



老虎工作室
www.laohu.net



从 开始



C语言程序设计 基础培训教程

老虎工作室

孙海侠 赖积滨 姜继红 编著



人民邮电出版社
POSTS & TELECOM PRESS

从零开始

—— C 语言程序设计基础培训教程

孙海侠
老虎工作室 赖积滨 编著
姜继红



人民邮电出版社

图书在版编目 (CIP) 数据

从零开始——C 语言程序设计基础培训教程/孙海侠, 赖积滨, 姜继红编著.

—北京: 人民邮电出版社, 2004.7

ISBN 7-115-12341-1

I. 从... II. ①孙...②赖...③姜... III. C 语言—程序设计—技术

培训—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2004) 第 058295 号

内 容 提 要

本书详细介绍了 C 语言的数据类型、变量、各种基本语句和函数的功能、数组、指针、结构体、位运算、文件操作以及程序设计的方法, 是学习 C 语言的入门教程。

本书通俗易懂, 充分考虑了初学者的实际需要, 真正“从零开始”, 可以使对 C 语言一点都不懂的读者, 通过学习本书而掌握 C 语言的应用方法和技巧。本书每一章的最后都有相应的练习题, 可帮助读者检验学习效果。

为了方便读者学习, 书中用到的程序代码已经上载到老虎工作室网站 (<http://www.laohu.net>), 读者可以根据需要下载使用。

本书既可以作为各类专业技术人员学习 C 语言程序设计的培训教材, 也可作为相关专业大、中专及本科生的教学参考书。

从零开始——C 语言程序设计基础培训教程

◆ 编 著 老虎工作室 孙海侠 赖积滨 姜继红

责任编辑 李永涛

执行编辑 徐宝姝

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

邮编 100061 电子函件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

北京隆昌伟业印刷有限公司印刷

新华书店总店北京发行所经销

◆ 开本: 787×1092 1/16

印张: 20

字数: 482 千字

印数: 13 001—15 000 册

2004 年 7 月第 1 版

2006 年 2 月北京第 5 次印刷

ISBN 7-115-12341-1/TP · 4019

定价: 28.00 元

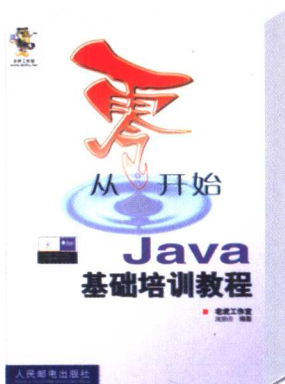
读者服务热线: (010) 67132692 印装质量热线: (010) 67129223



