



语言 程序设计

金連甫 編
浙江大学出版社



C语言程序设计

金连甫 编

浙江大学出版社

1989年·杭州

内容简介

本书详细介绍了C语言及其程序设计方法,提供了丰富的程序实例,使初学者容易理解和掌握,有利于培养实际编程能力。本书既重视C语言的语法和语义的描述,又突出C语言的应用,加强编程能力的训练,最后还较详细地针对C语言程序设计讲授了程序设计风格和方法。

本书可作为大专院校程序设计课的教课书,也可供从事计算机的技术人员参考。

C 语言 程 序 设 计

金连甫 编

责任编辑 尤建忠

* * *

浙江大学出版社出版

德清雷甸印刷厂印刷

浙江省新华书店经销

* * *

开本 850×1168 1/32 印张 12 字数 279 千

1989年10月第1版 1989年10月第1次印刷

印数: 00001-3000

ISBN 7-308-00321-3

TP·021 定价: 3.05元

前 言

C程序设计语言是贝尔实验室于70年代初期在已开发的程序设计语言基础上发展起来的。它与UNIX操作系统的发展有着密切的关系。UNIX操作系统早期是用汇编编写的，后来用C语言重写，使新的操作系统的规模仅为旧的三分之一。这一尝试的成功，确信C语言是一种有效的程序设计工具。UNIX操作系统的普及促进了C语言的发展和推广。由于C语言有着经济实用的表达式、丰富的控制流和数据结构，语言简单、灵活、易于实现，因此C语言在大型机、中型机、小型机以及微型机上都得到实现，且不仅UNIX操作系统支撑的计算机上有C语言，而且在许多非UNIX操作系统支撑的计算机上也实现了C语言，使C语言成为一种当今主流程序设计语言。

C语言是通用的程序设计语言，它不针对任何具体的机器，语言本身考虑到了可移植性，因此有着良好的可移植的性能。

C语言容易学习，容易编写。用C语言编制的程序清晰紧凑，有助于模块化和好的程序结构。

C语言在提高目标程序的效率方面作了努力，使编译出的目标质量好，适应于编制各种系统软件、应用软件，是程序设计的一种理想的语言工具。

195123/02

本书为使初学者易于入门，使已经熟悉C语言的读者又能有新的收获，书中既介绍C语言的各种特征和组成部份，又介绍编程方法和技巧，并也包含了它的最新进展。在内容安排上深入浅出，由浅入深，简明易懂，突出了C语言特点和实用性。

本书共分十章，第一章介绍C语言的特点和概念，使初学者对C语言有一个初貌了解。以后六章介绍C语言的基本语法成分和语义，突出数据的构造类型，并给出了较多的实际例子。第八、九章介绍C语言的库函数，着重叙述标准文件的输入输出和一般文件的操作。最后一章介绍了程序设计风格和方法，使读者能养成一个好的程序设计风格和采用好的程序设计方法。书中给出较多的应用实例，有利于读者自学和加深对C语言的语法、语义的理解。每章均有一定的练习，以便读者复习巩固。

在编写过程中，浙江大学计算机系陈增武、王泽兵同志为编写本书提供了非常有益的建议，并仔细审阅了手稿，提出了宝贵的意见，借此表示衷心的感谢。

由于时间仓促，加之本人水平有限，书中缺点有所难免，恳望得到批评和指教。

金连甫

1989年1月于杭州

目 录

第一章 导论

§ 1.1 引言	1
§ 1.2 C语言的特点	3
§ 1.3 C语言概述	6
§ 1.4 C语言的编辑、编译和运行	15
练习	20

第二章 数和表达式

§ 2.1 标识符和变量	21
§ 2.2 基本数据类型	23
§ 2.3 常量	27
§ 2.4 表达式	30
练习	42

第三章 语句和控制流

§ 3.1 简单语句和复合语句	44
§ 3.2 条件语句	45
§ 3.3 开关语句	53
§ 3.4 循环语句	57
§ 3.5 break和continue语句	67
§ 3.6 goto语句与标号	69
§ 3.7 空语句	72
练习	73

