



程序设计教程

Thomas Plum 著
金茂忠 张子让 译
周伯生 校

科学普及出版社

TP312
71

软 件 工 程 丛 书

C 程序设计教程

〔美〕 Thomas Plum 著

金茂忠 张子让 译

周伯生 校

科 学 普 及 出 版 社

内 容 提 要

C语言是一种通用的结构程序设计语言,它表达简洁,具有先进的控制结构 and 数据结构及丰富的操作符,并反映了现代计算机系统的公共特征,具有效率高、可移植性好等特点,C语言可用于系统程序设计和应用程序设计,用C语言成功地编写了UNIX操作系统,从而使它得到广泛流行。

本书通俗、清晰、详尽地阐述了C语言的基本特征和C语言程序设计方法,并通过程序设计实例介绍了软件开发基本技术和软件工程基本概念。

本书可作为大专院校有关专业的教科书或教学参考书,也可供科学技术人员及其它从事计算机程序设计的人员参考。

Bees / 2

Learning to Program in C
Thomas Plum

Prentice-Hall, Inc. 1983

软件 工 程 丛 书

C 程 序 设 计 教 程

金茂忠 译
张子让
周伯生 校

*
科学普及出版社出版(北京海淀区魏公村白石桥路32号)
新华书店北京发行所发行 各地新华书店经售
北京市密云县印刷厂印刷

*
开本: 850×1168毫米 1/32 印张: 11.5 字数: 300千字
1986年8月第一版 1986年8月第一次印刷
印数: 1—7000 定价3.10元
统一书号: 15051·1203 本社书号: 1324

“C程序设计教程”介绍

C语言的特点是紧紧地把握住了现代计算机系统结构的公共特征,使其适用于从微型机、小型机直至大型机等各类处理机。这种“实现者的语言”使小巧、快速的程序不做任何改动即可运行在所有这些处理机上。不论以前是否有过程序设计经验,你都可以学会这个强有力的语言的基本原理,并将其应用于实时程序设计、信号处理、电子工程、应用程序包设计或是复杂的个人计算。

Thomas Plum

前 言

本书试图通过C程序设计语言向读者介绍计算机的程序设计技术。它并不要求读者先前具有程序设计的经验，但是熟悉其他程序语言的读者将会发现本书的许多阐述对他们也非常有用，这一点已在以往的许多C语言培训班上得到证明。

本书将C语言作为一种通用的计算机语言介绍给那些对可移植性及效率感兴趣的程序员。因而并不侧重在某一专门的应用领域，例题的选择完全是为了解释C语言的性质。本书着重在教学目的而不是作为参考手册；也没有试图在语法细节上代替编译程序手册。现在的内容是根据五年多C语言教学经验不断演化而来的。我竭力在讲解程序设计的基本原理（它独立于语言和环境）和介绍程序库、编译过程及软件维护等有用的细节之间做出合理安排。目的是给出在一个真实的软件工程环境中工作的一名合格的程序员所必备的知识。

我希望本书能够交流那些除了技术细节以外的使用C语言的乐趣。C是适用于不同类型计算机

程序设计的一种极好的设计工具，它朴素优美，对很多应用领域又有足够的功能。使用它，你可以写出与任一特定计算机硬件无关的精巧、高效的应用程序。当你在你的计算项目中使用C时，就会体会到为什么许多专职程序员如此热烈地喜爱C语言。

附录A给出了一个关于C语言及UNIX（及许多其他）系统上通常提供的程序库的简明指南。这些指南并不是要取代你的随机的编译程序手册中所包含的语法参考的细节，而是一个为书写**可读**C程序的规则的总结。

附录B详尽地讨论了环境问题，例如如何编译一个程序、字节和字的大小，以及依赖编译的特征。这个附录同时也罗列了本书正文中所有的程序清单。在附录B中也给出了以**黑体字问题**开头的思考题的答案。

