

<<数据结构>>

图书基本信息

书名：<<数据结构>>

13位ISBN编号：9787030104571

10位ISBN编号：7030104579

出版时间：2002-7-1

出版时间：科学出版社

作者：缪淮扣,顾训穰,沈俊

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>

## &lt;&lt;数据结构&gt;&gt;

## 前言

作为计算机程序组成部分的数据结构和算法的研究，一直受到计算机领域工作者的高度重视。数据结构是计算机专业教学计划中的一门核心课程，也是信息管理、通信电子、自动控制等与计算机技术关系密切的专业的一门基础课程。

要从事与计算机科学与技术相关的工作，尤其是计算机应用领域的开发和研制工作，必须具备坚实的数据结构的基础。

数据结构课程的教学目的是使学生学会分析研究计算机所要加工处理的数据的特征，掌握组织数据、存储数据和处理数据的基本方法，并加强在实际应用中选择合适的数据结构和相应算法的训练。

面向对象技术是软件工程领域中的重要技术，它不仅是一种程序设计方法，更重要的是一种对真实世界的抽象思维方式。

目前，面向对象的软件分析和设计技术已发展成为软件开发的主流方法。

为了适应软件开发方法与技术的发展以及应用领域的要求，就有必要改进和充实数据结构的教学内容。

因此，用面向对象的观点来描述数据结构就成为一种既顺理成章又紧迫的选择。

用面向对象的观点来描述数据结构，要涉及到面向对象程序设计语言的选用问题。

目前被广泛采用作为程序设计语言教学的是C语言，C++是以C语言为基础、使用比较普遍的面向对象的程序设计语言。

因此本书采用C++作为数据结构的描述语言。

数据结构课程内容丰富，学习量大；隐藏在各部分内容中的方法和技术多，贯穿于全书的动态链表存储结构和递归技术令不少初学者望而生畏。

本书的编写者长期以来从事数据结构课程的教学，对该课程的教学特点和难点有比较深切的体会。

作者在认真总结二十多年讲授“数据结构”课程的基础上，参考了美国ACM / IEEE-CS公布的“计算2001教程”，吸收了国内外各种数据结构教材的优点，对多年来形成的数据结构课程的教学内容进行了合理的剪裁，既强调了数据结构的原理和方法，又注重了其实践性，使之适应于现代大学生的学习特点和要求。

