

高等院校计算机系列教材

数据结构(C++版)

主 编 王艳华 戴小鹏

副主编 贺文华 徐雨明 乐晓波 侯利娟

E m a i



WUHAN UNIVERSITY PRESS

武汉大学出版社

高等院校计算机系列教材

数据结构(C++版)

主 编 王艳华 戴小鹏

副主编 贺文华 徐雨明
乐晓波 侯利娟

参 编 朱雅莉 邓红卫
蒋瀚洋 杨建雄



WUHAN UNIVERSITY PRESS

武汉大学出版社

图书在版编目(CIP)数据

数据结构(C++版)/王艳华,戴小鹏主编. —武汉:武汉大学出版社,
2007.4

(高等院校计算机系列教材)

ISBN 978-7-307-05440-0

I. 数… I. ①王… ②戴… 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 TP312

中国版本图书馆 CIP 数据核字(2007)第 023119 号

责任编辑:黄金文 史新奎 张敏 责任校对:黄添生 版式设计:支笛

出版发行:武汉大学出版社 (430072 武昌 珞珈山)

(电子邮件:wdp4@whu.edu.cn 网址:www.wdp.com.cn)

印刷:武汉中科兴业印务有限公司

开本:787×1092 1/16 印张:14.5 字数:338千字

版次:2007年4月第1版 2007年4月第1次印刷

ISBN 978-7-307-05440-0/TP·238 定价:23.00元

版权所有,不得翻印;凡购买我社的图书,如有缺页、倒页、脱页等质量问题,请与当地图书销售部门联系调换。

高等院校计算机系列教材书目

计算机网络	李勇帆等
数据库系统原理与应用	刘先锋等
C++程序设计教程	刘 宏等
计算机组成与系统结构	陈书开等
计算机操作系统	王志刚等
Delphi 2005 数据库基础教程	徐长梅等
网页制作与网站设计	阳西述等
C 语言程序设计 (公共课)	杨克昌等
汇编语言程序设计	李 浪等
数据结构 (C++版)	王艳华等

高等院校计算机系列教材

编 委 会

主 任：刘 宏，湖南师范大学数学与计算机科学学院计算机系主任，教授

编 委：（以姓氏笔画为序）

王 毅，湘潭大学信息工程学院副院长，副教授

乐晓波，长沙理工大学计算机与通信工程学院计算机科学与技术系主任，教授

许又泉，邵阳学院信息电气工程系

羊四清，湖南人文科技学院计算机系主任，副教授

刘先锋，湖南师范大学数学与计算机科学学院，教授

刘连浩，中南大学信息工程学院计算机系教授

全惠云，湖南师范大学数学与计算机科学学院信息与计算科学系主任，教授

沈 岳，湖南农业大学信息科学技术学院院长，副教授

张小梅，凯里学院数学与计算机科学系副主任，副教授

杨克昌，湖南理工学院计算机与信息工程系教授

何迎生，吉首大学数学与计算机科学学院信息与计算科学系副主任

邱建雄，长沙学院计算机科学与技术系副教授

李勇帆，湖南第一师范学院信息技术系主任，教授

周 昱，吉首大学师范学院数学与计算机科学系副主任

罗新密，湖南商学院计算机与电子工程系副教授

徐雨明，衡阳师范学院计算机科学系副主任，副教授

郭国强，湖南文理学院计算机科学与技术系主任，教授

晏峻峰，湖南中医药大学计算机系副教授

龚德良，湘南学院计算机科学系副主任，副教授

蒋伟进，湖南工业大学计算机科学与技术系副主任，副教授

熊 江，重庆三峡学院数学与计算机科学学院副教授

谭敏生，南华大学计算机学院院长，副教授

戴祖雄，湖南科技大学计算机科学与工程学院

执行编委：黄金文，武汉大学出版社计算机图书事业部主任，副编审

内 容 简 介

本书对常用数据结构进行了系统的介绍,包括线性表、栈、队列、串、数组、树、图等,详细讨论了查找和排序的各种实现方法和算法,阐明了各种数据结构的内在逻辑关系及其在计算机中的存储表示,给出了每种数据结构的运算及算法描述,并进行了初步的算法分析。全书采用 C++ 语言进行数据结构和算法的描述。