第四章 函数与程序结构

§ 4.1 函数	75
§ 4.2 程序结构	90
§ 4.3 作用域	98
§ 4.4 初始化	108
§ 4.5 C 预处理器	110
练习	119

第五章 构造类型——数组和指针

§ 5.1 数组	122
§ 5.2 指针	134
§ 5.3 指针与数组	140
§ 5.4 指针与函数	151
练习	165

第六章 构造类型结构

§ 6.1 结构及结构变量	167
§ 6.2 结构和函数	177
§ 6.3 结构和数组	183
§ 6.4 结构嵌套	193
练习	198

第七章 其它构造类型

§ 7.1 位段	201
§ 7.2 联合	208
§ 7.3 枚举类型	211
§ 7.4 定义类型	216

§ 7.5 数据类型转换	218
练习	221

第八章 库函数及标准文件的输入输出

§ 8.1 利用标准库的方法	223
§ 8.2 UNIX的C库	232
§ 8.3 标准文件的字符输入输出	235
§ 8.4 格式化输出函数printf	238
§ 8.5 格式化输入函数scanf	244
§ 8.6 内存中的格式转换	249
练习	250

第九章 文件操作

§ 9.1 UNIX文件系统简介	252
§ 9.2 低级输入输出	259
§ 9.3 文件处理的库函数	274
练习	291

第十章 程序设计风格和方法

§ 10.1 引言	293
§ 10.2 程序设计风格准则	296
§ 10.3 健全程序的设计	308
§ 10.4 自顶向下的程序设计方法	316
§ 10.5 程序设计组	325
§ 10.6 程序模块化	328

附录A C语言汇总	337
-----------	-----

附录B ASCII码集合	367
--------------	-----

导 论

§ 1.1 引言

C 程序设计语言是一种通用的程序设计语言，是七十年代初期由美国贝尔实验室 Dennis M. Ritchie 设计的，正式发表于1978年。

C 语言既不是“非常高级”的语言，也不是一个很大的语言，它不是为任何特定的应用领域而专门设计的。它的通用性和无限制性使得它对许多程序设计来说都比想象中的功能的语言更加通俗、更加有效。目前C语言已用于各方面的程序设计，无论设计系统软件（如操作系统、编译系统等），还是应用软件（如图形处理）、数据处理（如企业管理）以及数值计算等都可以很方便地使用C语言。

今天所用的C语言是几年来演化的产物。它的许多重要思想来源于Martin Richards研制的且比较老的但有生命力的BCPL语言。BCPL语言对C语言的影响是通过B语言进行的，1970年Ken Thompson用B语言写了PDP-7上第一个UNIX系统。

虽然C语言与B和BCPL语言有若干共性，但不能认为C就是B的一种改进型，而是具有较大的差异。BCPL和B语言都是“无类型的”语言：唯一的数据类型是机器字，欲存取其

它种类的对象要通过专门的操作符或函数调用来进行。在C语言中,基本的数据对象是字符、不同长度的整数和浮点数。此外还有用指针、数组、结构、联合和函数建立起来的导出数据类型的体系。

C语言与UNIX操作系统紧密地联系在一起,这是因为C语言是在UNIX上研制的,而且UNIX操作系统及其软件又是用C语言书写的。UNIX操作系统的早期文本是用汇编书写的,在1973年夏季又用C语言重写,新系统的规模约为旧系统的三分之一,这一尝试的成功,使人确信C语言可以作为系统程序设计的有效工具。随着C语言的完善和发展,它的充分表达能力及其有效性使它几乎完全取代了UNIX上的汇编语言。事实上除UNIX操作系统本身之外,UNIX系统的所有例行程序都是用C语言写的。因此,C语言常常被称为“系统程序设计语言”(因为它在编写系统程序方面发挥了巨大的威力)。但是,如果认为它仅应用于系统程序设计方面则是十分片面的,其实UNIX系统下的应用程序几乎都是用C语言编写的,它所提供的相当完备的数据结构(利用指针和结构可以十分方便地构造出链表、双链表、树、图等几乎所有的基本数据结构)以及十分丰富的运算(包括一般高级语言缺乏的位处理操作),使得它大量地应用于数值计算、文字处理、数据库管理系统等方面。

