



普通高等教育“十一五”国家级规划教材

C语言程序设计

C YUYAN CHENGXU SHEJI

黄迪明 许家珩 胡德昆 编著



电子科技大学出版社

普通高等教育“十一五”国家级规划教材

C 语言程序设计

黄迪明 许家珩 胡德昆 编著

电子科技大学出版社

图书在版编目 (CIP) 数据

C 语方程序设计 / 黄迪明等编著. —成都: 电子科技大学出版社, 2008.7

普通高等教育“十一五”国家级规划教材

ISBN 978-7-81114-817-6

I. C… II. 黄… III. C 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2008) 第 075622 号

内 容 简 介

本书详细介绍 C 语言及其程序设计方法。全书共 10 章, 主要内容包括: C 语言程序设计基础; 基本数据类型与运算; 控制语句; 数组与结构; 指针; 函数; 变量的存储类型; 位域、联合、枚举及定义类型; 输入输出及文件管理。此外, 本书还为读者介绍了 Turbo C 及 Visual C++ 编译系统的使用及标准库函数在动态内存分配、BIOS 接口及 DOS 系统调用和声音中的各类实例。

本书为读者展示了 C 语言灵活、精致的编程方法和在工程、科研中的应用, 力求做到 C 语言知识和应用开发能力的融会贯通。为了帮助读者学习, 每章设有小结和习题, 并配有程序设计题解与上机指导辅助教材。

本书是高等学校程序设计方法的入门教材, 亦可作为学习 C 语言人员的自学教材。

普通高等教育“十一五”国家级规划教材

C 语言程序设计

黄迪明 许家珩 胡德昆 编著

出 版: 电子科技大学出版社 (成都市一环路东一段 159 号电子信息产业大厦 邮编: 610051)

策划编辑: 吴艳玲

责任编辑: 吴艳玲

主 页: www.uestcp.com.cn

电子邮箱: uestcp@uestcp.com.cn

发 行: 新华书店经销

印 刷: 成都蜀通印务有限责任公司

成品尺寸: 185mm×260mm 印张 26 字数 650 千字

版 次: 2008 年 7 月第一版

印 次: 2008 年 7 月第一次印刷

书 号: ISBN 978-7-81114-817-6

定 价: 39.00 元

■ 版权所有 侵权必究 ■

◆ 本社发行部电话: 028-83202463; 本社邮购电话: 028-83208003。

◆ 本书如有缺页、破损、装订错误, 请寄回印刷厂调换。

◆ 课件下载在我社主页“下载专区”。

前 言

C 语言是一种通用的程序设计语言。它的结构简单,数据类型丰富,运算灵活方便。用它编写的程序,具有速度快、效率高、代码紧凑、可移植性好等优点,能够有效地用来编制各种系统软件和应用软件,是当今最为流行的计算机编程语言之一。

本教材以美国国家标准 C 语言(ANSI C)为基本内容,以当前广为使用的 Visual C++6.0 编译系统为实现的版本,全面系统地介绍了 C 语言及其程序设计方法。本书的第一版是国家九五电子信息类规划教材,第二版是国家“十一五”计算机类本科规划教材。全书共 10 章,主要内容包括:C 语言程序设计基础;基本数据类型与运算;控制语句;数组与结构;指针;函数;变量的存储类型;位域、联合、枚举及定义类型;输入输出及文件管理。此外,本书还为读者介绍了 Turbo C 及 Visual C++ 编译系统的使用及标准库函数在动态内存分配、BIOS 接口及 DOS 系统调用和声音中的各类实例。本书为读者展示了 C 语言灵活、精致的编程方法和在工程、科研中的应用,力求做到 C 语言知识和应用开发能力的融会贯通。

本教材是作为程序设计的入门教材而编写的,如果读者对第 1 章中的第一节和第二节内容已经了解,可直接阅读后面的内容。本教材的参考学时数为 68 学时(含上机 20 学时),书中标注“*”的内容,可根据教学实际情况进行取舍,既可作为基本教学内容的扩展,亦可作为自学内容。此外,本书在各章节重要知识点添加了良好编程习惯和编程错误提示,以便初学者能借鉴从而提高学习效率和编程能力。

为了帮助读者更好地理解 C 语言,提高读者开发应用程序的能力,本教材以典型案例图书管理系统应用程序贯穿各章内容,使读者循序渐进地学习和掌握 C 语言开发应用程序的方法与技巧。

