

725

0.83

7366(2)

全国计算机等级考试

二 级 教 程

基础知识和 QBasic 程序设计

(第二版)

刘瑞挺 主编

方文祺 高福成 编著

南开大学出版社

·天 津·

内容简介

本书是根据国家教育部考试中心 1998 年制定的“全国计算机等级考试”新大纲中“二级考试大纲”的要求编写的。主要介绍了计算机的基础知识、操作系统的基本概念与基本操作、QBasic 语言的基本概念、语法规则和程序设计的有关知识。

本书覆盖了考试大纲要求的知识范围,系统扼要地介绍了考生应掌握的内容。每章均有小结与题型示例,并且附有习题。书后附有习题参考答案。

本书可作为学习 QBasic 语言程序设计、参加全国计算机等级考试(二级)考生的辅导教材。

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 QBASIC 程序设计 / 方文祺主编. —天津: 南开大学出版社, 2001. 12

(全国计算机等级考试系列丛书)

ISBN 7-310-01585-1

I. 全... II. 方... III. ①电子计算机—水平考试—教材②BASIC 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(2001)第 046582 号

出版发行 南开大学出版社

地址:天津市南开区卫津路 94 号

邮编:300071 电话:(022)23508542

出版人 肖占鹏

承印 南开大学印刷厂印刷

经销 全国各地新华书店

版次 2001 年 12 月第 2 版

印次 2001 年 12 月第 4 次印刷

开本 787mm×1092mm 1/16

印张 19.5

字数 486 千字

印数 15001 — 20000

定价 27.00 元

前 言

1994 年,国家教委考试中心推出了面向社会的“全国计算机等级考试”。这是一种重视应试人员对计算机和软件的实际掌握能力的考试。它不限制报考人员的学历背景,为培养各行各业计算机的应用人才创出了一条新路。随着计算机知识的普及和应用,全国计算机等级考试越来越受到人们的重视和认可。

计算机知识有别于其他基础性理论,它总是随着计算机技术的发展而不断更新的。1994 年制定的考试大纲已经不能适应形势的需要。国家教育部考试中心于 1998 年组织专家对“全国计算机等级考试大纲”进行了修订,使计算机等级考试更能反映当前计算机与软件技术的应用实际。新大纲根据近年来计算机技术的发展和我国计算机的普及情况,在原大纲的基础上增加了 Windows、网络以及多媒体技术等知识,特别是在二级考试中,用结构化的 QBasic 程序设计取代 BASIC 程序设计。

全国计算机等级考试根据计算机应用能力水平的不同分为四个等级。其中二级主要考核应试者软、硬件基础知识和使用某种高级计算机程序设计语言编制程序、上机调试的能力。

本书是根据国家教育部考试中心 1998 年制定的“全国计算机等级考试”新大纲中“二级考试大纲”关于计算机基础知识、基本操作以及 QBasic 语言程序设计的要求编写的。主要介绍了计算机基础知识、DOS 基本操作、Windows 基本操作和 QBasic 语言的基本概念、语法规则及程序设计的有关知识。

本书的内容覆盖了考试大纲要求的知识范围,系统扼要地介绍了考生应掌握的内容。在每章均有小结与题型示例,并且附有习题。考虑到考试的形式大多采用“客观题”,所以习题大多数采用选择题的形式,并配有部分填空题,部分章节安排有少量的编程题。习题是根据历次考试试题精选编排的,有很强的针对性。书后附有绝大部分习题的参考答案。要掌握一门计算机程序设计语言,不上机实践是不行的。本书最后专门安排一章上机操作的内容。除了程序的编辑、调试与运行以外,还通过重点题型针对上机考试内容加以分析。

本书可作为学习计算机基础知识、基本操作和 QBasic 语言程序设计,参加全国计算机等级考试(二级)考生的辅导教材。

参加本书编写的有方文祺、黄福铭、沈楠和鲁雪梅等,最后由方文祺统稿。

编 者

2001. 6.

