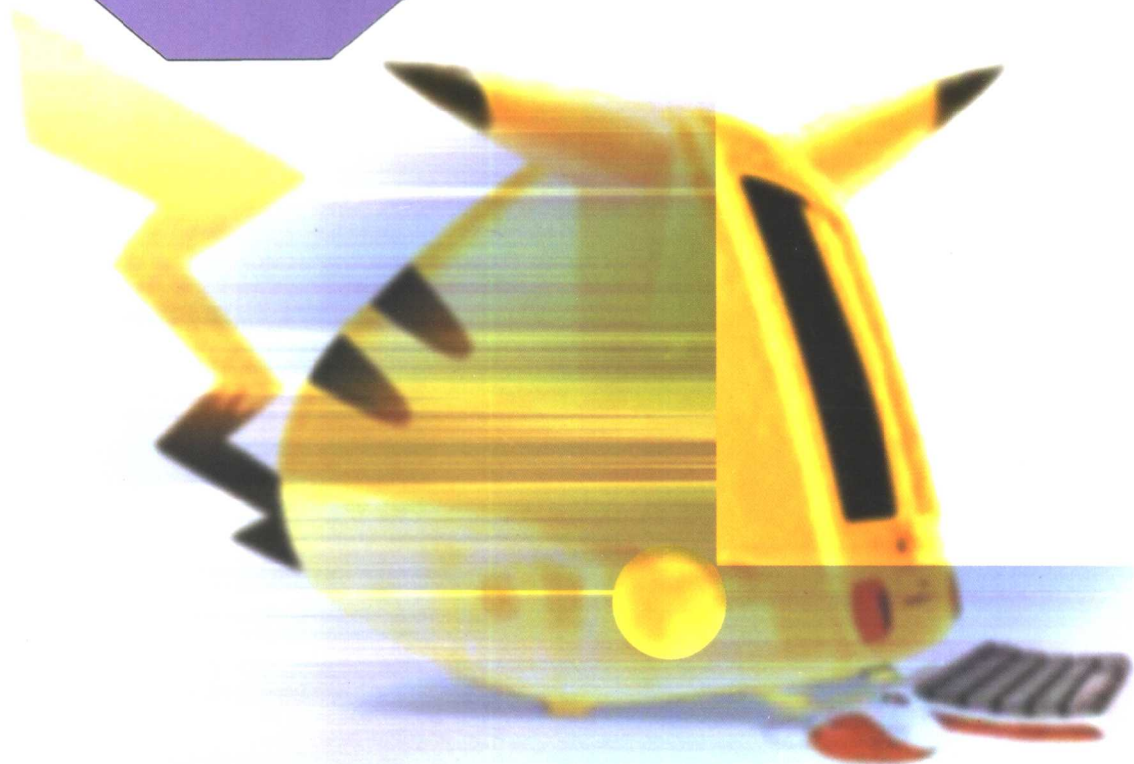


National Computer Rank Examination
全国计算机等级考试

二级教程

基础知识和Java语言程序设计

李季 赵成璧 编著



南开大学出版社

全国计算机等级考试系列

二级教程

基础知识和 Java 语言程序设计

李 季 赵成璧 编著

南开大学出版社

天津

图书在版编目(CIP)数据

全国计算机等级考试二级教程. 基础知识和 Java 语言程序设计 / 李季, 赵成壁编著. —天津: 南开大学出版社, 2004. 11

(全国计算机等级考试系列)

ISBN 7-310-02141-X

I. 全... II. ①李... ②赵... III. ①电子计算机—水平考试—教材 ②JAVA 语言—程序设计—水平考试—教材 IV. TP3

中国版本图书馆 CIP 数据核字(2004)第 069431 号

版权所有 翻印必究

南开大学出版社出版发行

出版人: 肖占鹏

地址: 天津市南开区卫津路 94 号 邮政编码: 300071

营销部电话: (022)23508339 23500755

营销部传真: (022)23508542 邮购部电话: (022)23502200

*

河北昌黎太阳红彩色印刷有限责任公司印刷

全国各地新华书店经销

*

2004 年 10 月第 1 版 2004 年 10 月第 1 次印刷

787×1092 毫米 16 开本 18.75 印张 476 千字

定价: 26.00 元

如遇图书印装质量问题, 请与本社营销部联系调换, 电话: (022)23507125

前 言

1994年,国家教育部推出了面向社会的全国计算机等级考试。10年来,为了促进计算机知识的普及和计算机应用技术的推广,满足用人单位考核工作人员应用计算机的水平,全国计算机等级考试也在不断自我完善和修订,最新版考纲在2004年开始推行。新考纲中,除了取消过时的考试科目和调整一些科目的考试内容之外,在二级考试科目中,新增了目前较为流行的语言Java、C++和Access。

