

<<算法：C语言实现>>

图书基本信息

书名：<<算法：C语言实现>>

13位ISBN编号：9787111275718

10位ISBN编号：7111275713

出版时间：2009-10

出版时间：机械工业出版社

作者：塞奇威克

页数：456

译者：霍红卫

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>

<<算法：C语言实现>>

前言

写本书的目的是为了对当今使用最为重要的计算机算法做一综述，并为需要学习这方面知识的越来越多的读者提供基础的技术。

本书可以在学生掌握了所需的基本程序设计技巧，熟悉了计算机系统，但还未学过计算机科学或计算机应用高级领域的专业课程的时候，用作计算机科学的第二。

第三或第四门课程的教科书。

此外，由于本书包含了大量有用算法的实现，以及关于这些算法的性能特征的详细信息，因而它还可用于自学，或者作为从事计算机系统或应用程序开发人员的参考手册。

宽广的视角使得本书成为计算机算法领域最合适的入门读物。

对于新的一版，我不仅完全重写了它的内容，而且还添加了一千多个练习。

一百多幅图表和数十个新程序。

我还给所有图表和程序添加了详细的注释。

新的素材不仅涵盖了新的主题，而且还包含对经典算法的更完整解释。

抽象数据类型是这本书的重点，这使得程序应用更广泛，并且与现代面向对象的程序设计环境更紧密。

读过本书旧版本的人一定会发现，新版本包含了更为丰富的新信息，所有读者将发现大量的教学资料为掌握基本概念提供了有效途径。

由于新的素材数量过多，所以我们把新版本分为两卷（每一卷的容量都大约为旧版本的大小），本书是第一卷。

这卷书中包含了基本概念。

数据结构。

排序算法和搜索算法，第二卷涵盖的高级算法及应用是以第一卷的基本抽象概念和方法为基础的。

这个新版中的关于基本原理和数据结构的所有素材几乎都是新的。

这本书不仅适合于程序员和计算机科学专业的学生，而且也适合于想利用计算机并想使它运行更快或是想要解决更大问题的人们。

这本书中的算法代表了过去50年来所研究的知识主体。

对于大量应用问题，这些知识主体已经成为有效使用计算机的不可缺少的部分。

从物理学中的N-体模拟问题到分子生物学中的序列分析问题，在此所描述的基本方法在科学研究中已日显重要。

另外，对于从数据库系统到Internet搜索引擎，这些方法已经成为现代软件系统的重要组成部分。

随着计算机应用的覆盖面越来越广，基本算法的影响也日益显著。

本书的目标是要提供一种资源，使广大学生以及专业人士可以了解并有效利用这些算法解决计算机应用中出现的问题。

<<算法：C语言实现>>

内容概要

本书细腻讲解计算机算法的C语言实现。

全书分为四部分，共16章。

包括基本算法分析原理，基本数据结构、抽象数据结构、递归和树等数据结构知识，选择排序、插入排序、冒泡排序、希尔排序、快速排序方法、归并和归并排序方法、优先队列与堆排序方法、基数排序方法以及特殊用途的排序方法，并比较了各种排序方法的性能特征，在进一步讲解符号表、树等抽象数据类型的基础上，重点讨论散列方法、基数搜索以及外部搜索方法。

书中提供了用C语言描述的完整算法源程序，并且配有丰富的插图和练习，还包含大量简洁的实现将理论和实践成功地相结合，这些实现均可用在真实应用上。

本书内容丰富，具有很强的实用价值，适合作为高等院校计算机及相关专业本科生算法课程的教材，也是广大研究人员的极佳参考读物。

<<算法：C语言实现>>

作者简介

塞奇威克 (Robert Sedgewick)，拥有斯坦福大学博士学位（导师为donald E.Knuth），普林斯顿大学计算机科学系教授，Adobe Systems公司董事，曾是Xerox PARC的研究人员，还曾就职于美国国防防御分析研究所以及INRIA。
除本书外，他还与Philippe Flajolet合著了《算法分析导论》