UNIX系统是C语言发展的摇篮,同时C语言又为UNIX系统的扩充与发展提供了极为有利的条件,两者相互依赖,相互促进,不断完善,不断发展。当今UNIX系统已在全世界范围内取得了巨大成功,它几乎成了16位机的标准操作系统,在PDP-11小型机,VAX-II超级小型机,甚至像IBM370,Honeywell等大型机上也有它的踪迹。随着微机的发展,8位机上也移植,如Z80为CPU的Cromemco,8088为CPU的

IBM-PC机上。C语言最初是附属于UNIX系统且在PDP-11上实现,但是目前C语言却独立于UNIX操作系统,独立于PDP-11机而蓬勃发展,有的甚至在一种机型下提供多种C语言编译系统。例如,当前社会应用甚广的IBM-PC机,该机上在DOS支持下的C语言编译就有C86、C88、MSC、TURBO C和解释C语言等,显示出C语言的生命力及社会对其的关注。

§ 1.2 C语言的特点

当今世界上通用的程序设计语言很多,有BASIC、FORTRAN、PASCAL、ALGOL、COBOL、ADA、LISP…等,各种语言都有其各自的特点和形式,对熟悉多种语言的读者,只有搞清相应语言的特点,才能避免在使用中造成混淆。

C语言的特点是多方面的,人们从不同角度总结出众多特点。下面从几种语言比较角度来讨论C语言的特点:

第一,C语言吸取了汇编语言的长处,使C语言相对高级语言来讲是“低级”语言。

众所周知,汇编语言是一种面向机器的程序设计语言,尽管它的编程相对高级语言来要麻烦得多,但由于它具有描述准确和目标程序质量高的优点,汇编语言仍然有很强的生命力。C语言吸取了汇编语言中的精华。

1) C语言中提供对地址的操作。C中提供了“指针”概念,引进了二个一目运算符“&”和“*”。“&”给出了一个目标地址,如:

$$px = \&x;$$

将变量X的地址赋给指针变量px; “*”使得操作数当作地址,

如:

```
yc = *px;
```

将指针px中的内容当作地址,然后将该地址的内容赋给yc。

这样程序就可以直接对内存、指定寄存器进行操作。

2) 提供寄存器类型,可以把程序中使用频率高的变量说明为寄存器类型,使系统尽可能对该变量使用寄存器,以提高运行速度。

3) 提供了&(与)、|(或)、^(异或)、<<(左移)、>>(右移)、~(求反)等位操作,使C语言操作准确到位。

4) 吸取了宏汇编技术中的某些灵活的处理方法,在C语言中设置了二个宏:宏代换#define和文件蕴含#include。

5) C语言能够很方便地与汇编语言连接,C程序中引用汇编例程与引用C语言函数一样,有的机器甚至允许C程序中插入一段汇编程序,这为某些特殊程序的编写提供方便。

第二,C语言继承发扬了高级语言的长处,使C语言相对汇编来讲又是“高级”语言。

1) 吸取了ALGOL 60的分程序结构思想,在C语言中设置了用花括号括起的复合语句(分程序),在花括号内可以对变量进行定义,但不能对函数进行定义。

2) 吸取了FORTRAN的模块结构思想,C语言的每个函数都是独立的,能够对各函数进行独立编译,这样对设计一个大程序有利于分工、编程、调试等。

3) 继承发扬了高级语言中的控制语句结构,C的控制语句基本上类似于ALGOL、PASCAL等高级语言。

4) 继承发扬了PASCAL语言中的数据类型,提供了相当完备的数据结构。这样可以十分方便地构造出链表、树等。

5) C语言中任何函数都允许递归调用,这样对某些算法

实现起来就十分方便、清晰。

第三，C语言的规模保持在适中的程度上，却也确实带来了不少益处。C语言的简洁，使用户可以很快地被掌握。由于C语言的规模保持在适中的程度上，其编译程序也可以变得简单而紧凑。利用目前的编译技术，为一台新机器写一个C编译程序是相当容易的。其规模适中具体表现在：

