10M11CI211: Advanced Algorithms

Course Credit: 3
Semester: M.Tech, II

Introduction

This is a graduate course on the design and analysis of algorithms, covering several advanced Topics not studied in typical introductory courses on algorithms.

Course Objectives (Post-conditions)

Knowledge objectives:

The need for efficient algorithms arises in nearly every area of computer science. But the type of problem to be solved, the notion of what algorithms are "efficient," and even the model of computation can vary widely from area to area. In advance algorithms course, we will survey many of the techniques that apply broadly in the design of efficient algorithms, and study their application in a wide range of application domains and computational models. Techniques to be covered include network flow, linear programming, fixed-parameter algorithms, and approximation algorithms.

Application objectives:

- Some familiarity with several of the main thrusts of work in algorithms-sufficient to give you some context for formulating and seeking known solutions to an algorithmic problem.
- Sufficient background and facility to let you read current research publications in the area of algorithms.
- A set of tools for design and analysis of new algorithms for new problems that you encounter.

Expected Student Background (Preconditions)

Prerequisites include "Introduction to algorithms", linear algebra, and mathematical maturity (since we'll be doing some correctness proofs). The course is especially designed for students interested in theoretical computer science.

Topics Outline:

S NO	Topics	Hrs
1	Analysis of Algorithms: computational models, order notation, amortized complexity	5
2	Techniques for designing efficient algorithms: recursion, divide-and-conquer, dynamic	6

	programming, balancing and backtracking, branch and bound.	
3	Network flows (max flow and min-cost flow/circulation)	4
4	Graph algorithms (use of Fibonacci heaps lazy delete)	5
5	Linear programming and duality (structural results, algorithms)	4
6	Dealing with intractability: approximation algorithms (techniques for design and analysis)	4
7	Dealing with large data sets (compression, streaming algorithms, compressed sensing)	5
8	Computational geometry	5
9	Spectral algorithms	4
	Total	42

References

1. Ahuja, R. K., T. L. Magnanti, and J. B. Orlin. Network Flows: Theory, Algorithms, and Applications. Upper Saddle River, NJ: Prentice Hall, 1993. ISBN: 9780136175490.

2. Cormen, T.H., C.E. Leiserson, R.L. Rivest, and C. Stein. Introduction to Algorithms. 2nd ed. Cambridge, MA: MIT Press, 2001. ISBN: 9780262032933
3. Chvatal, V. Linear Programming. New York, NY: W.H. Freeman and Company, 1983, appendix. ISBN: 9780716715870. [An easy to read description without all the details.]
4. Korte, B. H., and J. Vygen. Combinatorial Optimization. New York, NY: Springer-Verlag, 2002, chapter 4. ISBN: 9783540431541. [A detailed description.]
5. Boyd, Stephen, and Lieven Vandenberghe. Convex Optimization . Cambridge, UK: Cambridge Univ. Press, 2005. ISBN: 9780521833783
6. Nemirovski, Arkadi. "Lectures on Modern Convex Optimization." (PDF - 2.7 MB)
7. Approximation algorithms. Vazirani, V. Approximation Algorithms. New York, NY: Springer-Verlag, 2004. ISBN: 9783540653677.
8. de Berg, Mark, O. Cheong, M. van Kreveld, and M. Overmars. Computational Geometry. 3rd ed. New York, NY: Springer-Verlag, 2008. ISBN: 9783540779735

9. S. Muthukrishnan, "Data streams: Algorithms and applications", Foundations and Trends in Theoretical Computer Science, Volume 1, issue 2, 2005.
10. Bach, E., and J. Shallit. Algorithmic Number Theory. Vol. 1. Cambridge, MA: MIT Press, August 26, 1996. ISBN: 9780262024051.

Evaluation Scheme:

S.No	Examination	Marks
1	T-1	15
2	T-2	25
3	T-3	35
4	*Internal Marks	25

*Internal Marks Breakdown:

Assignments 9 marks (3x3)

Quizzes 12 marks (3x4)

Regularity 4 Marks