

普通高等教育“十一五”国家级规划教材配套教材

《数据结构——C语言描述(第二版)》 学习指导和习题解析

陈慧南 编著



西安电子科技大学出版社
<http://www.xduph.com>

普通高等教育“十一五”国家级规划教材配套教材

《数据结构——C 语言描述(第二版)》 学习指导和习题解析

陈慧南 编著

西安电子科技大学出版社

2009

内 容 简 介

本书是普通高等教育“十一五”国家级规划教材《数据结构——C语言描述(第二版)》(陈慧南编著,西安电子科技大学出版社2009年出版)的教学配套书。本书在编写时考虑了计算机类专业全国硕士研究生入学统一考试的新情况,加入了考研试题解析的内容,因此本书也可作为计算机类专业基础综合考试数据结构部分的考研辅导书。

本书涉及配套教材各章内容,涵盖硕士研究生入学考试大纲的各个知识单元,包括典型题解、考研试题解析和配书习题选解。本书例题和习题均经过精心选择和编写,具有代表性,题型多样,覆盖面广。本书还包括完整的模拟试卷和试卷解析。本书题解简明扼要,深入浅出,易于学习和理解。相信本书对以《数据结构——C语言描述(第二版)》为教材或教学参考书的读者会有很大帮助,也会对本科毕业生参加计算机类专业全国硕士研究生入学统一考试有很好的辅导作用。

图书在版编目(CIP)数据

《数据结构——C语言描述(第二版)》学习指导和习题解析 / 陈慧南编著.

—西安:西安电子科技大学出版社,2009.6

普通高等教育“十一五”国家级规划教材配套教材

ISBN 978-7-5606-2237-8

I. 数… II. 陈… III. ① 数据结构—高等学校—教学参考资料 ② C语言—程序设计—高等学校—教学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 052069 号

策 划 马乐惠

责任编辑 阎 彬 马乐惠

出版发行 西安电子科技大学出版社(西安市太白南路2号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfb001@163.com

