第一图书网, tushu007.com

<<算法设计>>

图书基本信息

书名：<<算法设计>>

13位ISBN编号：9787302122609

10位ISBN编号：7302122601

出版时间：2006-1

出版时间：清华大学出版社

作者：[美]克莱因伯格（KleinbergJ.） é vaTardos

页数：838

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：http://www.tushu007.com

第一图书网, tushu007.com

<<算法设计>>

内容概要

《算法设计》(影印版)是近年来关于算法设计和分析的不可多得的优秀教材。
《算法设计》(影印版)围绕算法设计技术组织素材，对每种算法技术选择了多个典型范例进行分析。
《算法设计》(影印版)将直观性与严谨性完美地结合起来。
每章从实际问题出发，经过具体、深入、细致的分析，自然且富有启发性地引出相应的算法设计思想，并对算法的正确性、复杂性进行恰当的分析、论证。
《算法设计》(影印版)覆盖的面较宽，凡属串行算法的经典论题都有涉及，并且论述深入有新意。
全书共200多道丰富而精彩的习题是《算法设计》(影印版)的重要组成部分，也是《算法设计》(影印版)的突出特色之一。

第一图书网, tushu007.com

作者简介

Jon Kleinberg is a professor of Computer Science at Cornell University. He received his Ph.D. from M.I.T. in 1996. He is the recipient of an NSF Career Award, an ONR Young Investigator Award, an IBM Outstanding Innovation Award, the National Academy of Sciences Award for Initiatives in Research, research fellowships from the Packard and Sloan Foundations, and teaching awards from the Cornell Engineering College and Computer Science Department.

Kleinberg's research is centered around algorithms, particularly those concerned with the structure of networks and information, and with applications to information science, optimization, data mining, and computational biology. His work on network analysis using hubs and authorities helped form the foundation for the current generation of Intern

第一图书网, tushu007.com

书籍目录

About the Authors
Preface
Introduction: Some Representative Problems
1.1 A First Problem: Stable Matching
1.2 Five Representative Problems
Solved Exercises
Exercises
Notes and Further Reading
Basics of Algorithm Ana/ys/s
2.1 Computational Tractability
2.2 Asymptotic Order of Growth
2.3 Implementing the Stable Matching Algorithm Using Lists and Arrays
2.4 A Survey of Common Running Times
2.5 A More Complex Data Structure: Priority Queues
Solved Exercises
Exercises
Notes and Further Reading
3 Graphs
3.1 Basic Definitions and Applications
3.2 Graph Connectivity and Graph Traversal
3.3 Implementing Graph Traversal Using Queues and Stacks
3.4 Testing Bipaniteness: An Application of Breadth-First Search
3.5 Connectivity in Directed Graphs
3.6 Directed Acyclic Graphs and Topological Ordering
Solved Exercises
Exercises
Notes and Further Reading
4 Greedy Algorithms
4.1 Interval Scheduling: The Greedy Algorithm Stays Ahead
4.2 Scheduling to Minimize Lateness: An Exchange Argument
4.3 Optimal Caching: A More Complex Exchange Argument
4.4 Shortest Paths in a Graph
4.5 The Minimum Spanning Tree Problem
4.6 Implementing Kruskal's Algorithm: The Union-Find Data Structure
4.7 Clustering
4.8 Huffman Codes and Data Compression
* 4.9 Minimum-Cost Arborescences: A Multi-Phase Greedy Algorithm
Solved Exercises
Exercises
Notes and Further Reading
5 D/v/de and Corn/net

第一图书网, tushu007.com

5.1 A First Recurrence: The Mergesort Algorithm
5.2 Further Recurrence Relations
5.3 Counting Inversions
5.4 Finding the Closest Pair of Points
5.5 Integer Multiplication
5.6 Convolutions and the Fast Fourier Transform
Solved Exercises
Exercises
Notes and Further Reading
6 Dynamic Programming
6.1 Weighted Interval Scheduling: A Recursive Procedure
6.2 Principles of Dynamic Programming: Memoization or Iteration over Subproblems
6.3 Segmented Least Squares: Multi-way Choices
6.4 Subset Sums and Knapsacks: Adding a Variable
6.5 RNA Secondary Structure: Dynamic Programming over Intervals
6.6 Sequence Alignment
6.7 Sequence Alignment in Linear Space via Divide and Conquer
6.8 Shortest Paths in a Graph
6.9 Shortest Paths and Distance Vector Protocols
* 6.10 Negative Cycles in a Graph
Solved Exercises
Exercises
Notes and Further Reading
Network Flora
7.1 The Maximum-Flow Problem and the Ford-Fulkerson Algorithm
7.2 Maximum Flows and Minimum Cuts in a Network
7.3 Choosing Good Augmenting Paths
* 7.4 The Preflow-Push Maximum-Flow Algorithm
7.5 A First Application: The Bipartite Matching Problem
7.6 Disjoint Paths in Directed and Undirected Graphs
7.7 Extensions to the Maximum-Flow Problem
7.8 Survey Design
7.9 Airline Scheduling
7.10 Image Segmentation
7.11 Project Selection
7.12 Baseball Elimination
* 7.1.3 A Further Direction: Adding Costs to the Matching Problem
Solved Exercises
Exercises
Notes and Further Reading
NP and Computational Intractability
8.1 Polynomial-Time Reductions
8.2 Reductions via "Gadgets": The Safisfiability Problem
8.3 Efficient Certification and the Definition of NP
8.4 NP-Complete Problems

