CS 3510: Design and analysis of algorithms
Spring 2017
Section B
MWF 2:05pm - 2:55pm

General Description: Efficient algorithms for fundamental computational problems. NP-Completeness.

Prerequisites: A course in constructing proofs (equivalent to cs2050), introduction to programming (equivalent to cs1312), background in elementary data structures, and a discrete mathematics course such as Math 3012 or permission of the instructor.

Instructor: H. Venkateswaran, KACB 3366, venkat@cc.gatech.edu, Office Hours: Tue/Thu 1pm - 2pm and by appointment.

Text book: Title: Algorithms

Authors: Sanjoy Dasgupta, Christos Papadimitriou, and Umesh Vazirani

Grading: Best of the following two options:

- Option 1: Five homework assignments worth 20%, best three of four tests worth 60%, and a final examination worth 20%.
- Option 2: Five homework assignments worth 20%, four tests worth 80%, and no final examination.

Homework Guidelines: Homework guidelines posted on T-Square.

Syllabus:

1. Introduction. (Chapter 0)
2. Divide and Conquer: integer multiplication, merge-sort, divide and conquer recurrences, a lower bound for sorting, Strassen's matrix multiplication, linear-time algorithm for median-finding, Fast-Fourier transform, butterfly network (Chapter 2).
3. Elementary graph algorithms: graph traversals and applications such as topological sorting and strongly connected components. (Chapters 3, 4).
4. Greedy algorithms: minimum spanning trees, binary heaps, union by rank and path compression, shortest paths, finding negative cycles, Huffman codes, approximation for set-cover (Chapters 4, 5).
5. Dynamic programming: matrix-chain multiplication, longest increasing subsequences, edit-distance, Floyd-Warshall algorithm for all-pairs shortest paths and transitive closure, knapsack (Chapter 6).
6. Maximum-flow, Ford-Fulkerson algorithm, max-flow min-cut theorem, maximum matching in bipartite graphs (Chapter 7).
7. NP-Completeness: Basic notions such as reducibility and completeness. Assume SAT is NP-Complete. Examples of NP-Complete problems: 3SAT, independent set, vertex cover, clique, subset sum (reduction from 3sat - notes), 3-coloring (Chapter 8).
8. Simple approximation algorithms: vertex cover, TSP with triangle inequalities. (Section 9.2).
9. Other topics as time permits.