

新世纪计算机专业系列教材

数据结构

——C++实现

缪淮扣 顾训穰 沈俊 编著

新世纪计算机专业系列教材

数据结构—C++实现

缪淮扣 顾训穰 沈 俊 编著

科学出版社

2002

内 容 简 介

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。

图书在版编目(CIP)数据

数据结构:C++实现/缪淮扣,顾训穰,沈俊编著. —北京:科学出版社, 2002

(新世纪计算机专业系列教材)

ISBN 7-03-010457-9

I. 数… I. ①缪… ②顾… ③沈… 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 N. TP311.12

中国版本图书馆 CIP 数据核字(2002)第 033707 号

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

双青印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2002 年 7 月第 一 版 开本:787×1092 1/16

2002 年 7 月第一次印刷 印张:21 1/4

印数:1—5 000 字数:483 000

定价:29.00 元

(如有印装质量问题,我社负责调换<环伟>)

新世纪计算机专业教材

编委会

顾问编委

施伯乐教授	复旦大学
白英彩教授	上海交通大学

主 任

左孝凌教授	上海交通大学
-------	--------

编 委

刘 璟教授	南开大学
宋方敏教授	南京大学
何炎祥教授	武汉大学
余雪丽教授	太原理工大学
阮家栋教授	上海工程技术大学
顾训穰教授	上海大学
徐汀荣教授	苏州大学
曾 明教授	西安交通大学
曹元大教授	北京理工大学
曹文君教授	复旦大学
陶树平教授	同济大学
缪淮扣教授	上海大学
谢康林教授	上海交通大学

总 序

20 年来, 计算机学科的发展日新月异, 促使现代科学在各个领域突飞猛进。目前, 计算机科学技术已应用在实时控制、信息处理、通信传输、企事业管理等领域, 成为人们工作、学习、生活必不可少的工具。计算机技术的发展瞬息万变, 具有以下三方面特点:

(一) 传统的工、理、文、医、商、农在计算机的应用方面都有着各自专业的需要, 例如, 经济、艺术、法律、管理、医学等各种学科都需要依赖于计算机技术的应用。除了各自领域的专业实践外, 应用计算机已是各个专业提高效率、发挥潜能、促进发展的必不可少的手段。因此现在很难用传统的工、理、文、医、商、农等去界定学科的分类。

(二) 计算机网络改变了计算机通信的时空距离。计算机应用的发展是与计算机网络的发展紧密相连的。从最初的局域网 (LAN) 到广域网 (WAN), 以至用一种新的方法将 LAN 和 WAN 互联起来, 即成为网际网 (Internetwork)。这种网际网的实验原型 Internetwork, 通常缩写为 Internet。计算机网络将计算机互连起来, 从而使计算机之间可以交换信息, 而且这种信息交换可以在几分钟内就影响到世界各地。计算机网络的发展, 带动了计算机学科在很多领域的拓展。

(三) 现代计算机学科向综合性发展。计算技术发展伊始, 每种学科均以软硬件分类, 泾渭分明。但自网络发展以来, Internet 软件中的两部分变得特别重要和特别具有开创性, 即网际协议 (Internet Protocol, 简称 IP) 和传输控制协议 (Transmission Control Protocol, 简称 TCP)。这些协议是必不可少的软件系统。但是在网络系统中, 网络的互连必须依靠路由器、服务器、接口插座、调制解调器等硬件设施, 所以计算机网络很难归结为软件或硬件的单一体系。

随着计算机技术的发展, 计算机与通讯、视频、声音等密不可分; 随着多媒体的发展和应用, 计算机科学已经愈来愈成为与数字传输、视频、声、光、电等综合的学科。

尽管计算机技术的发展如此神速、新异, 但像一切新学科的发展一样, 计算机教育水平仍滞后于计算机技术的发展。为了适应计算机教学改革的需要, 我们国内部分重点院校的教授、学者, 在科学出版社的积极鼓励和支持下, 成立了新世纪计算机专业教材编委会。自 2000 年 10 月以来, 我们群策群力, 多次探讨了当前教育与技术进展之间的差距, 并且仔细研讨了美国 ACM/IEEE-CS 公布的 *Computing Curricula 2001* 的优点与不足, 结合我国计算机教育的实际情况, 提出了编著一套适用于计算机本科专业的励精图治的教材计划。这套教材的选题、定位乃至作者的遴选, 都得到了国内很多著名教授和学者的认同, 并且有很多选题都争取到了一些著名教授亲自参与编写。这套教材立意着重基础, 反映导向, 注重实践。