标记为**黑体字练习**的编程实习留给读者以发挥他们的创造力。

我非常感谢那些在本书写作过程中给予我鼓励与帮助的人们。首先，我要对P.J.Plauger表示特殊的感谢，既应感谢他在成书过程中的作用，又应感谢他将C语言应用到种种不同的环境中，事实上，如果不是因为其他工作占用了他很多时间，他会成为本书的作者之一。书中出现的“我们”及“我们的”并非虚拟—我保留这些字是希

望他们能与我合作产生本书未来的版本。

如上所述，“我们”也经常指Plum Hall公司的其他讨论班教员，特别是S.Schustack和D.Graham，他们对C语言的清晰表达的琢磨使得许多例题更加典型。对各次草稿进行过评述的还有C.Gelber、J.Justice、B.Kernighan、I.Macleod、E.Rathje、G.Rose、T.Whyman和我的母亲R.Schunior。我对他们的感谢当然并不意味着他们对本书的各方面都持赞成态度——对一个写作、编程、测试并自己排版打印的作者来说是没有理由为尚存的错误责备任何其他人的。

对于我的家庭成员——Halls、Pritchetts、Richters、Schuniors、Schanzes和Whynmans——的坚定支持和鼓励，我表示由衷的感谢。

我还要特别感谢D.Plauger、T.Plauger以及Whitesmiths公司的其他工作人员。D.Bolinger编写了正文格式化程序ctext，并给出了供排版打印用的样本。Whitesmiths公司提供了可靠的编译程序以测试程序。这些程序不经改动就可在三种不同结构的机器上运行。他们的Idris操作系统为准备本书中程序的运行提供了可靠的支持。我同样要感谢VenturCom公司提供UNIX的Venix版本，在本书写作过程中多次使用了Venix。

C语言的原始作者，贝尔实验室的D.Ritche，应得到C语言界所有人的感激。C语言并不是由某

个委员会设计出来的，这一点对于我们中的许多人来说具有很大的吸引力。

我非常高兴地向Plum Hall公司杰出的职员S. Whynman和L. Deutsch为使本书得以出版所做的一切努力表示感谢。

我尤其要感谢J. Hall—我的妻子和Plum Hall公司的董事长，她使我从一个助理教授成长为一个好的指导者。

Thomas Plum

目 录

第 0 章 引言.....	(1)
第 1 章 计算机和C	(5)
1.1 伊哥和数字游戏.....	(5)
1.2 实际计算机的基本知识.....	(8)
1.3 基本结构.....	(10)
1.4 公用环境.....	(12)
1.5 一个程序例题.....	(13)
第 2 章 数据.....	(18)
2.1 二进制位和数.....	(18)
2.2 整型变量.....	(25)
2.3 浮点变量.....	(28)
2.4 常量.....	(29)
2.5 八进制和十六进制数.....	(31)
2.6 C 程序的基本元素.....	(33)
2.7 控制流.....	(35)
2.8 输出和打印.....	(38)
2.9 程序大小.....	(41)
2.10 定义和包含.....	(42)

第3章 运算符..... (44)

3.1	算术运算符.....	(44)
3.2	关系运算符.....	(48)
3.3	逻辑运算符.....	(51)
3.4	字符的输入/输出.....	(54)
3.5	字符类型测试.....	(58)
3.6	按位逻辑运算符.....	(60)
3.7	移位运算符.....	(62)
3.8	函数.....	(64)
3.9	左值, 右值, 增1, 减1.....	(71)
3.10	赋值运算符.....	(73)
3.11	运算符的嵌套.....	(74)
3.12	寻址运算符和格式化输入函数.....	(75)
3.13	条件运算符.....	(77)
3.14	数组和下标.....	(79)
3.15	逗号运算符.....	(85)
3.16	求值顺序.....	(87)
3.17	浮点计算.....	(89)
3.18	优先级和结合性.....	(92)
3.19	转换.....	(96)
3.20	溢出.....	(100)
3.21	定义类型和定义常量.....	(101)
3.22	关于输入/输出的进一步论述.....	(104)
3.23	程序计时.....	(109)

第4章 语句和控制流.....(111)