书号: 10944
定价: 28元



书号: 10941
定价: 28元



书号: 11229
定价: 28元



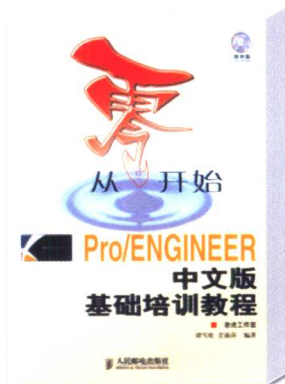
书号: 11267
定价: 36元



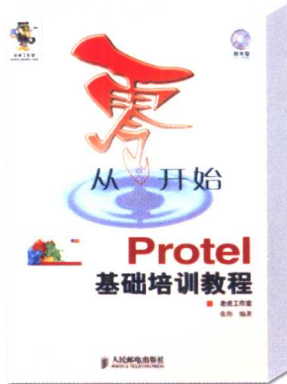
书号: 11730
定价: 24元



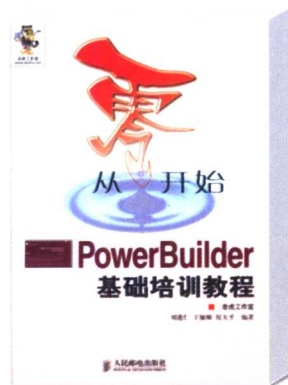
书号: 11747
定价: 28元



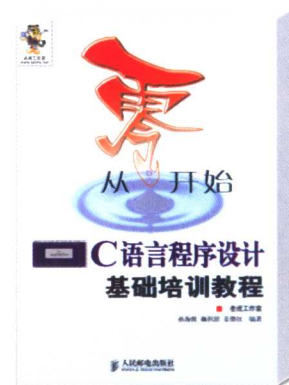
书号: 11733
定价: 36元



书号: 11725
定价: 30元



书号: 11735
定价: 28元



书号: 12341
定价: 28元



老虎工作室

主 编：沈精虎

编 委：许曰滨 黄业清 杜俭业 姜 勇 宋一兵
向先波 谭雪松 孙海侠 冯 辉 蔡汉明
高长铎 詹 翔 宋雪岩 郭万军 周 锦
田博文 张 琴 赖积滨 姜继红 李 仲

内容和特点

C 语言是被广泛使用的计算机高级语言，它既可以用来编写系统程序，也可以用来编写不同领域的应用程序。C 语言编译系统几乎可以在所有的操作系统和各种类型的机器上运行，从个人计算机到大型计算机，有相当大比例的软件是用 C 语言编写的。

C 语言具有数据类型和运算符类型丰富、结构灵活、代码质量高、可移植性强、编程风格自由、兼有汇编语言和高级语言的优势等特点，深受广大程序设计爱好者的喜爱，也是一般相关专业的大、中专和本科学生必修的程序设计语言。有了 C 语言程序设计基础，对进一步学习 C++ 面向对象编程语言会有很大的帮助。

本书在内容的选取及章节的设置上充分考虑了初学者的实际需要，真正“从零开始”。通过对本书的深入学习，读者基本上能够掌握 C 语言的使用，并能够联系实际，编写出有价值的应用程序。本书简明清晰、概念清楚、逻辑性强，在叙述上做到了深入浅出、通俗易懂。

全书包括 14 章和 1 个附录，各章内容简介如下。

- 第 1 章：介绍 C 语言的起源、发展、特点、基本结构以及 C 程序的上机步骤。
- 第 2 章：介绍常量和变量的定义和使用。
- 第 3 章：介绍运算符的功能、表达式的构成以及语句的作用。
- 第 4 章：介绍字符输入输出函数和格式化输入输出函数。
- 第 5 章：介绍数据类型转换和顺序结构程序的设计。
- 第 6 章：介绍关系、逻辑运算符的功能和选择结构程序的设计。
- 第 7 章：介绍循环语句的使用和循环结构程序的设计。
- 第 8 章：介绍数组的概念、定义和应用。
- 第 9 章：介绍指针的概念、运算和应用。
- 第 10 章：介绍函数的概念、定义和应用。
- 第 11 章：介绍预处理的概念和使用方法。
- 第 12 章：介绍结构体和共用体的概念、定义和应用。
- 第 13 章：介绍位运算符的功能和应用。
- 第 14 章：介绍文件的概念和使用方法。
- 附录：介绍了关于 C 语言的一些常用资料。

书中每章的最后都附有练习题，读者可以通过这些练习检验学习效果。

读者对象

本书既可以作为各类专业技术人员学习 C 语言程序设计的培训教材，也可作为相关专业大、中专及本科生的教学参考书。

关于本书实例的源代码

本书所有操作实例的源代码都存放在老虎工作室网站 (<http://www.laohu.net>) 上，需要的读者可以直接下载使用。如果读者朋友在学习过程中遇到困难或有其他技术问题需要和作者讨论，可以登录“老虎论坛” (<http://www.laohu.net/bbs/INDEX.ASP>)。

感谢您选择了本书，也请您把对本书的意见和建议告诉我们。

老虎工作室网站 <http://www.laohu.net>，电子函件地址 postmaster@laohu.net。

老虎工作室

2004 年 6 月

第 1 章 C 语言概述.....	1
1.1 C 语言的诞生与发展.....	1
1.2 C 语言的特点.....	2
1.3 C 程序的基本结构.....	3
1.4 建立与运行 C 程序的过程.....	8
1.5 用 Turbo C 运行 C 程序的上机步骤.....	9
1.6 小结.....	13
1.7 习题.....	14
第 2 章 数据类型.....	15
2.1 常量.....	15
2.1.1 字面常量.....	15
2.1.2 符号常量.....	18
2.2 变量.....	19
2.2.1 变量的定义.....	19
2.2.2 变量的初始化.....	24
2.3 小结.....	24
2.4 习题.....	24
第 3 章 运算符、表达式和语句.....	27
3.1 运算符和表达式.....	27
3.1.1 赋值运算符和赋值表达式.....	28
3.1.2 算术运算符和算术表达式.....	29
3.1.3 逗号运算符和逗号表达式.....	34
3.1.4 求字节数运算符和负值运算符.....	35
3.2 语句.....	36
3.2.1 空白对语句的影响.....	36
3.2.2 表达式语句.....	36
3.2.3 复合语句.....	37
3.3 小结.....	38