因此我们在基础课目方面, 首先列选了数据库原理、操作系统、编译程序原理、智能基础等基础教程。这些基础课教材都由一些国内著名学者执笔, 论述内容既注意打好扎实

基础,又注意要反映最新导向,高屋建瓴,使读者迅速接近最新领域。

同时,为了反映导向,我们抓住网络课程作为计算机专业学生的应用基础,编写了一本实用性极强的《计算网络教程》。这本教材的编著思想是以基础—理论—应用为主线,通信是基础,协议是核心,互连是重点,应用是目标。

其次,为了拓展学生的网络应用本领,我们还安排了电子商务、多媒体应用以及 Web 数据库技术三门应用课程。电子商务和多媒体应用是计算机应用中最为热门的课程,也是拓展性极广的计算机应用领域,应用前景极为广阔。

Web 数据库技术是一种随着互联网技术发展起来的应用技术。它涉及网络、HTTP 协议、Script 语言、动态网页开发平台、远程数据访问技术等各种网络应用技术。目前国内外尚无适合教材,因此,编写 Web 数据库技术的教材,可以说是填补了应用领域的一个空白。

在研究美国公布的“计算 2001-CS 教程”中,我们仔细探讨了数据结构这一课程的变化。在“计算 1991 教程”中,数据结构内容明确放在算法与数据结构之中,而“2001-CS 教程”却无数据结构的课程名称,代之以程序设计基础(Programming Fundamentals)。文件中提到了基本数据结构和抽象数据类型以及面向对象的程序设计等内容。从这里可以看出,数据结构是以程序设计基础作为研究对象的。另外该教程把算法与复杂性作为一个单独课程列出,这一方面说明算法是一种问题求解的策略,另一方面也说明基本算法及复杂性的讨论对于程序设计是多么重要。

为此在这套丛书中我们安排了一个软件课程系列,即开设从语言、数据结构、算法到软件工程的课程。首先我们从面向对象的 C++ 语言入手,进一步讲解语言学概论。主要内容是分析语法结构,掌握语言构成规律,读懂语言文本。任何计算机语言均可触类旁通,这种从结构规律来学会应用的方法,就是以不变应万变,因为从根本上说,尽管计算机语言千变万化,但万变不离其宗。在搞通语言基础上,我们组编了数据结构,或者说是研究程序设计基础。然后是学习基本算法,也就是为了程序设计需要,而进行问题求解,即进行常用算法讨论。为了使开发软件遵循工程管理方法,软件工程的学习将是计算机专业学生规范软件开发的必不可少的训练课程。

我们筹组这套丛书时,希望每本教材都有创意,能引起共鸣,能被关注,能被采纳,能被推广。但是我们也注意到,由于各个学校情况不同,各人观点不同,理解角度也有所不同,所以对教材的选用和编著,不易一致认同。不过我们希望这套教材能够反映当前学校动向,在促进学以致用等方面有所促进、有所推动,更希望兄弟院校的教师、学者能够积极使用,参与讨论,以使本套丛书能够不断修改,日臻完善。

最后我要感谢科学出版社的领导对本套丛书的列选、报审、出版所给予的鼓励和支持。

左孝凌

2001 年 7 月 30 日

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观念。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。

本书第1~5章由缪淮扣编写,第6~9章由沈俊和顾训穰共同编写。占学德和刘玲调

试了部分算法。全书由缪淮扣和顾训穰统稿。

在本书的写作过程中,本丛书的编委会、科学出版社、上海大学教务处和计算机学院给予了很大支持。在此致以诚挚感谢。

由于时间仓促和作者水平有限,本书难免存在疏漏和缺点,敬请广大读者批评指正。

缪淮扣

2002年3月于上海

目 录

1 绪论	1
1.1 (算法+数据结构)=程序	1
1.2 数据结构的基本概念	2
1.2.1 两个简单的数据结构实例	2
1.2.2 什么是数据结构	3
1.3 C++语言基础	4
1.3.1 程序结构	5
1.3.2 数据声明和作用域	6
1.3.3 输入/输出	7
1.3.4 函数	9
1.3.5 参数传递	10
1.3.6 函数名重载	11
1.3.7 动态内存分配	11
1.3.8 结构与联合	12
1.4 算法性能与复杂度	16
1.4.1 算法的定义	16
1.4.2 算法的性能标准	17
1.4.3 算法的复杂度	17
习题 1	21
2 抽象数据类型和 C++类	23
2.1 抽象数据类型	23
2.1.1 从数据类型到抽象数据类型	23
2.1.2 封装和信息隐藏	24
2.1.3 抽象数据类型描述	25
2.2 类与对象的基本概念	26
2.2.1 类与对象	26
2.2.2 消息与合作	28
2.2.3 多态性	28
2.3 面向对象的程序设计方法	28
2.4 C++类与对象	29
2.5 构造函数和析构函数	31