本书力求概念清晰,注重理论联系实际。每一章后面均列举了典型应用实例,并配有算法和程序以供教学和实践使用。

本书可作为高等学校计算机类专业或信息类相关专业的本科或专科教材,也可供从事计算机工程与应用工作的科技工作者参考。



前言

数据结构是计算机及其相关专业的一门专业基础课,主要研究数据的逻辑结构、存储结构以及相应运算的算法。数据的逻辑结构概括为线性结构、树形结构和图形结构三大类。数据的存储结构概括为顺序结构、链式结构、散列结构和索引结构四大类,每一类都有各自的优点和缺点,分别适用于不同的应用场合。对数据的基本运算概括为查找、插入、删除、排序和遍历等,当数据的逻辑结构、存储结构及程序设计方法不同时,相应运算的算法也就不同。

本书遵循数据结构课程教学大纲的要求,从内容上分为9章:第1章绪论,介绍了数据结构的主要内容和基本概念、算法与算法分析方法。第2章至第4章介绍了几种常用的线性结构,包括线性表、栈和队列、串、数组和广义表。着重讨论这些数据结构的内在逻辑关系、在计算机中的存储表示以及基于其存储表示之上的基本运算和算法实现。第5章主要介绍了树形结构(树与二叉树),包括树和二叉树的定义、存储表示及其基本操作、二叉树的性质、树和二叉树的遍历、树和二叉树的转换以及线索二叉树和哈夫曼树等。第6章详细介绍了图形结构及其应用,包括图的定义和术语、图的存储结构、图的基本运算与算法和几个最优化问题。第7章和第8章介绍了查找和排序,较全面地介绍了查找和排序的各种常用方法和实现算法,并给出了简单的算法分析。第9章介绍了常用的文件结构。全书对每一种数据结构都给出了数据类型定义或描述,并对其基本运算给出了C++语言描述的实现方法。

本书力求概念清晰,注重理论联系实际。每一章后面均列举了典型应用实例,并配了算法和程序以供教学和实践使用。

本书可作为高等学校计算机类专业或信息类相关专业的本科或专科教材,也可供从事计算机工程与应用工作的科技工作者参考。与本书配套的还有《数据结构(C++版)复习提要与实践指导》,由武汉大学出版社出版,书中包括各章的内容回顾、实验指导和题解,并可作为独立的学习指导手册。

本书的第1章由长沙理工大学乐晓波教授编写,第2章、第3章由湖南农业大学戴小鹏副教授编写,第4章由长沙理工大学侯利娟老师编写,第5章由衡阳师范学院徐雨明副教授、朱雅莉老师编写,第6章由衡阳师范学院邓红卫、蒋瀚洋老师编写,第7章和第8章由长沙理工大学王艳华副教授编写,第9章由湖南人文科技学院贺文华教授编写。长沙理工大学在读研究生杨建雄参与了部分编写工作。本教材最后由王艳华副教授统一定稿。

本书在写作过程中,得到了许多专家的大力支持,参考了大量的文献资料和国内外优秀的教材,在此表示诚挚的谢意!

由于时间仓促和作者水平有限,书中难免有不妥甚至错误之处,恳请同行专家和读者批评指正,使本书在使用中改进和完善。

作者

2007年1月

目 录

第1章 绪 论	1
本章主要知识点.....	1
1.1 引言.....	1
1.2 常用术语和基本概念.....	1
1.3 算法与算法分析.....	4
1.3.1 算法的重要特性.....	4
1.3.2 算法设计的基本要求.....	5
1.3.3 算法的描述方法.....	5
1.3.4 算法分析.....	6
习题 1.....	7
第2章 线性表	8
本章主要知识点.....	8
2.1 线性表的定义及逻辑结构.....	8
2.1.1 线性表的定义.....	8
2.1.2 线性表的逻辑结构特征.....	8
2.1.3 常见线性表的基本操作.....	9
2.2 线性表的顺序存储结构.....	9
2.2.1 线性表的顺序存储结构及特点.....	9
2.2.2 顺序存储的类型定义与运算实现.....	10
2.2.3 算法效率分析.....	14
2.3 线性表的链式存储结构.....	14
2.3.1 线性表的链式存储结构及特点.....	15
2.3.2 链式存储的类型定义及运算实现.....	16
2.3.3 循环链表和双向链表.....	22
2.3.4 静态链表.....	25
2.4 应用举例及分析——多项式抽象数据类型.....	26
2.4.1 多项式表示.....	26
2.4.2 多项式相加.....	27
习题 2.....	29
第3章 栈和队列	31
本章主要知识点.....	31
3.1 栈.....	31