为了满足考生的要求,在全国计算机等级考试中心有关专家的指导下,我们严格按照新考试大纲的要求,组织了既熟悉等级考试又精通专业技术的强大编写队伍,专门针对新增的Java考试科目编写了本书。

本教程针对准备参加全国计算机等级考试二级Java语言程序设计的考生,帮助您快速掌握Java语言的设计方法,提高学习效率,在较短的时间内轻松通过考试。

考试基本要求

全国计算机等级考试二级(Java语言程序设计)分为上机考试和笔试两部分,主要内容涉及公共基础知识和Java语言程序设计。

公共基础知识部分对于考生的要求是:掌握算法的基本概念;掌握基本数据结构及其操作;掌握基本排序和查找算法;掌握逐步求精的结构化程序设计方法;掌握软件工程的基本方法,具有初步应用相关技术进行软件开发的能力;掌握数据库的基本知识,了解关系数据库的设计。

Java语言程序设计部分对于考生的基本要求是:掌握Java语言的特点、实现机制和体系结构;掌握Java语言中面向对象的特性;掌握Java语言提供的数据类型和结构;掌握Java语言编程的基本技术;会编写Java用户界面程序;会编写Java简单应用程序;会编写Java小应用程序(Applet);了解Java的应用。

笔试时间为90分钟,满分100分(公共基础知识的考试方式为笔试,与Java语言程序设计的笔试部分合为一张试卷,占30分);上机考试90分钟,满分100分。

本书的特点

专家构成的作者队伍

本书由长期从事全国计算机等级考试授课、辅导和Java语言程序设计的专家共同编写,并得到考试中心有关专家的详细指点,内容紧扣考试大纲,书中的习题模拟考试题,避免读者走弯路,提高学习效率。

知识讲解详细清晰

本书严格按照等级考试的大纲要求，由浅入深、循序渐进地介绍 Java 语言程序设计技术，并针对考点、重点和难点，进行详细的讲解。

大量的例题分析和讲解

为了让读者真正掌握 Java 语言的编程技能，本书在介绍理论知识的同时，辅以大量程序设计例题，并进行详尽的分析和说明。

课后习题测试掌握程度

在每章的最后，都有一个习题部分，在此可以测试读者对该章内容的掌握程度。在附录中，给出了各章习题的参考答案。

本书作者

本书由李季、赵成璧老师主笔，编写过程中，邱代燕、宫晓琳、高磊、黄志雄、董岚枫、徐增辉、刘刚、刘朋、尚卫平、孙宏、于樊鹏、陈河南、贺军、梁德成、梁彩隆、廖明武、倪永智、贺民、李志云、戴军、陈安南、李晓春、王春桥、王雷、郭涛、韦笑、龚亚萍、孟丽艳、石丽霞等人在资料整理、查错、代码调试、预读等方面做了大量的工作。

本书的所有例题代码，均经过上机调试通过。读者如果需要，或者有其他意见和建议，可以给如下地址发邮件。

xiaoxiang-007@sohu.com

目 录

第一部分 公共基础知识

第 1 章 基本数据结构与算法	1
1.1 基本数据结构.....	1
1.1.1 什么是数据结构.....	1
1.1.2 逻辑结构.....	2
1.1.3 存储结构（物理结构）.....	2
1.1.4 数据的运算.....	3
1.1.5 数据结构的图形表示.....	3
1.2 算法.....	3
1.2.1 什么是算法.....	3
1.2.2 算法复杂度.....	4
1.3 线性表.....	4
1.3.1 什么是线性表.....	4
1.3.2 线性表的顺序存储结构.....	4
1.3.3 线性表的插入与删除运算.....	5
1.4 栈和队列.....	5
1.4.1 什么是栈和队列.....	5
1.4.2 栈的顺序存储结构及其基本运算.....	5
1.4.3 队列的顺序存储结构及其基本运算.....	6
1.5 线性单链表、循环链表与双向链表.....	6
1.5.1 线性单链表.....	6
1.5.2 循环链表.....	7
1.5.3 双向链表.....	7
1.5.4 链表的基本操作.....	7
1.6 树.....	8
1.6.1 什么是树.....	8
1.6.2 二叉树及其存储结构.....	9
1.6.3 二叉树的遍历.....	11
1.7 查找.....	11
1.7.1 顺序查找.....	11
1.7.2 二分查找算法.....	11
1.8 排序.....	12
1.8.1 交换排序.....	12
1.8.2 选择类排序.....	12