3.4 习题	38
第 4 章 数据的输入与输出	41
4.1 格式输出函数 printf()和格式输入函数 scanf()	41
4.1.1 printf()函数	41
4.1.2 scanf()函数	48
4.2 字符输出函数 putchar()和字符输入函数 getchar()	52
4.2.1 putchar()函数	52
4.2.2 getchar()函数	53
4.3 小结	54
4.4 习题	54
第 5 章 顺序结构程序设计	57
5.1 程序设计	57
5.1.1 算法	58
5.1.2 结构化程序设计	60
5.2 类型转换	63
5.2.1 自动类型转换	63
5.2.2 强制类型转换	66
5.3 顺序结构程序举例	68
5.4 小结	70
5.5 习题	71
第 6 章 选择程序结构设计	73
6.1 关系运算符和关系表达式	73
6.2 逻辑运算符和逻辑表达式	75
6.3 if 语句	77
6.3.1 if 语句的 3 种形式	77
6.3.2 选择语句的嵌套	82
6.3.3 条件运算符和条件表达式	85
6.4 switch 语句	87
6.5 程序举例	91
6.6 小结	94
6.7 习题	94
第 7 章 循环结构程序设计	97
7.1 while 语句	97

7.2	for 语句	100
7.3	do-while 语句	104
7.4	循环的嵌套	106
7.5	循环的辅助语句	110
7.6	goto 语句	114
7.7	两种常用的算法	116
7.8	程序举例	118
7.9	小结	120
7.10	习题	121
第 8 章	数组	123
8.1	数组的定义和引用	123
8.1.1	数组的定义	123
8.1.2	数组的引用	124
8.2	数组元素在内存中的存储方法	126
8.3	数组的初始化	127
8.3.1	一维数组的初始化	127
8.3.2	二维数组的初始化	128
8.4	数组应用举例	129
8.5	字符数组	135
8.5.1	字符数组的定义和初始化	135
8.5.2	字符数组的引用	136
8.5.3	字符串	137
8.5.4	字符数组的输入和输出	138
8.5.5	字符串处理函数	140
8.5.6	应用举例	142
8.6	小结	144
8.7	习题	144
第 9 章	指针	147
9.1	地址和指针	147
9.2	指针的运算	148
9.2.1	指针的定义	148
9.2.2	使指针指向对象的方法	149
9.2.3	取地址运算符和访问地址运算符	150
9.2.4	指针的引用	151
9.2.5	指针的运算	152
9.3	指针和数组	154

9.3.1 指针与一维数组.....	154
9.3.2 指针与二维数组.....	156
9.4 指针和字符串.....	160
9.5 指针数组和二级指针.....	164
9.5.1 指针数组.....	164
9.5.2 二级指针.....	166
9.6 小结.....	168
9.7 习题.....	168
第 10 章 函数	171
10.1 概述.....	171
10.2 函数的定义.....	172
10.3 函数的调用.....	173
10.3.1 函数调用的定义和声明.....	173
10.3.2 函数的返回.....	177
10.4 变量作用域.....	179
10.4.1 局部变量.....	179
10.4.2 全局变量.....	180
10.5 变量的存储类型.....	182
10.6 函数间数据传递方式.....	190
10.6.1 函数间的参数传递.....	190
10.6.2 函数返回值.....	195
10.6.3 全局变量.....	197
10.7 函数的嵌套调用.....	198
10.8 函数的递归调用.....	200
10.9 函数指针和函数指针数组.....	204
10.9.1 函数指针.....	204
10.9.2 函数指针数组.....	208
10.10 main()函数的参数和返回值.....	209
10.11 内部函数和外部函数.....	211
10.12 小结.....	212
10.13 习题.....	212
第 11 章 预处理	215
11.1 宏定义.....	215
11.1.1 不带参数的宏定义.....	215
11.1.2 带参数的宏定义.....	218
11.2 文件包含.....	220