为了帮助读者学习,每章设有小结和习题,并配有程序设计题解与上机指导辅助教材,重点介绍了编译系统的使用方法,使学生在课堂学习过程中能迅速掌握 C 语言程序的编制、编译、调试和运行方法。

本教材由黄迪明、许家珩、胡德昆编写。黄迪明编写第 1 章、第 6 章、第 7 章、第 8 章及第 9 章;许家珩编写第 3 章、第 4 章及第 5 章;胡德昆编写第 2 章、第 10 章及附录;阿都建华负责编写全书案例。电子科技大学李玉柏教授、杨国炜教授、张建中副教授对本书的编写提出了各种有益的建议。本书在编写过程中,还得到了杜海涛、王波、刘家芬、陈琼、张大愚、邹波、曾烨等人的热情帮助。在此对他们及所有为本书的出版付出了辛勤劳动的同志表示衷心的感谢。

由于编者水平有限,书中难免存在一些缺点和错误,殷切希望广大读者批评指正。

编著者

2008 年 5 月于电子科技大学

目 录

第 1 章 C 语言程序设计基础知识	1
1.1 计算机基础知识概述	1
1.1.1 计算机与信息社会	1
1.1.2 计算机中信息的表示	2
1.1.3 计算机系统的组成	7
1.2 软件开发过程	10
1.2.1 计算机求解问题的步骤	10
1.2.2 算法的表示	14
1.3 C 语言概述	17
1.3.1 C 语言简史及特点	17
1.3.2 基本程序结构	19
1.3.3 基本语法单位	22
1.4 C 语言程序的编写和运行	24
1.4.1 C 程序的编写和运行步骤	24
1.4.2 Visual C++ 6.0 介绍	26
1.5 案例研究	29
1.6 常见的编程错误	31
小结一	32
习题一	33
第 2 章 基本数据类型及运算	35
2.1 基本数据类型	35
2.1.1 整型	36
2.1.2 浮点型	37
2.1.3 字符型	37
2.2 常量	37
2.2.1 整型常量	37
2.2.2 浮点型常量	38

2.2.3	字符型常量	39
2.2.4	字符串常量	40
2.2.5	符号常量	41
2.3	变量	42
2.3.1	变量的定义	42
2.3.2	变量的初始化	43
2.3.3	变量地址	43
2.4	运算符与表达式	44
2.4.1	算术运算符和算术表达式	45
2.4.2	赋值运算符和赋值表达式	47
2.4.3	关系运算符和关系表达式	49
2.4.4	逻辑运算符和逻辑表达式	52
2.4.5	位运算符和位表达式	54
2.4.6	条件运算符和条件表达式	59
2.4.7	逗号运算符和逗号表达式	61
2.4.8	其他运算符	62
2.5	混合运算与类型转换	63
2.5.1	自动类型转换	63
2.5.2	强制类型转换	64
2.5.3	赋值运算中的类型转换	65
2.6	运算的优先级与结合性	68
2.6.1	运算符汇总	68
2.6.2	运算符嵌套	69
2.6.3	表达式的运算顺序	70
2.7	数据的输入输出	71
2.7.1	字符输出函数 putchar()和格式输出函数 printf()	72
2.7.2	字符输入函数 getchar()和格式输入函数 scanf()	78
2.8	案例研究	82
小结二		83
习题二		84
第3章	控制语句	90
3.1	程序的三种基本结构	90
3.2	复合语句	91
3.3	if 条件分支语句	92

3.3.1 if 流程	92
3.3.2 if else 流程	94
3.3.3 else if 流程	96
3.3.4 if 语句嵌套	98
3.4 switch 多路开关语句	100
3.5 for 循环语句	105
3.6 while 语句和 do while 语句	110
3.6.1 while 语句	110
3.6.2 do while 语句	116
3.7 循环嵌套	119
3.8 break, continue 和 goto 语句	120
3.8.1 break 语句	121
3.8.2 continue 语句	122
3.8.3 goto 语句	123
3.9 案例研究	125
小结三	130
习题三	131
第 4 章 数组和结构	133
4.1 一维数组	133
4.1.1 一维数组的定义	134
4.1.2 一维数组元素的引用	135
4.1.3 一维数组的初始化	136
4.1.4 一维数组程序举例	137
4.2 二维数组	142
4.2.1 二维数组的定义	142
4.2.2 二维数组元素的引用	143
4.2.3 二维数组的初始化	144
4.3 字符数组	145
4.3.1 字符数组的定义和初始化	146
4.3.2 字符数组的输入输出	147
4.3.3 与字符串处理有关的几个函数	149
4.3.4 字符串应用举例	154
4.4 结构及结构变量的定义与访问	156
4.4.1 结构及结构变量的定义	156