1.8.3 插入类排序	12
1.8.4 各种排序方法	12
本章小结	13
习题一	13
第 2 章 程序设计基础	16
2.1 程序设计	16
2.1.1 程序设计的方法	16
2.1.2 程序设计的风格	16
2.2 结构化程序设计	19
2.2.1 结构化程序设计的基本特征	19
2.2.2 结构化程序设计的主要原则	19
2.3 面向对象的程序设计方法	20
2.3.1 面向对象设计	20
2.3.2 面向对象方法的特点	20
2.3.3 面向对象程序设计的相关概念	20
2.3.4 面向对象分析与模型化	21
本章小结	22
习题二	22
第 3 章 软件工程基础	25
3.1 软件工程概述	25
3.1.1 软件工程	25
3.1.2 软件生命周期	25
3.1.3 软件工具与软件开发环境	25
3.2 结构化分析与设计	25
3.2.1 结构化分析方法	26
3.2.2 数据流图	26
3.2.3 数据字典	27
3.2.4 软件需求	27
3.2.5 结构化设计	27
3.2.6 总体设计	28
3.2.7 详细设计(过程设计)	28
3.3 软件测试与程序调试	28
3.3.1 软件测试的目的	28
3.3.2 测试用例设计	28
3.3.3 软件测试的实施	29
3.3.4 程序调试	30
3.3.5 常见调试方法	30
本章小结	30
习题三	31

第 4 章 数据库设计基础	33
4.1 数据库与数据模型.....	33
4.1.1 数据库.....	33
4.1.2 数据库管理系统.....	33
4.1.3 数据库系统.....	33
4.1.4 实体联系模型.....	33
4.1.5 E-R 图.....	33
4.1.6 关系数据模型.....	35
4.1.7 E-R 模型到关系模型的转换.....	35
4.1.8 关系代数运算.....	36
4.1.9 数据库的规范化.....	36
4.2 数据库设计.....	37
4.2.1 需求分析.....	37
4.2.2 概念设计.....	38
4.2.3 逻辑设计.....	38
4.2.4 物理设计.....	38
本章小结.....	39
习题四.....	40

第二部分 Java 语言程序设计

第 5 章 Java 概述	42
5.1 Java 语言简介.....	42
5.1.1 Java 语言的起源与发展.....	42
5.1.2 Java 语言的特点.....	43
5.1.3 Java 程序类型.....	44
5.1.4 Java 的实现机制.....	45
5.2 Java 体系结构.....	47
5.2.1 JDK 目录结构.....	48
5.2.2 Java 的 API 结构.....	53
5.2.3 开发环境设置、程序开发步骤以及开发工具简介.....	54
5.2.4 Java 程序结构.....	61
5.2.5 Java 程序举例.....	62
本章小结.....	67
习题五.....	67
第 6 章 Java 语言中的面向对象特性	70
6.1 面向对象编程的基本概念和特征.....	70
6.1.1 面向对象编程简介.....	70
6.1.2 对象与类的基本概念.....	71

6.1.3	消息与方法的基本概念	73
6.1.4	面向对象编程的基本特征	73
6.2	类	76
6.2.1	类的定义	77
6.2.2	成员变量	79
6.2.3	成员方法	82
6.2.4	方法重载	85
6.2.5	构造方法	85
6.2.6	main 方法	88
6.3	对象	89
6.3.1	对象的生成	89
6.3.2	对象的使用	91
6.3.3	对象的清除	92
6.4	类的继承和多态	93
6.4.1	类的继承	93
6.4.2	运行时多态	95
6.4.3	抽象类和抽象方法	97
6.5	接口与包	98
6.5.1	接口	98
6.5.2	包(Package)	101
6.6	Java 类库中的常用类和接口	103
6.6.1	常用类	103
6.6.2	常用接口	107
	本章小结	108
	习题六	108
第 7 章	简单数据类型及运算	112
7.1	变量和常量	112
7.1.1	变量	112
7.1.2	常量	112
7.2	基本数据类型及转换	112
7.2.1	基本数据类型	112
7.2.2	基本数据类型的缺省值	117
7.2.3	基本数据类型的转换	117
7.3	Java 类库中对简单类型数据的类包装	118
7.3.1	Java 类库介绍	118
7.3.2	标准 Java 包	119
7.3.3	简单类型数据的类包装	119
7.4	运算符和表达式运算	120
7.4.1	运算符分类	120

