国外计算机科学教材系列

算法设计技巧与分析

Algorithms Design Techniques and Analysis

英文版

[沙特] M. H. Alsuwaiyel 著

電子工業出版社
Publishing House of Electronics Industry
www.phei.com.cn

国外计算机科学教材系列

算法设计技巧与分析

（英文版）

Algorithms Design Techniques and Analysis

[沙特] M.H. Alsuwaiyel 著

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

版 权 声 明

版权贸易合同登记号：01-2002-5160

图书在版编目（CIP）数据

算法设计技巧与分析（英文版）/（沙特）阿苏外耶（Alsuwaiyel，M. H.）著.—北京：电子工业出版社，2003.1

书名原文：Algorithms Design Techniques and Analysis

ISBN 7-5053-8084-2

Ⅰ. 算… Ⅱ. 阿… Ⅲ. 算法设计—英文 Ⅳ. TP301.6

中国版本图书馆CIP数据核字（2002）第083117号

责任编辑：吴 源

印 刷：李家史山印刷厂

出版发行：电子工业出版社 http://www.phei.com.cn

北京市海淀区万寿路173信箱 邮编：100036

北京市海淀区翠微东里甲2号 邮编：100036

经 销：各地新华书店

开 本：850×1168 1/32 印张：17 字数：440 千字

版 次：2003年1月第1版 2003年1月第1次印刷

定 价：40.00元

凡购买电子工业出版社的图书，如有缺损问题，请向购买书店调换，若书店售缺，请与本社发行部联系。联系电话：（010）68279077

出 版 前 言

多年来，我一直想寻找一本适合中国计算机系学生用的算法方面的国外教材。尽管有些不错的国外教材在中国出版，但总有篇幅过多、内容略显陈旧或数据结构内容夹杂其中等等这样或那样的不甚满意之处。

去年我有幸看到世界科学图书出版社出版的由M.H.Alsuwaiyel撰写的《Algorithms Design Techniques and Analysis》，它是以国际著名算法专家，我国台湾出身的李德财教授所主编的系列丛书——Lecture Notes Series on Computing——中的一本。虽然此书不是美国的大学教材，而是沙特阿拉伯的大学计算机系教材。但是我很快就被该书的组织简明、概括，且包含当前市面上算法书较少涉及的概率算法和近似算法的基本内容所吸引。它是一本适合本科生学习算法的好书。

该书涉及数据结构的部分较少，即使有一些，表述上也很快与算法中比较复杂的集合查找和合并运算等相结合，使读者不会感到和已经学过的数据结构相重复。这比较适合中国大学计算机系中数据结构和算法分成两门开设的实际状况。

对于想了解NP完全问题基本概念的读者，本书的篇幅给了他们基本的但又清楚的描绘。本书还包括计算几何一章，其取材也是适中的。

概率算法和近似算法是近20年算法研究迅猛发展的领域，本书给予了足够的重视。这也是本书特色之一，是我向中国学生特别推荐的主要原因。

本书的另一个特色是以算法的设计技术为纲，讲述一个又一个的算法技术，然后分析其算法复杂性。

我希望本书的出版能满足短期内暂时无合适中文算法教材的空白。诚挚地推荐给中国的广大算法老师采用本书作为教材。

朱　洪

2002年10月1日

复旦大学

Preface

The field of computer algorithms has flourished since the early 1960's when the first users of electronic computers started to pay attention to the performance of programs. The limited resources of computers at that time resulted in additional impetus for devising efficient computer algorithms. After extensive research in this field, numerous efficient algorithms for different problems emerged. The similarities among different algorithms for certain classes of problems have resulted in general algorithm design techniques. This book emphasizes most of these algorithm design techniques that have proved their utility in the solution to many problems. It may be considered as an attempt to cover the most common techniques in the design of sequential algorithms. Each technique is presented as follows. First, the context in which that technique can be applied. Second, the special characteristics of that technique that set it apart. Third, comparison with other techniques, whenever possible; finally, and most importantly, illustration of the technique by applying it to several problems.