本书的一个重要特点就是将程序设计的基础与数据结构的方法尽可能地结合起来。

第1、2章介绍C++语言时尽可能给出比较完整的程序，使学生能对C++语言有比较全面和深入的了解，也便于上机实习，从而为数据结构课程的实验建立良好的基础。

全书共分9章，第1、2章介绍了数据结构、算法及其复杂度的基本概念，对C++作了简单介绍，并叙述了抽象数据类型和面向对象的概念。

第3~5章介绍了线性结构——线性表、栈、队列、数组、广义表；第6章和第7章介绍了非线性结构——树和图；第8章和第9章分别介绍了查找和排序的方法。

## <<数据结构>>

### 内容概要

数据结构是计算机专业教学计划中的一门核心课程，也是信息管理、通信电子、自动控制等与计算机技术关系密切的专业的一门基础课程。

要从事和计算机科学与技术相关的工作，尤其是计算机应用领域的开发和研制工作，必须具备坚实的数据结构的基础。

本书对C++语言作了简单介绍，叙述了抽象数据类型和面向对象的概念，介绍了线性表、栈、队列、数组、广义表、树和图等数据结构，并且介绍了查找和排序的方法。

全书用C++语言描述并实现了所有数据结构的类和程序，并附有习题，便于教学。

本书是为高等院校开设“数据结构”课程编写的教材，可作为计算机专业本科生教材使用，也可供从事计算机软件开发和应用的工程技术人员阅读、参考。

## &lt;&lt;数据结构&gt;&gt;

## 书籍目录

## 1 绪论

1.1 (算法+数据结构) = 程序

1.2 数据结构的基本概念

1.2.1 两个简单的数据结构实例

1.2.2 什么是数据结构

1.3 C++语言基础

1.3.1 程序结构

1.3.2 数据声明和作用域

1.3.3 输入 / 输出

1.3.4 函数

1.3.5 参数传递

1.3.6 函数各重载

1.3.7 动态内存分配

1.3.8 结构与联合

1.4 算法性能与复杂度

1.4.1 算法的定义

1.4.2 算法的性能标准

1.4.3 算法的复杂度

习题1

## 2 抽象数据类型和C++类

2.1 抽象数据类型

2.1.1 从数据类型到抽象数据类型

2.1.2 封装和信息隐藏

2.1.3 抽象数据类型描述

2.2 类与对象的基本概念

2.2.1 类与对象

2.2.2 消息与合作

2.2.3 多态性

2.3 面向对象的程序设计方法

2.4 C++类与对象

2.5 构造函数和析构函数

2.6 工具函数

2.7 继承

2.8 this指针的使用

2.9 虚函数、多态性以及动态联编

2.9.1 虚函数和多态性

2.9.2 动态联编

2.10 模板类

习题2

## 3 线性表

3.1 线性表的定义

3.2 线性表的顺序表示

3.2.1 顺序表的类定义

3.2.2 顺序表插入、删除算法的复杂度分析

3.2.3 顺序表的应用

## &lt;&lt;数据结构&gt;&gt;

## 3.3 线性表的链表表示

## 3.3.1 单链表

## 3.3.2 单循环链表

## 3.3.3 双向循环链表

## 3.3.4 静态链表

## 3.4 多项式抽象数据类型

## 3.4.1 多项式表示

## 3.4.2 多项式相加

## 习题3

## 4 栈、队列和递归

## 4.1 栈

## 4.1.1 顺序栈

## 4.1.2 链式栈

## 4.1.3 表达式的计算

## 4.2 队列

## 4.2.1 循环队列

## 4.2.2 链队列

## 4.3 递归

## 4.3.1 递归的概念

## 4.3.2 递归过程与递归工作栈

## 4.3.3 消除递归

## 4.3.4 迷宫问题

## 习题4

## 5 串、数组和广义表

## 5.1 字符串

## 5.1.1 字符串的定义、存储结构和操作

## 5.1.2 串的操作

## 5.1.3 常用的C++字符串函数

## 5.1.4 串类及其实现

## 5.1.5 模式匹配算法

## 5.2 数组

## 5.2.1 C++中数组的定义

## 5.2.2 数组的抽象数据类型表示

## 5.2.3 数组的顺序存储结构

## 5.3 稀疏矩阵

## 5.3.1 三元组顺序表

## 5.3.2 十字链表

## 5.4 广义表

## 5.4.1 广义表的定义

## 5.4.2 广义表的存储结构

5.4.3  $n$ 元多项式的表示

## 5.4.4 广义表的递归算法

## 习题5

## 6 树和森林

## 6.1 树的概念

## 6.1.1 树的定义

## 6.1.2 树的术语

## &lt;&lt;数据结构&gt;&gt;

- 6.1.3 树的表示形式
- 6.1.4 树的基本操作和抽象数据类型
- 6.2 二叉树
  - 6.2.1 二叉树的定义
  - 6.2.2 二叉树的性质
  - 6.2.3 二叉树的基本操作和抽象数据类型
- 6.3 二叉树的存储结构
  - 6.3.1 数组表示法
  - 6.3.2 链表表示法
  - 6.3.3 二叉树的二叉链表类声明
- 6.4 遍历二叉树
  - 6.4.1 前序遍历
  - 6.4.2 中序遍历
  - 6.4.3 后序遍历
  - 6.4.4 层序遍历
- 6.5 线索二叉树
  - 6.5.1 线索二叉树的定义
  - 6.5.2 线索二叉树的类定义
  - 6.5.3 中序线索二叉树
- 6.6 二叉树的应用
  - 6.6.1 堆
  - 6.6.2 哈夫曼树
- 6.7 树和森林
  - 6.7.1 树的存储结构
  - 6.7.2 树、森林和二叉树的转换
  - 6.7.3 树的遍历
  - 6.7.4 森林的遍历
- 6.8 等价类及其表示
  - 6.8.1 等价关系与等价类
  - 6.8.2 并查集
- 习题6
- 7 图
  - 7.1 图的基本概念
    - 7.1.1 图的定义
    - 7.1.2 图的术语
    - 7.1.3 图的基本操作和抽象数据类型
  - 7.2 图的存储结构
    - 7.2.1 邻接矩阵
    - 7.2.2 邻接表
    - 7.2.3 邻接多重表
    - 7.2.4 十字链表
  - 7.3 图的遍历与连通性
    - 7.3.1 深度优先遍历
    - 7.3.2 广度优先遍历
    - 7.3.3 连通分量
  - 7.4 最小生成树
    - 7.4.1 克鲁斯卡尔算法