7.4.2	运算符优先级	125
7.4.3	表达式	125
7.5	数组和字符串	127
7.5.1	一维数组的声明	127
7.5.2	一维数组的初始化	127
7.5.3	访问数组元素	128
7.5.4	多维数组	131
7.5.5	字符串的构造	132
7.5.6	字符串的基本操作	134
7.5.7	StringBuffer 类	138
	本章小结	139
	习题七	140
第 8 章	基本语句	144
8.1	表达式语句	144
8.2	条件语句	145
8.2.1	if 语句	145
8.2.2	switch 语句	152
8.3	循环语句	156
8.3.1	while 语句	156
8.3.2	do-while 语句	159
8.3.3	for 语句	161
8.3.4	循环嵌套	165
8.3.5	中断控制流程	166
8.4	注释语句	167
	本章小结	167
	习题八	168
第 9 章	异常、线程和对象串行化	173
9.1	异常	173
9.1.1	异常的概念	173
9.1.2	异常类的类层次	173
9.1.3	异常处理	177
9.2	线程	186
9.2.1	线程的概念	186
9.2.2	线程的使用	187
9.2.2	同步与共享	196
9.3	对象串行化	199
9.3.1	串行化的概念和目的	199
9.3.2	串行化方法	199
9.3.3	对象的串行化应用举例	200

本章小结.....	203
习题九.....	203
第 10 章 输入/输出处理和集合.....	207
10.1 Java 输入/输出流.....	207
10.1.1 文件类.....	208
10.1.2 InputStream 类.....	208
10.1.3 OutputStream 类.....	209
10.1.4 Reader 类.....	209
10.1.5 Writer 类.....	210
10.1.6 RandomAccessFile 类.....	211
10.2 文件和文件 I/O.....	211
10.2.1 文件.....	211
10.2.2 文件 I/O.....	218
10.3 集合.....	222
10.3.1 Collection 接口.....	223
10.3.2 Set 和 List 接口.....	224
10.3.3 集合中的其他类和接口.....	225
10.3.4 工具类.....	228
本章小结.....	229
习题十.....	229
第 11 章 用户界面.....	232
11.1 图形用户界面.....	232
11.2 AWT 库简介.....	232
11.2.1 AWT 概述.....	232
11.2.2 AWT 容器.....	233
11.2.3 AWT 基本组件.....	235
11.2.4 布局管理器.....	237
11.2.5 事件处理机制.....	242
11.3 Swing 简介.....	245
11.3.1 Swing 库简介.....	245
11.3.2 Swing 容器.....	246
11.3.3 Swing 基本组件.....	247
11.3.4 布局管理器.....	252
11.4 AWT 与 Swing 比较.....	254
本章小结.....	255
习题十一.....	256
第 12 章 编写小应用程序.....	259
12.1 小应用程序的概念.....	259
12.2 安全机制.....	260

12.3	Applet 的执行过程.....	261
12.3.1	编译和运行 Applet 程序.....	261
12.3.2	Applet 的生命周期和主要的方法.....	262
12.4	Applet 的图形绘制.....	265
12.4.1	Applet 中与 AWT 图形绘制有关的方法.....	265
12.4.2	java.awt.Graphics 类.....	265
12.4.3	Java 2D 图形类.....	268
12.5	Applet 的窗口.....	268
12.6	Applet 的工作环境.....	272
12.6.1	同页 Applet 间的通信.....	272
12.6.2	Applet 和浏览器间的通信.....	274
12.6.3	Applet 的网络通信.....	276
12.7	Java Application 和 Applet.....	276
12.8	Java 的应用.....	278
	本章小结.....	278
	习题十二.....	279
附录	习题参考答案.....	282

