



教育部高职高专规划教材

C语言 程序设计

邵士媛 主编
徐维祥 主审

12C



化学工业出版社
教材出版中心

出版说明

高职高专教材建设工作是整个高职高专教学工作中的重要组成部分。改革开放以来,在各级教育行政部门、有关学校和出版社的共同努力下,各地先后出版了一些高职高专教育教材。但从整体上看,具有高职高专教育特色的教材极其匮乏,不少院校尚在借用本科或中专教材,教材建设落后于高职高专教育的发展需要。为此,1999年教育部组织制定了《高职高专教育专门课课程基本要求》(以下简称《基本要求》)和《高职高专教育专业人才培养目标及规格》(以下简称《培养规格》),通过推荐、招标及遴选,组织了一批学术水平高、教学经验丰富、实践能力强的教师,成立了“教育部高职高专规划教材”编写队伍,并在有关出版社的积极配合下,推出一批“教育部高职高专规划教材”。

“教育部高职高专规划教材”计划出版500种,用5年左右时间完成。这500种教材中,专门课(专业基础课、专业理论与专业能力课)教材将占很高的比例。专门课教材建设在很大程度上影响着高职高专教学质量。专门课教材是按照《培养规格》的要求,在对有关专业的人才培养模式和教学内容体系改革进行充分调查研究和论证的基础上,充分吸取高职、高专和成人高等学校在探索培养技术应用性专门人才方面取得的成功经验和教学成果编写而成的。这套教材充分体现了高等职业教育的应用特色和能力本位,调整了新世纪人才必须具备的文化基础和技术基础,突出了人才的创新素质和创新能力的培养。在有关课程开发委员会组织下,专门课教材建设得到了举办高职高专教育的广大院校的积极支持。我们计划先用2~3年的时间,在继承原有高职高专和成人高等学校教材建设成果的基础上,充分汲取近几年来各类学校在探索培养技术应用性专门人才方面取得的成功经验,解决新形势下高职高专教育教材的有无问题;然后再用2~3年的时间,在《新世纪高职高专教育人才培养模式和教学内容体系改革与建设项目计划》立项研究的基础上,通过研究、改革和建设,推出一大批教育部高职高专规划教材,从而形成优化配套的高职高专教育教材体系。

本套教材适用于各级各类举办高职高专教育的院校使用。希望各用书学校积极选用这批经过系统论证、严格审查、正式出版的规划教材,并组织本校教师以对事业的责任感对教材教学开展研究工作,不断推动规划教材建设工作的发展与提高。

教育部高等教育司

2001年4月3日

前 言

为了贯彻《教育部关于加强高职高专教育人才培养工作的意见》(教高[2000]2号)有关精神,积极支持教育部面向21世纪高职高专教材建设,在教育部领导直接关怀下,全国高等职业教育院校协作会专门课开发指导委员会确定了编写电子类专业的10门主干课程(《电路分析》、《模拟电子技术》、《数字电子技术》、《电子技术实训》、《高频电子线路》、《电子测量与仪器》、《电视接收技术》、《电子设计自动化(EDA技术)》、《单片机应用技术》和《C语言》)供电子技术应用、应用电子技术、电子工程、通信、电子设备制造与维修等相关专业使用。

本套教材紧密结合高职高专教育特点,主动适应社会实际需要,突出应用性、针对性,加强实践能力的培养。内容叙述力求深入浅出,将知识点与能力点有机结合,注重培养学生的工程应用能力和解决现场实际问题的能力;内容编排力求简洁明快、形式新颖、目标明确,利于促进学生的求知欲和学习主动性。

随着计算机的不断普及,用于程序设计的语言也得到迅速发展。C语言是一种结构化、可编译的通用程序设计语言,具有功能丰富、表达能力强、使用灵活方便、应用面广、可移植性好、编程效率高等特点,既具有高级语言的形式,又具有低级语言的功能。因此,C语言是当前国内外最流行的一门语言,已被广大计算机专业或非计算机专业人员所喜爱。目前很多学校已把C语言作为学习程序设计的入门语言,有关C语言的教材很多,但多数教材都偏重于较强的理论学习,并在程序算法中过多地使用高等数学知识,起点高、难度大,这使初学者和作为入门语言学习的人望而却步。

