

<<数据结构与算法>>

图书基本信息

书名：<<数据结构与算法>>

13位ISBN编号：9787302307501

10位ISBN编号：7302307504

出版时间：2013-2

出版时间：熊岳山、祝恩 清华大学出版社 (2013-02出版)

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>

<<数据结构与算法>>

内容概要

<<数据结构与算法>>

作者简介

熊岳山，男，1963年4月出生，国防科技大学计算机学院教授，全军优秀教师。

计算机应用技术、计算数学博士生导师。

主要研究兴趣为虚拟手术、图形图像处理和数值计算方法。

主持多项国家自然科学基金项目、“863”高技术项目、国防实验技术研究项目的研究。

曾主讲“数据结构”、“数值分析及其应用”、“计算方法”、“偏微分方程数值解”、“矩阵迭代分析”、“计算几何”等课程，多次获教学优秀奖。

编著的《数据结构》教材获湖南省高等院校优秀教材，先后获国防科技大学教学优秀二等奖2次，校教学成果一、二等奖各1次；所从事的研究工作受到国内外同行的关注，1995年、1997年、2002年曾三次应邀到香港理工大学从事访问研究，先后在国内外有影响的刊物上发表研究论文80余篇，其中16篇进入SCI检索、20余篇进入EI检索。

祝恩，1976年出生于湖南益阳，2005年6月毕业于国防科技大学计算科学与技术专业，获博士学位，现为国防科技大学副教授。

2006年起从事数据结构与算法、计算机图形学、人工智能、模式识别等课程的教学工作。

主要研究领域为模式识别、图像分析、生物认证等。

在国际国内刊物和会议上发表学术论文90余篇。

博士学位论文评为全国百篇优秀博士学位论文。

是中国计算机学会理论计算机科学专业委员会委员、模式识别与人工智能专业委员会委员。

<<数据结构与算法>>

书籍目录

第1章数据结构概述 1.1基本概念 1.1.1数据、数据元素、数据对象 1.1.2数据结构 1.2数据结构的分类 1.3数据类型 1.3.1基本类型、组合类型 1.3.2抽象数据类型 1.4算法和算法分析 1.4.1算法概念 1.4.2算法分析习题 第2章 向量、栈和队列 2.1线性表 2.1.1线性表的抽象数据类型 2.1.2线性表的结构表示 2.2向量 2.2.1向量的抽象数据类型 2.2.2 向量的插入和删除 2.2.3 向量的应用 2.3栈 2.3.1 栈的抽象数据类型及其实现 2.3.2栈的应用 2.4递归效率分析 2.4.1递归方程求解 2.4.2生成函数求解递归方程 2.4.3特征方程求解递归方程 2.4.4递归树方法 2.5 队列 2.5.1 队列的抽象数据类型及其实现 2.5.2 队列的应用——模拟银行活动习题 第3章链表 3.1单链表 3.1.1基本概念 3.1.2单链表结点结构 3.1.3单链表结构 3.1.4栈的单链表实现 3.1.5队列的单链表实现 3.1.6单链表的应用举例 3.2循环链表 3.3双链表 习题 第4章 串 4.1基本概念 4.2串的存储 4.3串结构和串的运算 4.4模式匹配 4.4.1朴素的模式匹配算法 4.4.2 KMP匹配算法 4.4.3 BM匹配算法 习题 第5章排序 5.1基本概念 5.2插入排序 5.2.1直接插入排序 5.2.2折半插入排序 5.2.3 Shell排序 5.3选择排序 5.3.1直接选择排序 5.3.2树形选择排序 5.4交换排序 5.4.1起泡排序 5.4.2快速排序 5.5分配排序 5.5.1基本思想 5.5.2基数排序 5.6归并排序 5.7外部排序 5.7.1二路合并排序 5.7.2 多路替代选择合并排序 5.7.3最佳合并排序 习题 第6章查找 6.1基本概念 6.2顺序查找 6.3折半查找 6.4分块查找 6.5散列查找 6.5.1概述 6.5.2散列函数 6.5.3 冲突的处理 6.5.4散列查找的效率 习题 第7章树和二叉树 7.1树的概念 7.2二叉树 7.2.1二叉树的概念 7.2.2二叉树的性质 7.2.3二叉树的存储方式 7.2.4树（树林）与二叉树的相互转换 7.3树（树林）、二叉树的遍历 7.3.1树（树林）的遍历 7.3.2二叉树的遍历 7.4抽象数据类型BinaryTree以及BinaryTree结构 7.4.1抽象数据类型BinaryTree 7.4.2一个完整的包含构建二叉树与遍历实现的例子 7.5二叉树的遍历算法 7.5.1 非递归（使用栈）的遍历算法 7.5.2线索化二叉树的遍历 习题 第8章树状结构的应用 8.1二叉排序树 8.1.1二叉排序树与BinarySTree结构 8.1.2二叉排序树的检索、插入、删除运算 8.1.3等概率查找对应的最佳二叉排序树 8.2平衡的二叉排序树 8.2.1平衡的二叉排序树 8.2.2平衡的二叉排序树的插入、删除 8.2.3 AVL树高度 8.3 B—树、B+—树 8.4键树和2—3树 8.4.1键树 第9章图 第10章算法设计与分析 图目录 算法目录 关键字索引 参考文献

