

高等院校
信息技术应用型
规划教材

数据结构实用教程

于莉莉 白振林 主编

清华大学出版社



高等院校
信息技术应用型
规划教材

数据结构实用教程

于莉莉 白振林 主 编
平金珍 李新强 王伟锋 许 伟 副主编

清华大学出版社
北京

内 容 简 介

本书注重培养学生的实践能力。以实践项目为中心组织本书的内容,将所有知识点融入实践项目,理论内容的设置遵循“实用为主、必需和够用为度”的准则。本书采用 Java 语言作为算法的描述语言,共由 8 章组成,分别讲解数据结构的基本概念、线性表、栈和队列、串和数组、树和二叉树、图、查找、排序等内容。根据线性表、栈和队列、串和数组、树和二叉树、图这几种常用的数据结构的特点,结合猴子选大王、一元多项式加法运算、舞伴问题、文本加密器、哈夫曼编码器、教学计划的编制等具体项目,强化了数据结构思想的应用。

全书按照“任务说明→知识点说明→任务分析→任务实现→小结→巩固与拓展”的总体思路来组织任务的内容,线索清晰,知识内容由浅入深、循序渐进,符合学习者的认知规律。

本书可作为高职高专院校计算机专业的教材,也可供计算机相关专业学生参考。

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

数据结构实用教程/于莉莉,白振林主编.--北京:清华大学出版社,2014

高等院校信息技术应用型规划教材

ISBN 978-7-302-34839-9

I. ①数… II. ①于… ②白… III. ①数据结构—高等学校—教材 IV. ①TP311.12

中国版本图书馆 CIP 数据核字(2013)第 310953 号

责任编辑:陈砺川

封面设计:傅瑞学

责任校对:袁 芳

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

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

质 量 反 馈:010-6277015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795764

印 装 者:北京密云胶印厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:13.25

字 数:304 千字

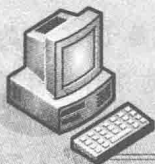
版 次:2014 年 7 月第 1 版

印 次:2014 年 7 月第 1 次印刷

印 数:1~2500

定 价:30.00 元

产品编号:055930-01



前言

“数据结构”是计算机程序设计的重要理论基础,它揭示了数据的逻辑结构、存储结构和相应的操作,理论性较强。“数据结构”又是一门实践性很强的课程,主要用于培养学生分析数据、组织数据的能力,从而使他们能够编写出效率高、结构好的程序。在工程开发的各阶段、软件工程开发的实施过程中能否正确、灵活地运用数据结构的思维设计出解决问题的算法,是检验“数据结构”课程学习效果的重要标准。

目前市面上有很多版本的“数据结构”教材,但这些教材大多采用串讲的方式来组织理论知识,即“提出概念→解释概念→举例说明”,这种形式不能很好地结合实际应用来讲解理论,不太符合高职院校学生从具体实例开始再到一般的认知过程,使他们很难理解数据结构在程序设计过程中的重要作用。此外,这些教材基本采用 C 或 C++ 语言作为数据结构和算法的描述语言,偏向于基础的应用。

本书弥补了上述数据结构教材的缺陷,注重培养学生的实践能力,理论知识传授遵循“实用为主、必需和够用为度”的准则,以实践任务为中心组织课程内容,将所有知识点融入实践任务。本书共分 8 章,每章的内容都按照“任务说明→知识点说明→任务分析→任务实现→小结→巩固与拓展”的总体思路来呈现,引导学生从了解项目需求开始,有针对性地准备相应知识,通过任务分析将知识点与任务相融合,从而设计实现任务效果,最后对整个过程进行总结,达到提高能力和加深理解的目的。书中每个任务的最后都有配套的知识巩固与能力拓展内容,通过基础知识巩固题目检验学习效果,通过对应的能力拓展题目提高学生的应用能力。

本书的内容分别如下:第 1 章初识数据结构,主要介绍数据结构的相关术语、学习数据结构的意义、算法的度量等知识;第 2~6 章介绍线性表、栈和队列、串和数组、树和二叉树、图等常用的数据结构,根据几种数据结构的特点,结合猴子选大王、一元多项式加法运算、舞伴问题、文本加密器、哈夫曼编码器、教学计划的编制等项目,将理论与实践结合起来;第 7、8 章介绍了经典的查找和排序算法,并结合简单的学生成绩管理系统说明查找和排序的具体应用。

