

724

724
13305(2)

全国计算机等级考试

二 级 教 程

基础知识和 C 程序设计

(第二版)

刘瑞挺 主编
高福成 编著

南开大学出版社

• 天津 •

内容提要

本书是根据教育部考试中心制定的《全国计算机等级考试二级考试大纲》中有关C语言程序设计的要求编写的。本书以应用为目的,以程序设计为主线,系统介绍了C语言及其程序设计技术。全书共十一章,包括C语言的基本数据类型、数据运算、程序控制结构、数组、函数、指针、复合数据类型、文件和编译预处理等。书中突出了重点,分解了难点,配以大量的应用实例和多种类型的习题,方便自学,是应试人员考前必读教材,也可供软件开发人员参考。

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C程序设计 / 刘瑞挺主编; 高福成, 李宗耀, 李兰友编著. —2版. —天津: 南开大学出版社, 2001. 12 (2002. 3重印)
(全国计算机等级考试系列丛书)
ISBN 7-310-01583-5

I. 全... II. ①刘... ②高... ③李... ④李...
III. ①电子计算机—水平考试—教材 ②C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070841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 河北省昌黎县人民胶印厂印刷

经销 全国各地新华书店

版次 2001年12月第2版

印次 2002年6月第5次印刷

开本 787mm×1092mm 1/16

印张 21.75

字数 539千字

印数 20001—25000

定价 30.00元

前 言

C 语言是计算机程序设计语言中的小字辈,但由于其卓越的性能,使它风靡全世界,成为最受欢迎的语言之一。当今流行的面向对象语言 C++ 及 Internet 网上的 JAVA 语言就来源于 C,参加全国计算机等级考试三级(偏软)的应试者也必须掌握 C 语言。

目前参加计算机等级考试的应试者中,选择 C 语言的人数不断增加。学习计算机语言的唯一目的是应用,而应用要通过程序设计来体现。进行程序设计,需要很强的逻辑思维能力,是一种极富创造性的智力劳动。可以这样认为,语言是一种技能,程序设计是一门科学。因此,任何计算机语言及其程序设计的基本特点就是理论性和实践性并重,教学上应强调科学训练与技能培养并存。基于这一认识,本书的宗旨是以应用为目的、以程序设计为主线,系统介绍 C 语言及其程序设计技术,让读者在大量的程序设计实践中自然而然地熟悉和掌握语言。

本书是根据教育部考试中心制定的《全国计算机等级考试二级考试大纲》中有关 C 语言程序设计的要求编写的,从选材上注意了以下几点:

1. 立足基点,掌握制高点。基点是会用语言来进行程序设计,学会一大批基本的程序设计单元;制高点是掌握程序设计的基本思维方法和一定的技巧。因此,本书根据实用和适用的原则,精选大量的例题对 C 语言的各种语法现象加以阐述。这些例题不是零星的程序段,而是完整的实际题目,以便于读者建立整体的观点。

2. 突出重点,分解难点。本书的重点章节是第 4、6、7、10 章,书中对此进行了较大篇幅的阐述。对那些难点问题,如指针、数据在函数间的传递、存储类型等,则分散在多个章节内讨论。

3. 为了使读者能及时考查自己的学习效果,每一章都安排了大量的习题,包括选择题、填空题、编程题和程序调试题共四种题型,可满足读者参加等级考试、发挥程序设计的创造性以及上机实践的需要。

根据 C 语言自身的特点,本书从内容上作了如下安排:

第 1 章介绍简单的 C 程序设计;第 2、5、9 章从易到难介绍 C 语言的各种数据结构;第 3 章专门介绍 C 语言的运算功能;第 4 章集中介绍 C 语言的控制结构,体现结构化程序设计的特点;第 7、10 章介绍函数和文件,这是模块化

程序设计的需要;第6章介绍指针,这是C语言的精华和特色,也是学习的难中之难;第8章介绍了C语言环境中的特殊问题,即编译预处理和分割编译。附录部分给出了C语言的运算符、ASCII代码表、Turbo C常用库函数,便于读者查阅。习题(选择题、填空题)参考答案。