第一部分 公共基础知识

第 1 章 基本数据结构与算法

如果问一个木匠的学徒工：“你工作时用什么工具呢？”他可能回答：“一把锤子和一把锯子”。但如果去问老木匠这个问题，他会说：“我需要一些精确的工具。”同样，由于计算机所解决的问题都是从生活中抽象出来的问题，其复杂性与多样性不言而喻，所以我们需要精确有效的工具去解决现实生活中的复杂问题。算法、数据结构、程序设计语言都是这样的工具。瑞士的计算机学者 Niklaus Wirth 早在 1976 年就曾这样定义程序：

算法 + 数据结构 = 程序设计

很快，这就成了在计算机工作者之间流传的一句名言，从中不难看出算法与数据结构的重要性。本章将介绍有关数据结构与算法的基本知识。

1.1 基本数据结构

1.1.1 什么是数据结构

简单地说，数据结构就是数据的数据模型。一般说来，用计算机解决一个具体问题时，大致需要经历下列几个步骤：

- (1) 首先要从具体问题抽象出一个适当的数学模型。
- (2) 然后，设计一个解此数学模型的算法。
- (3) 最后编出程序，进行测试、调整，直至得到最终解答。

寻求数学模型的实质是分析问题，从中提取操作的对象，并找出这些操作对象之间含有的关系，然后用数学的语言加以描述。例如，求解梁架结构应力的数学模型为线性方程组；预报人口增长情况的数学模型为微分方程。然而，更多的非数值计算问题无法用数学方程加以描述。请看下面一个非数值计算问题的例子。

【例 1】铺设城市煤气管道的问题。

假设要在某个城市的 n 个居民区之间铺设煤气管道，则在这 n 个居民区之间只要铺设 $n-1$ 条管道即可。假设任意两个居民区之间都可以铺设管道，但由于地理环境的不同，所需经费也不同，如何规划才能使得总投资花费最少，这个问题即为“求图的最小生成树”的问题。其数学模型为如图 1.1 所示的“图”，图中“顶点”表示居民区，顶点之间的连线及其上的数值表示可以架设的管道及所需经费。求解的算法为在可能假设的 m 条管道中选

取 $n-1$ 条,既能连通 $n-1$ 个居民区,又使总投资达到“最小”。

通常,这类涉及道路、线路问题的数学模型是一种所谓“图”的数据结构。对于其他的非数值计算问题,还可能用其他诸如表、树之类的数据结构来描述和解决。

在给出数据结构的定义之前,首先要知道什么是数据。所谓数据,就是所有能被输入到计算机中,且能被计算机处理的符号(数值、字符等)的集合,是计算机操作的对象总称。

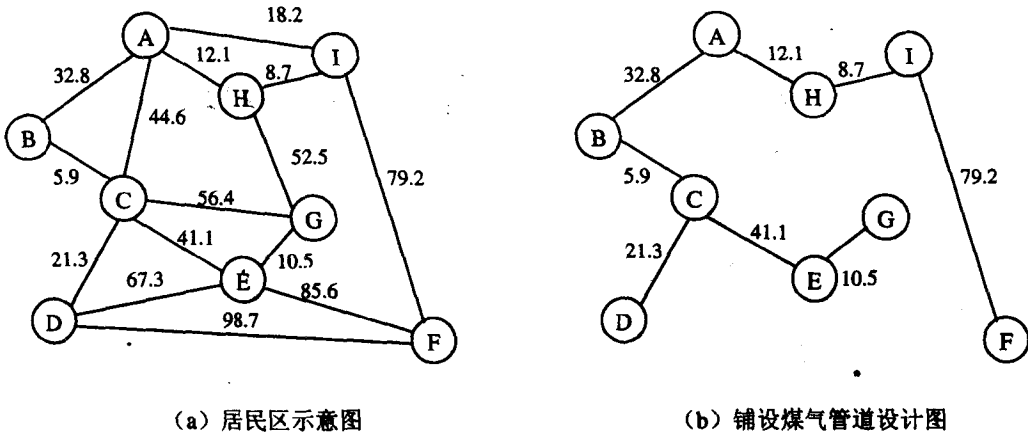


图 1.1 图及最小生成树示例

而数据结构是相互之间存在一种或多种特定关系的数据元素的集合。数据结构一般包括以下三方面的内容:数据的逻辑结构、数据的物理结构、数据的运算。

1.1.2 逻辑结构

逻辑结构即数据元素之间的逻辑关系。它可以用一个数据元素的集合和定义在此集合上的若干关系来表示。

数据的逻辑结构是从逻辑关系上描述数据,它与数据的存储无关,是独立于计算机的。它有两大类:线性结构和存储结构。

1. 线性结构

线性结构的逻辑特征是,在数据元素的非空有限集中:

- 存在惟一的一个被称作“第一个”的数据元素;
- 存在惟一的一个被称作“最后一个”的数据元素;
- 除第一个之外,集合中的每个数据元素均只有一个前驱;
- 除最后一个之外,集合中的每个数据元素均只有一个后继。

例如,线性表是一个线性结构。

2. 非线性结构

非线性结构的逻辑特征是:一个结点可能有多个直接前驱和直接后继。例如:树和图都是非线性结构。

1.1.3 存储结构(物理结构)

数据元素及其关系在计算机存储器内的表示——数据的存储结构是逻辑结构用计算机语言的实现,即建立数据的机内表示。存储结构主要内容指在存储空间中使用一个存储结

点来存储一个数据元素；在存储空间中建立各存储结点之间的关联，来表示数据元素之间的逻辑关系。其中存储结点是指一个数据元素在存储结构中的存储。

数据的存储结构有如下的4种基本方式：

- 顺序存储方式 每一个存储结点只含一个数据元素。所有的存储结点相继存储在一个连续的存储区里。用存储结点之间的位置关系表示数据元素之间的逻辑关系。
- 链式存储方式 每一个存储结点不仅含有一个数据元素，还包括指针。每一个指针指向一个与本结点有逻辑关系的结点，即用指针表示逻辑关系。
- 索引存储方式 每一个存储结点仅含一个数据元素，所有的存储结点都连续存放。此外，增设一个索引表。
- 散列存储方式 每一个存储结点仅含一个数据元素，数据元素按散列函数确定存储位置。

1.1.4 数据的运算

数据运算即对数据施加的操作。常用的运算如下：

- 查找运算——从结构中找出满足某种条件的结点的位置。
- 读取运算——读出结构中指定位置上的内容。
- 插入运算——在结构中的某指定位置上增加一个新的结点。
- 删除运算——撤销结构中指定位置上的结点。
- 更新运算——修改结构中某指定结点的内容。

1.1.5 数据结构的图形表示

图 1.2 是一些常见数据结构的图形表示示例。

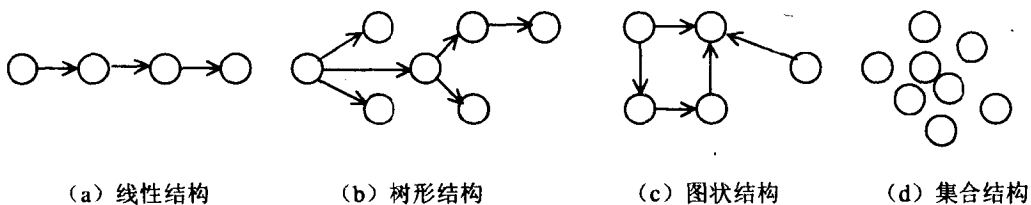


图 1.2 常见数据结构的图形表示示例

1.2 算法

1.2.1 什么是算法

算法是对特定问题求解步骤的一种描述，它是指令的有限序列，其中每条指令表示一个或是多个操作。严格说来，一个算法必须具有下列五个主要特性。

- 有穷性 一个算法必须总是（对任何合法的输入值）在执行有穷步之后结束，而且每一步都可在有穷时间内完成。
- 确定性 算法中每天指令必须有确切含义，且在任何条件下，算法只有惟一的一

条执行路径。

- 可行性 算法中描述的操作都是可以通过已经实现的基本运算执行有限次来实现的。
- 有输入 一个算法有零个或多个输入，这些输入取自于某个特定的对象集合。
- 有输出 一个算法有零个或多个输出，这些输出是同输入有着某些特定关系的量。

1.2.2 算法复杂度

选用算法首先考虑正确性，还要考虑执行算法所耗费的时间和存储空间，同时，算法应易于理解、编码、调试等。算法的复杂度可分为时间复杂度和空间复杂度，是衡量算法优劣的量度。