3.1.1 栈的定义及基本操作	31
3.1.2 栈的顺序存储结构	31
3.1.3 栈的链式存储结构	33
3.1.4 栈的应用举例	35
3.2 队列	42
3.2.1 队列的定义及基本操作	42
3.2.2 队列的顺序存储结构	43
3.2.3 队列的链式存储结构	46
习题 3	48
第 4 章 其他线性数据结构	50
本章主要知识点	50
4.1 串	50
4.1.1 串的定义及基本操作	50
4.1.2 串的存储结构	51
4.1.3 串的基本操作的实现	53
4.1.4 串的匹配	55
4.2 多维数组	57
4.2.1 数组的定义	57
4.2.2 数组的顺序表示和实现	58
4.2.3 稀疏矩阵的压缩存储	60
4.2.4 稀疏矩阵的转置算法	64
4.3 广义表	67
4.3.1 广义表的定义	67
4.3.2 广义表的存储结构	68
4.3.3 广义表的递归算法	69
4.4 应用举例及分析	74
习题 4	76
第 5 章 树与二叉树	77
本章主要知识点	77
5.1 树的定义和基本操作	77
5.1.1 树的定义	77
5.1.2 基本术语	78
5.1.3 树的基本操作	80
5.2 二叉树	80
5.2.1 二叉树的定义及基本操作	80
5.2.2 二叉树的性质	83
5.2.3 二叉树的存储结构	84
5.2.4 遍历二叉树	87

5.2.5 线索二叉树	91
5.3 树和森林	96
5.3.1 树的存储结构	96
5.3.2 树、森林与二叉树的转换	99
5.3.3 树和森林的遍历	102
5.4 哈夫曼树和判定树	103
5.4.1 哈夫曼树的定义与构造方法	103
5.4.2 最优树的构造方法	104
5.4.3 哈夫曼编码	104
5.5 应用举例及分析	107
习题 5	114
第 6 章 图	116
本章主要知识点	116
6.1 图的定义和术语	116
6.1.1 图的定义	116
6.1.2 图的相关术语	117
6.1.3 图的基本操作	119
6.2 图的存储结构	119
6.2.1 数组表示法	119
6.2.2 邻接表	122
6.2.3 十字链表	123
6.2.4 邻接多重表	123
6.3 图的遍历	124
6.3.1 深度优先搜索	125
6.3.2 广度优先搜索	126
6.4 图的应用	128
6.4.1 生成树和最小生成树	128
6.4.2 拓扑排序	133
6.4.3 最短路径	135
6.5 应用举例及分析	139
习题 6	145
第 7 章 查 找	148
本章主要知识点	148
7.1 查找表概念	148
7.2 静态查找表	149
7.2.1 顺序查找	149
7.2.2 有序表的查找	150
7.2.3 索引顺序表的查找	153



7.3 动态查找表	154
7.3.1 二叉排序树	154
7.3.2 平衡二叉树	159
7.3.3 B. 树	165
7.3.4 B ⁺ 树	169
7.4 哈希表	170
7.4.1 哈希表和哈希函数	170
7.4.2 哈希函数的构造方法	171
7.4.3 处理冲突的方法	173
7.4.4 哈希表的查找及其分析	175
习题 7	178
第8章 排 序	180
本章主要知识点	180
8.1 基本概念	180
8.2 插入排序	181
8.2.1 直接插入排序	181
8.2.2 其他插入排序	182
8.2.3 表插入排序	183
8.2.4 希尔排序	186
8.3 快速排序	188
8.3.1 快速排序的基本原理	188
8.3.2 快速排序(Quick Sort)	189
8.4 选择排序	191
8.4.1 简单选择排序	192
8.4.2 树形选择排序	192
8.4.3 堆排序	193
8.5 归并排序	196
8.6 基数排序	197
8.6.1 多关键字的排序	197
8.6.2 链式基数排序	198
8.7 各种内部排序方法的比较	200
8.8 外部排序	201
8.8.1 问题的提出	201
8.8.2 外部排序的基本过程	202
习题 8	203
第9章 文 件	205
本章主要知识点	205
9.1 基本概念	205



9.1.1 文件及其类别	205
9.1.2 记录的逻辑结构和物理结构	206
9.1.3 文件的操作(运算)	206
9.1.4 文件的物理结构	207
9.2 顺序文件	207
9.2.1 顺序文件的基本概念	207
9.2.2 顺序文件的查找方法	208
9.3 索引文件	208
9.3.1 索引文件的基本概念	208
9.3.2 ISAM 文件和 VSAM 文件	210
9.4 直接存取文件(散列文件)	211
9.5 多关键字文件	212
9.5.1 多重表文件	212
9.5.2 倒排文件	213
习题 9	214
参考文献	215



