16:198:512 - Introduction to Data Structures and Algorithms

Course Number: 16:198:512

Course Type: Graduate

Semester 1: Fall

Semester 2: Spring

Credits: 3

Description:

This course is required for all students joining the Computer Science M.Sc. program. Students from other departments can request special permission numbers provided they meet the prerequisites as stated below.

The course studies a variety of useful algorithms and analyze their complexity; students will gain insight into principles and data-structures useful in algorithm design.

This course counts as category A for the M.Sc. degree requirement. This course does NOT count as category A for Ph.D. students. Ph.D. students should take 16:198:513 instead.

Category: A (M.S.)

Prerequisite Information:

Calculus and Discrete Math, Ch 0 of the Textbook and Chapters 1, 2, 3 of the reference below.

Topics:

1. Complexity Measures. Methods for expressing and comparing complexity of algorithms: worst and average cases, lower bounds, and asymptotic analysis.

2. Searching, Sorting. Lower bounds for comparison-based sorting; merge sort, quick sort, heapsort, insertion sort (binsort, radix sort).

3. Divide and Conquer. Fast integer multiplication; recurrences; the master theorem; randomized median and selection algorithms; quicksort; fast matrix multiplication.

4. Graph Search Algorithms. Graphs representations; depth first search; topological search; strongly connected components. Breadth first search and layered DAGs.

5. Greedy Algorithms. Spanning trees and cuts, union-find and path compression; minimum spanning tree (MST) algorithms; Sample of randomized algorithms.

6. Shortest Paths (SPs) in Digraphs. Single-source SPs for nonnegative edge weights; priority queues and Dijkstra's; SPs in DAGs; single-source SPs for general edge weights.

7. Dynamic Programming. Paradigm of SPs in DAGs; longest increasing subsequence; (approximate) string matching; integer and (0, 1) knapsack problems; chain matrix multiplication; single-pair reliable SPs, all-pairs SPs; independent sets.

8. Introduction to Linear Programming

9. Network Flows. Max flow min cut theorem; bipartite matching; Menger's theorem and disjoint dipaths. Global minimum cuts.

10. NP-completeness and Problem Reductions and Coping with NP-Completeness. Approximation Algorithms and Fix Parameter Tractability.

11. Algorithm Sampler* Some more advanced topics of current interest like Page Rank, External Memory Algorithms, Streaming Algorithms, Parallel Algorithms, Distributed Algorithms, and Quantum Computing.

Course Material:

Textbook: Algorithms by Dasgupta, Papadimitriou, and Vazirani, McGraw Hill

Expected Work: Weekly Homeworks and a Programming Project (about 25% of the final Grade).

Exams: Weekly Quizzes plus two Midterms and a cumulative Final

Learning Goals:

Students will be prepared to contribute to a rapidly changing field by acquiring a thorough grounding in the core principles and foundations of computer science (e.g., algorithmic principles, techniques of program design, creation, testing; and key aspects of computer hardware).

Students will acquire a deeper understanding on selected topics of more specialized interest, and be able to critically review, assess, and communicate current developments in the field.