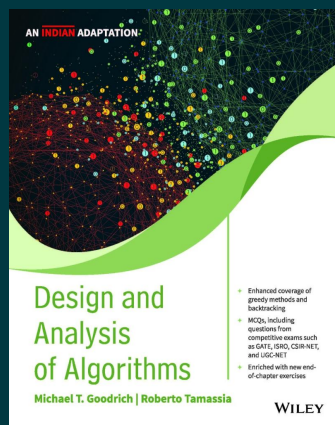




Design and Analysis of Algorithms (An Indian Adaptation)

By Michael T. Goodrich , Roberto Tamassia

Paperback

9789354248481

₹999.00

Pages: 624

Description

Design and Analysis of Algorithms offers an exhaustive coverage of the design and analysis of algorithms and data structures. Integrating application with theory, the book presents algorithmic topics in a context that is motivated from applications to uses in society, computer games, computing industry, science, engineering, and the Internet. Beginning with the basic framework and tools needed to analyze algorithms, the book then presents fundamental data structures, such as stacks, queues, lists, and advanced data structures such as trees, search trees, priority queues, heaps, hash tables, union-find structures, and graphs.

About the Author

Michael T. Goodrich , Roberto Tamassia

Michael T. Goodrich is a Chancellor's Professor in the Department of Computer Science at University of California, Irvine. Previously, he was a professor at Johns Hopkins University. His research interests include analysis, design, and implementation of algorithms, data security, cloud computing, graph drawing, and computational geometry. He is a Fulbright scholar and a fellow of the American Association for the Advancement of Science (AAAS), Association for Computing Machinery (ACM)

Table of Contents

Table of Contents

1 Algorithm Analysis

1.1 Analyzing Algorithms

1.2 A Quick Mathematical Review

1.3 A Case Study in Algorithm Analysis

1.4 A Case Study in Designing an Algorithm

1.5 Amortization

2 Basic Data Structures

2.1 Stacks and Queues

2.2 Vectors, Lists, and Sequences

2.3 Trees

3 Binary Search Trees

3.1 Searches and Updates

3.2 Range Queries

3.3 Index-Based Searching

4 Balanced Binary Search Trees

4.1 Ranks and Rotations

4.2 AVL Trees

4.3 Red-Black Trees

4.4 Weak AVL Trees

4.5 Splay Trees

5 Priority Queues and Heaps

5.1 Priority Queues

5.2 PQ-Sort, Selection-Sort, and Insertion-Sort

5.3 Heaps

5.4 Extending Priority Queues

6 Hash Tables

6.1 Maps

6.2 Hash Functions

6.3 Handling Collisions and Rehashing

6.4 Cuckoo Hashing

6.5 Universal Hashing

7 Union-Find Structures

7.1 Union-Find and Its Applications

7.2 A List-Based Implementation

7.3 A Tree-Based Implementation

8 Graphs and Traversals

8.1 Graph Terminology and Representations

8.2 Depth-First Search

8.3 Breadth-First Search

8.4 Directed Graphs

8.5 Biconnected Components

8.6 Single-Source Shortest Paths

9 Sorting Algorithms

9.1 Merge-Sort

9.2 Quick-Sort

9.3 A Lower Bound on Comparison-Based Sorting

9.4 Heap-Sort

10 Fast Sorting and Selection

10.1 Bucket-Sort and Radix-Sort

10.2 Selection

10.3 Weighted Medians

11 Divide-and-Conquer

11.1 Recurrences and the Master Theorem

11.2 Maxima and Minima Problem

11.3 Integer Multiplication

11.4 Matrix Multiplication

11.5 The Maxima-Set Problem

11.6 Order Statistics – Selection Problem

11.7 Shortest Paths in Directed Acyclic Graphs

12 The Greedy Method

12.1 The Fractional Knapsack Problem

12.2 Bin Packing

12.3 Task Scheduling

12.4 Text Compression and Huffman Coding

12.5 Coin Change Problem

12.6 Optimal Tape Storage Problem

13 Dynamic Programming

13.1 Binomial Coefficient

13.2 Matrix Chain-Products

13.3 The General Technique

13.4 Telescope Scheduling

13.5 Game Strategies

13.6 The Longest Common Subsequence Problem

13.7 The 0-1 Knapsack Problem

13.8 The Bellman-Ford Algorithm

13.9 All-Pairs Shortest Paths

13.10 Dijkstra's Algorithm

14 Minimum Spanning Trees

14.1 Properties of Minimum Spanning Trees

14.2 Kruskal's Algorithm

14.3 The Prim-Jarník Algorithm

14.4 Baruvka's Algorithm

15 NP-Completeness

15.1 P and NP

15.2 NP-Completeness

15.3 CNF-SAT and 3SAT

15.4 VERTEX-COVER, CLIQUE, and SET-COVER

15.5 SUBSET-SUM and KNAPSACK

15.6 HAMILTONIAN-CYCLE and TSP

16 Approximation Algorithms

16.1 The Metric Traveling Salesperson Problem

16.2 Approximations for Covering Problems

16.3 Polynomial-Time Approximation Schemes

16.4 Backtracking and Branch-and-Bound

17 Text Processing

17.1 Strings and Pattern Matching Algorithms

17.2 Tries

18 Number Theory, Cryptography, and Fast Fourier Transform

18.1 Fundamental Algorithms Involving Numbers

18.2 Cryptographic Computations

18.3 Information Security Algorithms and Protocols

18.4 The Fast Fourier Transform

19 Network Flow and Matching

19.1 Flows and Cuts

19.2 Maximum Flow Algorithms

19.3 Maximum Bipartite Matching

19.4 Baseball Elimination

19.5 Minimum-Cost Flow

20 Randomized Algorithms

20.1 Generating Random Permutations

20.2 Stable Marriages and Coupon Collecting

20.3 Minimum Cuts

20.4 Finding Prime Numbers

20.5 Chernoff Bounds

20.6 Skip Lists

Appendix A Backtracking

A.1 Generating Binary Strings

A.2 Fundamental Concept of Effective Backtracking Paradigm

A.3 Hamiltonian Cycle Problem

A.4 Traveling Salesman Problem

A.5 Eight Queens Puzzle

A.6 Combination as a Base Object for Backtracking

A.7 Clique Problem

Appendix B Useful Mathematical Facts

B.1 Logarithms and Exponents

B.2 Integer Functions and Relations

B.3 Summations

B.4 Basic Probability

B.5 Useful Mathematical Techniques

Bibliography

Index

Supplements

To purchase this product, please visit:

<https://www.wileyindia.com/design-and-analysis-of-algorithms-an-indian-adaptation.html>



Scan to buy