11.3	条件编译.....	222
11.4	小结.....	226
11.5	习题.....	227
第 12 章 结构体与共用体.....		229
12.1	结构体类型的声明.....	229
12.2	结构体类型的变量.....	230
12.2.1	结构体类型变量的定义.....	230
12.2.2	结构体类型变量的初始化	233
12.2.3	结构体类型变量的引用.....	234
12.3	结构体数组.....	235
12.4	指向结构体变量和结构体数组的指针	239
12.5	结构体与函数.....	242
12.5.1	结构体在函数间的传递.....	242
12.5.2	结构体类型函数或结构指针型函数	246
12.6	链表.....	248
12.6.1	链表的基本知识.....	248
12.6.2	动态存储分配函数.....	249
12.6.3	链表的操作.....	250
12.7	共用体.....	257
12.7.1	共用体类型的声明.....	257
12.7.2	共用体类型的变量.....	258
12.7.3	程序举例.....	260
12.8	枚举类型.....	262
12.9	用 typedef 定义类型.....	265
12.10	小结.....	267
12.11	习题.....	268
第 13 章 位运算.....		271
13.1	二进制数、位和字节.....	271
13.2	位运算符和位运算.....	272
13.3	程序举例.....	276
13.4	位段.....	277
13.5	小结.....	279
13.6	习题.....	279

第 14 章 文件	281
14.1 文件概述	281
14.2 文件类型指针	282
14.3 文件的打开与关闭	283
14.3.1 文件的打开	283
14.3.2 文件的关闭	285
14.4 文件读写函数	285
14.4.1 字符读写函数	286
14.4.2 字符串读写函数	288
14.4.3 格式化读写函数	290
14.4.4 数据块读写函数	290
14.5 文件检测函数	292
14.6 文件的定位	294
14.7 文件管理函数	297
14.8 小结	298
14.9 习题	298
附录 常用资料	301
F.1 ASCII 码	301
F.2 C 运算的优先级和结合性	302
F.3 常用的 C 语言库函数	303

第1章 C 语言概述

C 语言是国际上广泛流行的计算机高级语言，既可以用来编写系统程序，也可以用来编写不同领域的应用程序。现在，C 语言编译系统几乎可以在所有的操作系统，以及各种类型的计算机上运行。从个人计算机到大型计算机，有相当大比例的软件是用 C 语言编写的。C 语言突出的优点备受计算机爱好者及广大学生的青睐。

1.1 C 语言的诞生与发展

在 C 语言诞生以前，系统软件主要是用汇编语言编写的，包括 UNIX 操作系统在内。由于汇编语言程序依赖于计算机硬件，其可读性和可移植性都很差，但一般的高级语言又难以实现对计算机硬件的直接操作，而这正是汇编语言的优势。人们盼望有一种兼有汇编语言和高级语言特性的新语言，于是，C 语言就在这种情况下应运而生了。

C 语言的祖先是 ALGOL 60 (ALGO^rithmic Language 60)。ALGOL 60 是 1960 年由国际计算机委员会设计的一种面向过程的高级语言。它是一种结构化语言，用它编写的程序具有可读性和可移植性好的特点。但是，它不能直接对硬件进行操作，不宜用来编写系统程序。

1963 年，英国的剑桥大学和伦敦大学首先将 ALGOL 60 发展成 CPL (Combined Programming Language) 语言。CPL 语言在 ALGOL 60 的基础上接近硬件一些，但规模比较大，难以实现。

1967 年，英国剑桥大学的 Martin Richards 将 CPL 改为 BCPL (Basic Combined Programming Language) 语言。BCPL 语言比 CPL 语言大为简化。

1970 年，美国贝尔实验室的 Ken Thompson 以 BCPL 语言为基础，又做了进一步简化，设计出了很简单且很接近硬件的 B 语言，并用 B 语言写了第一个 UNIX 操作系统，1971 年在 PDP-7 小型机上实现。但 B 语言过于简单，功能有限。