本教材从实际出发,针对高职高专计算机及应用专业而编写,同时兼顾其他专业、计算机培训班、参加计算机等级考试人员及自学者的使用。书中标有“*”号的章节,因涉及C语言和程序设计能力方面比较深的内容,供选学。第十四章图形编程基础是为了让读者了解计算机作图程序的编制方法而编写的,其内容虽开辟了一个新的应用领域,然而都是由一些基本函数实现的,并有规范的编程方法,读者只需模仿实例程序,通过自学掌握有关内容。

本教材根据计算机技术的最新发展,在取材的深度和广度方面作了精心的优化选择。具体特点如下。

(1)在详细介绍C语言的同时,特别注重于C语言的正确使用,使读者通过对本书的学习,不仅能正确了解C语言的理论知识结构,还能掌握初步的程序设计方法和技巧,在程序设计能力方面受到良好的训练。

(2)在组织内容方面,力求以基本概念和基本方法为主线,深入浅出、通俗易懂,并特别注重实例的选择和说明。为了加深对基本概念的理解和掌握,各章均安排了大量习题供学生练习,同时,附有实验题目和设计题目配合学生上机实习和综合设计使用,最后配有模拟试题,从而验证学生的学习掌握情况。

(3)为了适应社会的需求,满足学生“多证书”的愿望,许多学生希望在学了C语言后能通过全国计算机等级考试,因此,本教材在编写过程中参考了全国计算机等级考试二级考试大纲,并增加了相关内容的练习。

本教材的第一、二章、综合应用部分由陈享成编写；第三、十四章由郜亚丽编写；第四、五章由孙凌编写；第六、十一章由李丹编写；第七章、实验部分由侯丽敏编写；第八、十三章、上机指导部分由王桂芝编写；第十、十二章由曹红玲编写；第九章、附录、模拟试题部分由邵士媛编写，并负责全书的统稿工作，徐维祥博士审阅了全稿内容。习题部分均由各章编写人员提供。

由于编者水平有限，加之编写时间紧促，书中错漏之处恳请有关专家和广大读者批评指正。

编 者

2002 年 10 月

内 容 提 要

本书是以 Turbo C 环境为基础,详细介绍了适用于 PC 系列计算机的 C 语言程序设计的基本概念、方法和技巧。全教材分为上下两册,上册《C 语言程序设计》共十四章,主要介绍 C 语言的基本概念、数据及运算、程序结构、数组、函数、指针、结构体与共用体、编译预处理、位运算、文件操作、图形编程基础。书中给出了大量典型程序实例,每章后面附有小结。下册《C 语言习题与上机指导》共分为习题、上机指导、实验、综合应用、模拟试题五部分,便于学生练习与教学参考使用。

本书是针对高职高专的教学特点组织编写的,对内容做了精心安排,其特点是通俗易懂、深入浅出、循序渐进、实用性强,注重程序设计能力的培养。

本收可作为高职高专院校计算机及应用专业的教材和教学参考书,同时可供其他专业、计算机培训班、参加计算机等级考试人员及自学者使用。

目 录

第一章 C语言概述	1	第三节 字符数组	87
第一节 C语言简介	1	本章小结	95
第二节 C语言的程序结构及特点	2	第八章 函数	96
第三节 C程序的上机过程	4	第一节 函数的概念和函数的定义	96
本章小结	5	第二节 函数的调用	99
第二章 数据类型与基本输入输出	6	第三节 函数的嵌套调用和递归调用	105
第一节 C语言的数据类型	6	第四节 数组作函数的参数	110
第二节 常量	6	第五节 局部变量和全局变量	114
第三节 变量	11	第六节 变量的存储类型	118
第四节 数据的基本输入与输出	17	第七节 函数的存储类型和程序的分割 编译	122
本章小结	26	本章小结	125
第三章 数据运算与表达式	28	第九章 指针	127
第一节 算术运算	28	第一节 变量的指针	127
第二节 赋值运算	32	第二节 数组的指针	134
第三节 不同类型数据间的转换	34	第三节 字符串的指针	144
第四节 关系运算与逻辑运算	39	第四节 返回指针值的函数	148
第五节 条件运算与逗号运算	42	第五节 指针数组与多级指针	150
第六节 sizeof 运算与 & 运算	44	第六节 函数的指针	155
本章小结	45	本章小结	158
第四章 顺序结构程序设计	47	第十章 结构体、共用体和枚举类型	160
第一节 C语句概述	47	第一节 结构体	160
第二节 程序的基本算法与基本结构	48	第二节 链表	173
第三节 顺序结构程序设计	51	第三节 共用体	179
本章小结	52	第四节 枚举类型和自定义类型	182
第五章 选择结构程序设计	54	本章小结	185
第一节 if 语句	54	第十一章 编译预处理	186
第二节 switch 语句	59	第一节 宏定义	186
第三节 程序举例	60	第二节 文件包含	190
本章小结	64	第三节 条件编译	192
第六章 循环结构程序设计	65	本章小结	194
第一节 循环语句	65	第十二章 位运算	195
第二节 转移语句	69	第一节 位运算	195
第三节 循环语句的嵌套	71	第二节 位段	201
第四节 程序举例	72	本章小结	203
本章小结	75	第十三章 文件	204
第七章 数组	77	第一节 文件概述	204
第一节 一维数组	77	第二节 文件的打开和关闭	207
第二节 二维数组	82		