4.4.2 结构成员的访问	159
4.4.3 结构变量的初始化	160
4.5 结构数组	161
4.6 程序举例	164
4.7 案例研究	172
小结四	175
习题四	176
第 5 章 指针	180
5.1 指针的概念和定义	180
5.1.1 指针的概念	180
5.1.2 指针的定义	181
5.1.3 指针的赋值	181
5.2 指针运算	185
5.3 指针和数组	189
5.3.1 指针与一维数组	189
5.3.2 指针与结构数组	193
5.4 字符串指针	197
5.4.1 指向字符数组的指针	197
5.4.2 指向字符串常量的指针	200
5.5 指针数组	202
5.6 指向指针的指针	206
5.7 程序举例	208
5.8 案例研究	214
小结五	217
习题五	218
第 6 章 函数	221
一、模块化程序设计方法	221
二、函数的分类	222
三、主函数	223
6.1 函数定义和调用	223
6.1.1 函数定义	223
6.1.2 函数调用	227
6.2 函数参数传递	229
6.2.1 传值调用	229

6.2.2 传址调用	230
6.3 函数与数组	232
6.3.1 数组元素作函数实参	232
6.3.2 数组名作为函数参数	233
6.4 函数与指针	237
6.4.1 返回指针的函数	239
*6.4.2 指向函数的指针	240
6.5 函数与结构	241
6.5.1 结构指针及结构变量的传址调用	242
6.5.2 结构型函数	243
6.5.3 结构指针型函数	244
6.6 递归函数	245
*6.7 命令行参数	251
6.8 标准库函数	253
6.9 程序举例	255
6.10 案例研究	260
小结六	263
习题六	264
第 7 章 变量的存储类型	266
7.1 C 程序的结构	266
7.1.1 C 程序的组成	266
7.1.2 变量的作用域	267
7.1.3 变量的存储类型	267
7.2 内部变量	268
7.3 外部变量	269
7.3.1 在同一个源程序文件中使用外部变量	269
7.3.2 在不同源程序文件中使用外部变量	273
7.4 静态变量	274
7.4.1 静态局部变量	274
7.4.2 静态全局变量	276
7.5 寄存器变量	276
7.6 变量的初始化	277
7.7 动态内存分配函数	278
7.8 预处理功能	279

7.8.1 宏替换——#define	280
7.8.2 包含文件——#include	286
7.8.3 条件编译——#if、#ifdef、#ifndef	287
7.9 程序举例	290
小结七	294
习题七	294
第 8 章 位域、联合、枚举和定义类型	298
8.1 位域及结构嵌套	298
8.1.1 位域	298
8.1.2 结构嵌套	301
8.2 联合	302
8.3 枚举	308
8.4 定义类型——typedef	311
小结八	313
习题八	313
第 9 章 输入、输出及文件管理	316
9.1 流和文件	316
9.2 控制台 I/O	317
9.2.1 字符输入输出——getchar()、putchar()	317
9.2.2 字符串输入输出——gets、puts	319
9.3 文件	320
9.3.1 打开文件函数——fopen	321
9.3.2 关闭文件函数——fclose	322
9.3.3 标准流式文件 stdin、stdout 和 stderr	322
9.4 用于文件的输入输出函数	323
9.4.1 单字符输入输出——getc()、putc()	324
9.4.2 行输入输出——fgets()、fputs()	326
9.4.3 数据块的输入输出——fread()、fwrite()	327
9.4.4 流式文件数据的格式化输入输出——fprintf()、fscanf()	329
9.4.5 文件的随机访问——fseek()	329
9.5 程序举例	331
9.6 案例研究	334
小结九	342
习题九	343

