Design and Analysis of Algorithms @ MTech-I (1^{st} semester)

Syllabus, Evaluation and Teaching Plan

August 15, 2022

Devesh C Jinwala,

Professor in CSE, SVNIT, Surat and Adjunct Professor, IITJammu & Dean (R&C), SVNIT

Department of Computer Science and Engineering, SVNIT, Surat

Topics of Study

- Introduction: Key focus on Analysis Techniques: Mathematical, Empirical and Asymptotic analysis.
  - Review of Abstract Machines - Analysis Techniques: Mathematical, Empirical and Asymptotic analysis. Justifying the use of Asymptotic Analysis. Review of the notations in asymptotic analysis. Proof Techniques - Illustrations [4 Hours]
- Divide and Conquer approach
  - Sorting & order statistics: Divide and Conquer technique - The Comparison based Sorts – Bubble, Insertion, Merge, Heap, Quicksort - Analysis of the Worst-case, Average-case and the Best-cases. Recurrence Relations and Solving Recurrences. Applications of Divide and Conquer approach: Counting Inversions - Finding the closest pair of points. Lower Bound on Sorting - Non-comparison based sorts. Medians and Order Statistics. The Union-Find problem. [6 Hours]

Topics of Study...

- Greedy Design Approach
  - Motivation – The Thirsty Baby Problem - Formalization. Basic Greedy Control Abstraction. Case Studies : Activity Selection Problem & variants. Huffman Coding – Horn Formulas. The Knapsack Problem. Graph Algorithms. Minimum Spanning Trees: Kruskal and Prim's algorithms and Applications. The k-Clustering problem. Minimum-Cost Arborescences. Multi-phase Greedy algorithms. Single Source Shortest Paths. Maximum Bipartite Cover Problem. Applications. [8 hours]
- The Dynamic Programming
  - Motivation - The Coin Changing problem, The 0/1 Knapscak problem. Shortest Paths, Constructing Optimal Binary Search Trees. Memoization and the Dynamic Programming Control Abstraction. Distance Vector Protocols. The Longest Common Subsequence. Dynamic Programming over Intervals - Floyd-Warshall's algorithm for All-pairs Shortest Path Problem. [10 hours]

Topics of Study...

- Backtracking & Branch-and-Bound
  - Basic concept - Branch & Bound. Illustrative problems [2 hours]
- NP - Theory
  - Classifying Problems: Motivation - Polynomial time, exponential time algorithms. The reductions. Classes of problems, The Concept of Nondeterminism. The polynomial time verification - NP-completeness & the Search Problems. Dealing with NP-completeness. Local Search Heuristics. The Approximation Algorithms. [8 hours]
- Probabilistic Algorithms
  - Motivation - Probabilistic Analysis - Indicator Random Variables - Four main design categories - Randomization of deterministic algorithms - Monte Carlo Algorithms - Las Vegas Algorithms - Numerical Probabilistic Algorithms [2 hours]

Topics of Study...: time permitting

- Network Flow
  - The Maximum-Flow problem - Ford Fulkerson algorithm - Maximum Flows and Minimum Cuts in a Network - Applications: Bipartite Cover problem - Airline Scheduling [2 hours]
- Miscellaneous Topics
  - Algorithms for String Matching. Amortized Analysis. Bloom Filters & their applications.

Texts and References

- Cormen, Leiserson , Rivest , Stein: *Introduction to Algorithms*, 3/E, the MIT Press, 2001.
- Knuth, Donald E. : *The Art of Computer Programming, Vol I &III* , 3/E, Pearson Education, 1997.
- Sara Baase , Allen van Gelder : *Computer Algorithms* , 3/E, Pearson Education, 1999
- Ellis Horowitz, Sartaj Sahni: *Data Structures, Algorithms and Applications in C++*, 2/E, Universities Press/Orient Longman, 2005.
- Ellis Horowitz, Sartaj Sahni, Rajsekaran: *Fudamentals of Computer Algorithms*, Computer Science PRess,W. H. Freeman and Company New York. 2005.
- J. Kleinberg, E. Tardos: *Algorithm Design* 1/E, Pearson Education, 2005
- Aho, Hopcroft, Ullman, *Design & Analysis of Algorithms* , Pearson, 2001
- Research Papers prescribed in class

Texts and References: Book Information

- Cormen, Leiserson, and Rivest [CLR90] - This is the one (other) book on algorithms you must own, with its comprehensive treatment of most of the problems we discuss here, including data structures, graph algorithms, and seminumerical algorithms.