## &lt;&lt;数据结构&gt;&gt;

- 7.4.2 普里姆算法
- 7.5 最短路径
  - 7.5.1 弧上权值为非负情形的单源点最短路径问题
  - 7.5.2 弧上权值为任意值的单源点最短路径问题
  - 7.5.3 所有顶点之间的最短路径
- 7.6 活动网络
  - 7.6.1 用顶点表示活动的网络
  - 7.6.2 用边表示活动的网络(AOE网络)
- 习题7
- 8 查找
  - 8.1 基本概念
  - 8.2 顺序表
    - 8.2.1 顺序表的查找
    - 8.2.2 有序表的折半查找
  - 8.3 索引顺序表
    - 8.3.1 索引顺序表
    - 8.3.2 倒排表
  - 8.4 二叉排序树
    - 8.4.1 二叉排序树定义
    - 8.4.2 二叉排序树上的查找
    - 8.4.3 二叉排序树的插入
    - 8.4.4 二叉排序树的删除
    - 8.4.5 二叉排序树查找的性能分析
  - 8.5 平衡二叉树
    - 8.5.1 平衡二叉树的定义
    - 8.5.2 平衡旋转
    - 8.5.3 平衡二叉树的插入和删除
  - 8.6 B-树
    - 8.6.1 动态的m路查找树
    - 8.6.2 B-树
    - 8.6.3 B-树的插入
    - 8.6.4 B-树的删除
    - 8.6.5 B+树
  - 8.7 散列表查找
    - 8.7.1 散列表的基本概念
    - 8.7.2 散列函数
    - 8.7.3 处理溢出的闭散列方法
    - 8.7.4 处理溢出的开散列方法——链地址法
    - 8.7.5 散列表分析
  - 习题8
  - 9 排序
    - 9.1 基础知识
      - 9.1.1 基本概念
      - 9.1.2 排序表的抽象数据类型描述和类定义
    - 9.2 交换排序
      - 9.2.1 冒泡排序
      - 9.2.2 快速排序

<<数据结构>>

9.3 插入排序

9.3.1 直接插入排序

9.3.2 折半插入排序

9.3.3 希尔排序

9.4 选择排序

9.4.1 直接选择排序

9.4.2 锦标赛排序

9.4.3 堆排序

9.5 归并排序

9.5.1 归并

9.5.2 两路归并排序

9.5.3 递归的归并排序

9.6 基数排序

9.6.1 多关键字排序

9.6.2 链式基数排序

9.7 各种排序方法的选择和使用

习题9

主要参考文献



## &lt;&lt;数据结构&gt;&gt;

## 章节摘录

插图：1.2 数据结构的基本概念 计算机科学是一门研究信息表示和处理的科学，人们是用程序来处理信息的。

信息的表示和组织直接关系到处理信息的程序的效率。

由于许多系统软件和应用软件的程序规模很大，结构又相当复杂，因而必须对程序设计方法进行系统的研究。

这不仅涉及到研究程序结构和算法，同时也涉及到研究程序加工的对象。

用计算机解题，首先应从具体问题抽象出一个适当的数学模型，然后才能设计算法和编制程序。

而构建数学模型的过程就是分析和概要设计的过程，要从对问题的分析中提取操作的对象，并找出这些操作对象之间的关系，然后用数学的语言加以描述。

例如，许多工程中的数值计算问题采用的数学模型是线性方程组或微分方程。

但更多的非数值计算问题是难以用数学方程来描述的。

1.2.1 两个简单的数据结构实例 为了获得对数据结构的感性认识，我们先来看两个简单的例子。

【例1-1】人事登记表。

在任何一个单位，人事登记表是人事部门关于职工信息的必不可少的表格。

例如，某个单位人事部门的工作人员想要查找当年退休的人员，或要查找基本工资在800元以下的人员，如果该单位有几千名职工，那么由人工来查找显然是很费时的；但如果在计算机中有一个关于职工信息的登记表，由有关的检索系统来查找，那是很方便的。

表1-1所示就是一张简单的人事登记表。

## <<数据结构>>

### 编辑推荐

《数据结构(C++实现)》为科学出版社出版发行。

<<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>