4.1	语句和分程序.....	(111)
-----	-------------	-------

4.2	If	(113)
4.3	If-else	(115)
4.4	Else-If	(117)
4.5	Switch	(119)
4.6	While	(121)
4.7	For	(124)
4.8	Do While	(126)
4.9	控制结构的设计	(127)
4.10	Break和Continue	(133)
4.11	Goto	(135)

第5章 函数 (138)

5.1	语法规则和易读形式	(138)
5.2	变元传递	(141)
5.3	参数和自动变量	(144)
5.4	数组变元	(147)
5.5	递归函数	(151)
5.6	自动纯量的初始化	(154)
5.7	存贮类和内部静态变量	(156)
5.8	分别编译和连接	(158)
5.9	外部静态存贮	(161)
5.10	静态存贮数组的初始化	(165)
5.11	二维数组	(167)
5.12	外部变量.....	(172)
5.13	寄存器存贮类	(174)
5.14	作用域规则	(176)
5.15	初始化小结	(177)
5.16	空括号：三种情况	(178)
5.17	带参数的宏定义	(179)

5.18	条件编译	(180)
第 6 章	软件开发	(183)
6.1	软件开发生命周期	(183)
6.2	分析	(184)
6.3	设计	(189)
6.4	实现: 编写程序	(196)
6.5	实现: 后面阶段	(208)
6.6	维护	(212)
第 7 章	指针	(214)
7.1	基础	(214)
7.2	说明和使用指针	(215)
7.3	指针用作函数参数	(218)
7.4	指针和数组	(219)
7.5	使用指针的函数	(222)
7.6	指针的算术运算	(224)
7.7	指针数组	(225)
7.8	命令行变元	(227)
第 8 章	结构	(230)
8.1	基础	(230)
8.2	成员	(232)
8.3	初始化	(232)
8.4	嵌套的结构	(233)
8.5	结构数组	(234)
8.6	结构指针	(235)

结束语.....	(238)
附录.....	(236)
参考文献.....	(346)
索引.....	(348)

第0章 引言

“计算机”：被“时代”杂志推选为1982年的“风云人物”。（实际上是这一年的“风云机器”）。

0.1 引言

“计算机”对很多人来说是一个可以触知的硬件，这对C语言的爱好者来说具有特殊的意义。直至1976年，人们还曾经预言计算将变得愈来愈集中，通过分散在各处的终端对巨型计算中心进行存取。然而微型机和小型机的成本不断降低已造成完全不同的局面。计算机用户通常知道他们所用的机器的内存大小和处理器速度，同时计算机作为**机器**（而不再是作为**遥远的抽象物**）已出现在多数的办公室以及许多家庭中。这种情况对于象C那样的语言是有利的，因为它以一种优雅的方法把现代计算机许多共同特征具体化了。

的确，受过我们C语言培训的许多人不是职业程序员，而是工程师，他们来学习C时对电路和处理器已有丰富的知识。对他们来说，C程序设计成为他们的专用工具包中又一个重要的工具。

对C感兴趣的另一类人是系统程序员，他们编写为其他程序员使用的程序—值得注意的是编译程序和操作系统。这些我们将在下一节讨论。在这样的工作中，效率—使程序小巧快速是主要的考虑因素。通常还要考虑一个能使程序在各种不同机器上运行

的可移植性问题。

还有一些编写应用程序包的程序员使计算机满足用户的特殊需要。这里同样由于 C 程序的效率和可移植性，使它成为软件生产者的理想工具。

所有这些 C 用户都分享了这种接近计算机特征的小型语言所起作用的好处。C 从一开始就有这些优点。它是从 B 语言演变而来的，B 是 K. Thompson 于 1970 年对 M. Richards 的 BCPL 的改编。UNIX 操作系统最初是用 PDP-11 的汇编语言编写的，在 1972 年创建 C 以后不久，它就用 C 进行改写。这种改写得到完全的肯定，在很短时间内，C 就在 UNIX 环境中取代了汇编程序设计。（关于 C 的历史的详细阐述，请看 Ritchie 等著的书 [1978]。）

