



C 语言初级教程

孟庆昌 刘振英 编著

宇航出版社

C语言初级教程

孟庆昌 刘振英 编著

学 苑 出 版 社

(京)新登字181号

内 容 简 介

本书用通俗易懂的语言和众多有趣的实例由浅入深、阶梯上升地介绍了C语言的基本知识和初级应用。全书分为四章和四个附录，主要内容包括程序设计基础知识；C语言的标识符、变量、常量和基本数据类型，运算符和各种语句等。对C语言的语法要点（如数组、函数、指针、结构、存储类、预处理和输入/输出库函数等）分别在单独章节中进行讲解。每章最后都有相应的习题，并在附录中给出参考答案，以便满足广大中学生学习和计算机初学者自学的需要。本书既考虑到青少年的知识水平，又照顾到C语言的完整性。

本书可作为初中二、三年级和高中学生（包括中专、中技生）计算机选修课的教材，也可供计算机知识初学者和非计算机专业的计算机应用人员使用。

C语言初级教程

编著：孟庆昌 刘振英

责任编辑：廖寿琪

*

宇航出版社出版

北京和平门内大街1号 邮政编码100013

新华书店北京发行所发行

各地新华书店经销

北京密云华都印刷厂印刷

*

开本：787×1092 1/32 印张：11 字数：256千字

1992年5月第1版第1次印刷 印数：1—15000册

ISBN 7-80034-434-7/TP·027 定价：6.60元

前 言

当今已进入信息时代，电子计算机的应用遍及社会生活的各个角落，从航天飞机、巡航导弹，到儿童玩具、智力游戏，处处都可见到微电子技术的作用。中国在十年规划和“八五”计划中把发展电子工业放在突出地位，这就对广大教育工作者提出一个如何培养合格人才的大问题。具体到一小点上，就是在有条件的中学开设电子计算机选修课，使学生尽早了解计算机、使用计算机。

应用计算机就少不了程序设计，进行程序设计就离不开程序设计语言。目前程序设计语言已有一百多种，较常用的就有二十多种，它们各有特色。对计算机初学者说来，学什么较适宜呢？我们认为，这种语言应该简洁易学、有实用价值、适应发展潮流，并且利于培养初学者，具有良好的程序设计风格。按此评选，C语言可名列前茅。

C语言是70年代初由美国电话电报公司(AT & T)所属的贝尔实验室的D·里奇(Dennis Ritchie)为开发UNIX操作系统而独立设计并实现的。随着UNIX操作系统的广泛流行及微型计算机的普及，C语言也以其众多优点——效率高、简洁、可移植性好、功能强等博得计算机界的交口赞誉，倍受软件工程技术人员的青睐。目前C语言已在不同的操作系统环境中以及几乎所有的计算机上运行了，它已成为国际上最受欢迎、最有实用价值、应用面最广的程序设计语

言。

国际上软件行业的科研、教学、开发与交流等许多任务，几乎都涉及到C语言。可以看到：在90年代计算机开发应用的各行业中，将是C语言盛行的时代，也是中国软件产业打入国际先进水平的时代。

我们在UNIX系统和C语言的科研、教学和开发工作中已有十五六年时间，深深感到C语言的重要性。1983年为了满足教学需要，曾编写了《C语言程序设计教程》，受到国内广大读者的欢迎。1988年宇航出版社出版了《C语言及其应用》，并在全国软件行业水平考试中初露头角，也受到国内广大读者的肯定和欢迎。四年来先后印了4次，可见读者迫切需要掌握C语言之心多么真诚啊！最近，有不少初学者希望有一本介绍C语言基本知识的通俗读物，从宇航出版社反馈来的信息也是要组一份初级教稿方面的稿件，本书试图满足这种要求。

本书针对初学者的特点，考虑到一般工程技术人员、管理人员及中学生的数学程度，用通俗易懂的语言介绍了程序设计基础知识、C语言的基本语法和一般应用及语法要点。采用分层次的知识结构、由浅入深地介绍，通过众多的有趣味的示例讲解程序设计的方法和技巧，力图使读者在较短的时间内能使用C语言解决一般问题，达到“入门”目的。在书后的附录中给出C语言语法汇总，这为读者进一步提高编程能力奠定良好基础。

