# <<数据结构>>

#### 图书基本信息

书名:<<数据结构>>

13位ISBN编号: 9787560623764

10位ISBN编号:756062376X

出版时间:2010-2

出版时间:西安电子科技大学出版社

作者:刘肖编

页数:210

字数:319000

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>>

#### 前言

- "数据结构"是计算机程序设计的重要理论基础,是计算机相关专业必修的一门重要基础课程和核心课程。
- "数据结构"主要介绍各种类型的数据结构及在其上进行的各种运算,包含了较多的理论知识,也有较高的实践要求。

为了达到理论与实践的结合统一,本书秉承以应用为主体,注重培养实践能力的指导思想,在强调理 论知识的理解和运用的同时,更加注重结合高职高专教学的要求,体现对学生应用能力的培养。

本书主要面向高职高专院校计算机相关专业的学生。

全书共分8章。

第1章为概述,主要阐述数据结构的相关概念、算法的描述和算法分析方法;第2章为线性表,主要讨论线性结构的顺序和链式表示方法、存储方法和各种操作方法以及应用:第3章为栈和队列,主要讨论栈和队列这两种限定性的线性表的表示、存储方法和操作方法以及应用:第4章为串与数组,主要讨论串与数组的存储方法和各种应用;第5章为树与二叉树,主要讨论树与二叉树的定义、存储方法、运算方法及应用;第6章为图,主要讨论图的各种概念、存储方法、遍历方法以及图的多种实际应用:第7章为查找,主要讨论查找的概念、各种查找方法及应用;第8章为排序,主要讨论排序的概念、各种排序方法及应用。

## <<数据结构>>

#### 内容概要

本书从实际应用的角度出发,介绍了数据结构的基本知识和各种数据结构的实际应用。 全书共分8章,主要内容包括线性表、栈与队列、串与数组、树、图、查找及排序等。 各部分内容均从实际应用问题引入基本知识的讲解和描述,使读者更容易理解所学知识的应用目标, 并在讲解中使用大量的实例来说明基本知识的应用。 除第1章外,每章还包括了两个实训项目,配置了多种类型的习题,以突出实际应用能力的培养。

本书可作为高职高专学校计算机类专业学生学习"数据结构"的教材,也可作为软件技术人员的参考用书。

为方便读者学习,本书的算法部分均采用c语言描述,实训项目也是完整的 c语言程序,读者可以很方便地对书中的算法进行上机测试。

## <<数据结构>>

#### 书籍目录

#### 第1章 概述

- 1.1 引言
- 1.2 基本术语及概念
- 1.2.1 基本术语
- 1.2.2 数据结构
- 1.3 算法描述与算法分析
- 1.3.1 算法与算法描述
- 1.3.2 算法分析

小结

习题

#### 第2章 线性表

- 2.1 线性表的逻辑结构及基本运算
- 2.1.1 线性表的逻辑结构
- 2.1.2 线性表的基本运算
- 2.2 线性表的顺序存储及运算
- 2.2.1 线性表的顺序存储——顺序表
- 2.2.2 顺序表的基本运算
- 2.3 线性表的链式存储及运算
- 2.3.1 单链表
- 2.3.2 循环链表
- 2.3.3 双向链表
- 2.3.4 静态链表

小结

习题

实训指导

第3章 栈与队列

- 3.1 栈
- 3.1.1 栈的定义及基本运算
- 3.1.2 栈的顺序存储及运算
- 3.1.3 栈的链式存储及运算
- 3.1.4 栈的应用
- 3.2 队列
- 3.2.1 队列的定义及基本运算
- 3.2.2 队列的顺序存储及运算
- 3.2.3 队列的链式存储及运算
- 3.2.4 队列的应用