目 录

第 1 部分 基础知识

第 1 章 计算机基础知识	(3)	防治	(26)
1.1 计算机概述	(3)	2.5.1 微型计算机系统的安全 ..	(26)
1.1.1 什么是计算机	(3)	2.5.2 微型计算机系统病毒	
1.1.2 计算机的主要应用领域	(3)	防治	(27)
1.2 计算机中数的表示	(4)	习题二	(29)
1.2.1 数制的概念	(4)	第 3 章 DOS 操作系统	(31)
1.2.2 计算机中几种常用的进位		3.1 DOS 操作系统概述	(31)
计数制	(5)	3.1.1 DOS 操作系统的功能和	
1.2.3 不同数制间的转换	(6)	组成	(31)
1.2.4 数在计算机中的表示	(9)	3.1.2 DOS 的启动	(32)
1.2.5 字符和汉字编码	(10)	3.1.3 与 DOS 操作有关的基本	
1.2.6 数据的单位	(12)	知识	(32)
习题一	(13)	3.1.4 DOS 命令	(37)
第 2 章 微型计算机系统	(15)	3.2 DOS 常用命令	(38)
2.1 微型计算机系统的组成	(15)	3.2.1 目录操作命令	(38)
2.1.1 计算机系统的组成	(15)	3.2.2 文件操作命令	(40)
2.1.2 微型计算机系统的组成 ..	(15)	3.2.3 磁盘操作命令	(44)
2.2 微型计算机的硬件系统	(16)	3.2.4 功能操作命令	(47)
2.2.1 微处理器	(16)	3.3 批处理文件	(49)
2.2.2 内存储器	(17)	3.3.1 批处理文件的基本概念 ..	(49)
2.2.3 外存储器	(18)	3.3.2 批处理文件的建立与	
2.2.4 输入设备	(20)	执行	(49)
2.2.5 输出设备	(21)	3.3.3 自动批处理文件	
2.3 微型计算机的软件系统	(22)	(AUTOEXEC. BAT)	(51)
2.3.1 软件系统的组成	(22)	3.4 输入输出重定向	(52)
2.3.2 系统软件	(22)	3.4.1 输出改向	(52)
2.3.3 应用软件	(24)	3.4.2 输入改向	(53)
2.4 微型计算机的性能指标及系统		习题三	(53)
基本配置	(24)	第 4 章 多媒体技术和计算机网络	
2.4.1 微型计算机的性能指标 ..	(24)	基础	(58)
2.4.2 微型计算机系统的基本		4.1 多媒体技术	(58)
配置	(25)	4.1.1 多媒体的概念	(58)
2.5 微型计算机系统的安全与病毒		4.1.2 多媒体技术	(58)

4.1.3 多媒体计算机	(59)	5.1.4 中文 Windows 的启动 和退出	(78)
4.1.4 多媒体技术的应用	(60)	5.2 中文 Windows 的基础操作	(80)
4.2 计算机网络	(60)	5.2.1 鼠标	(80)
4.2.1 计算机网络的概念	(61)	5.2.2 桌面	(81)
4.2.2 计算机局域网基础知识	(65)	5.2.3 “开始”菜单	(81)
4.3 Internet 简介	(70)	5.2.4 窗口	(83)
4.3.1 什么是 Internet	(70)	5.2.5 窗口命令菜单	(86)
4.3.2 如何接入 Internet	(71)	5.2.6 对话框的组成及其操作	(88)
4.3.3 Internet 服务的功能	(72)	5.2.7 图标及其操作	(89)
习题四	(74)	5.3 资源管理器	(90)
第 5 章 中文 Windows 操作系统的功能 和使用	(77)	5.3.1 启动资源管理器的方法	(90)
5.1 Windows 操作系统简介	(77)	5.3.2 资源管理器的窗口	(90)
5.1.1 Windows 操作系统 的发展	(77)	5.3.3 文件和文件夹的操作	(92)
5.1.2 Windows 的功能和 特点	(77)	5.4 应用程序的操作	(97)
5.1.3 中文 Windows 系统的 安装	(78)	5.4.1 运行和退出应用程序	(97)
		5.4.2 应用程序的安装和删除	(97)
		习题五	(98)
		附录一 第 1 部分习题答案	(100)

第 2 部分 语言程序设计

第 6 章 QBasic 的基本概念	(105)	6.4.3 表达式	(112)
6.1 QBasic 集成环境	(105)	6.5 本章小结与题例分析	(113)
6.1.1 从 BASIC 到 QBasic	(105)	6.5.1 本章小结	(113)
6.1.2 QBasic 集成环境	(105)	6.5.2 题例分析	(113)
6.1.3 窗口	(106)	习题六	(114)
6.1.4 菜单	(106)	第 7 章 顺序结构程序设计	(116)
6.1.5 对话框	(107)	7.1 程序设计的基础知识	(116)
6.2 QBasic 程序的结构与执行 方式	(108)	7.1.1 程序设计的概念	(116)
6.2.1 QBasic 程序及其结构	(108)	7.1.2 什么是算法	(116)
6.2.2 QBasic 的两种执行方式	(109)	7.1.3 程序框图	(116)
6.3 常量和变量	(110)	7.1.4 利用计算机解题的步骤	(118)
6.3.1 常量	(110)	7.2 输出语句	(118)
6.3.2 变量	(110)	7.3 赋值	(119)
6.4 标准函数和表达式	(111)	7.3.1 赋值语句	(119)
6.4.1 运算符	(111)	7.3.2 变量的存取特征	(120)
6.4.2 标准函数	(112)	7.3.3 交换语句	(121)
		7.4 键盘输入	(121)