第 10 章 C 高级程序应用	345
*10.1 链表	345
10.1.1 引用自身的结构	345
10.1.2 单向链表	345
10.1.3 双向链表	348
10.1.4 循环链表	350
10.1.5 链表应用程序举例	350
*10.2 与系统有关的库函数	353
10.2.1 BIOS 接口调用函数	355
10.2.2 DOS 系统调用函数	356
10.2.3 案例研究	361
10.3 声音程序	366
10.3.1 声音函数	366
10.3.2 音乐	367
10.3.3 应用举例	368
10.4 案例研究	369
附录	395
附录 A C 语言的关键字	395
附录 B 运算符的优先级与结合性 (见表 B-1)	395
附录 C 常用字符 ASCII 表 (见表 C-1)	396
附录 D C 语言中常用库函数	397
参考文献	404

第1章 C 语言程序设计基础知识

C 语言是一种通用的程序设计语言。随着 C 语言在开发系统软件和应用软件中的广泛应用，它已成为当今世界上最流行的语言之一。

本章简要介绍与 C 语言程序设计相关的基础知识。

1.1 计算机基础知识概述

1.1.1 计算机与信息社会

电子计算机的出现和发展是当代科学技术的最伟大成就之一。从第一台计算机问世以来，计算机的发展取得了令人瞩目的成就。今天，计算机科学与技术已作为一门先进的学科独立存在；计算机工业已成为改造传统工业、振兴国民经济的重要支柱产业；计算机在科学研究、工农业生产、国防建设以及社会各个领域的广泛应用已成为国家现代化的一个重要标志。

人类在改造客观世界的过程中，已经认识到文字、物质材料和能源是构成世界的三大要素。在人类社会文明的发展过程中离不开信息交流，计算机作为信息处理工具，在信息存储、处理和传播方面起着重要的作用。人类历史上曾经历了四次信息革命。第一次信息革命是语言的使用；第二次信息革命是文字的使用；第三次信息革命是印刷术的发明；第四次信息革命是电话、广播电视的使用。而从 20 世纪 60 年代开始的第五次信息革命产生的社会技术——信息技术，则是计算机、通信与控制技术相结合的技术，它标志着人类正迈向信息社会。

文化是人类在社会历史发展中所创造的物质财富和精神财富的总和。可以认为，文化离不开语言。计算机技术已经创造并且还在继续创造出不同于传统自然语言的计算机语言。这种计算机语言已从简单的应用发展到了多种复杂的对话，并逐步发展到能像传统自然语言一样地表达和传递信息。可以说，计算机技术引起了语言的重构。同时，一个社会的文化模式是以它的记忆为基础的。数据库和网络技术的诞生使知识和信息的存储，在数量上与性质上都发生了质的变化，人们获得知识的方式也因此而发生了变化。文字的出现曾改变了人类历史的进程和文明的面貌，而数据库和网络技术的出现，则从根本上改变了静态的信息存储方式和局部的信息交换方式，人类开始进入了信息社会。

计算机技术使语言、知识及它们间的相互交流发生了根本性的变化，因此引起了思维概念和推理的改变。在 1981 年召开的第三次世界计算机教育会议上，第一次提出了计算机文化（Computer Literacy）的术语。即为了区别传统的人类文化，把人类具备的对自然语言的阅读和写作能力称为“第一文化”，把人类具备的对计算语言的阅读和编程能力称为“第二

文化”，也称为“计算机文化”。可见，在当今社会，掌握计算机知识，提高应用计算机能力应当成为对人才素质最基本的要求。

计算机在信息社会中具有如此重要的地位，那么，什么是计算机？简单说，计算机是一种在事先存入的程序控制下，能够接收数据、存储数据、处理数据和提供处理结果的电子设备。如图 1.1.1 所示给出了计算机工作流程的简图，其中包括：输入、处理、输出和存储四个步骤。

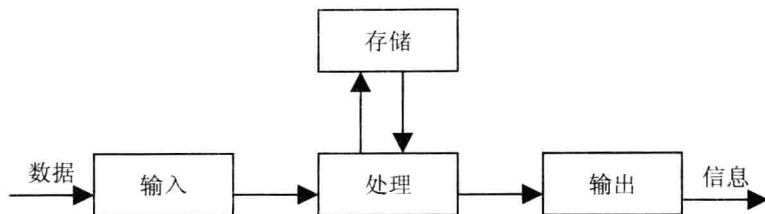


图 1.1.1 计算机工作流程

图 1.1.1 中，输入计算机的数据通常是指未经处理的原始数字、字符、图像或声音。经输入设备转换为二进制形式的数据在计算机中事先存入的程序控制下，按预定方式被加工成有意义或有用的形式。通常把经计算机处理形成的有用数据称为信息。信息可以按用户要求的方式存储或在输出设备中转换为用户可识别的形式，如打印成报表或图表，或在显示屏上显示等。从广义上说，在信息社会中，计算机是一种能以高速、精确和可靠的方式进行信息处理的工具。计算机技术对信息的产生、收集、处理、存储和传播将发挥越来越重要的作用，计算机作为一种崭新的生产力将推动信息社会更快地向前发展。