第三节 文件的读写	209	本章小结	234
第四节 文件的定位及随机读写	217	附录	235
第五节 文件的出错检测	219	附录 I 常用字符与 ASCII 代码对照表 ...	235
本章小结	219	附录 II 关键字	236
*第十四章 图形编程基础	221	附录 III 运算符	236
第一节 显示器和显示卡	221	附录 IV C 语言常用语法提要	237
第二节 字符屏幕操作	222	附录 V 常用 Turbo C 库函数	240
第三节 图形操作	226	主要参考文献	244

第一章

C 语言概述

C 语言是当前广泛流行的程序设计语言，它像其他高级语言一样，面向用户，面向解题的过程，编程者不必熟悉具体的计算机内部结构和指令；同时，它又可内嵌汇编指令，将汇编指令当作它的语句，所以 C 语言又像低级语言一样，可以对计算机硬件进行控制，如进行端口 I/O 操作、位操作、地址操作。因此，人们把它称为介于高级语言与低级语言之间的一种中级语言。由于 C 语言的这种特点，它不但用于编一般应用程序，而且许多大的操作系统、编译系统也是由 C 语言编写的。

第一节 C 语言简介

C 语言是由一种早期的编程语言 BCPL 发展演变而来的。1970 年美国贝尔实验室的 Ken Thompson 和 Dennis Ritchie 完成了 UNIX 的初版，与此同时，他们还改写了由 Martin Richards 开发的 BCPC 语言，形成了一种称为 B 的语言，此后 B 语言又进一步被进行了改进和完善，形成了称之为 C 的语言。

C 语言大约形成于 1972 年。1973 年 Dennis Ritchie 把 UNIX 系统中的 90% 用 C 语言进行了改写，并在 PDP-11 小型机上完成了用 C 语言及汇编语言编写的 UNIX 操作系统的调试，将其投入运行。随着 UNIX 的移植、推广，C 语言也得到移植和推广，DOS 支持下的，甚至 Windows 支持下的 C 语言都相继出现。如目前广泛使用的在 PC 微机上的 MS-DOS 支持下的 Turbo C 和 Microsoft C 就是典型的 C 语言版本。

以 1978 年发表的 UNIX 第 7 版中的 C 编译程序为基础，Brian W. Kernighan 和 Dennis M. Ritchie（合称 K&R）合著了影响深远的名著《The C Programming Language》，这本书中介绍的 C 语言成为后来广泛使用的 C 语言版本的基础，它被称为标准 C。1983 年，美国国家标准化协会（ANSI）根据 C 语言问世以来各种版本对 C 的发展和扩充，制定了新的标准，称为 ANSI C。ANSI C 比原来的标准 C 有了很大的发展。K&R 在 1988 年修改了他们的经典著作《The C Programming Language》，按照 ANSI C 标准重新写了该书。1987 年，ANSI 又公布了新标准——87ANSI C。1990 年，国际标准化组织 ISO（International Standard Organization）接受 87ANSI C 为 ISO C 的标准（ISO 9899—1990）。目前流行的 C 编译系统都是以它为基础的。本书的叙述基本上以 ANSI 为基础。

C 语言是一种结构程序设计的好语言，但随着软件处理对象从简单的数字和字符串发展到目前的图、文、声、像，信息量愈来愈大，愈来愈复杂，因而对程序的设计方法提出了更高的要求，随之在 20 世纪 80 年代出现了一种崭新的程序设计方法——面向对象的程序设计方法。结构程序设计的基本单位是模块，而面向对象的程序设计基本单位则是