全书分为四章和四个附录。第一章介绍程序设计基础知识，包括数的表示和流程图等；第二章是C语言概述，展示

C程序的一般结构；第三章为C语言（一），这是C语言基础知识介绍，包括简单数据类型、初级运算符、循环语句和初步的数组内容；第四章是C语言（二），是上一章内容的扩展和提高，比较完整、系统地介绍了C语言的基本知识和应用，包括各种运算符、语句和函数、指针、结构等。每一章后面都有习题，书后附有参考答案，以便自学检验。附录A给出C语法汇总，是对C语言的归纳总结。为便于初学者上机实习，附录D中给出上机操作要点。

1990年ISO（国际标准化组织）发布了《语言的国际标准》。本书作为一本初级教程只能介绍C语言的基本知识。读者在学完本书后，如果需要进一步全面系统地了解C语言，或者是需要掌握标准C语言的读者，或者是期望日后尽快通过程序员级和高级程序员级水平考试的读者，请参阅宇航出版社将要出版的新书——《标准C语言程序设计》。这些书奉献给读者，将会起到抛砖引玉作用。

在本书编写过程中始终得到宇航出版社廖寿琪等许多同志的大力支持。在此对给予帮助的各位同志表示诚挚谢意。

由于编者的经验和水平有限，书中肯定会存在不足之处，敬请广大读者多提宝贵意见和建议。

编 者

1990年12月

于北京信息工程学院

目 录

第一章 程序设计基础知识.....	(1)
第一节 数的表示.....	(1)
一、二进制.....	(2)
二、八进制.....	(4)
三、十六进制.....	(6)
四、二-八与二-十六进制数的转换.....	(6)
第二节 高级语言与程序设计.....	(8)
一、什么是高级程序设计语言.....	(8)
二、怎样进行程序设计.....	(9)
三、用户程序的主要处理阶段.....	(11)
习题.....	(13)
第二章 C语言概述.....	(15)
第一节 C语言的基本特点.....	(15)
第二节 C程序示例.....	(17)
习题.....	(22)
第三章 C语言(一).....	(23)
第一节 词法及简单数据类型.....	(23)
一、基本词法.....	(23)
二、标识符和关键字.....	(24)
三、基本数据类型.....	(27)
四、变量.....	(30)
五、常量.....	(33)
第二节 运算符(一)和语句(一).....	(40)

一、算术运算符和赋值语句.....	(40)
二、关系运算符.....	(45)
三、if语句 (一)	(47)
四、条件运算符.....	(53)
五、增量、减量运算符和表达式语句.....	(56)
六、复合语句.....	(60)
七、while循环语句.....	(61)
八、break语句.....	(69)
九、for 循环语句 (一)	(71)
第三节 数组 (一)	(80)
一、数组的说明和表示.....	(81)
二、数组的下标和数组的应用.....	(82)
习题.....	(88)
第四章 C语言 (二)	(92)
第一节 运算符 (二) 和语句 (二)	(92)
一、运算符 (二)	(92)
二、运算符的优先级和结合性.....	(106)
三、语句 (二)	(109)
第二节 函数.....	(143)
一、函数定义.....	(144)
二、函数调用.....	(151)
三、函数类型说明.....	(160)
四、递归函数.....	(163)
第三节 数组 (二)	(171)
一、字符数组.....	(171)
二、多维数组.....	(177)
三、数组初始化.....	(181)
第四节 指针.....	(185)