第1章 绪 论



本章主要知识点

- 数据结构的常用术语及基本概念
- 集合、线性结构、树形结构、图形结构的逻辑特点
- 抽象数据类型
- 算法、算法描述及算法分析

1.1 引言

自 1946 年计算机诞生以来,计算机技术就以迅猛的速度发展起来。迄今为止,计算机的应用已经不再局限于科学计算,而是全面地深入到人类社会的各个领域。除了科学计算之外,人们更多地利用计算机进行控制、管理及数据处理等非数值计算的数据处理。要利用计算机对诸如字符、表格、图像等各种具有一定结构的数据进行处理,就要研究分析数据之间存在的关系,并考虑如何在计算机中有效地组织这些数据,从而有效地对数据进行各种处理。这就是数据结构这门课程要讨论的核心内容。

本章主要介绍数据结构课程中的一些常用的术语,并介绍数据结构中常见的几种结构,即集合结构、线性结构、树形结构和网状结构(或称图形结构)。同时,对数据结构这门课程学习中将要遇到的一些基本概念作概括性的叙述,这些基本概念将贯穿数据结构课程的整个学习过程。

1.2 常用术语和基本概念

数据(Data)是人们利用文字符号、数字符号以及其他规定的符号对客观现实世界的事物及其活动所作的抽象描述。它是计算机程序加工的“原料”。例如,某班甲同学的姓名为“张三”,乙同学的姓名为“李四”,就是关于班上同学姓名的描述。

表示一个事物的一组数据称为一个**数据元素(Data Element)**,它是数据的基本单位,在计算机中通常作为一个整体来进行考虑和处理。一般情况下,一个数据元素由若干个**数据项(Data Item)**构成。例如,描述一个学生信息的一个数据元素包含该学生的学号、姓名、性别、年龄、籍贯、通讯地址等若干数据项,如表 1-1 所示。

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:描述 N 个学生的有关信息的 N 个数据元素构成了一个数据对象。

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集合上的一组操作的总称。例如,高级语言中用到的整数数据类型,是指某个区间(如 $[-32768, 32767]$)上的整数数值构成的集合及定义在这个集合上的一组操作(加、减、乘、除、乘方等)的总称。



表 1-1

学生信息表

学号	姓名	性别	年龄	籍贯	通讯地址
01	张三	男	21	长沙	长沙市韶山路 129 号
02	李四	男	23	北京	北京市西单 235 号
03	王五	女	22	上海	上海市南京路 19 号
04	赵六	男	24	武汉	武汉市解放路 276 号

抽象数据类型 (Abstract Data Type) 通常是指由用户定义, 用以表示实际应用问题的数据模型, 一般由基本数据类型或其他已定义的抽象数据类型以及定义在该模型上的一组操作数据组成。在 C 或 C++ 语言中, 一般可用 struct 或直接用“类”来定义抽象数据类型。

本书中, 在具体使用 C++ 语言描述某种抽象数据类型之前, 我们将先采用如下 ADT 格式来定义该抽象数据类型:

```
DAT 抽象数据类型名{
    数据对象: <数据对象 D 的定义>
    数据关系: <数据关系的集合 R 的定义>
    基本操作: <基本操作的集合 P 的定义>
} DAT 抽象数据类型名
```

【例 1-1】 下面是一个关于抽象数据类型“三元组”的 DAT 定义:

```
DAT triplet {
    数据对象:  $D=\{e_1, e_2, e_3 | e_1, e_2, e_3 \in \text{ElemSet}(\text{ElemSet 为某数据元素的集合})\}$ 
    数据关系:  $R=\{R_1\}$ 
                $R_1=\{ \langle e_1, e_2 \rangle, \langle e_2, e_3 \rangle \}$ 
```

基本操作:

InitTriplet(&T, v1, v2, v3)

操作结果: 构造了三元组 T, 元素 e_1, e_2, e_3 分别被赋予以参数 v1, v2, v3 的值。

DestroyTriplet(&T)

操作结果: 三元组 T 被消除。

Get(T, i, &e)

初始条件: 三元组 T 已存在, $1 \leq i \leq 3$ 。

操作结果: 用 e 返回 T 的第 i 元的值。

Put(&T, i, e)

初始条件: 三元组 T 已存在, $1 \leq i \leq 3$ 。

操作结果: 改变 T 的第 i 元的值为 e。

IsAscending(T)

初始条件: 三元组 T 已存在。

操作结果: 如果 T 的三个元素按升序排列, 则返回 1, 否则返回 0。

IsDsending(T)