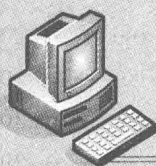
本书选择 Java 作为描述语言,因为相对于其他语言,Java 语言比较完整、彻底地体现了面向对象的设计思想,还能与面向对象的程序设计语言无缝对接。

本书由于莉莉、白振林主编并统稿,于莉莉、白振林和平金珍共同编写了第1~6章的内容,李新强、王伟锋和许伟共同编写了第7、8章的内容和配套习题。云捷软件科技有限公司总经理袁永强先生对本书的内容及案例提出了宝贵的修改意见。

百密难免一疏,若各位读者在阅读过程中发现任何问题、有任何疑问,都可以随时提出,我们定当虚心接受,尽力解决。

编 者

2014年6月



目 录

第 1 章 初识数据结构	1
1.1 学籍档案管理系统的数据库组织——什么是数据库结构	2
1.1.1 任务说明	2
1.1.2 什么是数据库结构	2
1.1.3 数据库结构的常用术语	2
1.1.4 任务分析	3
1.1.5 小结	3
1.2 五子棋人机对弈系统——数据库结构的用途	4
1.2.1 任务说明	4
1.2.2 学习数据库结构的意义	4
1.2.3 数据库结构的作用	4
1.2.4 任务分析	5
1.2.5 小结	6
1.3 N 个对象全排列——数据库结构与算法	6
1.3.1 任务说明	6
1.3.2 算法的含义	6
1.3.3 算法的设计要求	6
1.3.4 算法效率的度量	7
1.3.5 任务分析	8
1.3.6 小结	8
1.4 巩固与拓展	8
第 2 章 线性表	10
2.1 猴子选大王——线性表的顺序存储	10
2.1.1 任务说明	10
2.1.2 线性表的顺序存储	10
2.1.3 任务分析	16
2.1.4 任务实现	16
2.1.5 小结	19

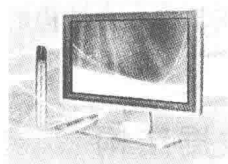
2.2	一元多项式加法运算——线性表的链式存储	20
2.2.1	任务说明	20
2.2.2	单链表	20
2.2.3	单循环链表	29
2.2.4	双向链表	32
2.2.5	循环双向链表	36
2.2.6	任务分析	37
2.2.7	任务实现	38
2.2.8	小结	42
2.3	巩固与拓展	43
第3章	栈和队列	46
3.1	分隔符匹配——顺序栈	47
3.1.1	任务说明	47
3.1.2	顺序栈	47
3.1.3	任务分析	51
3.1.4	任务实现	51
3.1.5	小结	53
3.2	表达式求值——链栈	53
3.2.1	任务说明	53
3.2.2	链栈	53
3.2.3	任务分析	56
3.2.4	任务实现	57
3.2.5	小结	60
3.3	舞伴问题——顺序队列	60
3.3.1	任务说明	60
3.3.2	顺序队列	61
3.3.3	任务分析	66
3.3.4	任务实现	66
3.3.5	小结	69
3.4	停车场管理问题——链队列	69
3.4.1	任务说明	69
3.4.2	链队列	70
3.4.3	任务分析	72
3.4.4	任务实现	72
3.4.5	小结	75
3.5	巩固与拓展	75

第 4 章 串和数组	78
4.1 文本加密器——串	78
4.1.1 任务说明	78
4.1.2 串的定义及相关术语	79
4.1.3 串的存储	79
4.1.4 串的基本操作	80
4.1.5 串的模式匹配	83
4.1.6 任务分析	87
4.1.7 任务实现	87
4.1.8 小结	89
4.2 求解 n 阶魔方——数组	90
4.2.1 任务说明	90
4.2.2 数组的基本概念	90
4.2.3 数组的顺序存储	90
4.2.4 特殊矩阵的压缩存储	91
4.2.5 任务分析	96
4.2.6 任务实现	96
4.2.7 小结	97
4.3 巩固与拓展	97
第 5 章 树和二叉树	100
5.1 随机生成树——树	101
5.1.1 任务说明	101
5.1.2 树的定义	101
5.1.3 树的相关术语	102
5.1.4 树的表示方法	102
5.1.5 树的存储结构	103
5.1.6 任务分析	104
5.1.7 任务实现	104
5.1.8 小结	106
5.2 哈夫曼编码器——二叉树	106
5.2.1 任务说明	107
5.2.2 二叉树的定义	107
5.2.3 二叉树的性质	109
5.2.4 二叉树的存储	110
5.2.5 二叉树的遍历	112
5.2.6 哈夫曼树及哈夫曼编码	113