一、指针说明.....	(186)
二、指针运算.....	(188)
三、指针和数组.....	(199)
第五节 结构.....	(204)
一、结构说明.....	(204)
二、结构成员的存取.....	(208)
三、结构的初始化.....	(214)
四、结构数组.....	(216)
第六节 变量存储类.....	(220)
一、自动变量.....	(220)
二、寄存器变量.....	(222)
三、外部变量.....	(223)
四、静态变量.....	(227)
第七节 预处理功能.....	(230)
一、简单宏定义和宏替换.....	(230)
二、带参数的宏.....	(235)
三、文件包含.....	(237)
第八节 输入/输出和库函数使用.....	(243)
一、字符输入/输出函数.....	(243)
二、格式输入/输出函数.....	(244)
三、基本文件操作.....	(251)
四、标准库函数.....	(255)
习题.....	(257)
附录A C语法汇总.....	(260)
附录B ASCII码字符集.....	(301)
附录C 习题参考答案.....	(305)
附录D 上机操作指南.....	(333)

第一章 程序设计基础知识

第一节 数的表示

人们在日常生活、社会活动中离不开信息的交流，例如彼此间的问候、对一个问题的讨论、对一件事情的介绍等等。人们交换信息的最主要的工具是自然语言。如，“你到哪儿去？”“我去上课。”“上什么课？”“上C语言。”这样一问一答，彼此知道对方的意思。但是当遇到一位外国人，他不会中国话，那么就无法用中国话进行交谈。倘若又不懂对方说的话，又要交流思想，那怎么办呢？大家都会想到：找一位懂双方语言的人当翻译。通过翻译，把对方的话变成自己能理解的语言。

人和计算机之间是怎样交换信息呢？也就是说，如何让计算机按人的意图去完成计算工作呢？通常的方法是由用户（这里指使用计算机的人）编写程序，再把它“翻译”成计算机的内部语言——机器语言，然后按照这一系列的机器语言，依次执行。计算机并不懂人类的自然语言，它只识别自己的机器语言。机器语言只用两个数字来表示，即“0”和“1”。这就是“二进制数”。二进制数有什么性质？它与十进制数之间是怎样转化的？下面就介绍这方面的问题。

一、二进制

平常使用的数，一般都是十进制的，用0, 1, 2, ..., 9共十个数字来表示，如9715。从小到大各相邻位之间是逢十进一。例如， $7+6=13$ 。

二进制数是计算机内部使用的，只用0和1两个数字来表示所有的数。二进制的基本规则是：逢二进一，即从小到大相邻数位之间的关系是2的倍数。例如，二进制数11（读作么么）等于十进制数3，即： $2+1=3$ 。二进制数1101（读作么么零么）等于十进制数13，即： $1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 8 + 4 + 0 + 1 = 13$ 。二进制数10110（读作么零么么零）等于

$$\begin{aligned} & 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\ & = 16 + 0 + 4 + 2 + 0 = 22。 \end{aligned}$$

为了区分是哪种进位制的数，采用一种记号：把数用圆括号括起来，并在右括号的下脚处标明是几进制。例如，

(1)₁₀表示十进制数1，(698)₁₀表示十进制数698，(101)₂表示二进制数，它等于十进制数5，可见与(101)₁₀是截然不同的。下面还会见到(37)₈（表示八进制数，等于十进制数31），(20)₁₆（表示十六进制数，等于十进制数32）这种形式的数。

1. 二-十进制转换

在上面已讲过二进制数转换成十进制数的例子，这里介绍一般的转换规则。

要把一个二进制数转化为十进制数，要按下列步骤去做：将它的倒数第一位（即最右边）数字乘以 2^0 ，倒数第二位数字乘以 2^1 ，倒数第三位数字乘以 2^2 ，……，倒数第n位数字

乘以 2^{n-1} ，依此类推；然后将各项的积加起来，就得到相应的十进制数。

例1.1 将二进制数11001化为十进制数。

$$\begin{aligned}\text{解: } (11001)_2 &= 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ &= 16 + 8 + 0 + 0 + 1 \\ &= (25)_{10}\end{aligned}$$

所得结果为 $(11001)_2 \longrightarrow (25)_{10}$ 。

例1.2 将二进制数1010转化为十进制数。

$$\begin{aligned}\text{解: } (1010)_2 &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\ &= 8 + 0 + 2 + 0 \\ &= (10)_{10}\end{aligned}$$