经 销 新华书店

印刷单位 陕西华沐印刷科技有限责任公司

版 次 2009年6月第1版 2009年6月第1次印刷

开 本 787毫米×1092毫米 1/16 印 张 13.375

字 数 313千字

印 数 1~4000册

定 价 19.00元

ISBN 978-7-5606-2237-8/TP·1140

XDUP 2529001-1

*** 如有印装问题可调换 ***

本社图书封面为激光防伪覆膜,谨防盗版。

欢迎选购西安电子科技大学出版社教材类图书

~~~~~“十一五”国家级规划教材~~~~~			数控机床故障分析与维修(高职)(第二版)	25.00
计算机系统结构(第四版)(李学干)	25.00		液压与气动技术(第二版)(朱梅)(高职)	23.00
计算机系统安全(第二版)(马建峰)	30.00		~~~~~计算机提高普及类~~~~~	
计算机网络(第三版)(蔡皖东)	27.00		计算机应用基础(第三版)(丁爱萍)(高职)	22.00
计算机应用基础教程(第四版)(陈建铎)			计算机文化基础(高职)(游鑫)	27.00
(for Windows XP/Office XP)	30.00		计算机文化基础上机实训及案例(高职)	15.00
计算机应用基础(冉崇善)(高职)			计算机科学与技术导论(吕辉)	22.00
(Windows XP & Office 2003 版)	23.00		计算机应用基础(高职)(赵钢)	29.00
《计算机应用基础》实践技能训练			计算机应用基础——信息处理技术教程	31.00
与案例分析(高职)(冉崇善)	18.00		《计算机应用基础——信息处理技术教程》	
微型计算机原理(第二版)(王忠民)	27.00		习题集与上机指导(张郭军)	14.00
微型计算机原理及接口技术(第二版)(裘雪红)	36.00		计算机组装与维修(中职)(董小莉)	23.00
微型计算机组成与接口技术(第二版)(高职)	28.00		微型机组装与维护实训教程(高职)(杨文诚)	22.00
微机原理与接口技术(第二版)(龚尚福)	37.00		~~~~~计算机网络类~~~~~	
单片机原理及应用(第二版)(李建忠)	32.00		计算机网络技术基础教程(高职)(董武)	18.00
单片机应用技术(第二版)(高职)(刘守义)	30.00		计算机网络管理(雷震甲)	20.00
Java程序设计(第二版)(高职)(陈圣国)	26.00		网络设备配置与管理(李飞)	23.00
编译原理基础(第二版)(刘坚)	29.00		网络安全与管理实验教程(谢晓燕)	35.00
人工智能技术导论(第三版)(廉师友)	24.00		网络安全技术(高职)(廖兴)	19.00
多媒体软件设计技术(第三版)(陈启安)	23.00		网络信息安全技术(周明全)	17.00
信息系统分析与设计(第二版)(卫红春)	25.00		动态网页设计实用教程(蒋理)	30.00
信息系统分析与设计(第三版)(陈圣国)(高职)	20.00		ASP动态网页制作基础教程(中职)(苏玉雄)	20.00
传感器原理及工程应用(第三版)	28.00		局域网组建实例教程(高职)(尹建璋)	20.00
数字图像处理(第二版)(何东健)	30.00		Windows Server 2003组网技术(高职)(陈伟达)	30.00
电路基础(第三版)(王松林)	39.00		组网技术(中职)(俞海英)	19.00
模拟电子电路及技术基础(第二版)(孙肖子)	35.00		综合布线技术(高职)(王趾成)	18.00
模拟电子技术(第三版)(江晓安)	25.00		计算机网络应用基础(武新华)	28.00
数字电子技术(第三版)(江晓安)	23.00		计算机网络基础及应用(高职)(向隅)	22.00
数字电路与系统设计(第二版)(邓元庆)	35.00		~~~~~计算机技术类~~~~~	
数字信号处理(第三版)(高西全)	29.00		计算机系统结构与组成(吕辉)	26.00
电磁场与电磁波(第二版)(郭辉萍)	28.00		电子商务基础与实务(第二版)(高职)	16.00
现代通信原理与技术(第二版)(张辉)	39.00		数据结构——使用 C++ 语言(第二版)(朱战立)	23.00
移动通信(第四版)(李建东)	30.00		数据结构(高职)(周岳山)	15.00
移动通信(第二版)(章坚武)	24.00		数据结构教程——Java 语言描述(朱振元)	29.00
物理光学与应用光学(第二版)(石顺祥)	42.00		离散数学(武波)	24.00

现代控制理论基础(舒欣梅)	14.00	数控加工与编程(第二版)(高职)(詹华西)	23.00
过程控制系统及工程(杨为民)	25.00	数控加工工艺学(任同)	29.00
控制系统仿真(党宏社)	21.00	数控加工工艺(高职)(赵长旭)	24.00
模糊控制技术(席爱民)	24.00	数控加工工艺课程设计指导书(赵长旭)	12.00
工程电动力学(修订版)(王一平)(研究生)	32.00	数控加工编程与操作(高职)(刘虹)	15.00
工程力学(张光伟)	21.00	数控机床与编程(高职)(饶军)	24.00
工程力学(皮智谋)(高职)	12.00	数控机床电气控制(高职)(姚勇刚)	21.00
理论力学(张功学)	26.00	数控应用专业英语(高职)(黄海)	17.00
材料力学(张功学)	27.00	机床电器与 PLC(高职)(李伟)	14.00
材料成型工艺基础(刘建华)	25.00	电机及拖动基础(高职)(孟宪芳)	17.00
工程材料及应用(汪传生)	31.00	电机与电气控制(高职)(冉文)	23.00
工程材料与应用(戈晓岚)	19.00	电机原理与维修(高职)(解建军)	20.00
工程实践训练(周桂莲)	16.00	供配电技术(高职)(杨洋)	25.00
工程实践训练基础(周桂莲)	18.00	金属切削与机床(高职)(聂建武)	22.00
工程制图(含习题集)(高职)(白福民)	33.00	模具制造技术(高职)(刘航)	24.00
工程制图(含习题集)(周明贵)	36.00	模具设计(高职)(曾霞文)	18.00
工程图学简明教程(含习题集)(尉朝闻)	28.00	冷冲压模具设计(高职)(刘庚武)	21.00