小结

习题

实训指导

第4章 串与数组

- 4.1 串
- 4.1.1 串的基本概念
- 4.1.2 串的存储结构
- 4.1.3 串运算的实现
- 4.2 数组

# <<数据结构>>

- 4.2.1 数组的定义和运算
- 4.2.2 数组的顺序存储和实现
- 4.2.3 特殊矩阵的压缩存储
- 小结
- 习题
- 实训指导
- 第5章 树与二叉树
- 5.1 树
- 5.2 二叉树
- 5.2.1 二叉树的定义
- 5.2.2 二叉树的性质
- 5.2.3 二叉树的存储结构
- 5.2.4 遍历二叉树
- 5.2.5 应用实例
- 5.3 树和森林
- 5.3.1 树的存储结构
- 5.3.2 树、森林与二叉树的转换
- 5.3.3 树和森林的遍历
- 5.4 最优二叉树——哈夫曼树
- 5.4.1 哈夫曼树的定义和构造方法.
- 5.4.2 哈夫曼编码
- 小结
- 习题
- 实训指导
- 第6章图
- 6.1 图的基本概念
- 6.2 图的存储表示
- 6.2.1 图的邻接矩阵
- 6.2.2 邻接表
- 6.3 图的遍历
- 6.3.1 深度优先搜索
- 6.3.2 广度优先搜索
- 6.4图的应用
- 6.4.1 生成树和最小生成树
- 6.4.2 最短路径
- 6.4.3 拓扑排序
- 小结
- 习题
- 实训指导
- 第7章 查找
- 7.1 查找的基本概念
- 7.2 静态查找表
- 7.2.1 顺序查找
- 7.2.2 折半查找
- 7.2.3 分块查找
- 7.3 动态查找表
- 7.3.1 二叉排序树

# <<数据结构>>

- 7.3.2 二叉排序树的插入和生成
- 7.3.3 二又排序树的删除
- 7.3.4 二叉排序树的查找
- 7.3.5 二叉排序树的查找性能
- 7.3.6 平衡二叉树
- 7.4 哈希表查找
- 7.4.1 哈希表与哈希查找
- 7.4.2 构造哈希函数的方法
- 7.4.3 处理冲突的方法
- 7.4.4 哈希表的查找分析
- 小结
- 习题
- 实训指导
- 第8章 排序
- 8.1 基本概念
- 8.2 插入排序
- 8.2.1 直接插入排序
- 8.2.2 希尔排序
- 8.3 交换排序
- 8.3.1 冒泡排序
- 8.3.2 快速排序
- 8.4 选择排序
- 8.4.1 简单选择排序
- 8.4.2 树形选择排序
- 8.4.3 堆排序
- 8.5 二路归并排序
- 8.6 基数排序
- 8.6.1 多关键字排序
- 8.6.2 链式基数排序
- 8.7 排序方法的比较
- 小结
- 习题
- 实训指导
- 参考文献

### <<数据结构>>

#### 章节摘录

对于同一个问题可以设计出不同的算法,它们之间肯定有是否适合和"好""差"之分。 衡量一个适合的、"好"的算法,在其必须是正确的算法的基础上,还应考虑下列几个方面:

- (1)易读性:算法应易于阅读和理解,以便于调试、修改和扩充。
- (2)高效性:算法应具有较高的时间效率和空间效率,即占用较短的执行时间和较少的存储空间。

当然,这两者都和问题的规模有关。

(3)健壮性:正确的输入能得到正确的输出,这是算法必须具有的特性之一。 但当遇到非法输入时,算法应能做出反应或处理(如提示信息等),而不会产生不需要的或不正确的 结果。

例如,我们在算法中总是采用较简单的方法以及模块化函数,以增强算法的易读性;采用步骤较少的求解方法,以提高算法的时间效率;对可能发生的各种现象及输入形式给出应对措施,以提高算法的健壮性。

要确定一个算法是适合的、"好"的算法,就需进行算法分析。

算法分析的两个主要方面是分析算法的时间效率和空间效率,目的是以求改进算法或对不同的算法进行比较。

在目前情况下,鉴于运算空间较为充足,我们把算法的时间效率分析作为主要内容。

算法运行的时间分析和程序运行的时间分析是有区别的。

同一算法由不同的程序员来编写,所编写的程序会有优劣之分,程序运行的时间也就有所不同;程序 在不同的机器上运行的速度又和机器本身的速度有关。

而我们感兴趣的是对解决问题的算法作时间上的度量分析,或对解决同一问题的两种或两种以上算法的运行时间加以比较。

估算算法运行时间需要考虑的基本因素是问题的"规模"和算法执行"基本操作"的次数。 一个算法的"规模"和"基本操作"要视具体问题而定。

- "规模"一般是指输入量的数目,比如在排序问题中,问题的规模可以是待排序的元素数目。 数据量越大,算法执行花费的时间就越多,它是影响时间的最主要的因素。
- "基本操作"一般是指算法在解决某个问题时必须涉及到的主要操作,它与操作数的具体取值无关, 比如在查找运算中,两个数据的比较就可视为基本操作。

显然,在一个算法中,执行基本运算的次数越少,其运行时间也就相对地越短;执行次数越多,其运行时间也就相对地越长。

. . . . . .

# <<数据结构>>

#### 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