第一图书网, tushu007.com

8.5 Sequencing Problems
8.6 Partitioning Problems
8.7 Graph Coloring
8.8 Numerical Problems
8.9 Co-NP and the Asymmetry of NP
8.10 A Partial Taxonomy of Hard Problems
Solved Exercises
Exercises
Notes and Further Reading
9 PSPACE: A Class of Problems beyond NP
9.1 PSPACE
9.2 Some Hard Problems in PSPACE
9.3 Solving Quantified Problems and Games in Polynomial Space
9.4 Solving the Planning Problem in Polynomial Space
9.5 Proving Problems PSPACE-Complete
Solved Exercises
Exercises
Notes and Further Reading
10 Extending the Limits of Tractability
10.1 Finding Small Vertex Covers
10.2 Solving NP-Hard Problems on Trees
10.3 Coloring a Set of Circular Arcs
* 10.4 Tree Decompositions of Graphs
* 10.5 Constructing a Tree Decomposition
Solved Exercises
Exercises
Notes and Further Reading
11 Approximation Algorithms
11.1 Greedy Algorithms and Bounds on the Optimum: A Load Balancing Problem
11.2 The Center Selection Problem
11.3 Set Cover: A General Greedy Heuristic
11.4 The Pricing Method: Vertex Cover
11.5 Maximization via the Pricing Method: The Disjoint Paths Problem
11.6 Linear Programming and Rounding: An Application to Vertex Cover
* 11.7 Load Balancing Revisited: A More Advanced LP Application
11.8 Arbitrarily Good Approximations: The Knapsack Problem
Solved Exercises
Exercises
Notes and Further Reading
Local Search
12.1 The Landscape of an Optimization Problem
12.2 The Metropolis Algorithm and Simulated Annealing
12.3 An Application of Local Search to Hopfield Neural

第一图书网, tushu007.com

Networks
12.4 Maximum-Cut Approximation via Local Search
12.5 Choosing a Neighbor Relation
12.6 Classification via Local Search
12.7 Best-Response Dynamics and Nash Equilibria
Solved Exercises
Exercises
Notes and Further Reading
Randomized Algorithms
13.1 A First Application: Contention Resolution
13.2 Finding the Global Minimum Cut
13.3 Random Variables and Their Expectations
13.4 A Randomized Approximation Algorithm for MAX 3-SAT
13.5 Randomized Divide and Conquer: Median-Finding and Quicksort
13.6 Hashing: A Randomized Implementation of Dictionaries
13.7 Finding the Closest Pair of Points: A Randomized Approach
13.8 Randomized Caching
13.9 Chernoff Bounds
13.10 Load Balancing
13.11 Packet Routing
13.12 Background: Some Basic Probability Definitions
Solved Exercises
Exercises
Notes and Further Reading
Epilogue: Algorithms That Run Forever
References
Index

第一图书网, tushu007.com

章节摘录

版权页： 插图： 2.5 A More Complex Data Structure:Priority Queues Our primary goal in this book was expressed at the outset of the chapter:we seek algorithms that improve qualitatively on brute-force search,and in general we use polynomial-time solvability as the concrete formulation of this."pically,achieving a polynomial-time solution to a nontrivial problem is not something that depends on fine-grained implementation details; rather,the difference between exponential and polynomial is based on overcoming higher-level obstacles.Once one has an efficient algorithm to solve a problem,however,it is often possible to achieve further improvements in running time by being careful with the implementation details,and sometimes by using more complex data structures. Some complex data structures are essentially tailored for use in a single kind of algorithm,while others are more generally applicable.In this section,we describe one of the most broadly useful sophisticated data structures,the priority queue.Priority queues will be useful when we describe how to implement some of the graph algorithms developed later in the book.For our purposes here,it is a useful illustration of the analysis of a data structure that,unlike lists and arrays,must perform some nontrivial processing,each time it is invoked. The Problem In the implementation of the Stable Matching algorithm in Section 2.3,we discussed the need to maintain a dynamically changing set S (such as the set of all free men in that case).In such situations,we want to be able to add elements to and delete elements from the set S,and we want to be able to select an element from S when the algorithm calls for it.A priority queue is designed for applications in which elements have a priority value,or key,and each time we need to select an element from S,we want to take the one with highest priority.

第一图书网, tushu007.com

<<算法设计>>

编辑推荐

《算法设计》(影印版)为英文原版教材，围绕算法设计技术组织素材，对每种算法技术选择了多个典型范例进行分析。

《算法设计》(影印版)适用于本科高年级学生以及研究生算法课的教材，也很适于具有计算机或相近专业本科水平的人自学算法的需要。

第一图书网, tushu007.com

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:http://www.tushu007.com