5.2.7	任务分析	115
5.2.8	任务实现	116
5.2.9	小结	117
5.3	二叉树管理器——树和森林、二叉树的相互转换	117
5.3.1	任务说明	118
5.3.2	树、森林和二叉树的相互转换	118
5.3.3	树和森林的遍历	119
5.3.4	任务分析	120
5.3.5	任务实现	120
5.3.6	小结	126
5.4	巩固与拓展	126
第6章	图	129
6.1	社区超市选址——图	130
6.1.1	任务说明	130
6.1.2	图的基本概念	130
6.1.3	图的存储	131
6.1.4	图的遍历	137
6.1.5	最短路径	139
6.1.6	任务分析	142
6.1.7	任务实现	142
6.1.8	小结	143
6.2	局域网络布线——最小生成树	143
6.2.1	任务说明	143
6.2.2	最小生成树的概念	144
6.2.3	用克鲁斯卡尔算法求解最小生成树	144
6.2.4	用普里姆算法求解最小生成树	145
6.2.5	任务分析	145
6.2.6	任务实现	145
6.2.7	小结	147
6.3	教学计划的编制——拓扑排序	147
6.3.1	任务说明	147
6.3.2	拓扑排序	147
6.3.3	任务分析	148
6.3.4	任务实现	149
6.3.5	小结	151
6.4	求解工程的关键路径——关键路径	151
6.4.1	任务说明	151

6.4.2	关键路径	151
6.4.3	任务分析	153
6.4.4	任务实现	153
6.4.5	小结	160
6.5	巩固与拓展	160
第 7 章	查找	163
7.1	顺序查找	164
7.1.1	算法说明	164
7.1.2	算法实现	165
7.2	折半查找	166
7.2.1	算法说明	166
7.2.2	算法实现	168
7.3	二叉排序树	168
7.3.1	算法说明	168
7.3.2	算法实现	174
7.4	哈希表	175
7.4.1	算法说明	175
7.4.2	算法实现	178
7.5	学生成绩分段查找——查找算法应用	180
7.5.1	任务说明	180
7.5.2	任务分析	180
7.5.3	任务实现	180
7.5.4	小结	181
7.6	巩固与拓展	182
第 8 章	排序	184
8.1	直接插入排序	188
8.1.1	算法说明	188
8.1.2	算法实现	188
8.2	简单选择排序	190
8.2.1	算法说明	190
8.2.2	算法实现	190
8.3	冒泡排序	191
8.3.1	算法说明	191
8.3.2	算法实现	192
8.4	快速顺序	192
8.4.1	算法说明	192

8.4.2 算法实现	193
8.5 归并排序	194
8.5.1 算法说明	194
8.5.2 算法实现	194
8.6 学生信息按照成绩关键字排序——排序算法应用	195
8.6.1 任务说明	195
8.6.2 任务分析	195
8.6.3 任务实现	195
8.6.4 小结	199
8.7 巩固与拓展	199
参考文献	201



第 1 章 初识数据结构

学习目标：

1. 掌握数据结构的定义及相关术语；
2. 理解学习数据结构的意义；
3. 理解数据结构在程序设计中的作用；
4. 了解算法的度量方法；
5. 会应用数据结构相关概念分析实际项目中的数据结构选取问题。

学习效果：

知 识 内 容		掌 握 程 度	存 在 的 疑 问
数据结构基础	什么是数据结构		
	数据结构的常用术语		
数据结构的用途	学习数据结构的意义		
	数据结构的作用		
数据结构与算法	算法的含义		
	算法的设计要求		
	算法效率的度量		

“数据结构”是计算机及相关专业的一门基础课程，理论性较强。在开始学习之前，要明确以下几个问题。

第一，数据结构的主要内容。以线性表(包括顺序表和链表)、栈、队列、串、权组、树、二叉树、图等几种数据结构的存储特点和操作方法为主线，将算法应用到编程实践之中，以解决各种实际问题，如一元多项式加法、停车场管理、哈夫曼编码器、社区超市选址等。

第二，数据结构的学习方法。首先，复习 Java 语言基础，因为本书中的所有算法都是用 Java 语言编写的。其次，养成自主分析项目的习惯，先深入思考，再探索解决问题的办法。再次，通过知识储备的介绍，了解相关概念、术语和算法。最后，通过编程实践，将理论与实践相结合，提高认知能力。整个过程中遇到疑问要主动想办法解决，不要仅仅满足于现状，这样对于知识的理解才能更加透彻，应用起来才会得心应手。