1) C语言没有整体处理字符串、集合、表格和数组的能力。例如它没有类似PL/1的处理数组和字符串的操作。

2) C语言本身不提供输入/输出机构，即没有在一般高级语言中常用的READ、WRITE语句和固定的文件存取方式。所有这些高级机构都须调用显式函数的方法来实现。

3) 除了对静态定义和函数局部变量要求的栈结构之外，C语言没有提供任何存储分配机构，没有类似于ALGOL68的全程区(heap)定义和碎片空间收集之类的机构。

4) 在控制方式上也只提供直接的单线控制流程结构。尽管C被大量地应用于书写操作系统和相应的软件，但是它并没有提供多道程序设计、并行操作、同步或协同程序等复杂控制机构。

第四，C语言在提高目标程序的效率方面所作的努力也是卓有成效的，使生成的代码质量高。用高级语言来编写程序显然比汇编语言方便且容易得多，程序调试工作也可以大大减少，这大大缩短了软件的研制周期。C中提供了位操作，不同长度的整数，寄存器类型和提供增减操作，为C的高效率作出具体的努力。例如，`++i;`语句确实能够产生比语句`i = i + 1;`速度快得多的机器的代码。所以C就成了人们描述系统软件和应用软件比较理想的工具。在代码效率方面的确可以和汇编语言媲美。

第五，C的可移植性好。这是指程序可以从一个环境不加

或稍加改动就搬到另一个完全不同的环境上运行。汇编语言因为依赖于机器软件，所以根本不可移植，而一些高级语言，如FORTRAN等的编译程序也不可移植。目前许多机器上几乎都配有FORTRAN，但这基本上都是根据国际标准重新实现的。而C语言目前在许多机器上出现，大部分却是由C语言编译移植得到的。统计资料表明，不同机器上的编译程序80%的代码是公共的。C语言的编译程序便于移植，也就是使一个环境上用C编写的许多程序可以很方便地移到另一个环境上。

第六，C语言格式自由、限制少，编写程序较自由，为编制程序带来了方便。

C语言的优点很多，但也有一些不足和缺点，虽然有些已得到了弥补。这些不足和缺点包括有：类型检验太弱，转换比较随便，所以不太安全（目前已有一个预处理程序 lint，它主要用来检验类型协调匹配，帮助解决移植中的一些问题）；还有运算符优先级太多，不便记忆；有些还与常规约定有所不同等。尽管如此，C语言仍不失为一个实用的通用程序设计语言，由于上述的几个突出优点而使人们对它产生越来越多的兴趣，以至当今国内外使用和研究C语言的人数迅速增多。实际应用经验表明，程序设计人员一旦接触了这种语言，并且有一定程序设计经验后就会对它爱不释手。

§ 1.3 C语言概述

本节我们对C语言作一简要介绍，使读者对C语言概貌有一个初步的了解和印象，以便以后各章可以深入地开展C语言的讨论。另外，对C语言的编程、编译、运行等也作适当的介绍，以便读者可以边学习边进行实际的上机训练。

一、基本符号

任何语言都有它自己特定的一套基本符号，C语言也不例外，由如下一些基本符号组成。

(1) 数字：0, 1, 2, ..., 9;

(2) 英文字母：A, B, C, ..., Z, a, b, c, ..., z。在书写源程序时一般习惯用小写英文字母，在内部处理时往往以小写字母为主，程序中只有对定义的常量等用大写字母。