7.4.1 简单 INPUT 语句	(121)	实例	(150)
7.4.2 复合 INPUT 语句	(122)	9.2 DO—LOOP 循环	(152)
7.4.3 LINE INPUT 语句	(122)	9.2.1 最简单的 DO—LOOP	
7.5 读数与置数	(123)	循环	(152)
7.5.1 读数与置数语句	(123)	9.2.2 带 WHILE 子句的	
7.5.2 恢复数据区语句	(124)	DO—LOOP 循环	(152)
7.6 输出格式	(125)	9.2.3 带 UNTIL 子句的	
7.6.1 制表格式	(125)	DO—LOOP 循环	(153)
7.6.2 PRINT USING 语句	(126)	9.3 循环结构程序设计举例	(154)
7.7 顺序结构程序设计举例	(127)	9.3.1 求和(积)问题	(154)
7.8 本章小结与题例分析	(128)	9.3.2 穷举法	(156)
7.8.1 本章小结	(128)	9.3.3 随机模拟	(158)
7.8.2 题例分析	(128)	9.4 本章小结与题例分析	(160)
习题七	(129)	9.4.1 本章小结	(160)
第 8 章 选择结构程序设计	(132)	9.4.2 题例分析	(160)
8.1 关系表达式和逻辑表达式	(132)	习题九	(163)
8.1.1 关系运算	(132)	第 10 章 数组	(170)
8.1.2 逻辑运算	(133)	10.1 数组与数组元素	(170)
8.2 行 IF 语句	(133)	10.1.1 数组与下标变量的概念	
8.3 块 IF 语句	(134)		(170)
8.3.1 块 IF 语句	(134)	10.1.2 数组的定义	(171)
8.3.2 IF 的嵌套	(135)	10.1.3 数组的引用	(173)
8.4 SELECT CASE 结构	(136)	10.2 一维数组	(173)
8.4.1 SELECT CASE 语句	(136)	10.2.1 一维数组的基本操作	
8.4.2 GOTO 语句	(139)	——输入与输出	(173)
8.5 选择结构程序设计举例	(140)	10.2.2 一维数组的复制	(174)
8.6 本章小结与题例分析	(141)	10.2.3 数组元素的查找(检索)	
8.6.1 本章小结	(141)		(175)
8.6.2 题例分析	(142)	10.2.4 求数组元素的最大值	(176)
习题八	(144)	10.2.5 排序	(176)
第 9 章 循环结构程序设计	(147)	10.3 二维数组	(178)
9.1 FOR—NEXT 循环	(147)	10.3.1 二维数组的基本操作	(178)
9.1.1 FOR—NEXT 循环结构		10.3.2 二维数组应用实例	(179)
.....	(147)	10.4 静态数组与动态数组	(180)
9.1.2 FOR—NEXT 循环的		10.4.1 静态数组与动态数组	
执行过程	(148)	的定义	(180)
9.1.3 FOR—NEXT 循环的		10.4.2 REDIM 语句	(181)
嵌套	(149)	10.4.3 ERASE 语句	(181)
9.1.4 FOR—NEXT 循环应用		10.4.4 LBOUND 与 UBOUND	