对象。因此，C 又进一步发展和充实，出现了支持面向对象的程序设计语言 C++，它实际上是 C 的超集，其基本核心仍是 C。例如美国 BorLand 公司 1989 年推出了 Turbo C2.0，进而，又在继承和发扬 Turbo C2.0 的集成开发环境的基础上，推出了面向对象的程序设计语言 Turbo C++，它实际上是 Turbo C 的一个超集，Turbo C++ 包含了所有 Turbo C 的内容，因而学好、用好 C，仍是面向对象程序设计的基础和前提，它甚至也可体现出面向对象设计的一些方法。

第二节 C 语言的程序结构及特点

下面先介绍几个简单的 C 程序，然后从中分析 C 程序的结构及特点。

1. C 语言程序举例

例 1-1

```
main ( )
{
    printf (" Good morning! \n");
}
```

本程序的作用是输出以下一行信息：

Good morning!

这是一个非常简单的 C 程序，其中 main 表示“主函数”。每一个 C 程序都必须有一个 main 函数，C 语言总是从主函数开始执行。函数体由大括号“{ }”括起来。本例中主函数内只有一个输出语句，printf 是 C 语言中的输出函数，‘\n’是换行符，即在屏幕上输出“Good morning!”后回车换行。每条语句最后有一分号。

例 1-2

```
main ( )                                /* 主函数：求两数之和 */
{
    int a, b, sum;                       /* 定义三个变量 a, b, sum */
    a=123; b=456;                        /* 给 a、b 赋值 */
    sum=a+b;                             /* 求和的结果赋给 sum */
    printf (" sum is %d\n", sum);         /* 输出 sum 的值 */
}
```

本程序的作用是求两个整数 a 和 b 之和 sum。/* */ 表示注释部分，为便于理解，我们用汉字表示注释，当然也可以用英语或汉字拼音作注释。注释只是给人看的，对编译和运行不起作用。注释可以加在程序中任何位置。第 3 行是声明部分，定义变量 a 和 b，指定 a 和 b 为整型 (int) 变量。第 4 行是两个赋值语句，使 a 和 b 的值分别为 123 和 456。第 5 行使 sum 的值为 a+b，第 6 行中“%d”是输入输出的“格式字符串”，用来指定输入输出时的数据类型和格式，“%d”表示“十进制整数类型”。在执行输出时，此位置上代以一个十进制整数值。printf 函数中括弧内最右端 sum 是要输出的变量，现在它的值为 579 (即 123+456 之值)。因此输出一行信息为

sum is 579

2. C 语言的结构格式归纳

① C 语言程序通常使用英文小写字母书写，只有符号常量或其他特殊用途的符号才使用大写。应该注意的是，C 语言对大小写是区分的，它们代表着不同的变量。

② C 语言程序是由一条条语句组成的，每个语句都具有规定的语法格式和特定的功能。上面程序中，`printf()` 语句是输出变量数值的函数调用语句；`a=123` 是赋值语句。

③ C 语言程序使用“;”号作为语句间的分隔符。

④ C 语言程序中，一个语句可以占用多行，一行也可以有多个语句。不过为了使程序易于阅读，一般一个语句占用一行。

⑤ C 语言程序中使用大括号“{”和“}”来表示程序的结构层次范围。一个完整的程序模块要用一对大括号括起来，以表示该程序模块的范围。应该注意的是，左大括号“{”和右大括号“}”应该对应使用。

⑥ 为了增加程序的可读性，可以使用适量的空格和空行。但是，变量名、函数名和 C 语言保留字中间不能加入空格。除此之外的空格和空行可以任意设置，C 语言编译系统是不会理会这些空格和空行的。

⑦ 可以用 `/*...*/` 对 C 程序中的任何部分作注释。一个好的、有使用价值的源程序都应当加上必要的注释，以增加程序的可读性。

另外，应该指出的是 C 语言程序的书写格式自由度较高，灵活性很强，有较大的随意性。但是为了避免程序的层次混乱不清，还是建议读者在平时编程中，养成良好的编程习惯，使自己编写的程序便于别人阅读和理解，也便于自己今后的修改。