C 的大部分历史是和 UNIX 操作系统紧密地联系在一起的，UNIX 是它最初的宿主和第一个主要的应用。当前对 C 感兴趣的大多数人是由于 UNIX 的广泛流行。这种联系主要是历史上的而不是固有的，最近的倾向已促成了 C 语言的独立地位。不少非 UNIX 环境的 C 编译程序已由数家独立的卖主，如有名的 White-Smiths 公司和贝尔实验室生产出来。的确，80 年代初期已经看到 C 的微处理器版本的广泛流行，很多业余爱好者也对该语言产生了兴趣。

为了建议一个再三考虑的修辞，我们可以把 C 叫做现代计算机的“可移植的汇编程序”。“可移植的”的意思是指同一个程序能在各种不同的计算机上运行；“汇编程序”的意思是指这些程序小巧快速，能对机器中的数据进行完全控制。因此，判断 C 的应用的适合性的一个简单准则就是对这两个问题的肯定回答：“汇编语言正在被考虑使用吗？”和“该程序能在不同的硬件上运行吗？”

给出 C 的各种用途之后，我们不打算对程序设计给出一本象“菜谱”那样的书。我们的主要目的是使你清楚地了解该语言及其在计算机中的主要特性。所选择的程序实例和练习也不是为了作为有用的软件（虽然其中有一些将是有用的），而是为了作为

C 语言特征的说明。本书的内容、阐述和编排在多年来对专业人员讲授 C 语言的过程中得到不断改进。很多受我们培训的人的第一个程序设计课题是用 C 语言编程序，所以本书既包括程序设计的基本知识，又讲授 C 语言的细节。因为书中的任何一个句子可能要在后面独立引用，所以我们将对涉及的内容讲解得尽可能透彻而不留有尾巴，然而某些题目（如函数的分别编译）将完全推迟，直到适当的地方再行讲解。

因此，本书的结构如下。第 1 章（“计算机和 C”）叙述了现代计算机的特点。用一种循序渐进地介绍的方法，只要求读者稍稍懂得一点计算机。有经验的计算机用户完全可以跳过这一章。

第 2 章（“数据”）对二进制位、二进制、八进制、十六进制和字符代码进行了介绍（或复习），并详细讨论了纯量变量的说明（本书中未包括新近编译程序中具有的枚举类型 enum）。本章还定义了程序的基本元素。然后对赋值、控制流、输出、程序大小、预定义（#define）和包含（#include）进行了介绍，为实际程序提供了例子的基础。

第 3 章（“运算符”）是最长的章，包括了 C 语言的大多数运算符。首先讲解字符输入/输出，以便编写交互式程序。优先级和结合性的叙述贯穿全章，并在最后做了总结。用于格式化输入函数 scanf 的寻址运算符的介绍为后面讨论数组和指针打下了基础。本章还提出了定义可移植类型的模式，最后以讨论输入和输出的环境信息而结束（打开和关闭命名文件、以及处理它们的许多库函数都没有包括在本书之中）。

第 4 章（“语句和控制流”）详细地叙述了全部 C 控制结构，描述了它们的语法和可读形式。本章还介绍了控制结构的设计，为有关可靠程序的精细设计提供了设施。

第 5 章（“函数”）描述了 C 函数的全部语法、分别编译、连接、以及扩充的语法检查器 lint。二维数组推迟到这一章来介绍，这样可以描述它们的初始化问题。

第6章（“软件开发”）讨论了分析、设计、实现和维护。并详细地叙述了一个实际的问题实例“Blackjack”（二十一点）。

第7章（“指针”）包括C中有关指针的基本内容。其高级的应用—动态存储分配、连接表和指向函数的指针—没有包括在这本入门书中。

第8章（“结构”）包括C语言的结构 struct。但不包括Union、自引用结构和位域。

附录A总结了语法的可读规则和最通用的C程序库，成为C的一个“袖珍指南”。附录A还包含 printf 和 scanf 的格式、C的常见错误、C的“习惯用语”和ASCII代码。

附录B给出了正文中全部检查性思考题的答案、程序清单以及与环境有关的注意事项。为了使编译程序的卖主能以简洁的结构描述他们的编译程序的特征以及和其他编译程序的区别，我们仔细地选择了附录B的结构。

放在首位的是基本内容。