函数	(182)	12.1.2 字符串变量	(212)
10.5 本章小结与题例分析	(183)	12.1.3 字符串数组	(213)
10.5.1 本章小结	(183)	12.2 字符串变量的赋值	(213)
10.5.2 题例分析	(183)	12.2.1 字符串变量的赋值	(213)
习题十	(186)	12.3 字符串的运算	(214)
第 11 章 函数与子程序	(191)	12.3.1 字符串连接	(214)
11.1 用户自定义函数——DEF		12.3.2 字符串的比较	(214)
函数	(191)	12.3.3 字符串表达式	(215)
11.1.1 自定义函数的概念	(191)	12.4 字符串函数	(215)
11.1.2 单行 DEF 函数	(191)	12.4.1 常用字符串函数	(215)
11.2 模块内子程序——子例程 ..	(193)	12.4.2 字符串函数应用举例 ..	(216)
11.2.1 子例程的概念	(193)	12.5 字符串处理举例	(218)
11.3 独立模块子程序——SUB		12.5.1 字符串排序	(218)
过程	(194)	12.5.2 查找	(219)
11.3.1 SUB 子程序的定义	(194)	12.5.3 其他	(219)
11.3.2 SUB 子程序的调用	(195)	12.6 本章小结与题例分析	(220)
11.4 模块化函数——FUNCTION		12.6.1 本章小结	(220)
函数	(196)	12.6.2 题例分析	(221)
11.4.1 FUNCTION 函数的定义		习题十二	(223)
.....	(196)	第 13 章 文件	(228)
11.4.2 FUNCTION 函数的调用		13.1 文件的概念	(228)
.....	(196)	13.1.1 什么是文件	(228)
11.5 参数传递	(197)	13.1.2 文件的分类	(228)
11.5.1 形式参数和实际参数 ..	(197)	13.1.3 QBasic 文件操作命令 ..	(229)
11.5.2 虚实结合	(197)	13.1.4 数据文件的结构	(229)
11.6 全局变量与局部变量	(200)	13.1.5 数据文件的读写	(229)
11.6.1 全局变量	(200)	13.2 顺序文件	(230)
11.6.2 局部变量	(201)	13.2.1 顺序文件的概念	(230)
11.6.3 静态变量	(201)	13.2.2 顺序文件的基本操作 ..	(230)
11.7 过程的嵌套与递归	(202)	13.3 随机文件	(235)
11.7.1 过程的嵌套调用	(202)	13.3.1 记录类型的定义	(235)
11.7.2 过程的递归调用	(203)	13.3.2 说明记录型变量	(236)
11.8 本章小结与题例分析	(205)	13.3.3 使用记录型变量	(236)
11.8.1 本章小结	(205)	13.3.4 随机文件的概念	(237)
11.8.2 题例分析	(205)	13.3.5 随机文件的基本操作 ..	(237)
习题十一	(208)	13.4 本章小结与题例分析	(239)
第 12 章 字符串处理	(212)	13.4.1 本章小结	(239)
12.1 字符串与字符串变量	(212)	13.4.2 题例分析	(239)
12.1.1 字符串常量	(212)	习题十三	(242)

第 14 章 屏幕控制与作图	(247)	15.1.6 从外存中读入源程序 ...	(257)
14.1 显示模式	(247)	15.1.7 程序的基本编辑	(258)
14.1.1 文本模式	(247)	15.1.8 文本块操作	(258)
14.1.2 图形模式	(247)	15.2 在 QBasic 集成环境中建立	
14.2 屏幕控制	(248)	过程	(259)
14.2.1 设置显示模式	(248)	15.2.1 在输入程序的同时	
14.2.2 屏幕定位	(248)	建立过程	(260)
14.2.3 颜色设置	(249)	15.2.2 使用菜单建立新过程	
14.3 基本图形的画法和着色	(249)	(或函数)	(260)
14.3.1 绝对坐标与相对坐标 ...	(249)	15.2.3 编辑程序模块	(261)
14.3.2 画点	(249)	15.2.4 在编辑窗口同时编辑两个	
14.3.3 画直线与矩形	(250)	程序模块	(261)
14.3.4 画圆与椭圆	(251)	15.3 QBasic 程序的调试与在线	
14.3.5 图形着色	(252)	帮助	(262)
14.4 本章小结与题例分析	(252)	15.3.1 程序的调试	(262)
14.4.1 本章小结	(252)	15.3.2 在线帮助	(263)
14.4.2 题例分析	(253)	15.4 上机考试内容与题例	(265)
习题十四	(254)	15.4.1 操作系统考试题例	(265)
第 15 章 上机指导	(256)	15.4.2 程序修改题题例	(266)
15.1 QBasic 程序的编辑	(256)	15.4.3 程序编制题题例	(268)
15.1.1 QBasic 的启动	(256)	习题十五	(270)
15.1.2 源程序的输入	(256)	附录一 二级考试大纲	(276)
15.1.3 程序的运行与保存	(256)	附录二 二级考试笔试模拟试卷	(278)
15.1.4 退出 QBasic 集成环境 ...	(257)	附录三 第 2 部分习题参考答案	(290)
15.1.5 输入一个新程序	(257)	参考书目	(296)