Texts and References: Book Information

- Cormen, Leiserson, and Rivest [CLR90] - This is the one (other) book on algorithms you must own, with its comprehensive treatment of most of the problems we discuss here, including data structures, graph algorithms, and seminumerical algorithms.
- Baase [Baa88] - This book is more accessible than [CLR90] for those without a strong mathematical background. It covers standard sorting, string, and graph algorithms, NP-completeness, and more exotically, an introduction to parallel algorithms.

Texts and References: Book Information

- Cormen, Leiserson, and Rivest [CLR90] - This is the one (other) book on algorithms you must own, with its comprehensive treatment of most of the problems we discuss here, including data structures, graph algorithms, and seminumerical algorithms.
- Baase [Baa88] - This book is more accessible than [CLR90] for those without a strong mathematical background. It covers standard sorting, string, and graph algorithms, NP-completeness, and more exotically, an introduction to parallel algorithms.
- Manber [Man89] - Built around the unconventional notion that induction is the fundamental paradigm of algorithm design, this book is especially good at teaching techniques for designing algorithms and has an outstanding collection of problems. Highly recommended.

Texts and References: Book Information

- Cormen, Leiserson, and Rivest [CLR90] - This is the one (other) book on algorithms you must own, with its comprehensive treatment of most of the problems we discuss here, including data structures, graph algorithms, and seminumerical algorithms.
- Baase [Baa88] - This book is more accessible than [CLR90] for those without a strong mathematical background. It covers standard sorting, string, and graph algorithms, NP-completeness, and more exotically, an introduction to parallel algorithms.
- Manber [Man89] - Built around the unconventional notion that induction is the fundamental paradigm of algorithm design, this book is especially good at teaching techniques for designing algorithms and has an outstanding collection of problems. Highly recommended.
- van Leeuwen [vL90b] - Not a textbook, but a collection of in-depth surveys on the state of the art in algorithms and computational complexity. Although the emphasis is on theoretical results, this book is perhaps the best single reference to point you to what is known about any given problem.

Texts and References: Book Information...

- Syslo, Deo, and Kowalik [SDK83] - This book includes printed Pascal implementations of 28 algorithms for discrete optimization problems, including mathematical programming, network optimization, and traditional operations research problems such as knapsack and TSP. Each algorithm is described in the book, and experimental timings (on a 1980s vintage machine) are provided. These codes are now available by ftp, as discussed in Section gif. Despite its age, this remains a useful reference, particularly with the programs now available on-line.

Texts and References: Book Information...

- Syslo, Deo, and Kowalik [SDK83] - This book includes printed Pascal implementations of 28 algorithms for discrete optimization problems, including mathematical programming, network optimization, and traditional operations research problems such as knapsack and TSP. Each algorithm is described in the book, and experimental timings (on a 1980s vintage machine) are provided. These codes are now available by ftp, as discussed in Section gif. Despite its age, this remains a useful reference, particularly with the programs now available on-line.
- Moret and Shapiro [MS91] - This algorithms text distinguishes itself by including Pascal implementations of all algorithms and by its careful experimental comparisons of different algorithms for such problems as sorting and minimum spanning tree. It provides a useful model for how to properly do empirical algorithm analysis. These programs are available by ftp - see Section gif for details.

Texts and References: Book Information...