3. C 语言的特点

(1) C 语言简洁、紧凑 C 语言兼容了其他计算机语言的一些优点，其程序结构清晰、规整，表达式简练、灵活、实用。用 C 语言编写的程序可读性强，编译效率高。

(2) C 语言具有丰富的数据类型 在系统软件中，特别是操作系统中，对计算机的所有软件、硬件资源要实施管理和调度，这就要求有相应的数据结构作为操作基础。C 语言具有五种基本的数据类型：`char`（字符型），`int`（整型），`float`（浮点单精度型），`double`（浮点双精度型），`void`（无值型）和多种构造数据类型（数组、结构、联合、枚举）以及复杂的导出类型，C 语言还提供了与地址密切相关的指针及其运算符。

(3) C 语言具有丰富的运算符 C 语言的运算符多达 44 种，丰富的数据类型与丰富的运算符相结合，使 C 语言具有表达灵活和效率高等特点。

(4) C 语言是结构化程序设计语言 特别适合于大型程序的模块化设计。C 语言具有编写结构化程序所必需的基本流程控制语句，C 语言程序是由函数集合构成的，函数各自独立，并且作为模块化设计的基本单位。C 语言的源文件，可以分割成多个源程序，分别进行编译，然后连接起来构成可执行的目标文件，为开发大型软件提供了极大的方便。C 语言还提供了多种存储属性，使数据可以按其需要在相应的作用域起作用，从而提高了程序的可靠性。

(5) C 语言是中级语言 C 语言处于汇编语言和高级语言之间，它把高级语言的基本结构和汇编语言的高效率结合起来。因此，它既具有高级语言面向用户、可读性强、容易编程和维护等特点，又具有汇编语言面向硬件和系统，可以直接访问硬件的功能，C 语言的运行效率可以与汇编语言媲美。

(6) C 语言具有较高的可移植性 在 C 语言的语句中，没有依赖于硬件的输入输出语句，程序的输入输出功能是通过调用输入输出函数实现的，而这些函数是由系统提供的独立

于 C 语言的程序模块。从而便于硬件结构不同的计算机之间实现程序的移植。

由于 C 语言具有上述诸多特点, 因此近年来 C 语言迅速得到了广泛的普及和应用。

第三节 C 程序的上机过程

C 语言是一种高级程序设计语言, 比较容易书写和阅读。用 C 语言编写出来的程序通常称为 C 语言源程序, 简称 C 程序, 其扩展名为 “.C”。

由于计算机硬件只能识别和执行二进制代码形式的机器语言程序, 不能直接执行源程序, 因此, 必须首先将源程序翻译成二进制代码。Turbo C 就是一种将 C 源程序翻译成机器语言的程序, 称为编译程序。编译后的程序称为目标程序, 其扩展名为 “.OBJ”。

C 语言的编程、运行、调试一般分为以下几个步骤。

- ① 上机输入与编辑源程序。
- ② 对源程序进行编译。
- ③ 与库函数连接。
- ④ 运行目标程序。

其中, 编译源程序、与库函数连接如果出现错误, 应回到第一步重新对源程序进行修改。

下面简单介绍一下用 Turbo C 2.0 运行 C 程序的步骤 (详细见习题与上机指导)。

例如, 运行下面的程序:

```
main ( )  
{  
    printf (" Welcome to Turbo C");  
}
```

具体步骤如下。

- ① 调用 Turbo C 程序。

进入 TC 子目录: C: \>CD \ TC ✓

执行 TC.EXE 文件: C: \ TC>TC ✓

屏幕上出现 Turbo C 集成环境, 在集成环境的上部, 有一行 “主菜单”, 其中包括下面几个菜单项:

File Edit Run Compile Project Options Debug Break/watch

以上 8 个菜单项分别代表: 文件操作、编辑、运行、编译、项目文件、选项、调试、中断/观察等功能。

② 编辑源文件。在 Turbo C 的集成环境下输入源程序并保存, 选择 “File” 菜单下的 “Save”, 起名并保存文件。比如起文件名为: W1.C。

③ 编译源程序。选择 “Compile” 菜单并在其下拉菜单中选择 “Compile to OBJ”, 则进行编译, 得到一个后缀为 .obj 的目标程序。

④ 连接目标文件。选择 “Compile” 菜单下的 “Link EXE file”, 进行连接操作, 可得到一个后缀为 .exe 的可执行文件。

⑤ 执行程序。选择 “Run” 菜单下的 “Run” 命令或直接按 “Ctrl+F9” 就可以执行 W1.exe 文件, 得到执行结果。

本章小结

本章主要介绍了C语言的来源、特点，以及C语言程序的基本结构和上机过程。通过程序举例，使学生了解了C程序的基本组成及特点。

1. C语言的特点

- ① C程序是由函数构成的。
- ② 一个C程序有且仅有一个main函数，程序总是从main函数开始执行，从main函数退出。

2. C上机的过程

C上机由编辑、编译、连接、执行四步组成。如果执行结果不正确，应回到开始重新编辑程序。

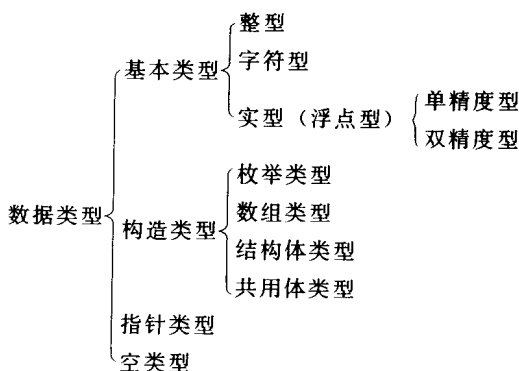
第二章

数据类型与基本输入输出

第一节 C 语言的数据类型

计算机的基本功能是进行数据处理。在 C 语言中，数据处理的基本对象是常量和变量。不管是常量还是变量，这样的数据都是以某种特定的形式存在的（例如整数、实数、字符等形式），不同数据类型的数据在计算机中具体存放形式也是不同的。另外，不同的数据之间往往还存在某些联系（例如由若干个整数组成一个整数数组）。所以，我们在使用不同类型的数据时，就应该先对数据进行类型说明。

C 语言提供的数据类型有以下几种：



例 2-1 C 语言中最简单的数据类型包括：

- (a) 整型、实型、逻辑型；(b) 整型、实型、字符型；
(c) 整型、字符型、逻辑型；(d) 整型、实型、字符型、逻辑型。

答案：b

分析：C 语言中没有逻辑型。最简单的数据类型就是指基本类型。

本章主要介绍基本数据类型。这种数据分类是从数据的存在形式上划分的，如果从变化的角度，C 语言中的数据还可分为常量和变量两类。

第二节 常 量

常量是在整个程序运行过程中其值保持不变的量。常量可分为字面常量（或直接常量）和符号常量。常量根据类型不同，也可分为整型常量、实型常量、字符常量和字符串常量。

一、标识符

Turbo C 可以为用户定义各种标识符作为常量、变量、函数、标号及用户定义对象的名称。组成 C 语言的用户标识符应符合以下规则。

① 标识符的有效长度为 1 至 32 个字符。但为了程序的可移植性，以及阅读程序方便，建议长度不要超过 8 个字符。

② 标识符的第 1 个字符必须是字母或下划线，后续字符可以是字母、数字或下划线。

Turbo C 也允许在标识符中使用字符 \$，但 \$ 是非标准字符，建议最好不用。以下列举了几个正确的和不正确的标识符名称。

正确	不正确
area	123area
test23	th+ere
x_point	high..balance

在第 1 个例子中由于 123area 是以数字开头，所以是非法的；第 2 个例子 th+ere 中“+”是不允许出现在标识符中的；而在第 3 个例子 high..balance 中“..”是非法标识符。

在 C 语言中，大小写字母是有区别的，所以，area、Area、AREA 是三个不同的标识符。标识符不能和 Turbo C 的关键字相同，也不应该和已定义的函数名或 Turbo C 的库函数名相同。

在选择标识符时，应注意做到“见名知意”即选有含意的英文单词（或缩写）、汉语拼音作标识符，如 area、count、name、sum 等，以增加程序的可读性，这是结构化程序的一个特征。

二、整型常量

整型常量即整型常数。C 整型常量可用以下三种形式表示。

① 十进制整数。这是最常用的一种形式，例如 345、-12、0。

② 八进制整数。以 0 开头的数是八进制数，例如 010 表示八进制数 10，等于十进制数 8。-0123 表示八进制数 -123，即十进制数 -83。

③ 十六进制整数。以 0x 或 0X 开头的数是十六进制数，其数码取值为 0~9，A~F 或 a~f。例如 0X2A、0XFFFF、0X10，其中 0X10 代表十六进制数 10，等于十进制数 16，-0X123 代表十六进制数 -123，等于十进制数 -291。