现代设计方法(李思益)	21.00	塑料成型模具设计(高职)(单小根)	37.00
液压与气压传动(刘军营)	34.00	液压传动技术(高职)(简引霞)	23.00
先进制造技术(高职)(孙燕华)	16.00	发动机构造与维修(高职)(王正键)	29.00
机械原理多媒体教学系统(资料)(书配盘)	120.00	机动车辆保险与理赔实务(高职)	23.00
机械工程科技英语(程安宁)	15.00	汽车典型电控系统结构与维修(李美娟)	31.00
机械设计基础(郑甲红)	27.00	汽车机械基础(高职)(娄万军)	29.00
机械设计基础(岳大鑫)	33.00	汽车底盘结构与维修(高职)(张红伟)	28.00
机械设计(王宁侠)	36.00	汽车车身电气设备系统及附属电气设备(高职)	23.00
机械设计基础(张京辉)(高职)	24.00	汽车单片机与车载网络技术(于万海)	20.00
机械基础(安美玲)(高职)	20.00	汽车故障诊断技术(高职)(王秀贞)	19.00
机械 CAD/CAM(葛友华)	20.00	汽车营销技术(高职)(孙华宪)	15.00
机械 CAD/CAM(欧长劲)	21.00	汽车使用性能与检测技术(高职)(郭彬)	22.00
机械 CAD/CAM 上机指导及练习教程(欧)	20.00	汽车电工电子技术(高职)(黄建华)	22.00
画法几何与机械制图(叶琳)	35.00	汽车电气设备与维修(高职)(李春明)	25.00
《画法几何与机械制图》习题集(邱龙辉)	22.00	汽车使用与技术管理(高职)(边伟)	25.00
机械制图(含习题集)(高职)(孙建东)	29.00	汽车空调(高职)(李祥峰)	16.00
机械设备制造技术(高职)(柳青松)	33.00	汽车概论(高职)(邓书涛)	20.00
机械制造基础(高职)(郑广花)	21.00	现代汽车典型电控系统结构原理与故障诊断	25.00

欢迎来函索取本社书目和教材介绍! 通信地址: 西安市太白南路2号 西安电子科技大学出版社发行部  
 邮政编码: 710071 邮购业务电话: (029)88201467 传真电话: (029)88213675。

# 前 言

---

本书是作者编写的普通高等教育“十一五”国家级规划教材《数据结构——C 语言描述(第二版)》(西安电子科技大学出版社 2009 年出版)的教学配套书。本书在编写时考虑了计算机类专业全国硕士研究生入学统一考试的新情况,参照研究生考试大纲,加入了考研试题解析的内容,因此本书也可作为计算机类专业基础综合考试数据结构部分的考研辅导书。

本书涉及配套教材各章内容,涵盖硕士研究生入学考试的各个知识单元。每章首先概述该章讨论的基本内容和要点,然后以典型题解、考研试题解析和配书习题选解三小节进行例题、试题和习题的解析。例题和习题均经过精心选择和编写,考研试题选自清华大学等全国高校考研试卷,具有代表性。这些题的题型丰富,有一定的深度且覆盖面广,包括判断题、选择题、填空题、解答题、算法分析和算法设计题。书中题解简明扼要,深入浅出,易于学习和理解。本书最后一章给出了本科“数据结构”课程期末和研究生入学考试的完整模拟试卷,并作了详细分析和解答。本书在附录中还附有某高校近四年的研究生入学“数据结构”试卷供读者参考。相信这本书会对以《数据结构——C 语言描述(第二版)》为教材或教学参考书的读者有很大帮助,也会对本科毕业生参加计算机类专业全国硕士研究生入学统一考试有很好的辅导作用。

本书的编写得到了南京邮电大学和计算机学院领导的推荐和关心,也得到了西安电子科技大学出版社马乐惠主任的大力推荐和支持,在此对他们表示衷心的感谢。

书中若有不当之处,敬请读者批评指正。

作 者

2008 年 12 月 30 日于南京

# 目 录

第 1 章 概论 .....	1	3.1 内容和要点 .....	26
1.1 内容和要点 .....	1	3.1.1 学习要求 .....	26
1.1.1 课程目的和内容 .....	1	3.1.2 堆栈 ADT .....	26
1.1.2 学习要求 .....	1	3.1.3 堆栈的实现 .....	27
1.1.3 基本概念 .....	2	3.1.4 队列 ADT .....	28
1.1.4 数据抽象和抽象数据类型 .....	2	3.1.5 队列的实现 .....	28
1.1.5 数据结构和算法描述 .....	3	3.1.6 表达式计算 .....	28
1.1.6 算法和算法分析 .....	3	3.1.7 递归 .....	29
1.2 典型题解 .....	4	3.2 典型题解 .....	29
1.2.1 基本题 .....	4	3.2.1 基本题 .....	29
1.2.2 算法分析题 .....	5	3.2.2 算法设计题 .....	31
1.2.3 算法设计题 .....	6	3.3 考研试题解析 .....	33
1.3 考研试题解析 .....	8	3.3.1 基本题 .....	33
1.3.1 基本题 .....	8	3.3.2 算法设计题 .....	35
1.3.2 算法分析题 .....	9	3.4 配书习题选解 .....	36
1.4 配书习题选解 .....	10		
第 2 章 数组和链表 .....	13	第 4 章 线性表和数组 ADT .....	43
2.1 内容和要点 .....	13	4.1 内容和要点 .....	43
2.1.1 学习要求 .....	13	4.1.1 学习要求 .....	43
2.1.2 数组 .....	13	4.1.2 线性表 ADT .....	44
2.1.3 指针和动态存储分配 .....	14	4.1.3 线性表的顺序表示 .....	44
2.1.4 单链表 .....	15	4.1.4 线性表的链接表示 .....	44
2.1.5 其他链表 .....	16	4.1.5 两种存储表示的比较 .....	45
2.2 典型题解 .....	16	4.1.6 多项式的算术运算 .....	45
2.2.1 基本题 .....	16	4.1.7 数组作为抽象数据类型 .....	46
2.2.2 算法设计题 .....	18	4.1.8 对称矩阵 .....	46