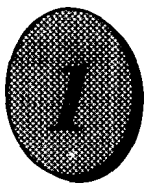
2.6	工具函数	35
2.7	继承	38
2.8	this 指针的使用	41
2.9	虚函数、多态性以及动态联编	42
2.9.1	虚函数和多态性	42
2.9.2	动态联编	50
2.10	模板类	52
	习题 2	54
3	线性表	56
3.1	线性表的定义	56
3.2	线性表的顺序表示	57
3.2.1	顺序表的类定义	57
3.2.2	顺序表插入、删除算法的复杂度分析	60
3.2.3	顺序表的应用	61
3.3	线性表的链表表示	62
3.3.1	单链表	62
3.3.2	单循环链表	73
3.3.3	双向循环链表	73
3.3.4	静态链表	79
3.4	多项式抽象数据类型	81
3.4.1	多项式表示	81
3.4.2	多项式相加	81
	习题 3	83
4	栈、队列和递归	85
4.1	栈	85
4.1.1	顺序栈	86
4.1.2	链式栈	88
4.1.3	表达式的计算	90
4.2	队列	94
4.2.1	循环队列	95
4.2.2	链队列	98
4.3	递归	100
4.3.1	递归的概念	100
4.3.2	递归过程与递归工作栈	101
4.3.3	消除递归	103
4.3.4	迷宫问题	107
	习题 4	109

5 串、数组和广义表	112
5.1 字符串	112
5.1.1 字符串的定义、存储结构和操作	112
5.1.2 串的操作	113
5.1.3 常用的 C++ 字符串函数	114
5.1.4 串类及其实现	115
5.1.5 模式匹配算法	121
5.2 数组	125
5.2.1 C++ 中数组的定义	126
5.2.2 数组的抽象数据类型表示	126
5.2.3 数组的顺序存储结构	128
5.3 稀疏矩阵	130
5.3.1 三元组顺序表	131
5.3.2 十字链表	133
5.4 广义表	135
5.4.1 广义表的定义	135
5.4.2 广义表的存储结构	136
5.4.3 n 元多项式的表示	139
5.4.4 广义表的递归算法	141
习题 5	144
6 树和森林	147
6.1 树的概念	147
6.1.1 树的定义	148
6.1.2 树的术语	148
6.1.3 树的表示形式	149
6.1.4 树的基本操作和抽象数据类型	149
6.2 二叉树	153
6.2.1 二叉树的定义	153
6.2.2 二叉树的性质	153
6.2.3 二叉树的基本操作和抽象数据类型	155
6.3 二叉树的存储结构	158
6.3.1 数组表示法	159
6.3.2 链表表示法	160
6.3.3 二叉树的二叉链表类声明	160
6.4 遍历二叉树	164
6.4.1 前序遍历	164
6.4.2 中序遍历	165

6.4.3	后序遍历	165
6.4.4	层序遍历	166
6.5	线索二叉树	168
6.5.1	线索二叉树的定义	168
6.5.2	线索二叉树的类定义	170
6.5.3	中序线索二叉树	173
6.6	二叉树的应用	177
6.6.1	堆	177
6.6.2	哈夫曼树	183
6.7	树和森林	187
6.7.1	树的存储结构	187
6.7.2	树、森林和二叉树的转换	190
6.7.3	树的遍历	192
6.7.4	森林的遍历	193
6.8	等价类及其表示	194
6.8.1	等价关系与等价类	194
6.8.2	并查集	195
	习题 6	199
7	图	203
7.1	图的基本概念	203
7.1.1	图的定义	203
7.1.2	图的术语	204
7.1.3	图的基本操作和抽象数据类型	207
7.2	图的存储结构	209
7.2.1	邻接矩阵	209
7.2.2	邻接表	212
7.2.3	邻接多重表	217
7.2.4	十字链表	219
7.3	图的遍历与连通性	220
7.3.1	深度优先遍历	220
7.3.2	广度优先遍历	221
7.3.3	连通分量	223
7.4	最小生成树	224
7.4.1	克鲁斯卡尔算法	225
7.4.2	普里姆算法	228
7.5	最短路径	230
7.5.1	弧上权值为非负情形的单源点最短路径问题	231