1. 时间复杂度

一般情况下，算法中的基本操作重复执行的次数是问题规模 n 的某个函数 $f(n)$ ，算法的时间量度记作：

$$T(n) = O(f(n))$$

它表示随着问题规模 n 的增大，算法执行时间的增长率和 $f(n)$ 的增长率相同，称作算法的渐进时间复杂度，简称时间复杂度。

2. 空间复杂度

类似时间复杂度，空间复杂度作为算法所需存储空间的量度：

$$S(n) = O(g(n))$$

其中， n 为问题的规模，表示随着问题规模 n 的增大，算法运行所需存储量的增长率与 $g(n)$ 的增长率相同。

1.3 线性表

1.3.1 什么是线性表

线性表是最常用且最简单的一种数据结构。所谓线性表是 n 个数据元素的有限序列。至于每个数据元素的具体含义在不同情况下各不相同，它可以是一个数，或是一个符号，也可以是一页书，甚至其他更复杂的信息。例如，26 个英文字母的字母表：

(A,B,C,D,⋯,Z)

是一个线性表，表中的数据元素是单个字母。又如，某校从 1998 年到 2004 年的计算机拥有量的变化情况，可以用线性表的形式给出：

(23, 35, 67, 156, 240, 287, 324)

表中的数据元素是整数。

1.3.2 线性表的顺序存储结构

线性表的顺序存储结构指的是用一组地址连续的存储单元依次存储线性表中的数据元素。其特点是：以元素在计算机内的“物理位置相邻”来表示线性表中的数据元素之间的逻辑关系，即以“存储位置相邻”表示有序对 $\langle a_{i-1}, a_i \rangle$ 。图 1.3 说明了数据元素在计算机

内的存储情况。其中 a_1, a_2, \dots, a_n 表示线性表中的数据元素。

所有数据元素的存储位置均取决于第一个数据元素的存储位置，即：

$$\text{LOC}(a_i) = \text{LOC}(a_1) + (i-1) \times C$$

↑ 基地址
↑ 一个数据元素所占存储量

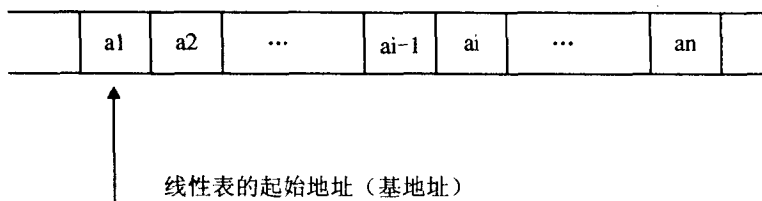


图 1.3 线性表的顺序存储结构示意图

1.3.3 线性表的插入与删除运算

插入和删除运算是线性表的基本操作。所谓插入运算，是指在结构中的某指定位置上增加一个新结点。而删除运算指撤销结构中某结点的内容。下面依次进行详细讨论。

1. 插入

线性表的插入操作是指在线性表的第 $i-1$ 个数据元素和第 i 个数据元素之间插入一个新的数据元素，就是要使长度为 n 的线性表

$(a_1, \dots, a_{i-1}, a_i, \dots, a_n)$

变成长度为 $n+1$ 的线性表

$(a_1, \dots, a_{i-1}, b, a_i, \dots, a_n)$

数据元素 a_{i-1} 和 a_i 之间的逻辑关系发生了变化。

2. 删除

相反，线性表的删除操作是使长度为 n 的线性表

$(a_1, \dots, a_{i-1}, a_i, a_{i+1}, \dots, a_n)$

变为长度为 $n-1$ 的线性表

$(a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_n)$

数据元素 a_{i-1}, a_i 和 a_{i+1} 之间的逻辑关系发生了变化。

1.4 栈和队列

1.4.1 什么是栈和队列

栈是限定仅在表尾进行插入和删除操作的线性表。因此，对栈来说，表尾端有其特殊的含义，称为栈顶，相应地，表头端称为栈底。

队列是限定仅在表的一端进行插入，而在表的另一端删除数据元素的线性表。在队列中，允许插入的一端叫做队尾，允许删除的一端则称队头。

1.4.2 栈的顺序存储结构及其基本运算

1. 顺序存储结构

栈的顺序存储结构是利用一组地址连续的存储单元依次存放自栈底到栈顶的数据元