2.3 考研试题解析 .....	20	4.1.9 稀疏矩阵 .....	46
2.3.1 基本题 .....	20	4.2 典型题解 .....	47
2.3.2 算法设计题 .....	21	4.2.1 基本题 .....	47
2.4 配书习题选解 .....	22	4.2.2 算法设计题 .....	49
		4.3 考研试题解析 .....	50
第 3 章 堆栈和队列 .....	26	4.3.1 基本题 .....	50
		4.3.2 算法设计题 .....	52

4.4 配书习题选解 .....	53	7.1.2 基本概念 .....	89
<b>第5章 字符串和广义表</b> .....	59	7.1.3 动态集 ADT .....	89
5.1 内容和要点 .....	59	7.1.4 集合的表示 .....	89
5.1.1 学习要求 .....	59	7.1.5 顺序搜索 .....	89
5.1.2 字符串 .....	59	7.1.6 二分搜索和对半搜索 .....	90
5.1.3 模式匹配 .....	60	7.1.7 二叉判定树 .....	90
5.1.4 广义表 .....	60	7.2 典型题解 .....	90
5.2 典型题解 .....	61	7.2.1 基本题 .....	90
5.2.1 基本题 .....	61	7.2.2 算法设计题 .....	92
5.2.2 算法设计题 .....	63	7.3 考研试题解析 .....	94
5.3 考研试题解析 .....	64	7.3.1 基本题 .....	94
5.3.1 基本题 .....	64	7.3.2 算法设计题 .....	95
5.3.2 算法设计题 .....	66	7.4 配书习题选解 .....	96
5.4 配书习题选解 .....	66	<b>第8章 搜索树</b> .....	99
<b>第6章 树</b> .....	70	8.1 内容和要点 .....	99
6.1 内容和要点 .....	70	8.1.1 学习要求 .....	99
6.1.1 学习要求 .....	70	8.1.2 二叉搜索树 .....	99
6.1.2 树的定义和基本术语 .....	71	8.1.3 二叉搜索树的搜索 .....	100
6.1.3 二叉树的定义和性质 .....	71	8.1.4 二叉平衡树 .....	100
6.1.4 二叉树 ADT 和二叉链表 .....	72	8.1.5 B 树 .....	101
6.1.5 二叉树遍历的递归算法 .....	72	8.2 典型题解 .....	102
6.1.6 二叉线索树 .....	73	8.2.1 基本题 .....	102
6.1.7 树和森林 .....	74	8.2.2 算法设计题 .....	106
6.1.8 堆和优先权队列 .....	74	8.3 考研试题解析 .....	107
6.1.9 哈夫曼树和哈夫曼编码 .....	74	8.3.1 基本题 .....	107
6.1.10 并查集和等价关系 .....	75	8.3.2 算法设计题 .....	110
6.2 典型题解 .....	75	8.4 配书习题选解 .....	111
6.2.1 基本题 .....	75	<b>第9章 散列表</b> .....	115
6.2.2 算法设计题 .....	77	9.1 内容和要点 .....	115
6.3 考研试题解析 .....	78	9.1.1 学习要求 .....	115
6.3.1 基本题 .....	78	9.1.2 字典 .....	115
6.3.2 算法设计题 .....	80	9.1.3 散列表 .....	115
6.4 配书习题选解 .....	81	9.1.4 拉链法 .....	116
<b>第7章 集合与搜索</b> .....	88	9.1.5 开地址法 .....	116
7.1 内容和要点 .....	88	9.2 典型题解 .....	117
7.1.1 学习要求 .....	88	9.2.1 基本题 .....	117
		9.2.2 算法设计题 .....	119

9.3 考研试题解析 .....	120	11.3.1 基本题 .....	155
9.3.1 基本题 .....	120	11.3.2 算法设计题 .....	157
9.3.2 算法设计题 .....	123	11.4 配书习题选解 .....	159
9.4 配书习题选解 .....	124		
<b>第 10 章 图</b> .....	126	<b>第 12 章 文件和外排序</b> .....	165
10.1 内容和要点 .....	126	12.1 内容和要点 .....	165
10.1.1 学习要求 .....	126	12.1.1 学习要求 .....	165
10.1.2 图的基本概念 .....	126	12.1.2 文件 .....	165
10.1.3 图 ADT .....	127	12.1.3 外排序 .....	166
10.1.4 图的存储结构 .....	128	12.2 典型题解 .....	167
10.1.5 图的遍历 .....	129	12.3 考研试题解析 .....	170
10.1.6 拓扑排序和关键路径 .....	129	12.4 配书习题选解 .....	172
10.1.7 最小代价生成树 .....	130		
10.1.8 最短路径 .....	130	<b>第 13 章 模拟试卷及解析</b> .....	174
10.2 典型题解 .....	131	13.1 本科“数据结构”期末模拟试卷 .....	174
10.2.1 基本题 .....	131	13.2 本科“数据结构”期末模拟 试卷解析 .....	178
10.2.2 算法设计题 .....	134	13.3 攻读硕士学位研究生入学考试 “数据结构”模拟试卷 .....	182
10.3 考研试题解析 .....	135	13.4 攻读硕士学位研究生入学考试 “数据结构”模拟试卷解析 .....	186
10.3.1 基本题 .....	135		
10.3.2 算法设计题 .....	138		
