

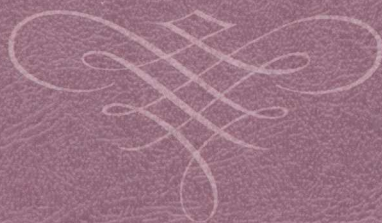
中国高等学校计算机科学与技术专业（应用型）规划教材

丛书主编 陈明

数据结构教程

（C++版）

陈明 编著



清华大学出版社



中国高等学校计算机科学与技术专业（应用型）规划教材

丛书主编 陈明

数据结构教程

（C++ 版）

陈明 编著



清华大学出版社

北京

内 容 简 介

本书系统地介绍了各种典型的数据结构,主要包括线性表、栈和队列、串、数组和广义表、树、图、查找、排序、递归和文件,为了加强对算法和C++ 语言的理解,还介绍了算法及算法分析、面向对象的程序设计与C++ 方面的内容。本书内容精选、概念清楚、注重实用、逻辑性强,各章中所涉及的数据结构与算法都给出了C++ 语言描述,并都附有大量习题,便于学生理解与掌握。

本书可作为高等院校计算机专业及相关专业的教材,也可作为计算机应用技术人员的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

数据结构教程(C++ 版) / 陈明编著. —北京:清华大学出版社, 2009.5

(中国高等学校计算机科学与技术专业(应用型)规划教材)

ISBN 978-7-302-18264-1

I. 数… II. 陈… III. ①数据结构—高等学校—教材 ②C 语言—程序设计—高等学校—教材 IV. ①TP311.12 ②TP312

中国版本图书馆 CIP 数据核字(2008)第 114005 号

责任编辑:谢琛 王冰飞

责任校对:时翠兰

责任印制:何芊

出版发行:清华大学出版社

<http://www.tup.com.cn>

社总机:010-62770175

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

地 址:北京清华大学学研大厦 A 座

邮 编:100084

邮 购:010-62786544

印 装 者:三河市春园印刷有限公司

经 销:全国新华书店

开 本:185×260

印 张:26.25

字 数:608 千字

版 次:2009 年 5 月第 1 版

印 次:2009 年 5 月第 1 次印刷

印 数:1~4000

定 价:36.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。
联系电话:010-62770177 转 3103 产品编号:030168-01

编委会

主任：陈 明

副主任：蒋宗礼 卢先和

委员：常 虹	陈国君	陈 峻	陈晓云	陈笑蓉
丛 琳	方路明	段友祥	高文胜	巩君华
关 永	郭 禾	郝 莹	何胜利	何晓新
贺安坤	胡巧多	李陶深	李仲麟	刘东升
刘贵龙	刘晓强	刘振华	路 游	马杰良
毛国君	苗凤君	宁 玲	施海虎	宋长龙
宋立军	孙践知	孙中胜	汤 庸	田俊峰
万本庭	王让定	王锁柱	王 新	王兆青
王智广	王志强	谢 琛	谢书良	徐孝凯
徐子珊	杨建刚	姚 琳	叶春蕾	叶俊民
袁 薇	张建林	张 杰	张 武	张晓明
张艳萍	周 苏	曾 一	訾秀玲	

序 言

应用是推动学科技术发展的原动力,计算机科学是实用科学,计算机科学技术广泛而深入的应用推动了计算机学科的飞速发展。应用型创新人才是科技人才的一种类型,应用型创新人才的重要特征是具有强大的系统开发能力和解决实际问题的能力。培养应用型人才的教學理念是教學过程中以培养学生的综合技术应用能力为主线,理论教学以够用为度,所选择的教學方法与手段要有利于培养学生的系统开发能力和解决实际问题的能力。

随着我国经济建设的发展,对计算机软件、计算机网络、信息系统、信息服务和计算机应用技术等专业技术方向的人才的需求日益增加,主要包括:软件设计师、软件评测师、网络工程师、信息系统监理师、信息系統管理工程师、数据库系統工程师、多媒体应用设计师、电子商务设计师、嵌入式系統设计师和计算机辅助设计师等。如何构建应用型人才培养的教學体系以及系統框架,是从事计算机教育工作者者的責任。为此,中国计算机学会计算机教育专业委员会和清华大学出版社共同组织启动了《中国高等学校计算机科学与技术专业(应用型)学科教程》的项目研究。参加本项目的研究人员全部来自国内高校教學一线具有丰富实践经验的专家和骨干教师。项目组对计算机科学与技术专业应用型学科的培养目标、内容、方法和意义,以及教學大纲和課程体系等进行了较深入、系統的研究,并编写了《中国高等学校计算机科学与技术专业(应用型)学科教程》(简称《学科教程》)。《学科教程》在编写上注意区分应用性人才与其他人才在培养上的不同,注重体现应用型学科的特征。在課程设计中,《学科教程》在依托学科设计的同时,更注意面向行业产业的实际需求。为了更好地体现《学科教程》的思想与内容,我们组织编写了《中国高等学校计算机科学与技术专业(应用型)规划教材》,旨在能为计算机专业应用型教學的課程设置、課程内容以及教學实践起到一个示范作用。本系列教材的主要特点如下:

1. 完全按照《学科教程》的体系组织编写本系列教材,特别是注意在教材设置、教材定位和教材内容的衔接上与《学科教程》保持一致。
2. 每門課程的教材内容都按照《学科教程》中设置的大纲精心编写,尽量体现应用型教材的特点。
3. 由各学校精品課程建设的骨干教师组成作者队伍,以課程研究为基础,將教學的研究成果引入教材中。
4. 在教材建设上,重点突出对计算机应用能力和应用技术的培养,注重教材的实践性。
5. 注重系列教材的立体配套,包括教参、教辅以及配套的教学资源、电子课件等。

高等院校应培养能为社会服务的应用型人才,以满足社会发展的需要。在培养模式、教学大纲、课程体系结构和教材都应适应培养应用型人才的目标。教材体现了培养目标和育人模式,是学科建设的结晶,也是教师水平的标志。本系列教材的作者均是多年从事计算机科学与技术专业教学的教师,在本领域的科学研究与教学中积累了丰富的经验,他们将教学研究和科学研究的成果融入教材中,增强了教材的先进性、实用性和实践性。

目前,我们对于应用型人才培养的模式还处于探索阶段,在教材组织与编写上还会有这样或那样的缺陷,我们将不断完善。同时,我们也希望广大应用型院校的教师给我们提出更好的建议。

《中国高等学校计算机科学与技术专业(应用型)规划教材》主编

陈明

2008年7月

前言

在非数值计算中,处理对象已从简单数值发展到具有一定结构的数据,这就需要讨论如何有效地组织计算机的存储,并在此基础上有效地实现对象间的运算,数据结构就是研究与解决这些问题的重要基础。

数据结构课是计算机科学与技术专业的一门必修的、重要的专业基础课,是计算机程序设计的重要理论技术基础。通过数据结构课的学习,不仅可以使同学们掌握数据结构的基本特性、数据的逻辑结构和数据的存储结构及典型算法和使用方法,而且能够训练学生应用数据结构和算法进行具体应用问题的程序设计。

全书分为 13 章,介绍最常用的数据结构、各种数据结构的逻辑关系、在计算机中的存储表示以及在数据结构上的运算等。主要内容包括算法及算法分析、面向对象的程序设计与 C++、线性表、栈和队列、串、数组和广义表、树、图、查找、排序、递归和文件等内容。

在结构上呈积木式,注重实践应用、各种常用数据结构的介绍从实际出发,避免抽象的理论论述和复杂的公式推导,在典型的算法介绍中深入浅出、简洁明了。每章都设有小结和习题,通过这些习题的练习,不仅能加深对基本概念和定义的理解,而且通过上机,能够提高编程能力和程序调试能力。

由于作者水平有限,书中不足之处在所难免,敬请读者批评指正。

陈 明
2009 年 1 月
于北京

目 录

第 1 章 绪论	1
1.1 数据结构的重要性	1
1.2 面向对象程序设计	2
1.2.1 面向对象方法	2
1.2.2 C++ 的特征及基本概念	3
1.3 基本术语	4
1.4 抽象数据类型	6
1.5 数据结构的概念	8
1.6 数据的逻辑结构	10
1.7 数据的存储结构	11
1.8 数据的运算	13
1.9 数据的逻辑结构、存储结构及数据的运算的关系	14
1.10 算法的描述和分析	14
1.10.1 算法描述	14
1.10.2 算法分析	15
小结	17
习题一	17
第 2 章 算法基础	19
2.1 算法的相关概念	19
2.1.1 算法的概念	19
2.1.2 算法与程序	19
2.1.3 数据结构与算法	20
2.2 算法分析的相关概念	20
2.2.1 算法分析的概念	21
2.2.2 算法的时间复杂度	22
2.2.3 算法的空间复杂度	25
2.3 算法分析举例	26

