<<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

<<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。

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

<<算法设计>>

书籍目录

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.

<<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。

版权说明