第 1 部分 基础知识

第 1 部分共五章。第 1 章介绍了计算机原理有关的概念和基本知识,第 2 章着重介绍计算机系统的组成和计算机安全使用常识,第 3 章以大量的实例介绍 DOS 操作系统的使用,第 4 章介绍考试大纲中新增加的多媒体技术、计算机网络和 Internet 的有关知识,最后一章介绍 Windows 操作系统的初步使用方法。每章后均附有大量的习题,形式与实际考题完全相同。附录中给出习题的参考答案,供读者进行自我测试和自我考核。

本书紧扣考试大纲要求,力求概念清楚,内容精练,注重实用,强调基础。

本书第 1、2 章由边莫英编写;第 3、4 章由高福成编写;第 5 章由曲建民编写。第 1 部分基础知识由高福成统稿。

第1章 计算机基础知识

1.1 计算机概述

计算机这种“人类通用智力工具”,已成为现代社会人类不可缺少的文化了。这种文化的兴起,使科学技术突飞猛进。在世纪之交,人们都在经历着计算机文化的洗礼。

1.1.1 什么是计算机

现代计算机是一种能按照事先编制好的程序(即指令序列)自动、快速、精确地对信息进行加工处理的通用电子设备。它处理的对象是信息,处理的结果也是信息。在这一方面,计算机和人脑有许多相似之处。因为人脑和各种感官也是采集、识别、转换、存储和处理信息的器官,所以人们通常又把计算机称为电脑。

1.1.2 计算机的主要应用领域

计算机的应用非常广泛,从大的方面讲,可以分为数值计算和非数值应用两个领域。非数值应用又包括工厂自动化、办公自动化、各种辅助系统、人工智能等。这里我们将其概括为以下几个方面。

1. 科学计算

以科学技术领域中的问题为主的数值计算。在这类计算中,计算的系数、常数、条件等比较多,输入的数据相对较少,而且计算的问题多数是微分方程、积分方程等,计算量大且数值变化范围宽。在科学技术现代化的今天,科学计算问题变得十分庞大而且复杂。如工程设计、地震预测、火箭发射等领域,都需要依赖计算机进行大量复杂的计算才能得以完成。

2. 数据处理

数据处理又称信息管理,是对信息(即各种形式的数据)进行收集、储存、转换、分组、排序、检索、计算和传输等一系列活动的总和。其基本目的是从大量的、杂乱无章的、难以理解的数据中抽取并推导出对于某些特定的人们来说有意义的、有价值的数据,借以作为决策的依据。数据处理是现代计算机应用最为广泛的领域,如人口统计、银行业务、情报检索以及各种信息管理等。数据处理的特点是,处理的数据量大但计算并不复杂。

3. 过程控制

过程控制又称实时控制,是采用计算机和自动化仪表对某一生产过程中有关工艺设备及操作等实现连续的或非连续的自动检测或监控。其目的是实现优质、高效、低耗、安全与省力的生产。计算机在过程控制中所做的主要工作是:巡回检测、自动记录、采集数据、分析数据、统计制表、监视报警、制定最佳方案并进行自动控制,如炼钢过程的自动控制、飞行器的自动控制、高射火炮的自动瞄准等。

4. 计算机辅助系统

(1) 计算机辅助设计(CAD)

用计算机辅助人们进行设计工作,称为计算机辅助设计(CAD)。计算机辅助设计广泛用于电路设计、机械设计、土木建筑设计、服装设计等。计算机辅助设计使设计工作自动化或半自动化。计算机辅助设计的实现,将是一个完善的包括硬件和软件的系统。

(2) 计算机辅助制造(CAM)

在机械制造业中,利用计算机高速处理和大容量存储的功能,通过各种数值控制机床和设备,自动完成离散产品的加工、装配、检测和包装等生产制造的过程,称为计算机辅助制造(CAM)。例如,在产品生产过程中,利用微型计算机控制机器的运行,处理生产过程中所需的数据,控制和处理材料的流动以及对产品的检测等,以达到提高产品质量、降低成本、缩短生产周期、改善劳动条件的目的。

(3) 计算机辅助工程(CAE)

利用计算机辅助工程建设,称为计算机辅助工程(CAE)。一个工程设计好后,在进行建设过程中,用计算机进行详细的预算、经营管理、原材料购入和仓库管理、工程指挥和调度项目进展情况、人员管理等,可大大提高工作效率、节省开支、推动工程进展。

(4) 计算机辅助教学(CAI)