1972 年，美国贝尔实验室的 D.M.Ritchie 在 B 语言的基础上设计出了 C 语言。C 语言既保持了 BCPL 和 B 语言的精练和接近于硬件的特点，也克服了它们过于简单、数据无类型等缺点。1973 年，Ken Thompson 和 D.M.Ritchie 再次合作将 1969 年他们用汇编语言编写的 UNIX 操作系统改用 C 语言编写，即 UNIX 第 5 版。其中 C 语言代码占 90% 以上，只保留了少量汇编语言代码。这样使 UNIX 操作系统向其他类型的机器上移植变得相当简单。到了 20 世纪 70 年代中期，UNIX 操作系统和 C 语言作为软件设计师的良好工具传遍了贝尔实验室，接着也传遍了几乎所有的美国大学校园，UNIX 和 C 开始风靡世界。

UNIX 操作系统的广泛使用，促进了 C 语言的迅速发展及普及，C 语言的发展与普及也促进了 UNIX 操作系统的推广。1978 年，出现了独立于 UNIX 和 PDP 计算机的 C 语言，从而，C 语言被迅速移植到大、中、小型机及微型机上。同年，以 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《The C Programming Language》一书。这本书介绍的 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。随着微型计算机的日益普及, 出现了许多 C 语言版本。由于没有统一的标准, 使得这些 C 语言之间出现了一些不一致的地方。从 1983 年开始, 为了改变这种情况, 美国国家标准研究所 (ANSI) 根据 C 语言问世以来各种版本对 C 语言的发展和扩充, 经过长达 5 年的努力, 制定了 ANSI C 标准。1988 年, K&R 修改了他们的经典著作《The C Programming Language》, 按 ANSI C 标准重新编写该书。现在一般称 ANSI C 为新标准, 称 K&R C 为旧标准。本书以 ANSI C 新标准来介绍。

C++语言是 C 语言发展的新阶段, 是应用广泛的面向对象的程序设计语言。

目前, 在微机上广泛使用的 C 语言编译系统有 Microsoft C、Turbo C、Borland C 等。虽然它们的基本部分都是相同的, 但还是有一些差异, 所以请读者注意自己所使用的 C 编译系统的特点和规定 (参阅相应的手册)。Turbo C 2.0 完全支持新标准, 并提供集成开发操作环境。本书选定的上机环境是 Turbo C 2.0。

1.2 C 语言的特点

C 语言发展如此迅速, 而且成为最受欢迎的语言之一, 主要因为它具有强大的功能。许多著名的系统软件, 如 dBASE III PLUS、dBASE IV 都是由 C 语言编写的。用 C 语言加上一些汇编语言子程序, 就更能显示 C 语言的优势了, 如 PC-DOS3.0、WordStar 等就是用这种方法编写的。归纳起来 C 语言具有以下特点。

(1) C 语言同时具有汇编语言和高级语言的优势

C 语言把高级语言的基本结构和语句与低级语言的实用性结合起来。C 语言可以像汇编语言一样对位、字节和地址进行操作, 而这 3 者是计算机最基本的工作单元。

(2) 语言简洁、使用灵活, 便于学习和使用

C 语言在表示方法上力求明了易懂、简单易行。例如, C 语言的书写格式没有限制, C 语言提供了预处理功能以及大量而有效的函数库等。

(3) 运算符类型丰富

C 语言的运算符包含的范围很广泛, 从而使 C 语言的运算类型极其丰富, 表达式类型多样化。灵活使用各种运算符可以实现在其他高级语言中难以实现的运算。

(4) 数据类型丰富, 并有多种数据结构

C 语言具有各种各样的数据类型, C 语言的数据类型有整型、实型、字符型、数组类型、指针类型、结构体类型和共用体类型等, 能用来实现各种复杂数据结构的运算, 尤其引入了指针概念, 可使程序效率更高。