<<算法：C语言实现>>

书籍目录

出版者的话

译者序

前言

第一部分 基础知识

第1章 引言

1.1 算法

1.2 典型问题——连通性

1.3 合并—查找算法

1.4 展望

1.5 主题概述

第2章 算法分析的原理

2.1 实现和经验分析

2.2 算法分析

2.3 函数的增长

2.4 大O符号

2.5 基本递归方程

2.6 算法分析示例

2.7 保证、预测及局限性

第二部分 数据结构

第3章 基本数据结构

3.1 构建组件

3.2 数组

3.3 链表

3.4 链表的基本处理操作

3.5 链表的内存分配

3.6 字符串

3.7 复合数据结构

第4章 抽象数据类型

4.1 抽象对象和对象集

4.2 下推栈ADT

4.3 栈ADT客户示例

4.4 栈ADT的实现

4.5 创建一个新ADT

4.6 FIFO队列和广义队列

4.7 复制和索引项

4.8 一级ADT

4.9 基于应用的ADT示例

4.10 展望

第5章 递归与树

5.1 递归算法

5.2 分治法

5.3 动态规划

5.4 树

5.5 树的数学性质

5.6 树的遍历

<<算法：C语言实现>>

5.7 递归二叉树算法

5.8 图的遍历

5.9 综述

第三部分 排序

第6章 基本排序方法

6.1 游戏规则

6.2 选择排序

6.3 插入排序

6.4 冒泡排序

6.5 基本排序方法的性能特征

6.6 希尔排序

6.7 对其他类型的数据进行排序

6.8 索引和指针排序

6.9 链表排序

6.10 关键字索引统计

第7章 快速排序

7.1 基本算法

7.2 快速排序算法的性能特征

7.3 栈大小

7.4 小的子文件

7.5 三者取中划分

7.6 重复关键字

7.7 字符串和向量

.....

第8章 归并与归并排序

第9章 优先队列和堆排序

第10章 基数排序

第11章 特殊用途的排序方法

第四部分 搜索

第12章 符号表和二叉搜索树

第13章 平衡树

第14章 散列

第15章 基数搜索

第16章 外部搜索

<<算法：C语言实现>>

章节摘录

插图：还有一个例子出现在某种程序设计环境中，连通性可用来断言两个变量名是否等价。问题是在经过这样的断言序列之后，能够确定两个给定的名字是否等价。

这个应用激发了我们打算考虑的几个算法的研制。

它直接将我们的问题与一种简单抽象关联起来，为使算法具有广泛应用而提供了一种方法。我们即将看到这一点。

像上一段描述的变量名等价问题这样的应用程序要求我们把每个不同的变量名与一个整数关联起来。这种关联关系也隐含在前面描述的网络连接和电路连接的应用中。

在第10章至第16章，我们将会以一种更高效的方法考虑提供这种连接关系的大量算法。

因此，不失一般性，本章假设有 N 个对象，每个都与 $0-N-1$ 之间的一个整数名对应。

我们正在寻求完成特定和良定义任务的程序，可能还想要解决其他许多相关的问题。

在研制算法时我们面对的首要任务之一是确信我们已经以合理的方式指定了问题。

我们要求算法的越多，它完成任务所需要的时间和空间越多。

不可能量化这个关系，并且我们在发现一个问题难以求解或是求解代价昂贵，或是在好的情况下，发现算法可以比原始说明提供更多有用的信息时，我们常常修改这个问题的说明。

例如，我们的连通问题的说明只要求我们的程序知道任意给定对 $p-q$ 是否是连通的，并不能够表明连接那个对的任何方式。

添加这样一个说明的要求会使问题更加困难，会涉及其他的算法，我们将在第5章简略讨论，并在第7章详细讨论。

前面这段提到的说明要比原始说明要求更多的信息，我们也可以要求更少的信息。

例如，我们可能只想回答这样的问题：“ M 个连接足以把 N 个对象都连接起来吗？”

这个问题表明，要研制一个高效的算法，常常需要我们对正在处理的抽象对象进行高级推理。

在这种情况下，由图论基本结果可以得出所有 N 个对象是连通的，当且仅当连通算法输出的对的个数恰好为 $N-1$ （见5.4节）。

换句话说，连通算法永远不会输出多于 $N-1$ 个对，这是因为一旦它输出 $N-1$ 个对，则它从那个时刻遇见的任何对将会是连通的。

因此，我们可以修改求解连通问题的程序，增加一个计数器就可以得到一个回答yes-no问题的程序，而不输出那些前面不连通的每个对，当计数器的值为 $N-1$ 时，程序回答“yes”，否则回答“no”。

这个问题只是我们希望回答关于连通性的许多问题中的一个例子。

输入对的集合称为图（graph），输出对的集合称为图的生成树，它连接了所有对象。

我们在第七部分考察图、生成树以及所有相关算法的性质。

<<算法：C语言实现>>

媒体关注与评论

对于在数学分析方面不算熟练且需要留意理论算法的普通程序员来说，本书是一本可读性很强的优秀读本。

他们应该会从中获益良多。

——Steve Summit，《C Programming FAQs》的作者Sedgewick有一种真正的天赋，可以用易于理解的方式来解释概念。

书中采用了一些易懂的实战程序，其篇幅仅有一页左右，这更是锦上添花。

而书中大量采用的图、程序、表格也会极大帮助读者的学习和理解，这使本书更显得与众不同。

——William A. Ward，南亚拉巴马大学

<<算法：C语言实现>>

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>