1.1.2 计算机中信息的表示

计算机是对信息（通常包括数字、字符、图像和声音（本书只讨论数字和字符信息）进行处理的机器。由于在计算机内部采用二进制数系统，所以无论何种类型的信息都必须以二进制数的形式在机器中进行处理。要了解计算机如何进行工作就必须了解二进制，及与其他数制之间的关系。

1. 进位计数制

在日常生活和工作中，人们在计数时使用不同的记写和命名数字的方法从而构成各种计数制。每一种计数制都使用一组特定的数字符号，通常把这些符号按序排列，由低位到高位进位，以表示一个数，这种计数方法称为进位计数制。人们最习惯和常用的是十进制。在计算机科学中除十进制外，常用的还有二进制、八进制和十六进制。

在采用进位计数的数字系统中，如果用 r 个基本符号（例如 0, 1, 2, ..., $r-1$ ）表示数值，则称其为 r 进制数， r 称为该数制的基。例如：我们日常生活中常用的十进制，其基 r 为 10，即基本符号为 0, 1, 2, ..., 9。若取基 $r=2$ ，则基本符号为 0 和 1，称为二进制数。不同的计数制具有的共同特点是：

（1）每一种计数制都有固定的符号集 如十进制数制，其符号有十个：0, 1, 2, ..., 9。二进制数制，其符号有两个：0 和 1。

（2）都使用位置表示法 即处于不同位置的数符所代表的值不同，其值与所在位置的

权值有关。

例如：十进制数 123.45 可表示为

$$123.45=1 \times 10^2+2 \times 10^1+3 \times 10^0+4 \times 10^{-1}+5 \times 10^{-2}$$

由此可以看出，各种进位计数制中的权值恰好是基数的 i 次幂，其中 $i \in (-\infty, +\infty)$ ， i 的大小与该位在数中的位置有关。因此，对任何一种进位计数制 r 表示的数都可以写成按其位权展开的多项式，任意一个 r 进制数 N 可表示为

$$N=\sum D_m r^i$$

式中， D_m 为该数制采用的基本数符， r^i 是权值， r 是基数，基数不同所表示的进位数制不同。计算机中常用的几种进位数制如表 1.1.1 所示。

表 1.1.1 计算机中常用的几种进位数制的表示

进位制	二进制	八进制	十进制	十六进制
规则	逢二进一	逢八进一	逢十进一	逢十六进一
基数 r	2	8	10	16
数符	0, 1	0, 1, ..., 7	0, 1, ..., 9	0, 1, ..., 9, A, ..., F
权	2^i	8^i	10^i	16^i
表示形式	B	O	D	H

前面我们已经介绍了常用的各种计数制，为什么计算机内采用二进制，而不采用我们熟悉的十进制呢？其主要原因是：

(1) 二进制只使用数字符号“0”和“1”，可用自然界存在的两种对立的物理状态表示。例如，晶体管导通为“1”，截止为“0”；高电压为“1”，低电压为“0”；灯亮为“1”，不亮为“0”；磁性器件磁化在一个方向为“1”，另一个方向为“0”；等等。计算机采用具有两种不同稳定状态的电子或磁性器件表示“0”和“1”。由于二进制状态简单，比十进制容易实现，数据传送不易出错，因此工作可靠。

(2) 二进制的运算比十进制数简单。二进制两个整数的“和”的运算规则只有三条：

加法 $0+0=0$
 $0+1=1$
 $1+1=10$

这种运算规则大大简化了计算机中实现运算的线路。实际上在计算机中减法、乘法及除法运算都可转化为加法这种最基本的运算来完成。

(3) 采用二进制可以进行逻辑运算，使逻辑代数和逻辑电路成为计算机电路设计的数学基础。

2. 不同进制之间的转换

(1) r 进制转换为十进制：采用以下权展开公式实现：

$$N=\sum D_m r^i$$

将 r 进制数转换为十进制数的方法是：将 r 进制数的基数与相应位置的权值相乘，然后相加即可。比如，把二进制数转换为相应的十进制数，只要将二进制数中出现 1 的位权相加即可。