所得结果为 $(1010)_2 \longrightarrow (10)_{10}$ 。

2. 十-二进制转换

十进制数是怎样转化为二进制数呢？这是上面过程的逆向过程。一般采取的办法是：将十进制数连续地被2除；每次相除都得到一个余数（不是1就是0）和一个商；再用2去除这个商，又得到一个余数和新商，这样一次次做下去，终于得到一串由1和0组成的余数（必须从最后一次除法所得的余数读回去），它就是相应的二进制数。

例1.3 将十进制数6转化为二进制数。

解：用下面的竖式形式进行除2运算：

$$\begin{array}{rcl} 2 \overline{) 6} & \cdots \cdots & \text{余} 0 \\ 2 \overline{) 3} & \cdots \cdots & \text{余} 1 \\ 2 \overline{) 1} & \cdots \cdots & \text{余} 1 \\ & & 0 \end{array} \quad \begin{array}{c} \uparrow \\ \text{读数方向} \end{array}$$

结果是 $(6)_{10} \longrightarrow (110)_2$ 。

例1.4 将十进制数10转化为二进制数。

解:

2		10		余0
2		5		余1
2		2		余0
2		1		余1
		0		

↑
读数方向

所得结果为 $(10)_{10} \rightarrow (1010)_2$ 。

例1.5 将十进制数8转化为二进制数。

解:

2		8		余0
2		4		余0
2		2		余0
2		1		余1
		0		

↑

结果是 $(8)_{10} \rightarrow (1000)_2$ 。

二、八进制

大家看到，计算机里使用的数字只有0和1。这样，要表示一个较大的数就得用一长串0、1码，书写起来不方便，又不易记忆。为此，在计算问题或者进行程序设计时，往往采用八进制或十六进制。这两种进制既克服了二进制的不足，又具有容易实现与二进制的转换的优点。

八进制数是以8为基底的数，用到的数字是0, 1, 2, ..., 7, 一共八个。八进制数的基本进位规则是：从小到大相邻数位是逢八进一。例如，八进制数0, 1, 2, ..., 7与十进制数0, 1, 2, ..., 7的表示形式相同，而十进制数8在八进制中表示为10，即 $(8)_{10} \rightarrow (10)_8$ ；十进制数9在八进制中表示为11，即 $(9)_{10} \rightarrow (11)_8$ ，依次类推，可得到 $(10)_{10} \rightarrow (12)_8$, $(15)_{10}$

$\rightarrow (17)_8, (16)_{10} \rightarrow (20)_8$ 等等。

1. 十-八进制转换

把十进制数转换成八进制数的方法和上面讲的十进制数转换成二进制数的方法一样，只不过这里是被8除、取余数。

例1.6 把十进制数10转换成八进制数。

$$\begin{array}{rcl} \text{解:} & 8 \overline{) 10} & \cdots \cdots \text{余} 2 \quad \uparrow \\ & 8 \overline{) 1} & \cdots \cdots \text{余} 1 \quad \uparrow \\ & 0 & \end{array}$$

所得结果是 $(10)_{10} \rightarrow (12)_8$ 。

例1.7 把十进制数54转换成八进制数。

$$\begin{array}{rcl} \text{解:} & 8 \overline{) 54} & \cdots \cdots \text{余} 6 \quad \uparrow \\ & 8 \overline{) 6} & \cdots \cdots \text{余} 6 \quad \uparrow \\ & 0 & \end{array}$$

转换结果是 $(54)_{10} \rightarrow (66)_8$ 。

2. 八-十进制转换

把八进制数转换成十进制数和把二进制数转换成十进制数的方法一样，只不过这里是乘以8的次幂。

例1.8 把八进制数14转换成十进制数。

$$\begin{aligned} \text{解: } (14)_8 &= 1 \times 8^1 + 4 \times 8^0 \\ &= 8 + 4 \\ &= (12)_{10} \end{aligned}$$