10.4 配书习题选解 .....	140	<b>附录 某高校研究生入学“数据结构” 试卷</b> .....	192
<b>第 11 章 内排序</b> .....	149	附录 A 2005 年攻读硕士学位研究生入学 考试试卷 .....	192
11.1 内容和要点 .....	149	附录 B 2006 年攻读硕士学位研究生入学 考试试卷 .....	195
11.1.1 学习要求 .....	149	附录 C 2007 年攻读硕士学位研究生入学 考试试卷 .....	199
11.1.2 基本概念 .....	149	附录 D 2008 年攻读硕士学位研究生入学 考试试卷 .....	202
11.1.3 排序使用的顺序表和链表结构 .....	150		
11.1.4 排序算法 .....	150	<b>参考文献</b> .....	206
11.1.5 算法分析 .....	151		
11.2 典型题解 .....	152		
11.2.1 基本题 .....	152		
11.2.2 算法设计题 .....	154		
11.3 考研试题解析 .....	155		

## 第1章 概 论

通过本章学习,读者可了解“数据结构”课程的学习目的和内容,并深刻理解有关数据结构的基本概念,理解将抽象数据类型应用于数据结构研究的方法,掌握算法分析的基本方法。

### 1.1 内容和要点^①

#### 1.1.1 课程目的和内容

“数据结构”是计算机科学与技术以及相关专业的学科基础课,是计算机软件设计的重要理论和实践基础。通过本课程的学习,学生应掌握计算机科学中的组织、表示和处理数据的基本方法,数据的逻辑结构、存储结构,以及各种基本运算及其实现;掌握算法设计和分析的基本方法,培养和增强运用数据结构和算法知识解决应用问题的能力。

本课程讨论线性表、堆栈和队列、数组和字符串、二叉树和树、搜索树、散列表、图和文件等常见的数据结构,还要讨论各种典型的搜索和内、外排序算法以及递归算法和算法分析的基本方法。

#### 1.1.2 学习要求

##### 1. 基本要求

- (1) 明确“数据结构”课程的学习目的和内容。
- (2) 掌握算法和数据结构的基本概念。
- (3) 理解抽象数据类型概念及其在数据结构研究中的应用。
- (4) 掌握算法分析的基本方法。

##### 2. 重点

- (1) 数据结构的基本概念。
- (2) 数据结构和算法的描述方法。

^① 请读者注意本书各章的脚注。由于各种数据结构教材在概念定义、算法描述等方面不尽相同,本书在每章的“内容和要点”一节中对此简明扼要地作了必要介绍,目的是使读者在阅读习题解答时不会误解。也希望读者在遇到疑惑时,查阅本书相关章节中关于数据结构知识的阐述,注意其与读者手边的参考书中相关知识的差异。本章内容源自作者编写的《数据结构——C语言描述(第二版)》一书第1章。

(3) 算法分析的基本方法。

### 3. 难点

(1) 数据抽象和抽象数据类型。

(2) 算法的时间复杂度和渐近复杂度表示。

## 1.1.3 基本概念

所谓**数据**,就是计算机加工处理的对象。一个数据可以由若干成分数据构成,并具有某种结构,我们称组成数据的成分数据为**数据元素**。数据一般分为两类:**数值数据**和**非数值数据**。一个**数据结构**是由数据元素依据某种逻辑联系组织起来的。对数据元素间的逻辑关系的描述称为数据的**逻辑结构**;数据必须在计算机内存储,数据的**存储结构**是数据结构的实现形式。研究数据结构是为了解决应用问题,讨论一个数据结构必须同时讨论在该数据结构上执行的相关**运算**及其**算法**才有意义。因此数据结构的研究包括三个方面:数据的逻辑结构、数据的存储结构和数据结构上的运算。

根据数据结构中数据元素之间的结构关系的不同特征,可形成四类基本的逻辑结构:**集合结构**、**线性结构**、**树形结构**和**图状结构**。集合结构的元素间除了同属于一个数据结构的联系外,没有别的关系;线性结构的元素间形成一对一的关系;树形结构的元素间存在一对多的关系;图状结构的元素间存在多对多的关系。

数据的逻辑结构是面向应用问题的,是从用户角度看到的数据结构;数据的存储结构是逻辑数据的存储映像,它是面向计算机的。四种常见的存储表示方式为顺序结构、链接结构、索引结构和散列结构。

**顺序结构**和**链接结构**是两种最基本的存储表示方法。在顺序存储表示下,数据元素间的逻辑关系由元素在计算机存储器内的相对位置表示,一般需要一块连续的存储空间;在链接存储表示下,存储一个元素的存储块由元素信息以及指向该元素逻辑上相关元素的结点的指针组成,元素间的逻辑关系表现在指针的链接联系上。**索引结构**是存储所有元素信息的数据文件的一个纲要,索引表的每个索引项只包含元素的关键字和相关元素的存储地址两部分信息。**散列结构**借助于散列函数建立关键字和存储地址的对应关系。

## 1.1.4 数据抽象和抽象数据类型

**数据抽象**使程序设计者可以将数据元素间的逻辑关系和数据在计算机内的具体表示分开考虑。而**过程抽象**可使程序设计者将一个运算的定义与实现运算的具体方法分开考虑。抽象的好处主要在于降低了问题求解的难度,也有助于提高程序的可靠性。**封装**是指把数据和操纵数据的运算组合成模块的机制。**信息隐蔽**是指对模块使用者隐藏了模块内部的数据结构和程序实现细节。采用封装和信息隐蔽原则来设计的模块被称为**黑盒子**。

一个**数据类型**定义一个值的集合以及作用于该值集的运算集合。一个**抽象数据类型(ADT)**是一个数据类型,其主要特征是数据对象及其运算的规范独立于它们的实现,实行封装和信息隐蔽,从而使 ADT 的使用与实现分离。

### 1.1.5 数据结构和算法描述

逻辑结构和运算的定义组成了数据结构的规范，而数据的存储表示和运算算法的描述则构成了数据结构的实现。规范是实现的准则和依据，规范指明“做什么”，而实现解决“怎样做”。本书从规范和实现两方面来描述一个数据结构，这体现了抽象数据类型的原则。本书中，数据的存储结构用 C 语言的类型描述，并使用 C 语言函数来表示数据结构运算的算法。