例 1 把二进制数 100110.101 转换成相应的十进制数。

$$(100110.101)_B = 1 \times 2^5 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^{-1} + 1 \times 2^{-3} \\ = (38.625)_D$$

例 2 把八进制数 103.2 转换成相应十进制数。

$$(103.2)_O = 1 \times 8^2 + 3 \times 8^0 + 2 \times 8^{-1} = (67.25)_D$$

(2) 十进制数转换为 r 进制

① 十进制数转换为 r 进制整数。将一个十进制整数反复除以基数 r ，直到商为 0 为止，并记下每次所得余数（注：余数始终是介于 0 到 $r-1$ 之间的数，包括 0 和 $r-1$ ），将最后一个余数到第一余数按从左到右的次序连起来，它们所组成的数字串即为相应的 r 进制整数。这种方法称为除 r 取余法。例如：把十进制整数转换成相应的二进制整数，只需将十进制整数反复除以 2，直到商为 0 为止。

例如，把十进制数 14 转换成二进制数，如下所示。

2		14		余数
2		7		0 第一个
2		3		1
2		1		1
		0		1 最后一个

所以 $(14)_D = (1110)_B$ 。

② 十进制小数转换为 r 进制小数。将十进制小数转换成 r 进制小数时，首先将十进制小数反复乘以基数 r ，并取其整数部分（注：整数部分始终是介于 0 到 $r-1$ 之间的数，包括 0 和 $r-1$ ）。将从第一个整数到最后一个整数按从左到右的顺序连起来，它们所组成的数字串即为相应的 r 进制小数，这种方法称为乘 r 取整法。

例如，将十进制数 0.375 转换成相应的二进制数。

$$\begin{array}{ll} 0.375 \times 2 = 0.75 & \underline{0} \quad \text{第一个整数} \\ 0.75 \times 2 = 1.5 & \underline{1} \\ 0.5 \times 2 = 1.0 & \underline{1} \quad \text{最后一个整数} \end{array}$$

所以 $(0.375)_D = (0.011)_B$ 。

如果十进制数包含整数和小数两部分，则必须将十进制数整数部分和小数部分分别按除 r 取余数和乘 r 取整数进行转换，然后，再把 r 进制整数和小数部分组合在一起。

例如，将十进制数 14.375 转换成二进制数，只要将上例整数部分和小数部分组合在一起即可，即 1110.011。

(3) 非十进制数间的转换

两个非十进制数之间的转换方法一般结合上述两种方法进行转换，即先把被转换数据转

换为相应的十进制数，然后再将十进制数转换为其他进制数。由于二进制、八进制和十六进制之间存在特殊关系，即 $8=2^3$ ， $16=2^4$ ，因此转换方法就比较容易，如表 1.1.2 所示。

表 1.1.2 二进制、八进制、十六进制之间的关系

二进制	八进制	二进制	十六进制	二进制	十六进制
000	0	0000	0	1000	8
001	1	0001	1	1001	9
010	2	0010	2	1010	A
011	3	0011	3	1011	B
100	4	0100	4	1100	C
101	5	0101	5	1101	D
110	6	0110	6	1110	E
111	7	0111	7	1111	F

根据这种对应关系，二进制转换到八进制十分简单，只需将二进制数以小数点为界，整数从右向左每 3 位一组，小数部分从左向右每 3 位一组，最后不足 3 位补零，然后根据表 1.1.2，即可完成转换。

例如，将二进制数 $(10100101.010111101)_B$ 转换成八进制数。

所以 $(\underline{010} \ \underline{100} \ \underline{101} . \underline{010} \ \underline{111} \ \underline{010})_B = (245.272)_O$ 。

将八进制转换成二进制的过程正好相反。

二进制同十六进制之间的转换就如同八进制同二进制之间一样，只是 4 位一组。

例如，将二进制 $(1111111000111.100101011)_B$ 转换成十六进制数。

所以 $(\underline{0001} \ \underline{1111} \ \underline{1100} \ \underline{0111} . \underline{1001} \ \underline{0101} \ \underline{1000})_B = (1FC7.958)_H$ 。