人们事先把教学内容编成“课件”并输入计算机,使计算机成为“老师”,学生通过与计算机“对话”选择适合自己程度的内容,可使教学内容多样化、形象化、直观化,便于因材施教,使学生由被动学习变为主动学习,大大提高教学效果和学习效果,并节省大批师资。

(5) 计算机辅助测试(CAT)

利用计算机作为辅助工具所进行的测试,称为计算机辅助测试(CAT)。人们把进行测试对象的特性参数及有关计算公式编入程序中,通过仪器、仪表把被测对象的有关参数送入计算机,通过分析运算得出各种有用的参数。

5. 人工智能(AI)

人工智能主要研究如何利用计算机来模拟、延伸和扩展人类的某些智力活动,使计算机具有“推理”和“学习”功能,是计算机应用的一个新领域。如图形识别、语音识别、学习过程、探索过程、逻辑推理以及环境适应等的有关理论和技术。

1.2 计算机中数的表示

1.2.1 数制的概念

数制又称计数制,是指用一组基本符号和一定的使用规则表示数值的方法。在日常生活和计算机中使用的是按进位的方法进行计数的进位计数制,如二进制计数制、十进制计数制、十六进制计数制等。

设数 N 用 $n+m$ 个数码 D_i ($-m \leq i \leq n-1$) 表示,从 D_{n-1} 到 D_{-m} 自左至右排列:

$$N = D_{n-1}D_{n-2} \cdots D_1D_0D_{-1} \cdots D_{-m}$$

小数点隐含在 D_0 与 D_{-1} 之间,则 $D_{n-1}D_{n-2} \cdots D_0$ 为整数部分, $D_{-1}D_{-2} \cdots D_{-m}$ 为小数部分。如果每一个 D_i 都赋一固定的数值 W_i ,则可写成:

$$N = \sum_{i=-m}^{n-1} D_i W_i$$

数码在一个数中的位置称为数位。

某种进位计数制中,每个数位上所能使用的数码符号的个数,称为该种进位计数制的基数。在每个数位上的数码符号所代表的数值大小等于该数位上的数码乘上一个固定的数值。这个固定的数值就是这种计数制的位权数。

例如,人们最常使用的十进制计数制中,有10个不同的数码符号0、1、2、3、4、5、6、7、8、9,即基数为10。每个数码根据它在该数中所处的位置不同,按“逢十进一”决定其实际数值。

例如十进制 $(1234.567)_{10}$,以小数点为界,往左是个位、十位、百位和千位,往右是十分位、百分位和千分位。可表示成:

$$(1234.567)_{10} = 1 \times 10^3 + 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0 + 5 \times 10^{-1} + 6 \times 10^{-2} + 7 \times 10^{-3}$$

1.2.2 计算机中几种常用的进位计数制

1. 二进制

二进制计数制中,只有两个不同的数码符号0和1,即基数为2。每个数码符号根据它在数中所处的位置,按“逢二进一”决定其实际数值。

例如:

$$\begin{aligned} (101011.101)_2 &= 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + \\ &\quad 1 \times 2^{-3} \\ &= (43.625)_{10} \end{aligned}$$

由此可归纳出任意一个二进制数N,可写成如下形式:

$$\begin{aligned} (N)_2 &= D_{n-1} \times 2^{n-1} + D_{n-2} \times 2^{n-2} + \cdots + D_1 \times 2^1 + D_0 \times 2^0 + D_{-1} \times 2^{-1} + \\ &\quad D_{-2} \times 2^{-2} + \cdots + D_{-m} \times 2^{-m} \\ &= \sum_{i=-m}^{n-1} D_i \times 2^i \end{aligned}$$

式中 D_i 为数位上的数码,取值为0或1; n 为整数位的个数; m 为小数位个数; i 为数位的编号;2是二进制的基数; $2^{n-1}, 2^{n-2}, \dots, 2^1, 2^0, \dots, 2^{-m}$ 是二进制的位权数。

在计算机中,数的存储、运算和传输都使用二进制,这是因为0和1两个数码可以十分容易地用电子的、磁的、光的或机械的元器件来表示和实现。

2. 八进制

八进制计数制中,使用8个不同的数码符号0、1、2、3、4、5、6、7,即基数为8。每一个数码符号根据它在该数中所处的位置,按“逢八进一”来决定其实际数值。例如:

$$(123.456)_8 = 1 \times 8^2 + 2 \times 8^1 + 3 \times 8^0 + 4 \times 8^{-1} + 5 \times 8^{-2} + 6 \times 8^{-3} = (83.58984375)_{10}$$