2.3.1	多项式问题	26
2.3.2	静态搜索问题	27
2.4	检验一个算法分析	30
	小结	31
	习题二	31
第3章	面向对象程序设计与C++	35
3.1	面向对象程序设计的概念	35
3.2	面向对象的程序设计与C++	35
3.3	变量、常量与数据类型	36
3.3.1	变量	36
3.3.2	常量	36
3.3.3	数据类型	37
3.4	控制语句	41
3.4.1	表达式语句和空语句	41
3.4.2	块语句	41
3.4.3	选择语句	42
3.4.4	循环语句	42
3.4.5	转移语句	42
3.5	函数	43
3.5.1	函数定义	43
3.5.2	函数声明	43
3.5.3	函数调用	44
3.5.4	参数传递	44
3.5.5	函数重载	46
3.5.6	构造函数和析构函数	47
3.5.7	友元函数	48
3.6	继承与派生	48
3.7	多态性、虚函数和纯虚函数	49
3.8	模板	51
3.8.1	模板的概念	51
3.8.2	函数模板与模板函数	52
3.8.3	类模板与模板类	54
3.9	输入与输出	55
	小结	56
	习题三	56
第4章	线性表	57
4.1	线性表及其抽象数据类型说明	57

4.1.1	线性表及其逻辑结构	57
4.1.2	线性表的抽象数据类型描述	61
4.2	线性表的顺序存储	62
4.2.1	顺序存储	62
4.2.2	顺序表(LinearList)类的定义	63
4.2.3	顺序表(LinearList)类的实现	64
4.3	线性表的链式存储	67
4.3.1	线性链表的存储结构	67
4.3.2	线性链表(Chain)类的定义	69
4.3.3	线性链表(Chain)类的实现	70
4.3.4	循环链表	74
4.3.5	循环链表 CircularList 类的实现	75
4.3.6	双向链表	76
4.3.7	可利用空间表	78
4.3.8	表遍历器	79
4.4	线性表的顺序存储和链式存储的比较	81
4.5	链式存储结构的应用	82
4.5.1	约瑟夫问题	82
4.5.2	一元多项式求和	83
	小结	87
	习题四	87
第5章 栈和队列		91
5.1	栈	91
5.1.1	栈的定义	91
5.1.2	栈的顺序存储结构	94
5.1.3	栈的链式存储结构	97
5.1.4	顺序栈和链式栈的比较	99
5.2	栈的应用	100
5.2.1	迷宫问题	100
5.2.2	表达式求值	103
5.2.3	汉诺塔问题	106
5.2.4	数制转换	108
5.2.5	行编辑	108
5.3	队列	110
5.3.1	队列的定义	110
5.3.2	队列的顺序存储	112
5.3.3	队列的链式存储	120

5.3.4	顺序队列与链式队列的比较	123
5.3.5	优先队列	124
5.4	队列的应用	124
5.4.1	解决设备速度不匹配问题	124
5.4.2	舞伴问题	125
5.4.3	火车车厢重排	126
小结		128
习题五		129
第6章	串	133
6.1	C++ 语言的字符和字符串	133
6.2	串的基本概念	134
6.3	串的存储结构	135
6.3.1	串的顺序存储结构	136
6.3.2	串的链式存储结构	137
6.3.3	串的索引存储结构	138
6.4	串的操作	139
6.4.1	常用的 C++ 字符串函数	139
6.4.2	串的抽象数据类型的描述	141
6.4.3	串的定义	142
6.4.4	部分成员函数的实现	143
6.5	串的基本运算与实现	146
6.5.1	串插入	146
6.5.2	串删除	147
6.6	模式匹配	149
6.6.1	模式匹配的 BF 算法	149
6.6.2	模式匹配的 KMP 算法	151
6.7	串在文本编辑中的应用	155
小结		156
习题六		156
第7章	数组和广义表	159
7.1	C++中数组的定义及抽象数据类型表示	159
7.1.1	C++中数组的定义	159
7.1.2	数组的抽象数据类型表示	160
7.2	数组的顺序存储结构	161
7.3	矩阵的压缩存储	162
7.3.1	特殊矩阵的压缩存储	163

