** In the weeks before Spring Break, we will give a take-home midterm in place of a homework assignment. The format for the midterm will be similar to the homeworks, except that it will be more comprehensive.

- **Final (30%)** This final will be in-class, and closed book. It is intended to test both breadth and depth of understanding, and will consists of a variety of both short answer and more involved questions.

- **Extra Credit (5%)** Most of the homeworks, and each of the exams will have an extra credit problem. Your scores on these questions will be kept *independently* of the rest of the scores, and your total extra credit score for the class will add up to 5% on top of the overall grade. In addition, class participation will have a positive effect on the final grade.

Textbooks

The only required textbook for this class is

- *Introduction to Algorithms* by Cormen, Leiserson, and Rivest, MIT Press, 1990.

However, the following books may also be very helpful in their presentation of material covered in this class, and will be held on reserve in the engineering library:

- *Computers and Intractability: A Guide to NP-completeness*, by Garey and Johnson, W. H. Freeman & Co., 1979.
- *Approximation Algorithms for NP-hard Problems*, by Dorit Hochbaum, PWS Publishers, 1997.
- *Introduction to the Theory of Computation*, by Michael Sipser, PWS Publishers, 1996.
- *Data Structures and Network Algorithms, CBMS 44*, by Robert Tarjan, SIAM, 1983.

Finally, a list of other useful sources for information about algorithms and data structures includes:

- *Design and Analysis of Computer Algorithms*, Aho and Ullman, Addison-Wesley, 1974.
- *Data Structures and Algorithms*, Aho and Ullman, Addison-Wesley, 1985.
- *The Art of Computer Programming*, Vol. 1–3, Knuth, Addison-Wesley, 1973–.
- *Randomized Algorithms*, Motwani and Raghavan, Cambridge Press, 1995.
- *Algorithms in C*, Sedgewick, third edition Parts 1-4, Addison-Wesley, 1998.
- *An Introduction to the Analysis of Algorithms*, Sedgewick and Flajolet, Addison-Wesley, 1996.

Syllabus (subject to change)

#	Date	Topic	Reading	Due	Out
1	Tues. Feb. 3	Introduction, Administration (example: closest pair)	Ch. 1, 35.4		hw1
2	Thur. Feb. 5	Analysis Techniques (asymptotics, recurrences, amortization)	Ch. 2 – Ch. 5, Ch. 18		
3	Tues. Feb. 10	Comparison Based Sorting, Selection	Ch. 9.1, 10.1, 10.3	hw1	hw2
4	Thur. Feb. 12	Minimum Spanning Trees	Ch. 24		
5	Tues. Feb. 17	Priority Queues and Binary Heaps	Ch. 7	hw2	hw3
6	Thur. Feb. 19	Binomial and Fibonnaci Heaps	Ch. 20, Ch. 21		
7	Tues. Feb. 24	Disjoint Set-Union	Ch. 22	hw3	hw4
8	Thur. Feb. 26	Augmented Search Trees (interval trees)	Ch. 15		
9	Tues. March 3	Dynamic Programming	Ch. 16	hw4	mid
10	Thur. March 5	Graph Searching (DFS, BFS, connectivity)	Ch. 23		
11	Tues. March 10	Shortest Paths (single-source)	Ch. 25		
12	Thur. March 12	Shortest Paths (all-pairs)	Ch. 26	mid	

Spring Break

#	Date	Topic	Reading	Due	Out
13	Tues. March 24	Randomization (quicksort, selection, treaps)	Ch. 6, Ch. 8, Ch. 10.2		hw5
14	Thur. March 26	Hashing	Ch. 11		
15	Tues. March 31	Cryptography, Primality	Ch. 33.7, 33.8	hw5	hw6
16	Thur. April 2	Parallel Computation	Ch. 30		
17	Tues. April 7	Complexity Classes: P and NP	Ch. 36	hw6	hw7
18	Thur. April 9	NP-Completeness, Cook-Levin Thm	Ch. 36		
19	Tues. April 14	NP-Completeness: Reductions	Ch. 36	hw7	hw8
20	Thur. April 16	Approximation Algorithms	Ch. 37		
21	Tues. April 21	Approximation Algorithms (cont.)	Ch. 37	hw8	hw9
22	Thur. April 23	Approximation Algorithms (cont.)	Ch. 37		
23	Tues. April 28	Online Algorithms, Competitve Analysis			
24	Thur. April 30	Online Algorithms (cont.)		hw9	

Reading Period

	TBA	Final Exam		