第三，数据结构的应用。“数据结构”课程的特殊性还在于，学习到的知识不像应用软件那样生动形象，也不像编程语言那样可以见到直观的运行结果。数据结构学习的是一

种思维方式,一种解决问题的方法。这种能力是潜移默化应用到编程实践之中的,不能牵强附会。需要学习者通过主动的学习、思考,并且举一反三,才能够真正掌握数据结构的精华所在。

1.1 学籍档案管理系统的数据库组织——什么是数据结构

1.1.1 任务说明

一个简单的学籍档案管理系统需要存储学生的学号、姓名、性别、出生日期、班级等信息,可以对信息进行增加、删除、修改或者查询操作,并且查询条件应该是多样化的,例如,按照班级查询、按照专业名称查询等。这些信息如何组织和存储才能最大限度地发挥系统优势,为使用者提供高效、快捷的服务呢?请根据系统需求,做好系统数据的组织和准备工作。

1.1.2 什么是数据结构

数据结构是在整个计算机科学与技术领域中被广泛使用的术语。它用来反映数据的内部构成,即数据由哪些成分构成,以什么方式构成,呈现什么样的结构。因此,可以将数据结构定义为相互之间存在一种或者多种特定关系的数据元素的集合。

数据的逻辑结构反映数据之间的逻辑关系,也就是数据的前驱和后继关系,这种关系与数据在计算机中的存储位置无关。逻辑结构包括集合结构、线性结构、树形结构和图形结构,如图 1.1 所示。

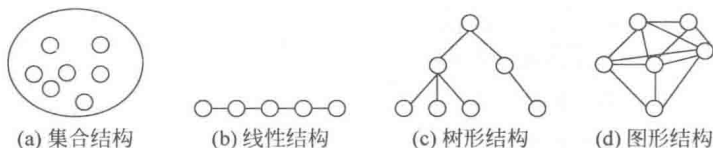


图 1.1 数据的逻辑结构

数据的物理结构反映数据在计算机内的存储方式,又称为存储结构。常见的有顺序存储结构、链式存储结构、索引存储结构和散列存储结构。

1.1.3 数据结构的常用术语

1. 数据

数据(Data)是外部世界信息的载体,是能够被计算机识别、加工、存储的符号的集合,在现实生活中可以比喻为人们加工某种产品的原材料。常见的数据分为数值数据和非数值数据。计算机中的数值数据包括整数、实数等,非数值数据包括字符、图片、影音资料等。

2. 数据元素

数据元素(Data Element)是数据的基本单位,在计算机处理的过程中通常作为一个整体来处理。

3. 数据项

数据项(Data Item)是一个数据元素中的一部分。也就是说,一个数据元素通常由一

个或多个数据项组成。例如,在存储学生学籍档案的数据表(见表 1.1)中,每行数据就是一个数据元素,其中学号、姓名等字段就是数据项。

表 1.1 学生学籍档案数据表

学 号	姓 名	性 别	出 生 日 期	班 级	...
20121160101	白谷京	男	1990.04.01	1 班	...
20121160102	朱芭杰	男	1990.06.01	1 班	...
20121160103	沙禾裳	女	1990.12.25	1 班	...
⋮	⋮	⋮	⋮	⋮	⋮

4. 数据对象

数据对象(Data Object)是性质相同的数据元素的集合,是数据的一个子集。例如,字符串集合{"aa","bb","cc","dd"}就是一个数据对象。

5. 数据类型

数据类型(Data Type)是一组性质相同的值的集合,以及定义在这个值上的一组操作的总称。数据类型与数据结构的关系是:数据结构是数据类型的抽象,数据类型是数据结构在计算机内部的具体表现。

1.1.4 任务分析

学籍档案管理系统的数据组织直接影响系统的性能。如果只是存储数据,那么将每个数据元素依次存储在顺序表中即可,顺序表可以按照存储位置方便、快捷地实现查询。如果要实现任务说明中提出的增加、删除、修改信息等功能,使用链表的效率会比较高,因为在链表上插入删除数据的时候,不需要像顺序表那样大量地移动元素,详细内容会在后续项目中进行说明。当按照不同条件查询数据时,就要想办法提高检索的效率,这种情况下采用树形结构是比较好的选择。

1.1.5 小结

数据结构主要研究 3 个方面的内容,即数据的逻辑结构、数据的物理(存储)结构和对数据的(或算法)运算。通常,算法的设计取决于数据的逻辑结构,算法的实现取决于数据的物理(存储)结构,具体如图 1.2 所示。