7.3.2 稀疏矩阵的压缩存储	166
7.4 广义表的概念	173
7.5 广义表的存储结构表示	174
7.6 广义表的运算	176
小结	183
习题七	183
第 8 章 树	187
8.1 树	187
8.1.1 树的定义	187
8.1.2 树的表示形式	188
8.1.3 树的常用术语	189
8.1.4 树的基本操作	189
8.1.5 一个树的接口	190
8.1.6 树的基本算法	191
8.2 二叉树	193
8.2.1 二叉树的定义	193
8.2.2 二叉树的性质	194
8.2.3 二叉树的接口	197
8.2.4 二叉树的存储结构	197
8.2.5 二叉树的遍历	204
8.2.6 二叉树遍历的应用	207
8.3 线索二叉树	208
8.3.1 线索二叉树的类定义	208
8.3.2 中序线索二叉树	212
8.4 树、森林和二叉树的关系	215
8.4.1 树的存储结构	215
8.4.2 森林与二叉树的转换	218
8.4.3 树和森林的遍历	220
8.5 哈夫曼树及其应用	222
8.5.1 哈夫曼树的定义	222
8.5.2 哈夫曼树的构造	223
8.5.3 哈夫曼树在编码问题中的应用	226
小结	227
习题八	228
第 9 章 图	231
9.1 图的基本概念	231

9.1.1	图的定义及基本概念	231
9.1.2	图的抽象数据类型	234
9.2	图的存储结构	236
9.2.1	邻接矩阵表示法	236
9.2.2	邻接表	242
9.2.3	十字链表	249
9.2.4	邻接多重表	250
9.3	图的遍历	252
9.3.1	深度优先搜索	252
9.3.2	广度优先搜索	254
9.3.3	欧拉回路	255
9.4	图的连通性	257
9.4.1	连通分量	257
9.4.2	重连通分量	259
9.5	生成树	259
9.5.1	普里姆(Prim)算法	261
9.5.2	克鲁斯卡尔(Kruskal)算法	264
9.6	最短路径	266
9.6.1	单源最短路径	266
9.6.2	求每一对顶点之间的最短路径	269
9.7	拓扑排序	270
9.8	关键路径	274
	小结	280
	习题九	280
第 10 章	查找	285
10.1	基本概念	285
10.2	线性表的查找	286
10.2.1	顺序查找	286
10.2.2	折半查找	288
10.2.3	索引查找	290
10.2.4	分块查找	294
10.3	树表查找	296
10.3.1	二叉查找树	296
10.3.2	平衡二叉树	303
10.3.3	B-树	307
10.4	哈希表的查找	309
10.4.1	哈希表	309

10.4.2	构造哈希表的基本方法	311
10.4.3	解决冲突的方法	313
10.4.4	哈希表的查找方法	315
10.5	各种查找方法的比较	316
小结		317
习题十		318
第 11 章	排序	321
11.1	基本概念	321
11.2	内部排序	324
11.2.1	插入排序	324
11.2.2	交换排序	329
11.2.3	选择排序	333
11.2.4	归并排序	341
11.2.5	基数排序	345
11.3	内部排序方法比较	349
11.4	外部排序简介	350
小结		351
习题十一		351
第 12 章	递归	355
12.1	递归的定义	355
12.2	常见递归问题	356
12.2.1	汉诺塔问题	356
12.2.2	八皇后问题	358
12.2.3	表达式树	360
12.3	递归的实现	362
12.4	消除递归	365
12.4.1	尾递归和单向递归的消除	365
12.4.2	用栈模拟系统运行时的栈	367
12.5	递归的评估	370
小结		371
习题十二		371
第 13 章	文件	375
13.1	外存储器的介绍	375
13.2	磁盘	376
13.3	有关文件的概念	377

13.3.1	文件及其类别	378
13.3.2	文件的操作	379
13.4	文件的组织	380
13.4.1	顺序文件	381
13.4.2	索引文件	382
13.4.3	散列文件	388
13.4.4	多关键字文件	389
13.5	外部排序	391
13.5.1	外部排序的简单方法	392
13.5.2	两路归并	392
13.5.3	多路归并	395
13.6	文件的索引结构	396
13.6.1	索引向量	396
13.6.2	树形索引结构	397
	小结	397
	习题十三	397
参考文献		399

第1章 绪 论

在深入学习数据结构之前,应首先了解学习数据结构的意义、什么是数据结构及数据结构的一些相关概念等。这对于深刻理解后面章节的内容将会有很大的帮助。

1.1 数据结构的重要性

在计算机发展的初期,人们使用计算机主要是处理数值的计算问题,程序设计人员也主要把精力集中在程序设计的技巧上,但随着计算机应用领域的扩大和软硬件的发展,计算机对信息的处理加工已从单一的数值计算发展到大量地解决非数值问题,其加工处理的信息也由简单的数值发展到字符、图像、声音等具有复杂结构的数据。而数据结构就是随着计算机的产生和发展而发展起来的一门较新的计算学科。