南开大学出版社的领导和编辑为本书的出版付出了大量的劳动,在此表示由衷的感谢。

编著者 谨识

1999.5

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

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 语言编程

第 6 章 C 程序设计初步知识	(105)	习题六	(117)
6.1 一个简单的 C 程序	(105)	第 7 章 基本数据类型、常量和变量	(119)
6.2 C 程序的基本结构	(105)	7.1 基本数据类型	(119)
6.2.1 基本单词	(106)	7.1.1 五种基本数据类型	(119)
6.2.2 C 语句	(107)	7.1.2 基本数据类型的存储方式 和取值范围	(119)
6.2.3 函数	(107)	7.1.3 基本数据类型的修饰	(120)
6.2.4 C 程序的书写风格	(108)	7.2 常量及其类型	(121)
6.2.5 库函数和标题文件	(108)	7.2.1 整型常数	(121)
6.3 简单的 C 程序设计	(109)	7.2.2 浮点常数及双精度型 常数	(122)
6.3.1 格式输出函数 printf() 简介	(109)	7.2.3 字符常量	(123)
6.3.2 格式输入函数 scanf() 简介	(110)	7.2.4 字符串常量	(124)
6.3.3 简单的赋值语句	(111)	7.3 变量及其类型	(125)
6.3.4 简单程序设计举例	(111)	7.3.1 变量名	(125)
6.4 C 程序的编译与运行	(112)		
6.5 Turbo C 2.0 集成环境	(113)		

7.3.2 变量的数据类型	(125)	习题八	(159)
7.3.3 变量的定义	(125)	第9章 结构化程序设计和C语言程序	
7.3.4 变量的存储类型	(127)	控制结构	(162)
7.3.5 变量的初始化	(129)	9.1 结构化程序设计的概念	(162)
7.4 符号常量	(130)	9.2 if 语句	(162)
7.4.1 宏定义	(130)	9.2.1 if-else 语句	(162)
7.4.2 const 修饰	(131)	9.2.2 if 语句的嵌套	(164)
7.5 不同类型数据的输入输出		9.2.3 if-else if-else 语句	(166)
.....	(132)	9.2.4 条件运算符和条件	
7.5.1 printf() 函数	(132)	表达式	(167)
7.5.2 scanf() 函数	(135)	9.3 switch 语句和 break 语句	(168)
7.5.3 字符输出输入函数		9.3.1 switch 语句	(168)
(putchar(), getchar())		9.3.2 break 语句	(170)
.....	(137)	9.3.3 switch 语句的使用	(170)
7.5.4 单字符输入函数(getche()		9.4 循环语句	(172)
和 getch())	(139)	9.4.1 for 语句	(172)
习题七	(139)	9.4.2 while 语句	(174)
第8章 运算符和表达式	(142)	9.4.3 do-while 语句	(176)
8.1 算术运算	(142)	9.4.4 循环的嵌套	(177)
8.1.1 二元算术运算符	(142)	9.5 转移控制语句(break、exit()、	
8.1.2 一元算术运算符	(143)	continue、goto)	(178)
8.1.3 算术表达式	(143)	9.5.1 break 语句	(178)
8.1.4 强制类型转换	(145)	9.5.2 exit() 函数	(180)
8.2 赋值运算	(146)	9.5.3 continue 语句	(180)
8.2.1 赋值运算符	(146)	9.5.4 goto 语句和标号	(181)
8.2.2 赋值表达式	(146)	习题九	(182)
8.3 逗号运算(顺序运算)	(148)	第10章 数组和字符串	(187)
8.4 关系运算和逻辑运算	(150)	10.1 数组的定义和初始化	(187)
8.4.1 关系运算符和逻辑		10.1.1 一维数组	(187)
运算符	(150)	10.1.2 二维数组	(187)
8.4.2 关系表达式、相等表达式		10.1.3 字符型数组和	
和逻辑表达式	(151)	字符串	(188)
8.5 测试数据长度运算符		10.1.4 多维数组	(188)
sizeof	(152)	10.1.5 数组的存储结构	(188)
8.6 位运算	(153)	10.1.6 数组的初始化	(189)
8.6.1 按位逻辑运算	(154)	10.1.7 隐含尺寸数组	(191)
8.6.2 移位运算	(155)	10.2 数组的赋值	(191)
8.6.3 位操作的应用	(156)	10.3 数组的输入和输出	(192)
8.7 数学函数	(157)	10.3.1 数值数组的输入和	