由于在多数情况下，数据元素的类型对于数据结构的讨论没有本质影响，不妨假定为类型 T。T 代表一种通用数据类型，它可以在实际应用时以 typedef 定义为具体的数据类型。例如，语句 typedef int T 将 T 定义为整型，typedef void* T 将 T 定义为 void 指针类型，该类型指针可指向不同具体类型的元素。在 C++ 语言环境下，T 可以用 C++ 语言的模板类方便地实现。

### 1.1.6 算法和算法分析

一个算法是对特定问题的求解步骤的一种描述，它是指令的有限序列。此外，算法具有下列五个特征：

- (1) 输入。算法有零个或多个输入。
- (2) 输出。算法至少产生一个输出。
- (3) 确定性。算法的每一条指令都有确切的定义，没有二义性。
- (4) 能行性。算法的每一条指令都足够基本，它们可以通过已经实现的基本运算执行有限次来实现。
- (5) 有穷性。算法必须在执行有限步之后终止。

衡量一个算法的性能，主要有以下几个标准：

- (1) 正确性。算法的执行结果应当满足预先规定的功能和性能要求。
- (2) 简明性。一个算法应当思路清晰、层次分明、简单明了、易读易懂。
- (3) 健壮性。当输入不合法数据时，算法应能作适当处理，不至于引起严重后果。
- (4) 效率。算法应有效使用存储空间，并有高的时间效率。

一个程序的时间复杂度是程序运行从开始到结束所需的时间。程序的一次运行是针对所求解的问题的某一特定实例而言的。判断算法性能要考虑的一个基本特征是问题实例的规模。一个程序步是指在语法上或语义上有意义的程序段，该程序段的执行时间与问题实例的特征无关。程序步数虽然并不能确切反映程序运行的实际时间，但程序步数在数量级上反映了一个程序的执行时间，依据它可对算法实施事前分析。程序的运行时间可通过输入选择的数据，实际运行程序来获取，这被称为算法的事后测试。

定义(大 O 记号): 设  $f(n)$  和  $g(n)$  是定义在正整数上的正函数，如果存在两个正常数  $c$  和  $n_0$ ，使得当  $n \geq n_0$  时，有  $f(n) \leq c g(n)$ ，则记作  $f(n) = O(g(n))$ 。

使用大 O 记号表示的算法的时间复杂度，称为算法的渐近时间复杂度。渐近时间复杂度也常简称为时间复杂度。通常用  $O(1)$  表示常数计算时间，即算法只需执行有限个程序步。算法的时间复杂度分为三种情况：最好情况、平均情况和最坏情况。

一个程序的空间复杂度是程序运行从开始到结束所需的存储量。应当注意的是，空间复杂度一般按最坏情况分析。空间复杂度也使用大 O 记号表示。

## 1.2 典型题解

### 1.2.1 基本题

#### 例 1-1 判断题

- (1) 数据的逻辑结构是数据结构在计算机中的表示。( )
- (2) C语言提供了实现抽象数据类型的机制。( )
- (3) 一个C语言程序是用C语言描述的一个算法。( )
- (4) 散列结构是一种特殊的存储结构方式。( )
- (5) 算法分析通常是指对算法实施事前分析。( )

#### 【解答】

- (1) 否。数据的逻辑结构是从用户角度看到的数据的结构形式。
- (2) 否。C语言并没有提供实现抽象数据类型的封装和信息隐蔽的机制，但这并不意味着当使用C语言描述数据结构时，不能运用抽象数据类型的原则。
- (3) 否。有穷性是算法的五个特征之一，对程序无此限制。一个算法使用程序设计语言描述便成为程序，反之不然。
- (4) 是。散列结构是四种基本的存储方式之一。
- (5) 是。算法分析是指对算法进行的时空效率分析，它并不要求写成实际的计算机程序和在具体计算机上运行。

#### 例 1-2 选择题

- (1) 一个数据结构DS可以用一个二元组 $DS=(D, R)$ 表示，其中，D是( )的有限集合，R是D中的( )集合。
 

(A) 数据元素，操作	(B) 结点，运算
(C) 数据元素，元素序偶	(D) 结点，结点关系
- (2) 算法的健壮性是指( )。
  - (A) 当输入不合法数据时，程序会采取应急措施
  - (B) 算法是正确的
  - (C) 程序不会被病毒所染
  - (D) 算法执行不会陷入无限循环
- (3) 下列表示中正确的是( )。
 

(A) $2n+O(n)=O(n^2)$	(B) $O(n)=3n$
(C) $3n^2+n\log_2 n=O(n\log n)$ ^①	(D) $6=O(6)$

#### 【解答】

- (1) 选(C)
- (2) 选(A)
- (3) 选(A)

^①  $\log_2 n$  即配套教材中的  $\lg n$ 。

## 例 1-3 填空题

- (1) 封装是指把数据和操纵数据的运算组合成( )的机制。
- (2) 一个程序步是一个在语法上或语义上有意义的程序段, 该程序段的执行时间与( )无关。
- (3) 使用大 O 记号表示的算法的时间复杂度, 称为算法的( )时间复杂度。

【解答】

- (1) 模块
- (2) 问题实例的特征
- (3) 渐近

## 例 1-4 解答题

- (1) 什么是数据类型, 什么是抽象数据类型?
- (2) 简述算法的确定性和能行性。
- (3) 算法有哪三种情况时间复杂度?

【解答】

- (1) 一个数据类型定义分为两部分: 值(可以是带结构的数据元素)的集合和一组运算。一个抽象数据类型(ADT)是一个数据类型, 它将规范和实现分离, 并实现封装和信息隐蔽。
- (2) 算法的确定性是指算法的每条指令都有确切的定义, 没有二义性; 而算法的能行性是指算法的每条指令原则上都可以由计算机指令执行有限步来实现。

(3) 算法分析一般可以分析一个算法的最好情况、最坏情况和平均情况三种时间复杂度。当对某个(些)问题实例执行算法时, 所需时间最少, 该时间值被称为此算法的最好情况时间复杂度; 当对另外的某个(些)实例执行算法时, 所需时间最多, 该时间值被称为此算法的最坏情况时间复杂度; 如果已知或假定问题的各种可能的实例出现的概率, 计算某算法执行时间的期望值, 则该值被称为此算法的平均时间复杂度。

## 1.2.2 算法分析题

例 1-5 设  $n$  是正整数, 下列程序(段)中, 语句 “ $y++$ ;” 的执行次数是多少?