在非数值计算问题中,数据之间的相互关系一般无法完全用数学方程式加以描述,并且数据的表示方法和组织形式直接关系到程序对数据的处理效率,而系统程序和许多应用程序的规模很大,结构复杂,这时人们考虑问题的关键已不再是分析数据和计算方法,而是是否能设计出合适的数据结构,有效地解决问题。

总之,计算机科学是一门研究用计算机进行信息表示和处理的科学。这里面涉及两个问题:信息的表示和信息的处理。而信息的表示和组成又直接关系到处理信息的程序的效率。随着计算机的普及、信息量的增加、信息范围的拓宽,使许多系统程序和应用程序的规模很大,结构又相当复杂。这就要求人们对计算机程序加工的对象进行系统的研究,即研究数据的特性以及数据之间存在的关系,而数据结构(Data Structure)正是描述数据的特性以及数据之间存在的关系的一门课程。

例 1-1 电话号码查询问题。

编写一个查询某个城市或单位的私人电话号码的程序。要求对任意给出的一个姓名,若有该人电话号码,则迅速找到其电话号码;否则指出没有该人电话号码。

要解此问题首先构造一张电话号码登记表。表中每个结点存放两个数据项:姓名和电话号码。

要写出好的查找算法,取决于这张表的结构及存储方式。最简单的方式是将表中结点顺序地存储在计算机中。查找时从头开始依次查对姓名,直到找出正确的姓名或是找遍整个表均没有找到为止。这种查找算法对于一个不大的单位或许是可行的,但对一个有成千

上万私人电话的城市就不实用了。若这张表是按姓氏排列的,则可另造一张姓氏索引表,采用如图 1-1 所示的存储结构。那么查找过程是先在索引表中查对姓氏,然后根据索引表中的地址到电话号码登记表中核查姓名,这样查找登记时就无需查找其他姓氏的名字了。因此,在这种新的结构上产生的查找算法就更为有效。

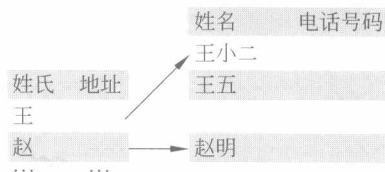


图 1-1

数据结构是计算机专业的核心课程之一,在众多的计算机系统软件和应用软件中都要用到各种数据结构。可以这样说,数据结构不仅是一般程序设计的基础,而且是实现编译程序、操作系统、数据库系统及其他系统程序和大型应用程序的基础。因此,仅掌握几种计算机语言是难以应付众多复杂的研究课题的,要想有效地使用计算机,还必须学习数据结构的知识。

瑞士计算机科学家 N. Wirth 教授曾提出这样一个等式:

$$\text{算法} + \text{数据结构} = \text{程序}$$

这个等式形象地描述了算法、数据结构和程序之间的关系,这里的数据结构指的是数据的逻辑结构和存储结构,而算法就是对数据运算的描述。由此可见,程序设计的实质就是对实际问题选取一种优秀的数据结构,加之设计一个优秀的算法,而且好的算法很大程度上取决于描述实际问题的数据结构。因此,对于一个程序设计人员来说,学好数据结构是十分重要的,那什么是数据结构呢?下面一节将对数据结构和一些相关概念进行解释。

1.2 面向对象程序设计

1.2.1 面向对象方法

1. 面向对象方法

面向对象方法是基于客观世界的对象模型的软件开发方法。当我们设计和实现一个客观系统时,如能在满足需求的条件下,把系统设计成由一些相对固定的部分组成的最小集合,则这个设计就是优秀的,而这些相对固定的部分就被看成是一些不同的对象。面向对象方法就是要建立与语言无关的对象模型,并围绕这些对象进行系统设计。

面向对象方法有如下基本特征:

- (1) 对象是数据和操作的封装体,较好地实现了数据的抽象。
- (2) 面向对象方法的继承性体现了概念分离抽象。下层对象继承上层对象的属性和操作,因而便于软件的演化和扩充。
- (3) 面向对象方法具有信息隐藏性。对象将其实现的细节隐藏在内部,所以无论是对对象功能的完善扩充,还是对象实现的修改,都不会对外界产生影响。这就保证了面向对象软件的可构造性和易维护性。
- (4) 面向对象方法用消息将对象动态链接在一起。面向对象方法采用了灵活的消息传