输出..... (192)	第 12 章 程序的模块结构和
10.3.2 字符数组的输入和	C 函数 (233)
输出..... (193)	12.1 C 程序的模块结构 (233)
10.3.3 字符串处理函数 (195)	12.1.1 模块化程序设计的
10.4 数组的应用 (197)	特点..... (233)
习题十 (201)	12.1.2 C 程序的模块结构..... (233)
第 11 章 指针..... (206)	12.2 C 函数的定义 (234)
11.1 地址和指针 (206)	12.3 C 函数的调用 (236)
11.1.1 地址..... (206)	12.4 函数间数据传递方式之一
11.1.2 取地址运算符和访问	——哑实结合 (238)
地址运算符..... (206)	12.4.1 哑实结合的概念..... (238)
11.1.3 指针的概念..... (209)	12.4.2 简单变量的哑实
11.2 用指针访问变量 (211)	结合..... (241)
11.3 指针的运算 (212)	12.4.3 数组(含字符串)的哑实
11.3.1 指针与整数相加减(指针	结合..... (242)
的移动) (212)	12.4.4 函数的哑实结合
11.3.2 两个同类型指针	(函数指针)..... (246)
相减..... (214)	12.5 函数间数据传递方式之二
11.3.3 指针的比较..... (214)	——函数返回值 (248)
11.4 用指针访问一维数组 (214)	12.5.1 返回数值和字符的基类型
11.5 用指针访问字符串 (216)	函数 (248)
11.6 用指针访问二维数组 (218)	12.5.2 返回地址的指针型
11.6.1 将二维数组视为一维	函数 (249)
数组处理..... (218)	12.6 函数间数据传递方式之三
11.6.2 行指针..... (220)	——全局变量 (250)
11.6.3 指针数组..... (221)	12.7 函数的嵌套调用和递归
11.7 二级指针 (224)	调用 (251)
11.7.1 一级指针和二级	12.7.1 函数的嵌套调用..... (251)
指针..... (224)	12.7.2 函数的递归调用..... (252)
11.7.2 用二级指针访问二维	12.8 main()函数的参数和
数组..... (224)	返回值 (255)
11.7.3 用二级指针访问一维	习题十二 (257)
数组..... (226)	第 13 章 编译预处理和分割
11.8 用指针进行内存动态	编译..... (262)
分配 (226)	13.1 宏定义 (262)
11.8.1 内存动态分配的	13.1.1 宏定义(#define)的
含义 (226)	形式和功能..... (262)
11.8.2 内存动态分配函数 (227)	13.1.2 宏定义的解除..... (265)
习题十一 (228)	13.2 文件包含 (266)

13.2.1 文件包含(#include)的形式和功能.....	(266)	15.1.1 文件的逻辑结构.....	(300)
13.2.2 文件包含的嵌套.....	(268)	15.1.2 文件的存取方式.....	(300)
13.3 条件编译	(268)	15.1.3 标准设备文件和数据文件.....	(301)
13.4 分割编译	(271)	15.2 文件的打开与关闭	(302)
习题十三	(273)	15.2.1 文件指针.....	(302)
第14章 结构(结构体)和联合(共用体).....	(275)	15.2.2 文件的打开.....	(302)
14.1 结构类型和结构	(275)	15.2.3 文件的关闭.....	(304)
14.1.1 结构类型和结构的定义.....	(275)	15.3 文件读写函数	(305)
14.1.2 结构成员的访问和结构的初始化.....	(277)	15.3.1 字符读写函数(getc()、fgetc()和putc()、fputc())	(305)
14.1.3 嵌套结构.....	(279)	15.3.2 字符串读写函数(fgets()和fputs()).....	(307)
14.1.4 结构的存储.....	(280)	15.3.3 格式读写函数(fscanf()和fprintf())	(308)
14.2 结构数组	(281)	15.3.4 数据块读写函数(fread()和fwrite())	(310)
14.3 指向结构变量和结构数组的指针	(281)	15.4 文件检测函数	(312)
14.4 结构在函数间的传递	(284)	15.5 文件的存取	(312)
14.4.1 结构在函数间的传递(哑实结合).....	(284)	15.5.1 文件的定位.....	(312)
14.4.2 结构型函数和结构指针型函数(函数返回值方式)	(287)	15.5.2 文件的顺序存取和随机存取.....	(314)
14.5 递归结构	(289)	习题十五	(317)
14.6 联合(共用体)	(291)	附录一 C运算符的优先级与结合性	(321)
14.7 类型定义 typedef)	(295)	附录二 ASCII 代码表	(322)
习题十四	(297)	附录三 Turbo C 2.0 常用库函数及其标题文件	(323)
第15章 文件.....	(300)	附录四 第2部分习题答案	(327)
15.1 文件概述	(300)		

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础知识,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情况进行情况、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:

$$\begin{aligned} (N)_{16} &= D_{n-1} \times 16^{n-1} + D_{n-2} \times 16^{n-2} + \cdots + D_1 \times 16^1 + D_0 \times 16^0 + \cdots + D_{-m} \times 16^{-m} \\ &= \sum_{i=-m}^{n-1} D_i \times 16^i \end{aligned}$$

式中 D_i 为数位上的数码,取值范围为 $0 \sim F$; 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。

$$\begin{aligned} (1) N_1 &= 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= (43.625)_{10} \end{aligned}$$

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

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

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

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

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

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 $0.D_1D_2\cdots D_m$ 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