转换结果为 $(14)_8 \rightarrow (12)_{10}$

例1.9 把八进制数106转换成十进制数。

$$\begin{aligned} \text{解: } (106)_8 &= 1 \times 8^2 + 0 \times 8^1 + 6 \times 8^0 \\ &= 64 + 0 + 6 \\ &= (70)_{10} \end{aligned}$$

转换结果为 $(106)_8 \rightarrow (70)_{10}$

三、十六进制

十六进制数是以16为基底，它是由数字0, 1, 2, ..., 9和字母A, B, C, D, E, F (或a, b, c, d, e, f) 构成，逢十六进1。其中字母A, B, C, D, E, F (或a, b, c, d, e, f) 对应十进制数10, 11, 12, 13, 14, 15。

十六进制数与十进制数之间的转换方法类似于八进制数与十进制数之间的转换方法。

例1.10 把十进制数186转换成十六进制数。

$$\begin{array}{r} \text{解: } 16 \overline{) 186} \cdots \cdots \text{余} 10 = A \uparrow \\ \underline{16 \overline{) 11}} \cdots \cdots \text{余} 11 = B \uparrow \\ 0 \end{array}$$

转换结果是: $(186)_{10} \rightarrow (BA)_{16}$ 或 $(ba)_{16}$ 。

例1.11 把十六进制数3AE转换成十进制数。

$$\begin{aligned} \text{解: } (3AE)_{16} &= 3 \times 16^2 + 10 \times 16^1 + 14 \times 16^0 \\ &= 768 + 160 + 14 \\ &= (942)_{10} \end{aligned}$$

转换结果为 $(3AE)_{16} \rightarrow (942)_{10}$

表1-1列出了十进制、二进制、八进制、十六进制的对应关系。

四、二-八与二-十六进制数的转换

1. 将二进制数化成八进制数是很容易的，因为八进制的基底8正好是2的三次幂，故对于一个二进制的整数来说，从右向左每三个数字为一组，这样很快就将一个二进制整数化为八进制数。

例1.12 把一个二进制数 101100010111011 转换成八进

表1-1 数的各种进制的对应关系

十进制 (基底=10)	二进制 (基底=2)	八进制 (基底=8)	十六进制 (基底=16)
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

制数。

解: $(101,100,010,111,011)_2$

5 4 2 7 3

转换结果为 $(101100010111011)_2 \rightarrow (54273)_8$

反过来, 将八进制数转换成二进制数, 只要将八进制数的每一位转换成三位的二进制数, 合在一起就是二进制数了。

例1.13 将八进制数43转换成二进制数。

解: $(43)_8 \rightarrow (100011)_2$

2. 二进制与十六进制数的转换

因为十六进制数的基底是16，正好是2的四次幂，所以对于一个二进制整数来说，从右向左每四个数字为一组，就立即把一个二进制数转换成十六进制数，反之亦然。

例1.14 把二进制数10011110转换成十六进制数。

解：(1001,1110)₂

9 E

转换结果为(10011110)₂ → (9E)₁₆

第二节 高级语言与程序设计

一、什么是高级程序设计语言

计算机的动作，如加、减、乘等是靠一系列指令指挥的。一条指令通常包括两部分：一是指出机器执行什么操作；二是指明操作数在什么地方。在机器中指令是用二进制数表示的。例如在某个机器中，010000101010是一条取数指令，前面4位“0100”表示取数操作，后面8位“00101010”表示存放所取数的地址。不同的机器有不同的指令格式，而且指令条数也不一样。

用计算机解决某个问题需要若干条指令，这样的一组指令就称为一个计算机程序。程序是由设计人员编写的。很明显，用上面所说的一大串二进制数来编写程序是非常烦人的；难学、难记、难写。后来，经过计算机专家的研究，产生了高级程序设计语言（简称高级语言），如FORTRAN、COBOL、C语言等。高级语言是与计算机具体特性无关的程序设计语言，它使编程人员能以更接近于人类的自然语言