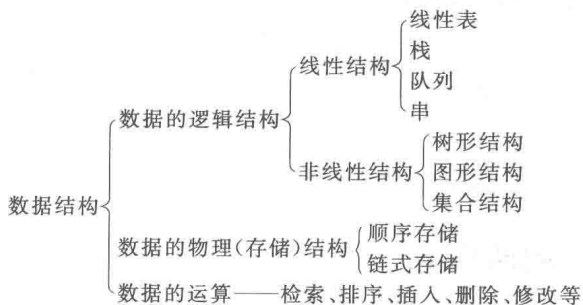


图 1.2 数据结构的研究内容

对于数据结构的定义及相关术语不需要死记硬背,重要的是理解。在理解相关概念的基础上可以尝试对一些问题进行初步的分析,将理论知识与解决实际问题紧密结合起来。例如,计划到几个城市旅游观光的时候,怎样设计一条最短路径,可以又省时间又省钱呢?为什么 Office 办公软件中撤销的功能能够把最近的一次操作恢复回来?大规模球类比赛中的赛制是如何安排的?多思考这些贴近人们现实生活的实例,对于学习数据结构会有很大的帮助。

1.2 五子棋人机对弈系统——数据结构的用途

1.2.1 任务说明

在人机对弈系统中,机器会根据人的每一步走子方案,按照某种算法进行分析和判断,最终得到对自己最有利的方案并且执行。尝试模拟这个过程,并且画出相应的数据结构示意图。

1.2.2 学习数据结构的意义

“数据结构”在计算机科学中是一门综合性的专业基础课。它不仅研究计算机的硬件(特别是编码理论、存储装置和存取方法等),而且和计算机软件有着更密切的关系,无论是编译程序还是操作系统,都涉及数据元素在存储器中的组织问题。在研究信息检索时也必须考虑如何组织数据,这样查找和存取数据元素会更为方便。因此,可以认为数据结构是介于数学、计算机硬件和计算机软件三者之间的一门核心课程。在计算机科学中,数据结构不仅是一般程序设计(特别是非数值计算的程序设计)的基础,而且是设计和实现编译程序、操作系统、数据库系统及其他系统程序和大型应用程序的重要基础。要想有效地使用计算机、充分发挥计算机的性能,还必须学习和掌握好数据结构的有关知识,目的是了解计算机处理对象的特性,通过算法描述出问题的解决方案,并最终交给计算机进行处理。具体如图 1.3 所示。

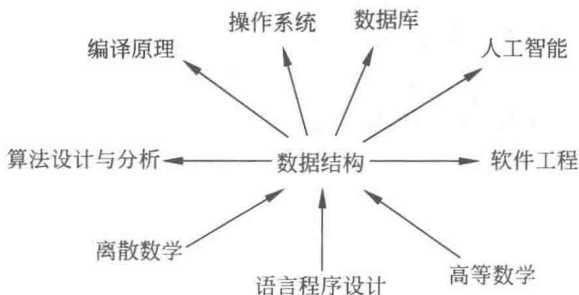


图 1.3 数据结构与其他课程的关系

1.2.3 数据结构的作用

在计算机发展的初期,人们使用计算机的目的主要是处理数值计算问题。当人们使

用计算机来解决一个具体问题时,一般需要经过下列几个步骤:首先要从该具体问题抽象出一个适当的数学模型,然后设计或选择一个关于此数学模型的算法,最后编出程序进行调试、测试,直至得到最终的解答。由于早期所涉及的运算对象是简单的整型、实型或布尔型数据,所以程序设计者的主要精力都集中于程序设计的技巧上,而没有重视数据结构。随着计算机应用领域的扩大和软、硬件的发展,非数值计算问题显得越来越重要。据统计,当今处理非数值计算性问题占用了85%以上的机器时间。这类问题涉及的数据结构更为复杂,数据元素之间的相互关系一般无法用简单的数学方程式加以描述。因此,解决这类问题的关键不再是数学分析和计算方法,而是要选择合适的数据结构,设计相应的算法,才能有效地解决问题。

举个简单的例子,要管理图书馆中众多的书籍,最直接的方式是按照书籍入馆的顺序依次摆放在书架上,并记录书籍的信息和摆放位置,但是如果有人来借书就会遇到麻烦了,因为书籍的摆放是完全随机的,非常不方便查找。也就是说,书籍这种数据没有经过必要的组织和处理。为了方便读者借书,就想到将书籍进行分类摆放来解决上述问题。但是按照什么规则来分类也是很重要的,如按照书名、按照出版社、按照首字母、按照内容类别等。很明显,目前绝大多数图书馆采用的都是按照书籍内容类别来组织的,因为这样可以帮助读者即使在不知道书名、出版社等信息的情况下也能最快地检索到自己想要的图书。