<<数据结构与算法>>

章节摘录

版权页：插图：第6章 查找 查找是数据处理领域最常用的一种重要运算，也称为检索。

查找的对象可以是线性表，也可以是复杂的树形结构和文件结构。

本章主要讨论基于线性表的查找。

6.1基本概念 所谓查找就是在给定的数据结构中搜索满足某种条件的结点。

最常见的查找是给出一个值，在数据结构中找出关键码等于指定值的结点。

例如，在学生成绩表中，查找指定学号的学生成绩，学号是学生成绩表的关键码，因为每个学生都有唯一的学号。

查找的结果有两种情况，第一种情况是学生成绩表中有相应学号的学生成绩，自然可以查找到该学生的成绩，称为“查找成功”；第二种情况是学生成绩表中没有相应学号的学生成绩，也就不可能查找对应的成绩，称为“查找失败”。

除了基于关键字的查找以外，还可能按其他属性值进行查找。

例如，可能需要查找学生成绩表中英语成绩为95分的学生。

显然，查找的结果有多种，可能没有任何一个学生的成绩为95分，可能有一个学生的成绩为98分，也可能有多个学生的成绩为95分。

对于有多个满足条件的结点，有些查找只要求给出一个结点即可，例如为了确定所有学生中是否有英语成绩为98分的学生；有些查找要求给出所有满足条件的结点，例如要找出所有学生中有哪些学生的英语成绩为98分。

一般来说，基于关键码的查找和基于属性值的查找没有太多的区别。

对不同的存储结构，可以采用的查找方法也不尽相同。

同时，为了提高查找的速度，也常常针对不同场合采用不同的存储结构。

衡量一个查找算法的好坏的依据主要是查找过程中需要执行的平均比较次数，或者称为平均查找长度，通常用 $E(n)$ 来表示，其中 n 为线性表中的结点个数。

此外，还要考虑算法所需要的附加存储空间，以及算法本身的复杂性等。

为方便起见，在本章以后的讨论中，均假设结点是等长的，查找都是基于关键码的查找，且关键码都是整数。

这些假设是合理的，因为，如果结点是不等长，则可以讨论结点的目录表；如果关键码不是整数，则可以在关键码和整数之间建立一一对应的关系。

6.2顺序查找 顺序查找是一种最简单也是效率比较低下的查找算法。

顺序查找时，将每个结点的关键码和给定的待查的关键码值进行比较，直到找出相等的结点或者找遍了所有结点。

执行顺序查找算法时，被查找的线性表可以是顺序存储的，也可以是链接存储的，对结点没有排序要求，因而顺序查找具有非常好的适应性。

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>