7.5.2	弧上权值为任意值的单源点最短路径问题	234
7.5.3	所有顶点之间的最短路径	236
7.6	活动网络	238
7.6.1	用顶点表示活动的网络	238
7.6.2	用边表示活动的网络(AOE 网络)	242
习题 7		246
8	查找	250
8.1	基本概念	250
8.2	顺序表	251
8.2.1	顺序表的查找	251
8.2.2	有序表的折半查找	252
8.3	索引顺序表	256
8.3.1	索引顺序表	256
8.3.2	倒排表	258
8.4	二叉排序树	260
8.4.1	二叉排序树定义	260
8.4.2	二叉排序树上的查找	262
8.4.3	二叉排序树的插入	263
8.4.4	二叉排序树的删除	265
8.4.5	二叉排序树查找的性能分析	266
8.5	平衡二叉树	266
8.5.1	平衡二叉树的定义	267
8.5.2	平衡旋转	267
8.5.3	平衡二叉树的插入和删除	269
8.6	B-树	273
8.6.1	动态的 m 路查找树	273
8.6.2	B-树	274
8.6.3	B-树的插入	274
8.6.4	B-树的删除	276
8.6.5	B+树	278
8.7	散列表查找	279
8.7.1	散列表的基本概念	279
8.7.2	散列函数	281
8.7.3	处理溢出的闭散列方法	282
8.7.4	处理溢出的开散列方法——链地址法	286
8.7.5	散列表分析	287
习题 8		288

9 排序	292
9.1 基础知识	292
9.1.1 基本概念	292
9.1.2 排序表的抽象数据类型描述和类定义	293
9.2 交换排序	299
9.2.1 冒泡排序	299
9.2.2 快速排序	300
9.3 插入排序	302
9.3.1 直接插入排序	302
9.3.2 折半插入排序	306
9.3.3 希尔排序	306
9.4 选择排序	308
9.4.1 直接选择排序	308
9.4.2 锦标赛排序	310
9.4.3 堆排序	312
9.5 归并排序	314
9.5.1 归并	314
9.5.2 两路归并排序	315
9.5.3 递归的归并排序	317
9.6 基数排序	319
9.6.1 多关键字排序	319
9.6.2 链式基数排序	320
9.7 各种排序方法的选择和使用	322
习题 9	323
主要参考文献.....	326



绪 论

1.1 (算法+数据结构)=程序

计算机能快速计算,能证明“四色定理”,能帮助工程师设计图纸,能帮助医生为病人诊断配药……它之所以如此神通广大、聪明能干,是因为人们教给了它本领。人是用“程序”来“教”计算机的。要想让计算机具备解题的能力,就要为计算机编写程序。随着计算机科学与技术的不断发展,计算机的应用领域也在不断扩大。“程序”越来越庞大、越来越复杂,因而解题的过程就不仅仅是编写程序了,而是一个包括编程序在内的软件开发过程,该过程包括对用户的需求进行分析,对要开发的系统进行软件设计,然后编程序,再进行测试和改错,最后再运行等一系列的工作。“软件”不单纯是指程序,而是指程序以及开发程序的过程中所产生的各种文档。软件开发的目的是产生能让计算机有效工作的程序,因此程序是软件的核心。

程序到底是什么呢? 计算机科学家,图灵奖获得者 N. Wirth 给出过一个著名的公式: 算法+数据结构=程序。从这个公式可以看到,数据结构和算法是构成程序的两个同样重要的组成部分。这个公式在软件开发的进程中产生了深远的影响。但是它并没有强调数据结构与解题的算法是一个整体。该公式现在已经受到了挑战。

我们知道,非面向对象的过程语言,如 FORTRAN、C 和 PASCAL,其数据结构是问题解的中心。一个软件系统的结构是围绕一个或几个关键数据结构为核心而组成的。“算法+数据结构=程序”完全体现了这种思想。但随着软件系统的规模越来越大、复杂性不断增长,人们不得不“对‘关键数据结构’重新评价。在这种系统中,许多过程和函数(子程序)的实现严格依赖于关键数据结构,如果这些关键数据结构的一个或几个数据有所改变,则涉及到整个软件系统,许多过程和函数必须重写,甚至因为几个关键数据结构的改变,导致软件系统整个结构彻底崩溃。

20 世纪 90 年代,面向对象的方法得到了很大的重视,并得到了比较广泛的推广和应用。在面向对象程序设计中,密切相关的数据与过程被定义为一个整体(即对象),而且一旦作为一个整体定义了之后,就可以使用它,而无需了解其内部的实现细节,从而提高软件开发的效率。

封装和数据隐藏是面向对象问题解和面向对象程序设计的基本要素。它可以使软件的许多维护问题简单化。一旦数据(类型)结构修改了,只要对实现模型的局部代码作适当修改,软件系统的其余部分无需修改。