三、实型常量

实型常量又称浮点型常量。实型常量也可称为实数或浮点数。在 C 语言中，实数只采用十进制，可有以下两种表示形式。

① 十进制小数形式。它由数字和小数点组成（注意必须有小数点）。例如 1.23、-32.4、0.0、.25、3. 等。

② 指数形式。它由十进制数和阶码标志“e”或“E”以及阶码组成。例如 123e3 或 123E3 都代表 123×10^3 ，123e-3 或 123E-3 都代表 123×10^{-3} 。但注意字母 e（或 E）之前必须有数字，之后的指数必须为整数。

一个实数可以有多种指数表示形式。例如 123.456 可以表示为 123.456e0，12.3456e1、

1.23456e2、0.123456e3、0.0123456e4 等。把其中的 1.23456e2 称为“规范化的指数形式”，即在字母 e（或 E）之前的小数部分中，小数点左边应有一位（且只能有一位）非零的数字。例如 2.3478e2、3.0999E5、6.46832e12 都属于规范化的指数形式，而 12.908e10、0.456e3、756e8 都不属于规范化的指数形式。一个实数在用指数形式输出时，是按规范化的指数形式输出的。例如，指定将实数 5689.65 按指数形式输出，必然输出 5.68965e+03。

四、字符常量

字符常量是用单引号括起来的一个字符。其中单引号只作为定界符使用，并不表示字符常量本身。例如 ‘a’，‘A’，‘5’，‘?’，‘+’ 等都是合法字符常量。

在 C 语言中，字符常量具有数值特性，字符常量的值就是该字符的 ASCII 码值。因此，可以说字符常量实际上是一个字节的整数。一般说来，字符常量分为以下两种类型。

① 单引号（即撇号）括起来的一个字符。如 ‘a’、‘*’、‘1’ 等。注意，‘a’ 和 ‘A’ 是不同的字符常量，整数 1 和字符 ‘1’ 也不是同一值。

② 转义字符。用反斜线 “\” 后面跟一个字符或一个数字表示，即以一个 “\” 开头的字符序列。例如，前面已经遇到过的，在 printf 函数中的 ‘\n’，它代表一个“换行”符。这是一种“控制字符”，在屏幕上是不能显示的。转义字符是 C 语言中使用字符的一种特殊表现形式。转义字符常用于表示 ASCII 字符集内的控制代码和某些用于功能定义的字符，如单引号、双引号和反斜杠等。

表 2-1 是常用的转义字符及其含义表。

表 2-1 转义字符及其含义

字符形式	含 义	ASCII 代码
\n	换行，将当前位置移到下一行的开头	10
\t	横向跳格，跳到下一个 tab 位置	9
\b	退格，将当前位置移到前一位	8
\r	回车，将当前位置移到本行的开头	13
\f	换页，将当前位置移到下页的开头	12
\\	反斜杠字符 “\”	92
\'	单引号字符	39
\"	双引号字符	34
\ddd	1 到 3 位 8 进制数所代表的字符	
\xhh	1 到 2 位 16 进制数所代表的字符	

表 2-1 中列出的字符称为“转义字符”，意思是将反斜杠（\）后面的字符转换成另外的意义。如 ‘\n’ 中的 “n” 不代表字母 n 而作为“换行”符。

‘\ddd’ 是用 ASCII 码（八进制数）表示一个字符，例如，‘\101’ 代表 ASCII 码（十进制数）为 65 的字符 ‘A’。原因是八进制的 101 转换成十进制就是 65。通过查 ASCII 码表（见附录），ASCII 码为 65 的字符是 ‘A’。‘\12’（十进制 ASCII 码为 10）代表“换行”。

‘\xhh’ 是用 ASCII 码（十六进制数）表示一个字符，例如，‘\x42’ 是 2 位十六进制数所代表的字符，十六进制的 42 转换成十进制就是 66。查表可知，ASCII 码为 66 的字符是 ‘B’。

利用表 2-1 中的方法可以表示任何可输出的字母字符、专用字符、图形字符和控制字符。

例 2-2 在 C 语言中，合法的字符常量是 ()。

(A) '\084' (B) '\x43' (C) 'ab' (D) "\0"