由以上例子可以看出,描述非数值计算问题的数学模型不再是数学方程,而是诸如线性表、树、图之类的数据结构。因此,数据结构课程的主要作用是研究非数值计算的程序设计问题中出现的计算机操作对象及它们之间的关系。

1.2.4 任务分析

人工智能是一门综合性很强的边缘科学。计算机人机对弈是人工智能的一个重要分支,对于五子棋人机对弈系统中机器决策算法的设计,直接影响着对弈的结果。结合图1.4来简单分析,在图中第一层的基础上,白子的走子位置有若干选择,于是形成了图中的第二层,从运筹的角度,针对白子的每一种选择,衍生出黑子的各种对策,即图中的第

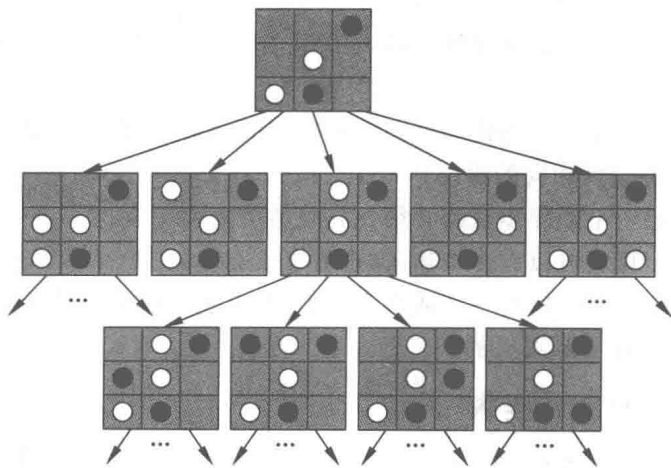


图 1.4 五子棋人机对弈决策树

三层,以此类推,计算机将创建一个表格来计算每种走法的分值,最后在所有决策中执行分值最高的。这个过程需要的数据结构就是树形结构。为了使人机对弈的效果更好,可以穷举决策树中的所有方案供机器选择,正好利用了树形结构一对多的特点。

1.2.5 小结

数据结构作为一门独立的学科,在计算机相关的知识体系中起着至关重要的作用。因此有这样一种说法——程序=数据结构+算法,可以毫不夸张地说没有数据结构支撑的程序就像没有灵魂的人一样苍白和无力。

通过数据结构的学习应该养成“遇见问题先列举解决方法,经过分析选择最佳方案,编写程序交给机器执行”的习惯。对于身边司空见惯、习以为常的问题要多思考、多分析、多比较,用“计算机”的思维方式去处理问题。

1.3 N 个对象全排列——数据结构与算法

1.3.1 任务说明

全排列的生成算法就是对于给定的字符集,用有效的方法将所有可能的全排列无重复、无遗漏地列举出来,本节要求设计出 3 个数字全排列的实现方法。

1.3.2 算法的含义

广义地说,算法就是为解决问题而采取的步骤和方法。在程序设计中,算法是在有限步骤内求解某一问题所使用的一组定义明确的指令序列,通俗点说,就是计算机解题的过程。每条指令表示一个或多个操作。在这个过程中,无论是形成解题思路还是编写程序,都是在实施某种算法,前者是算法实现的逻辑推理,后者是算法实现的具体操作。为了表示一个算法,可以采用多种不同的形式,如自然语言、传统流程图、结构化流程图、N-S 图、伪代码、计算机语言等。

算法应该具有如下特征。

(1) 有穷性。一个算法必须保证在执行有限步骤后结束,而且每一步都应该在有限时间内完成。

(2) 确定性。算法中每一条指令都必须有明确的含义,不能有歧义。在任何条件下,对于相同的输入只能得到相同的输出。

(3) 可行性。每一个操作步骤都必须在有限的时间内完成。

(4) 输入。一个算法可以有多个输入,也可以没有输入。

(5) 输出。一个算法可以有一个或多个输出,并且这些输出与输入是有特定关系的,没有输出的算法是没有实际意义的。

1.3.3 算法的设计要求

(1) 正确性。至少包含以下 4 个层次。

① 程序不含语法错误。