有人主张将 Wirth 的公式“算法+数据结构=程序”修改为“(算法+数据结构)=程序”,以体现面向对象的方法。

本书以面向对象的观点来介绍各种数据结构以及与这些数据结构有关的算法的知

识。

本章将介绍数据结构以及算法的基本概念,并介绍用来描述数据结构和算法的语言C++。

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。

表 1-1 人事登记表

编号	姓名	性别	出生日期	婚否	基本工资
0001	王 军	男	1960/5/30	已	650
0002	李 平	女	1953/6/2	已	710
0003	周丽娟	女	1948/7/8	已	980
0004	赵忠良	男	1950/12/2	已	950
0005	张国庆	男	1978/10/1	未	500
⋮	⋮	⋮	⋮	⋮	⋮

表 1-1 中,每个职工的信息放在一行中,我们称之为一个数据元素,所有职工的信息按照某一种顺序依次存放在表中。一般称这种由记录组成的线性表为文件。

类似这样表格在不同的计算机软件系统中是很多的,如学校的学生信息管理系统中的学生学籍登记表、图书馆管理系统中的图书目录表等。这类表格有一个共同的特性,就是各数据元素之间的关系是线性的关系,即一个元素的前面只有一个元素,后面也只有一个元素,第一个数据元素前面和最后一个数据元素后面没有元素。这样的一类表格就是一

种数据结构,称为线性数据结构。

【例 1-2】 一个典型的学校行政机构如图 1-1 所示,这是一个层次结构。顶层结点“学校”代表整个系统,它的下一层结点代表这个系统的各个子系统,即部处与学院。再下一层结点代表更小的机构,如教务科、计算机系等,直到最底层一个小组或一个教研室等。在该图中,每一个结点代表一个数据元素,每一个结点的下面可能有多个结点。这样的一种结构称为层次型数据结构。

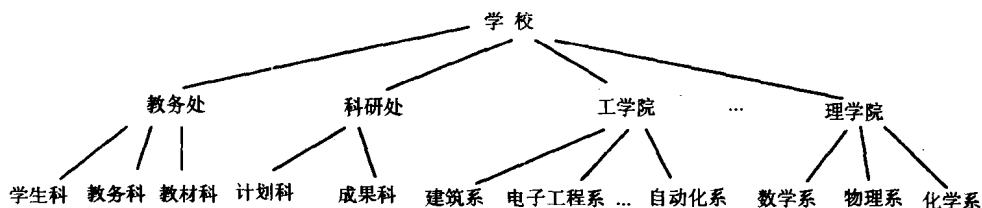


图 1-1 学校机构的“树结构”

在各种应用程序中会涉及到各种各样的数据,为了组织它们、存储它们,并且能对它们进行操作,就要考虑它们的归类以及它们之间的关系,从而建立相应的数据结构,并依此实现所要求的功能。

1.2.2 什么是数据结构

一个水平再高的厨师,如果不给他原料,他也无法做出色、香、味俱全的菜。这也就是人们常说的“巧妇难为无米之炊”。对一个程序来讲,数据就是“原料”。一个程序所要进行的计算或处理总是以某些数据为对象的。

大千世界中有各种各样的信息,如马路上的交通灯、进出地铁站的交通卡、股市投资者与证券商之间的交易、人们用语言交流的思想等。这些信息必须转换成数据才能在计算机中进行处理。让我们先来定义什么是数据以及与之相关的概念,然后再来回答“什么是数据结构”这个问题。

数据(data)是信息的载体,是描述客观事物的数、字符、图形、图像、声音以及所有能输入到计算机中并被计算机程序识别和处理的符号的集合。

例如,一个解代数方程的程序处理的对象只是整数和实数,一个 C++ 语言的编译程序处理的对象是由 ASCII 码字符组成的字符串等。

数据的基本单位是数据元素(data element)。一个数据元素可由若干个数据项(data item)组成,数据项是数据的最小单位。

为了解题的需要,具有相同特性的数据元素经常归类,被称之为数据对象的集合。数据对象是数据的子集。例如,所有的“数”构成了“数据”集合,而自然数集合 $N = \{0, 1, 2, \dots\}$ 是“数”的数据对象;所有的字符是数据,字母集合 $AS = \{A, B, \dots, Z, a, b, \dots, z\}$ 是该数据的数据对象。

在大多数情况下,一个数据元素不只是含有一个数据项,而是由多个数据项组成的。如在例 1-1 中,一个职工的记录就是一个数据元素,它包括编号、姓名、性别、出生日期、婚否和基本工资六个数据项。