3. 信息的单位及机器数

如前所述，在计算机中采用具有两种状态的电子器件表示“1”和“0”，每个电子器件代表二进制数中的一位。因此，位 (bit) 是计算机中的最小信息单位。通常将八位二进制位称为一个字节 (byte)，字节是信息的基本单位。一个字节可以表示 256 种状态，它可以存放 0~255 范围内的一个整数，或一个字符的编码。计算机中常以字节为单位表示文件或数据的长度以及存储容量的大小。例如，内存容量 128MB。其中， $1MB=2^{10}KB$ ，其他单位之间的关系如下：

1B=8bit

1KB=1024B

1MB=1024KB

1GB=1024MB

1TB=1024GB

二进制数在计算机中的表示形式称为机器数。由于计算机存放一个参与运算的机器数所使用的电子器件的基本位数是固定的，通常把具有固定位的这种二进制串称为字，而把字所包含的二进制数位数称为字长。通常所说多少位的计算机，就是指一个字长有多少位。例如，16 位机的字长为 16 位，能表示 2^{16} 个不同的信息；32 位机的字长为 32 位，能表示 2^{32} 个不

同的信息。一般来说,计算机的字长越长,性能也越高。大型机的字长高于 128 位。

机器数具有的重要特点如下:

(1) 机器数的位数固定,能表示的数值范围受到位数限制。例如,字长为 8 位的计算机,能表示的无符号整数的范围为 $0 \sim 255 (2^8 - 1)$; 字长为 16 位的计算机能表示无符号整数的范围为 $0 \sim 65535 (2^{16} - 1)$ 。由于字长的限制,如果计算机运算的结果超过了机器数能表示的范围,就会产生“溢出”,计算机便停止运行,进行溢出处理。

(2) 机器数的正、负用“0”和“1”表示。上述的二进制数没有考虑符号问题,所以是无符号的数。在实际应用中,数总是有正负的,在计算机中通常是把最高位作为符号位,其余作为数值位,并规定 0 表示正数,1 表示负数。因此,机器数是连同数据符号一起数字化了的数据。

(3) 机器数有定点和浮点两种表示法。

4. 字符的表示

为了符合人们的习惯,使用计算机时能用十进制数及常用的字母、字符完成信息的输入和输出,在机器内又能以二进制数进行处理,因此信息必须用二进制编码。所谓编码是用一串二进制数码代表一位十进制数字或一个字符。编码工作由计算机在输入、输出时自动进行。现在国际上广泛采用美国标准信息交换代码 (American Standard Code for Information Interchange) 表示字符,简称为 ASCII 码。ASCII 码基本字符集包括了 128 个字符,其中包括数字 (0~9), 英文大小写字母,一些在算式中常用的符号,以及控制字符,每个字符用一个字节表示。由于基本 ASCII 码的最高位为 0,因此 128 个字符的编码范围为 00000000~01111111,即十进制的 0~127。字符的二进制编码表如附录 C 所示。从表中可看出,从 A 到 Z 的 26 个大写字母,是由 01000001~01011010 (十进制的 65 到 90) 的 26 个连续代码来表示的,而 0 到 9 的数字,则由 00110000~00111001 (十进制的 48 到 57) 的 10 个连续代码来表示。从 NUL 到 US 的控制字符,用 00000000~00011111 (十进制的 0 到 31) 的 32 个连续代码来表示,控制字符(包括最后一个删除字符 DEL)是非显示和非打印字符,其他为可显示、可打印字符。

这些字符通常是用计算机的键盘输入的。键盘上的每个字符都由其 ASCII 码代表,通过这些字符的不同组合,就可以实现对各种信息的表示、传递和处理。由此可见,编码的作用就是把要计算机处理的数据(数值或字符)转换成二进制数字串,以便在机器中存储和处理,输出时再通过机器转换成对应符号。

例如,用键盘上按键输入“CHINA”的字符串,传送进计算机的,则是 01000011、01001000、01001001、01001110、01000001 这五个二进制数字串;反之,存储器内存储的二进制数字串 01010111、01010000、01010011 在显示器或打印机输出时,可转换成“WPS”字符串。

必须指出的是,由 7 位编码构成的 ASCII 码基本字符集能表示的字符只有 128 个,不能满足信息处理的需要,近年来,对 ASCII 码字符集进行了扩充。采用 8 个二进制位表示一个字符,编码范围:00000000~11111111,一共可表示 256 种字符和图形符号,这称为扩充的 ASCII 码字符集,但通常使用的仍是基本 ASCII 码字符集。