(5) 同汇编语言相比, 用 C 语言写的程序可移植性好

C 语言的一个突出的优点就是适合于在多种操作系统以及各种类型的计算机上运行, 因此移植起来很方便。

(6) 结构化程序设计语言

结构化程序设计语言的显著特点是代码和数据分隔化, 即程序的各个部分除了必要的信息交流外彼此独立。这种结构化设计的方式可使程序层次清晰, 便于使用、维护以及调



试。C 语言是以函数形式提供给用户的，这些函数可方便地调用，并具有多种循环语句、条件语句来控制程序的流向，从而使程序完全结构化。

(7) 生成的目标代码质量高，程序执行效率高

高级语言能否用来描述系统软件，特别是操作系统、编译程序等，除要求语言表达能力强之外，很重要的一个因素是语言生成的目标代码质量如何。许多实验表明，针对同一个问题，用 C 语言编写的程序，其目标代码的效率仅比汇编语言低 10%~20%。由于用高级语言描述问题比汇编语言编程迅速，工程量小，可读性和可移植性好，易于调试和修改，所以 C 语言成为描述系统软件和应用软件比较理想的工具。

(8) 程序设计较自由，语法限制不严格

对数组下标越界不做检查，对变量的类型使用比较灵活，C 语言允许程序编写者有较大的自由度，因此放宽了语法检查。

C 语言具有上述诸多优点，因而具有旺盛的生命力，但也有一些缺点和不足。如运算符的优先级太多，不便于记忆，类型转换比较随便，检测手段太弱等，这就要求程序员对程序设计更熟练、查错能力更强。但是，瑕不掩玉，多种程序设计的应用已经证明 C 语言是一种极为有效且表达能力很强的语言。

1.3 C 程序的基本结构

用 C 语言编写的程序，称为 C 语言源程序，简称为 C 程序。C 程序一般由一个或多个函数组成，这些函数可放在同一个文件中，也可分散放在几个文件中。

一、一个简单的例子

先看一个简单的例子，认识一下 C 程序的构成。

【例1-1】 一个计算梯形面积的 C 程序。

```
/*求梯形面积*/  
#include <stdio.h>           /*文件包含*/  
main()  
{  
    int  up,bot,high;         /*定义了上底、下底和高为整型变量*/  
    float area;               /*定义了面积为实型变量*/  
    up=3;                     /*上底等于 3*/  
    bot=5;                     /*下底等于 5*/  
    high=4;                    /*高等于 4*/  
    area=(up+bot)*high/2.0;    /*计算梯形面积 */  
    printf("area=%f\n",area); /*在屏幕上显示面积的值*/  
}
```

首先来看这个例子的构成。

- “/*”与“*/”之间的内容是注释。在本例中用了大量的注释，主要是提高程序的可读性，便于理解。在实际应用中，一个程序可以在一些难理解的地方适



当加上一些注释,对于程序的维护、理解有很大的帮助。注释不影响程序的执行,也不被编译。

- “#include”命令将使该程序把头文件“stdio.h”包含进来。
- main()叫做C程序的主函数。在1个完整的C程序中一定要有1个主函数main(),C程序总是从它开始执行。
- 一对花括号“{}”,是函数体的界限符。同时也是复合语句的界限符。
- “int up,bot,high;”和“float area;”是变量说明语句,用来通知计算机程序用到4个变量,并为它们分配存储单元。
- “up=3;”、“bot=5;”和“high=4;”是赋值语句,表示梯形的上底为3、下底为5和高为4。其中,“=”是赋值运算符,“3”、“4”和“5”是常量。
- “area=(up+bot)*high/2;”也是赋值语句,用来计算梯形的面积。其中,“+”、“*”和“/”是运算符,“()”是带有特定意义的运算符。
- “printf(“area=%f\n”,area);”是格式输出语句,“printf”是系统预定义的标准标识符,格式输出语句用来在屏幕上显示梯形的面积。
- 分号“;”表示语句结束。C程序中要求每个语句后面要加分号。语句是C程序的执行单位。
- 程序中的“main”、“int”、“float”是系统规定的关键字,“up”、“bot”、“high”、“area”是用户自己定义的名字,在C语言中它们属于标识符。

二、C程序的一般形式

C程序的一般形式为:

```
#include <文件名>           /*文件包含*/
...
#include "文件名"
...
#define 宏名 宏体           /*宏定义*/
...
全局变量说明语句
main()
{
    局部变量说明语句
    执行语句
}
存储类型 数据类型 函数名1(参数表)
{
    局部变量说明语句
    执行语句
}
...
```