Syllabus

COEN 279 Computer Algorithm
Department of Computer Engineering
Santa Clara University

Dr. Tunghwa Wang
Phone: (408) 886-8460
Course website:
Office Hours:

Fall Quarter 2016
Email address: twang1@cse.scu.edu
http://www.cse.scu.edu/~twang1
Tuesday & Thursday 9:00-9:30AM

Course Description

Techniques of design and analysis of algorithms: proof of correctness; running times of recursive algorithms; design strategies: brute-force, divide and conquer, dynamic programming, branch-and-bound, backtracking, and greedy technique; max flow/matching. Intractability: lower bounds; P, NP, and NP-completeness. (4 units). Also listed as AMTH 377.

Prerequisites

Basic programming experience and data structures (COEN 12): Most chapters for data structures will be skipped assuming such knowledge already learnt.

Required Textbooks

1. "Introduction to Algorithms Third Edition", by Thomas H. Corman, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein ISBN: 978-0-262-03384-8, The MIT Press 2009

References

1. Donald E. Knuth 2011 "The Art of Computer Programming", Vol. 1, Fundamental Algorithms, Vol. 2, Seminumerical Algorithms, Vol. 3 Sorting and Searching, Vol. 4A Combinatorial Algorithms, Addison-Wesley
2. S. Sahni 1998 "Data Structures, Algorithms, and Applications in C++" McGraw-Hill
3. Robert Sedgewick 2002 "Algorithms in C++ Part 5: Graph Algorithms" Addison-Wesley
4. Narasimha Karumanchi 2016 "Data Structure and Algorithmic Thinking with Python" CareerMonk Publications
5. Jens Clausen 1999 "Branch and Bound Algorithms – Principles and Examples", University of Copenhagen PDF
6. David A Bader, William E. Hart, Cynthia A. Phillips 2004 "Parallel Algorithm Design for Branch and Bound", in H. J. Greenberg "Tutorials on Emerging Methodologies and Applications in Operations Research" Kluwer Academic Press PDF
7. David G. Sullivan 2012 "Recursion and Recursive Backtracking" http://www.fas.harvard.edu/~cscie119/lectures/recursion.pdf
8. Richard E. Korf 1999 "Artificial intelligence search algorithms". In Atallah, Mikhail J. "*Handbook of Algorithms and Theory of Computation* " CRC Press.
9. Georg Becker 2008 "Merkle Signature Schemes, Merkle Trees and Their Cryptanalysis" Ruhr-Universit¨at Bochum. p. 16. Retrieved 2013-11-20 PDF
10. Matteo Frigo, Steven G. Johnson 2005 "The Design and Implementation of FFTW3" Proceedings of the IEEE 93(2):216-231
11. Paolo D'Alberto, Alexandru Nicolau 2007 "Adaptive Strassen's Matrix Multiplication" in Proceedings of the 21st Annual International Conference on Supercomputing Pages 284-292 June
12. Michael A. Bender, Erik D. Demaine, Martin Farach-Colton 2000 "Cache-oblivious B-trees" in Proceedings of the 41st Annual Symposium on Foundations of Computer Science page 399-409

Course Objectives

1. To learn and master advanced knowledge, design, analysis, and implementation in computing algorithms and underlying data structures.
2. To read and understand research publications in the technical area of computing algorithms, beyond that of the traditional textbook level.
3. To conduct group project and to equip for scholarly research in computing algorithms to solve intended application domain problem.
4. To develop capability to analyze, experiment, and then propose the best algorithm for any given application domain problem.

Expected Learning Outcomes

1. Demonstrate the knowledge of the basic algorithm design techniques and the impact on underlying data structure and vice versa.
2. Demonstrate the skill of algorithm analysis: proof of correctness, running time analysis, amortized analysis, time-space trade-off, and intractability: lower bounds, P, NP, and NP-completeness.
3. Demonstrate the knowledge of designing efficient algorithms and analyzing its computational complexity and time-space trade-off.
4. Demonstrate the ability to realize the best implementation of the designed algorithms for the given application domain.
5. Demonstrate the ability to read current research papers and implement example research group project in any given application domain.

Grading Policy

Course grade is determined based on the total score (maximum 1000 points + up to 200 optional bonus points for extra work) from:

1. Mid-term and final exams of 200 points each (close book with one A4 note, no sitting together, no wireless connection.) Makeup exams (must have a very good reason) are much difficult than normal exams.
2. Three programming assignments of 200 points each (late penalty: 30 points/day.) No makeup is allowed. You can call Design Center at 408/554-4909 for setup account or IT support, and ssh linux.scudc.scu.edu to work remotely.
3. A group (2-3 people in a team) programming term project of 300 points (late penalty: 60 points/day.) No makeup is allowed.
4. Bonus assignments will be assigned at each week with 20 points each. Due before next lecture begin by email to me with title "coen279 b*N*" (where *N* can be 2, 3, …, 10) and cc to the grader. (You automatically get 20 points for "coen279 b1" when you are on the roster.) The solution for bonus assignments will be posted on my protected web page. Please read solutions of bonus assignments before asking questions. No late work accepted for bonus assignments.

Table 1: Grade-score table

950 - 1200	900 - 949	850 - 899	800 - 849	750 - 799	700 - 749	650 - 699	600 - 649	0 - 599
A	A-	B+	B	B-	C+	C	C-	F

Course Schedule (Tuesday & Thursday 7:10pm-9:00pm)

Table 2: Course Schedule

#	Week	Readings	Remarks
1	09/20 09/22	Computational complexity, time-space trade-off, randomized algorithms	Ch. 2, 3, & 5
2	09/27 09/29	Divide-and-conquer	Ch. 4, 6, 7, & 30
3	10/04 10/06	Tree algorithms	Ch. 12 & 13
4	10/11 10/13	Dynamic programming	Ch. 15 program #1 due 10/13
5	10/18 10/20	Greedy algorithms	Ch. 16 mid-term exam 10/20
6	10/25 10/27	Greedy algorithms, Amortized analysis	Ch. 16 & 17
7	11/01 11/03	Graph algorithms	Ch. 22 program #2 due 11/01 last day to withdraw 11/04
8	11/08 11/10	Graph algorithms	Ch. 23, 26
9	11/15 11/17	P, NP, and NP-complete	Ch. 34 program #3 due 11/17
10	11/29 12/01	Approximation algorithms	Ch. 35 final exam 12/01
11	12/06	Review and summary	

Reminder

- No cheating, and no register complaint without talking to me first.
- No incomplete. No sit-in or audit the class except formally registered.
- Read files under /home/twang1/tips for help.
- Handouts,
- assignments, and solutions will be posted on the web. You should check the class web site at least once a week (and don't forget to refresh the webpage to get the latest versions). You are responsible for printing and bring the handout to the class if you prefer printed pages.
- Office hours: Tuesday & Thursday 9:00AM-9:30AM.

Honor Code

All students taking course in the school of engineering agree, individually and collectively, they will neither give nor receive unpermitted aid in examinations or other course work that is to be used by the instructor as a basis of grading.

Disability Accommodation Policy:

To request academic accommodations for a disability, students must contact Disability Resources located in The Drahmann Center in Benson, room 214, (408) 554-4111; TTY (408) 554-5445. Students must provide documentation of a disability to Disability Resources prior to receiving accommodations.