分析：很明显 (C) 和 (D) 都不是合法的字符常量，(A) 和 (B) 为转义字符常量，而 (A) 选项中的字符常量是用来表示一个八进制数，但八进制数只有 0~7 八个数字，所以“084”不能用来表示一个八进制数。(B) 选项的字符常量是一个十六进制数，'\x43' 代表字符 'C'。

因此，本题的答案是 (B)。

五、字符串常量

字符串常量是用一对双引号括起来的字符序列。其中双引号仅作为定界符使用，并不是字符串中的字符。例如，“How are you”，“china!”，“a”，“12345”都是字符串常量。

C 语言的字符串常量有与其他语言不同的独特性质。字符串常量在内存中存储时，系统自动在其尾部追加一个字符串结束标志——空操作字符 '\0'。在 ASCⅡ 码中，字符 '\0' 的值为 0，占内存一个字节。它不引起任何控制动作，也不可显示，只用于系统判断字符串是否结束。因此，长度为 n 个字符的字符串常量，在内存中占用 n+1 个字节的空。C 语言的字符串常量由于具有这种特性，所以一般称它为 C 字符串。

C 语言可以输出一个字符串，如

```
printf("How are you.");
```

例 2-3 写出下面输出结果。

```
main ( )
{
    printf(" \x30 \x49 \044 \012 \t \101");
}
```

运行结果如下：

0I\$

□□□□□□□□ A

分析：这是一个由转义字符组成的字符串，'\x30' 代表 '0' 字符，'\x49' 代表 'I' 字符，'\044' 代表 '\$' 字符，'\012' 代表 '换行' 控制符，'\t' 默认代表光标跳 8 列，'\101' 代表 'A' 字符。

不要将字符常量与字符串常量相混淆。'A' 是字符常量，“A”是字符串常量，二者不同。字符串常量与字符常量之间的主要区别是：

(1) 引号不同 字符常量由单引号括起来，字符串常量由双引号括起来。

(2) 容量不同 字符常量只能是单个字符，字符串常量则可以包含一个或多个字符。

(3) 占用内存大小不同 字符常量占一个字节，字符串常量占内存的字节数等于字符串中字符个数加 1 (增加一个放 '\0' 的字节)。

(4) 赋给变量不同 可把一个字符常量赋给一个字符变量，但不能把一个字符串常量赋给一个字符变量。在 C 语言中没有相应的字符串变量，要存放一个字符串常量可以使用字符数组。这将在后面章节介绍。

注意，C 规定在每一个字符串的结尾加一个字符 ‘\0’ 作为字符串结束标志。但在输出字符串时不输出 ‘\0’。在写字符串时也不必加 ‘\0’，否则会画蛇添足。‘\0’ 字符是系统自动加上的。字符串 “A”，实际上包含 2 个字符 ‘A’ 和 ‘\0’，因此，把它赋给只能容纳一个字符的字符变量 c：

```
c = " a";
```

显然是不行的。

六、符号常量

在编写程序时，对于许多常用的常量可以用一个标识符来代替。

例 2-4 计算半径为 3 的圆周长和圆面积。

方法一：

```
main ( )
{
    float r, cirle, area;
    r=3.0;
    cirle=2 * 3.1415926 * r;
    area=3.1415926 * r * r;
    printf (" cirle=%f, area=%f\n", cirle, area);
}
```

方法二：

```
#define PI 3.1415926
main ( )
{
    float r, cirle, area;
    r=3.0;
    cirle=2 * PI * r;
    area=PI * r * r;
    printf (" cirle=%f, area=%f\n", cirle, area);
}
```

这两种方法都是正确的。区别在于方法二中用 #define 命令行定义 PI 代表常量 3.1415926，此后凡在本程序中出现 PI 的地方都用 3.1415926 来代替，PI 可以和常量一样进行运算。

这种用一个标识符代表的一个常量，称为符号常量，即标识符形式的常量。请注意符号常量不同于变量，它的值在其作用域（在本例中为主函数）内不能改变，也不能再被赋值。如再用以下赋值语句给 PI 赋值是错误的。

```
PI=3.1415;
```

习惯上，符号常量名用大写，变量名用小写，以示区别。使用符号常量便于程序的阅读和维护，具体表现在：

- ① 含义清楚。如上面的程序中，看程序时从 PI 就可知道它代表圆周率，“见名知意”；
- ② 在需要改变一个常量时能做到“一改全改”。例如在程序中多处用到圆周率，如果圆