(3) 下划线字符：_，下划线字符在C语言中用来构成标识符，在C语言中以下划线打头是合法的。

(4) 特殊符号，主要包括下列一些运算符和关键字：

运算符有：+、-、*、/、%、=、<、>、<=、>=、!=、==、<<、>>、&、|、&&、||、^、~、()、[]、->、.、,、!、? 等。

关键字有 int, char, float, double, short, long, unsigned, struct, union, auto, extern, register, static, typedef, goto, return, sizeof, break, continue, if, else, do, while, switch, case, default, enum, for, void, entry, define, include, undef, ifdef, ifndef, endif 和 line 等。关键字 entry 目前还未由任何编译程序实现，但保留给将来使用。最后七个字是在C语言预处理程序中使用，所以不严格地属于关键字，但在编程中读者把它看作关键字是有好处的，它可以避免造成混淆。C语言中的关键字具有固定含义，不能作为一般标识符使用。

由于C语言的字符处理能力较强，一般来说C编译可不作改动在汉字操作系统支撑下，将汉字当作字符串处理，并能在注释中使用汉字或字符来进行注释。所以在汉字状态时，C语言的字符常量中还允许出现 GB2312-80 中规定的国标一二级

汉字等。

二、C 语言程序结构

C 语言程序一般是由若干个函数组成，而这些函数可以保存在一个或几个源程序文件中，这些文件都是以 .C 为文件扩展名。在组成一个程序的若干函数中必须有一个且只能有一个名为 main 的函数，在运行 C 程序时总是从 main 函数开始执行。一个函数在其名字之后一定要有一对圆括号，这是函数的标志。圆括号中的参数可有可无。

下面是一个打印 “The C programming Language” 的小而简单的程序：

```
/* ..... */  
/* 程序1.1   example.c      */  
/* C语言程序例              */  
/* ..... */  
main()  
{  
    printf("The C programming Language.");  
}
```

上述是一个完整的 C 程序。程序中，以 “/*” 开头到 “*/” 结尾之间的内容是一个注释，注释将帮助阅读和理解程序，但在编译时，注释行是要忽略掉的，即它不产生代码行。注释在 C 程序中是很有用的，对一个好程序设计者来说，应该在程序中使用注释来说明整个程序的功能和注意事项，以及有关算法等。

程序 1.1 中第 6、8 行实际上是一对花括号，类似于 PASCAL 中的 “begin” 和 “end”，是一个程序体。printf(“The C programming language.”); 语句行是一个函数调用，调用名叫 printf 的函数，双括号括起来的是所带的参数，即要打

印的内容。printf是库函数，它在标准输出设备（终端上）打印输出。

C中的函数等价于FORTRAN中的子程序或函数，等价于PL/1、PASCAL等中的过程。函数提供了一种方便方法，它把某些计算封装在黑盒子里，使用时无需考虑它的内部结构。函数实际上是对大程序所隐含的复杂性的唯一方法。正确地设计函数可能做到只要知道一个作业做什么就够了，不需要知道它究竟怎么做的。C语言函数的使用简单、方便、执行效率高。我们常常见到一个函数只有几行长，且只被调用一次，就是因为它使这几行代码看起来清晰。所以在C语言程序设计中，要养成良好的设计风格，即用多个小函数构成一个大程序。一个训练有素的C程序员应该把大程序分解成若干个独立完成单独一个功能的小程序。如果每个功能都能独立，那么程序员可以把这些功能都看成“零件”，需要时就可以把这些“零件”组成各种“大机器”（程序），而不必每次都重新从头开始构造大程序。下面例子中给出一个求指数函数power(m, n)及调用它的主程序。

```
/. ..... ./
/. 程序1.2      ./
/. 功能：指数函数      ./
/. ..... ./

main()    /*测试 POWER() 函数 */
{
    int i,
    for(i=0, i<10, ++i)
        printf ("%d %d %d\n", i, power(2, i), power(-3,i));
}

power(x, n) /*求X的N次幂 N>0 */
int x, n;
```