Although the main theme of the book is algorithm design techniques, it also emphasizes the other major component in algorithmic design: the analysis of algorithms. It covers in detail the analysis of most of the algorithms presented. Chapter 2 covers most of the mathematical tools that are helpful in analyzing algorithms. Chapter 11 is an introduction to the field of computational complexity, and Chapter 12 covers the basics of establishing lower bounds on the solution of various problems. These chapters are indispensable for the design of efficient algorithms.

The focus of the presentation is on practical applications of the design techniques. Each technique is illustrated by providing an adequate num-

ber of algorithms to solve some problems that quite often arise in many applications in science and engineering.

The style of presentation of algorithms is straightforward, and uses pseudocode that is similar to the syntax of structured programming languages, e.g. **if-then-else**, **for** and **while** constructs. The pseudocode is sometimes intermixed with English whenever necessary. Describing a portion of an algorithm in English is indeed instructive; it conveys the idea with minimum effort on the part of the reader. However, sometimes it is both easier and more formal to use a pseudocode statement. For example, the function of the assignment statement

$$B[1..n] \leftarrow A[1..n]$$

is to replace each entry $B[i]$ with $A[i]$ for all $i, 1 \leq i \leq n$. Neither the **for** ... **end for** construct nor plain English is more concise or easier to state than this notation.

The book is divided into seven parts. Each part consists of chapters that cover those design techniques that have common characteristics or objectives. Part 1 sets the stage for the rest of the book, in addition to providing the background material that is needed in subsequent chapters. Part 2 is devoted to the study of recursive design techniques, which are extremely important, as they emphasize a fundamental tool in the field of computer science: recursion. Part 3 covers two intuitive and natural design techniques: the greedy approach and graph traversals. Part 4 is concerned with those techniques needed to investigate a given problem and the possibility of either coming up with an efficient algorithm for that problem, or proving its intractability. This part covers NP-completeness, computational complexity and lower bounds. In Part 5, techniques for coping with hard problems are presented. These include backtracking, randomization and finding approximate solutions that are reasonable and acceptable using a reasonable amount of time. Part 6 introduces the concept of iterative improvement using two important problems that have received extensive attention, which resulted in increasingly efficient algorithms: the problem of finding a maximum flow in a network and the problem of finding a maximum matching in an undirected graph. Finally, Part 7 is an introduction to the relatively new field of computational geometry. In one chapter, the widely used technique of geometric sweeping is presented with examples of important problems in that field. In the other chapter, the versatile tool of

the Voronoi diagram is covered, and some of its applications are presented.

The book is intended as a text in the field of the design and analysis of algorithms. It includes adequate material for two courses in algorithms. Chapters 1 through 10 provide the core material for an undergraduate course in algorithms at the junior or senior level. Some of the material may be skipped such as the amortized analysis of the union-find algorithms, and the linear time algorithms in the case of dense graphs for the shortest path and minimum spanning tree problems. The instructor may find it useful to add some of the material in the following chapters such as backtracking, randomized algorithms, approximation algorithms or geometric sweeping. The rest of the material is intended for a graduate course in algorithms.

The prerequisites for this book have been kept to the minimum; only an elementary background in discrete mathematics and data structures are assumed.

The author is grateful to King Fahd University of Petroleum & Minerals (KFUPM) for their support and providing facilities for the preparation of the manuscript. This book writing project has been funded by KFUPM under Project ics/algorithm/182. The Author would like to thank those who have critically read various portions of the manuscript and offered many helpful suggestions, including the students of the undergraduate and graduate Algorithms courses at KFUPM. Special thanks go to S. Albassam, H. Almuallim, and S. Ghanta for their valuable comments.

Dhahran, Saudi Arabia M. H. Alsuwaiyel

Contents