由此可归纳出任意一个八进制数N,可写成如下形式:

$$\begin{aligned} (N)_8 &= D_{n-1} \times 8^{n-1} + D_{n-2} \times 8^{n-2} + \cdots + D_1 \times 8^1 + D_0 \times 8^0 + D_{-1} \times 8^{-1} + \cdots + D_{-m} \times 8^{-m} \\ &= \sum_{i=-m}^{n-1} D_i \times 8^i \end{aligned}$$

式中 D_i 为数位上的数码,取值范围为0~7; n 为整数位的个数; m 为小数位的个数; i 为数位的编号;8是八进制的基数; $8^{n-1}, 8^{n-2}, \dots, 8^1, 8^0, \dots, 8^{-m}$ 是八进制的位权数。

八进制是计算机中常用的一种计数制,它可以弥补二进制数书写位数过长的不足。

3. 十六进制

十六进制计数制中,需要 16 个不同的数码符号,它们是 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F,即基数为 16。每一个数码符号根据它在数中所处的位置,按“逢十六进一”来决定其实际数值。例如:

$$(B1C.F4)_{16} = B \times 16^2 + 1 \times 16^1 + C \times 16^0 + F \times 16^{-1} + 4 \times 16^{-2} = (2844.953125)_{10}$$

由此可归纳出任意一个十六进制数 N,可写成如下形式:

$$(N)_{16} = D_{n-1} \times 16^{n-1} + D_{n-2} \times 16^{n-2} + \dots + D_1 \times 16^1 + D_0 \times 16^0 + \dots + D_{-m} \times 16^{-m}$$

$$= \sum_{i=-m}^{n-1} D_i \times 16^i$$

式中 D_i 为数位上的数码,取值范围为 0~F; n 为整数位的个数; m 为小数位的个数; i 为数位的编号; 16 是十六进制的基数; $16^{n-1}, 16^{n-2}, \dots, 16^1, 16^0, \dots, 16^{-m}$ 是十六进制的位权数。

十六进制是计算机中常用的一种计数制,它可以弥补二进制数书写位数过长的不足。

1.2.3 不同数制间的转换

计算机中使用的是二进制数,如果要处理其他进制的数,就必须先将其转换成二进制数再进行处理。当输出处理结果时,又要将二进制数转换成其他进制的数。

数制间的转换,其实就是基数间的转换。转换依据的原则是:如果两个有理数相等,则两个数的整数部分和小数部分一定分别相等。因此,各数制间的相互转换,一般总是将整数部分和小数部分按转换方法分别进行的。

1. 二进制、八进制和十六进制数转换成十进制数

把二进制数、八进制数和十六进制数转换为十进制数,通常利用下述公式:

$$N = \sum_{i=-m}^{n-1} D_i W^i$$

按位权展开相加的方法,式中 D_i 表示二进制、八进制或十六进制的数码符号, W^i 表示它们的位权数。公式右端采用十进制运算时,其结果 N 就是与二进制、八进制或十六进制数等值的十进制数。

例如:将下列不同进制的数 $(101011.101)_2$ 、 $(123.456)_8$ 和 $(AB1.B2)_{16}$ 转换成十进制数。

$$(1) N_1 = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$$

$$= (43.625)_{10}$$

$$\text{即 } (101011.101)_2 = (43.625)_{10}$$

$$(2) N_2 = 1 \times 8^2 + 2 \times 8^1 + 3 \times 8^0 + 4 \times 8^{-1} + 5 \times 8^{-2} + 6 \times 8^{-3} = (83.58984375)_{10}$$

$$\text{即 } (123.456)_8 = (83.58984375)_{10}$$

$$(3) N_3 = A \times 16^2 + B \times 16^1 + 1 \times 16^0 + B \times 16^{-1} + 2 \times 16^{-2} = (2737.6953125)_{10}$$

$$\text{即 } (AB1.B2)_{16} = (2737.6953125)_{10}$$

2. 十进制整数转换成二进制、八进制和十六进制整数

将十进制整数转换成二进制整数的规则是“除以 2 取余数”法,即不断地用 2 去除十进制整数,得余数为 0 或 1,则相应数位的数码即为 0 或 1。依次从低数位到高数位求得相应的数码,一直进行到用 2 去除得到的商为 0 时为止。最后一次得到的余数为最高位。然后,再依次从高位到低位顺序写出这个转换后的二进制数。

例如,将十进制整数 135 转换成二进制整数:

2	135		余数	
2	67	1		↑ 二进制数的低位
2	33	1		
2	16	1		
2	8	0		
2	4	0		
2	2	0		
2	1	0		↑ 二进制数的高位
	0	1		

得: $(135)_{10} = (10000111)_2$

类似地,把十进制整数转换成八进制整数和十六进制整数,其转换规则是“除以8取余数”法和“除以16取余数”法。

例如,将十进制整数756转换成八进制整数:

8	756		余数	
8	94	4		↑ 八进制数的低位
8	11	6		
8	1	3		
	0	1		↑ 八进制数的高位

得: $(756)_{10} = (1364)_8$

例如,将十进制整数3456转换成十六进制整数:

16	3456		余数	
16	216	0		↑ 十六进制数的低位
16	13	8		
	0	D		↑ 十六进制数的高位

得: $(3456)_{10} = (D80)_{16}$

3. 十进制小数转换成二进制、八进制和十六进制小数

将十进制小数转换为二进制小数的规则是“乘以2取整数”法,即不断地用2去乘十进制小数,每乘一次取其乘积的整数部分。整数部分为0或1,则相应数位的数码就是0或1。从小数的首位到末位依次进行,直至最后一次乘积为1.00...0或满足精度要求时为止。然后依次从首位到末位写成 $0.D_{-1}D_{-2}\cdots D_{-m}$ 的形式。

例如,将十进制小数0.6537转换成二进制小数,精确到小数点后第六位:

	整数	0.6537
二进制小数首位	1.....	$\times) \quad 2$ 1.3074 0.3074
	0.....	$\times) \quad 2$ 0.6148 0.6148
	1.....	$\times) \quad 2$ 1.2296 0.2296
	0.....	$\times) \quad 2$ 0.4592 0.4592
	0.....	$\times) \quad 2$ 0.9084 0.9084
二进制小数末位	1.....	$\times) \quad 2$ 1.8168 0.8168

得: $(0.6537)_{10} \approx (0.101001)_2$

类似地,可以把十进制小数转换成八进制和十六进制小数,其转换规则是:“乘以 8 取整数”法和“乘以 16 取整数”法。

例如,把十进制小数 0.6537 转换成八进制小数,精确到小数点后五位:

	整数	0.6537
八进制小数首位	5.....	$\times 8$ 5.2296 0.2296
	1.....	$\times 8$ 1.8368 0.8368
	6.....	$\times 8$ 6.6944 0.6944
	5.....	$\times 8$ 5.5552 0.5559
八进制小数末位	4.....	$\times 8$ 4.4416 0.4416

得: $(0.6537)_{10} \approx (0.51654)_8$ 。

例如:将十进制小数 0.53975 转换成十六进制小数,精确到小数点后四位:

	整数	0.53975
十六进制小数首位	8.....	$\times 16$ 8.63600 0.63600
	A.....	$\times 16$ 10.17600 0.17600
	2.....	$\times 16$ 2.81600 0.81600
十六进制小数末位	D.....	$\times 16$ 13.05600 0.05600

得: $(0.53975)_{10} \approx (0.8A2D)_{16}$ 。

如果要将一个十进制混合小数转换成二进制、八进制和十六进制混合小数,那么只需将十进制混合小数的整数部分和小数部分分别进行转换,然后再组合在一起即可,这里不再赘述。

4. 二进制数与八进制和十六进制数的互相转换

二进制的不足之处是书写时太长,且不易读,因此通常使用八进制或十六进制数来弥补。二进制的基数是 2,八进制和十六进制的基数分别是 8 和 16,它们与 2 有整数次幂的关系,即 $8=2^3$, $16=2^4$ 。所以 1 位八进制数正好表示 3 位二进制数,1 位十六进制数正好表示 4 位二进制数,因此,将二进制数转换成八进制和十六进制数非常容易:只需将二进制数以小数点为界,向左每相邻 3 位或 4 位为一组构成 1 位八进制或十六进制整数,向右每相邻 3 位或 4 位为一组构成 1 位八进制或十六进制小数。向左若不足 3 位或 4 位时,则左边添“0”到满 3 位或 4 位,向右不足 3 位或 4 位时,则右边添“0”到满 3 位或 4 位。

例如:将二进制数 1110101101.1010011011 转换成八进制和十六进制数。

转换成八进制数时,每 3 位二进制数对应一位八进制数,不足 3 位补“0”到满 3 位,即:

001	110	101	101.	101	001	101	100
↓	↓	↓	↓	↓	↓	↓	↓
1	6	5	5.	5	1	5	4