```
for(i=1; i<=n; i++)
    if (2*i<n) y++;
```

【解答】  $y++$ ; 语句的执行次数为  $\sum_{i=1}^{\lceil n/2 \rceil - 1} 1 = \lceil n/2 \rceil - 1$  次, 其中, 符号  $\lceil x \rceil$  是取不小于  $x$

的最小整数。

例 1-6 设  $n$  为正整数, 计算下列程序(段)的程序步数。给出此程序段的最好、平均和最坏情况时间复杂度。

```
for(i=n-1; i>0; i--)
    for(j=0; j<i; j++)
        if(a[j]>a[j+1]){
            e=a[j]; a[j]=a[j+1]; a[j+1]=e;
        }
```

【解答】可以用 if 语句的执行次数来计算此程序的程序步数，其计算公式为

$$\sum_{i=1}^{n-1} i = \frac{n(n-1)}{2}$$

其最好情况、平均情况和最坏情况的时间复杂度是相同的，可用大 O 记号表示为  $O(n^2)$ 。

例 1-7 分析下列函数的时间复杂度，给出 sum 的最终值。

```
void f(int n)
{
    int i,j,k,sum=0;
    for(i=1; i<n+1;i++)
        for(j=n;j>i-1;j--)
            for(k=1;k<j+1;k++) sum++;
    printf("sum=%d\n",sum);
}
```

【解答】函数 f 运行结束，变量 sum 的值如下：

$$\sum_{i=1}^n \sum_{j=i}^n j = \sum_{i=1}^n \frac{(n+i)(n-i+1)}{2} = \frac{2n^3 + 3n^2 + n}{6}$$

sum 值的阶与执行该算法的程序步数增长速度的阶相同。所以，此算法的渐近时间复杂度为  $O(n^3)$ 。

### 1.2.3 算法设计题

例 1-8 试设计一个算法，从包括 n 个元素的数组 a 中求最大元素和最小元素，使得当 n 个元素已经从小到大有序时，元素之间的比较次数仅为 n-1 次。

【解答】程序如下：