初始条件: 三元组 T 已存在。



操作结果：如果 T 的三个元素按降序排列，则返回 1，否则返回 0。

$\text{Max}(T, \&e)$

初始条件：三元组 T 已存在。

操作结果：用 e 返回 T 的三个元素中的最大值。

$\text{Min}(T, \&e)$

初始条件：三元组 T 已存在。

操作结果：用 e 返回 T 的三个元素中的最小值。

} DAT triplet

数据结构 (Data Structure) 是指相互之间存在一种或多种特定关系的数据元素的集合。具体来说，数据结构包含三个方面的内容，即数据的逻辑结构、数据的存储结构（或称物理结构）和对数据所施加的一组操作。

数据的**逻辑结构**是数据元素之间本身所固有的独立于计算机的一种结构，这种结构可以用数据元素之间固有的关系的集合来描述。

数据的**存储结构**（或**物理结构**）是逻辑结构在计算机存储器中的具体存放方式的体现，是逻辑结构在计算机存储器中的映象。

操作是指对数据对象进行处理的一组运算或处理的总称。操作的定义直接依赖于逻辑结构，但操作的实现必依赖于存储结构。

根据数据元素之间存在的关系的不同特性，数据结构通常可以分为如下 4 类基本结构：

(1) 线性结构

元素之间存在一个一对一的线性关系，即除了第一个元素和最后一个元素外，每个元素都有一个直接前驱和一个直接后继，第一个元素有一个直接后继，最后一个元素有一个直接前驱。例如表 1-1 就给出了一个线性结构的数据对象。线性结构示意图详见图 1-1 所示。

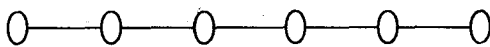


图 1-1 线性结构示意图

(2) 树形结构

数据元素之间存在着一个对多个的关系。例如，老师 T 指导 3 个硕士研究生 G_1, G_2, G_3 ；每个研究生 G_i 又分别指导 3 个本科生 S_{i1}, S_{i2}, S_{i3} ，则数据元素之间呈现树形结构，详见图 1-2 所示。

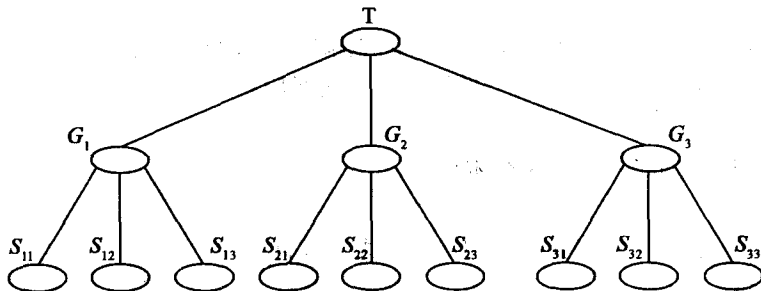


图 1-2 树形结构示意图



(3) 图形结构或网状结构

数据元素之间存在多个对多个的关系。网状结构示意图详见图 1-3 所示。

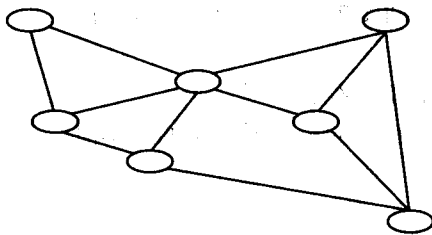


图 1-3 网状结构示意图

(4) 集合结构

数据元素之间无任何关系。集合结构示意图详见图 1-4 所示。

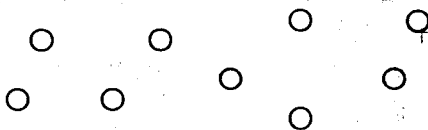


图 1-4 集合结构示意图

1.3 算法与算法分析

算法 (Algorithm) 是对于待定问题求解步骤的一种描述,它是指令的有限序列,其中每一条指令表示一个或多个操作。

算法分析 (Algorithm Analysis) 是指从“时间”和“空间”两个方面来分析算法的效率。

1.3.1 算法的重要特性

算法应具有如下 5 个重要特性:

(1) 输入性

具有 0 个或若干个输入量,这些输入量取自于某个特定的对象的集合。

(2) 输出性

具有一个或多个输出量,且当有一个以上的输入量时,该输出依赖于一组确定的输入量。也即对于某一组确定的输入量,有唯一的输出量与之对应。

(3) 有限性

构成算法的指令的序列一定是有限序列。

(4) 确定性

算法中的每一条指令必须有确切的含义,无二义性。

(5) 可行性

算法中所描述的操作都可以用已有的有限条指令或已实现的有限个基本运算来实现。