- Syslo, Deo, and Kowalik [SDK83] - This book includes printed Pascal implementations of 28 algorithms for discrete optimization problems, including mathematical programming, network optimization, and traditional operations research problems such as knapsack and TSP. Each algorithm is described in the book, and experimental timings (on a 1980s vintage machine) are provided. These codes are now available by ftp, as discussed in Section gif. Despite its age, this remains a useful reference, particularly with the programs now available on-line.
- Moret and Shapiro [MS91] - This algorithms text distinguishes itself by including Pascal implementations of all algorithms and by its careful experimental comparisons of different algorithms for such problems as sorting and minimum spanning tree. It provides a useful model for how to properly do empirical algorithm analysis. These programs are available by ftp - see Section gif for details.
- Knuth [Knu94] - This book presents the implementation of the Stanford GraphBase, a collection of programs for constructing different graphs and working with them. See Section gif for details. It is very intriguing to browse.

Texts and References: Book Information...

- Aho, Hopcroft, and Ullman [AHU74] - This was the first modern algorithms book, and it has had an enormous influence on how algorithms should be taught. Although it is now dated, it remains a useful guide to certain topics in vogue in the early 1970s, such as matrix multiplication, the fast Fourier transform, and arithmetic algorithms. A more elementary edition, focusing on data structures, is [AHU83].

Texts and References: Book Information...

- Aho, Hopcroft, and Ullman [AHU74] - This was the first modern algorithms book, and it has had an enormous influence on how algorithms should be taught. Although it is now dated, it remains a useful guide to certain topics in vogue in the early 1970s, such as matrix multiplication, the fast Fourier transform, and arithmetic algorithms. A more elementary edition, focusing on data structures, is [AHU83].
- Rawlins [Raw92] - This may well be the best self-study book available on algorithms. It is fun and inspiring, with built-in pauses so the reader can make sure they understand what is going on. The only drawback is a somewhat idiosyncratic set of topics, so you will miss certain important topics. But you can get that from here. Rawlins's book can teach you the proper mindset to think about algorithms.

Texts and References: Book Information...

- Aho, Hopcroft, and Ullman [AHU74] - This was the first modern algorithms book, and it has had an enormous influence on how algorithms should be taught. Although it is now dated, it remains a useful guide to certain topics in vogue in the early 1970s, such as matrix multiplication, the fast Fourier transform, and arithmetic algorithms. A more elementary edition, focusing on data structures, is [AHU83].
- Rawlins [Raw92] - This may well be the best self-study book available on algorithms. It is fun and inspiring, with built-in pauses so the reader can make sure they understand what is going on. The only drawback is a somewhat idiosyncratic set of topics, so you will miss certain important topics. But you can get that from here. Rawlins's book can teach you the proper mindset to think about algorithms.
- Papadimitriou and Steiglitz [PS82] - This book has more of an operations research emphasis than most algorithms texts, with a good coverage of mathematical programming, network flow, and combinatorial search.

Texts and References: Book Information...

- Lawler [Law76] - Particularly useful for its coverage of matroid theory, this book also provides a thorough treatment of the network flow, matching, and shortest path algorithms known by the mid-1970s.

Texts and References: Book Information...

- Lawler [Law76] - Particularly useful for its coverage of matroid theory, this book also provides a thorough treatment of the network flow, matching, and shortest path algorithms known by the mid-1970s.
- Binstock and Rex [BR95] - Although not a textbook, it includes C language implementations of an idiosyncratic variety of algorithms for programmers. Disks containing the code are available for a modest fee. The most interesting implementations are string and pattern matching algorithms, time and date routines, an arbitrary-precision calculator, and a nice section on checksums and cyclic-redundancy checks.

Evaluation: As per the institute rules and regulations

- Midsemester Exam - 30 percentage
- Quiz Tests, Total Nos 8 (each with at least 25 multiple choice questions) - 20 percentage, average
- EndSemester Exam - 50 percentage
- Lab Evaluations - Continuous evaluation of all the assignments, 30 percentage
- Final evaluation - lab exam OR viva voce at the end of the semester, 20 percentage

Attendance

- No specific requirement for this course, but.....

Attendance

- No specific requirement for this course, but.....
- the attendance policy aligns with that of the institute.