```
void MaxMin(T a[], T* max, T* min, int n)
{
    int i;
    if (n==0) return;
    *max=*min=a[0];
    for (i=1; i<n; i++)
        if(a[i]>*max) *max=a[i];
        else if(a[i]<*min) *min=a[i];
}
```

如果数组 a 中元素已经从小到大排序，则仅需 n-1 次元素间比较，就可得到最大元素和最小元素。如果数组 a 中元素次序呈逆序，则需要 2(n-1)次比较，才能完成计算。

假如将上述程序中的语句

```
if(a[i]>*max) *max=a[i];else if(a[i]<*min) *min=a[i];
```

用两个 if 语句

```
if(a[i]>*max) *max=a[i]; if(a[i]<*min) *min=a[i];
```

取代,则算法是正确的,但在最好、最坏和平均情况下,元素间的比较次数相同,均为  $2(n-1)$  次。程序中的类型 T, 可以用 `typedef int T;` 定义为整型或其他具体的元素类型。

**例 1-9** 实现抽象数据类型 ADT Complex, 它包括复数的加法和减法运算。设计主函数 main, 它使用复数类型 Complex 进行复数运算。

**【解答】**该抽象数据类型上除了需要定义加法和减法运算外,一般还需定义构造一个复数和输出一个复数的运算。

- (1) 使用一个 C 语言结构来存储一个复数, 定义如下:

```
typedef struct complex
```

```
{
```

```
    float x,y;
```

```
}Complex;
```

- (2) 可以设计一个构造函数 Comp 来构造一个复数:

```
Complex Comp(float x,float y)
```

```
{
```

```
    Complex c;
```

```
    c.x=x;c.y=y;
```

```
    return c;
```

```
}
```

- (3) 实现复数的加法和减法运算:

```
Complex Add (Complex a,Complex b)
```

```
{
```

```
    Complex c;
```

```
    c.x=a.x+b.x;
```

```
    c.y=a.y+b.y;
```

```
    return c;
```

```
}
```

```
:
```

/*减法运算与加法运算相似, 此处略*/

- (4) 设计主函数, 测试复数运算:

```
void main()
```

```
{
```

```
    Complex a,b,c,d;
```

```
    a=Comp(1.0f,2.0f); b=Comp(3.0f,4.0f); /*复数 a=(1+2i),b=(3+4i)*/
```

```
    c=Add(a,b); d=Sub(a,b); /*复数 c=a+b, 复数 d=a-b*/
```

```
    PrintComp(a);PrintComp(b); /*输出一个复数的运算, 请读者实现之*/
```

```
    PrintComp(c);PrintComp(d);
```

```
}
```

## 1.3 考研试题解析^①

### 1.3.1 基本题

#### 例 1-10 判断题

- (1) 数据的物理结构是指数据在计算机内的实际存储形式。( )
- (2) 数据结构的抽象操作的定义与具体实现有关。( )
- (3) 数据元素是数据的最小单位。( )

【解答】

(1) 是。数据的逻辑结构是面向应用问题的,数据的存储结构是数据在计算机内的组织方式,是面向计算机的。

(2) 否。操作的定义规定一个操作的功能,它与具体实现无关。

(3) 否。数据元素是组成数据的基本单位,并非数据的最小单位。数据项是数据的不可分割的最小单位。C 语言中原子类型数据是非结构的和不可分割的,结构类型的值是由若干成分按某种结构组成的。

#### 例 1-11 选择题

- (1) 以下与数据的存储结构无关的术语是( )。  
(A) 循环队列 (B) 单链表 (C) 散列表 (D) 堆栈
- (2) 计算机算法是求解特定问题的步骤,它必须满足下列( )特性。  
(A) 可执行性、可移植性、可扩充性 (B) 能行性、确定性、有穷性  
(C) 确定性、有穷性、稳定性 (D) 易读性、稳定性、安全性
- (3) 下列大 O 记号表示中,不正确的是( )。  
(A)  $n^2+2n=O(n^3)$  (B)  $n\log_2 n+2n=O(n^2)$   
(C)  $n^2+n\log n=O(n^2\log_2 n)$  (D)  $n^2+n\log_2 n=O(n\log_2 n)$

【解答】

- (1) 选(D)
- (2) 选(B)
- (3) 选(D)

#### 例 1-12 填空题

- (1) 对于给定的 n 个元素,可构成四种逻辑结构: ( )。
- (2) 一个数据结构在计算机中的( )称为存储结构。
- (3) 计算机执行下面的语句时,语句 s 的执行次数为( )。

```
for(i=1; i<n-1; i++)
    for(j=n;j>=i;j--) s;
```

^① 本书的考研试题选自全国几十所高校,不少试题被多个学校采用,出现在不同学校的试卷上。在选用时为符合本书的原配书的概念和方法,对试题在叙述上也作了一定改动。此外,由于数据结构描述语言不同,一些试题需从 Pascal 和 C++ 语言改编为 C 语言描述。因此,在本书中未列出试题的出处。