

【美】Stephen G. Kochan 著

贾洪峰 译

PEARSON

Broadview®
www.broadview.com.cn

Programming in C
(4th Edition)

C语言程序设计

(第4版)



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

Programming in C
(4th Edition)

C语言程序设计

(第4版)

【美】 Stephen G. Kochan 著
贾洪峰 译

电子工业出版社
Publishing House of Electronics Industry

内 容 简 介

本书全面介绍了 C 语言的各种特性,包括 C11 中增加的内容。书中包含大量完整的示例及详细的讲解。无论是否拥有编程经验,你都可以通过本书透彻地理解 C 语言。每章最后附有习题,便于课堂学习或自学。

Authorized translation from the English language edition, entitled Programming in C, 4E, 978-0321776419 by Stephen G. Kochan, published by PEARSON EDUCATION, INC., publishing as Addison-Wesley Professional, Copyright © 2014 Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

CHINESE SIMPLIFIED language edition published by PEARSON EDUCATION ASIA LTD., and PUBLISHING HOUSE OF ELECTRONICS INDUSTRY Copyright © 2015.

本书简体中文版专有出版权由 Pearson Education 培生教育出版亚洲有限公司授予电子工业出版社。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

本书简体中文版贴有 Pearson Education 培生教育出版集团激光防伪标签,无标签者不得销售。

版权贸易合同登记号:图字:01-2014-0174

图书在版编目(CIP)数据

C 语言程序设计:第 4 版 / (美)寇肯(Kochan,S.G.)著;贾洪峰译. —北京:电子工业出版社,2015.5

书名原文:Programming in C (4th Edition)

ISBN 978-7-121-25888-6

I. ①C… II. ①寇… ②贾… III. ①C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 074580 号

策划编辑:张春雨

责任编辑:张春雨

印 刷:三河市双峰印刷装订有限公司

装 订:三河市双峰印刷装订有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编:100036

开 本:720×1000 1/16 印张:34.5 字数:607.2 千字

版 次:2015 年 5 月第 1 版

印 次:2015 年 5 月第 1 次印刷

定 价:108.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

关于作者

Stephen G. Kochan 使用 C 语言设计软件已经有超过 30 年的时间。他是几本最畅销的 C 语言书籍的作者，包括 *Programming in C*、*Programming in Objective-C* 和 *Topics in C Programming*。他在 Unix 方面也有大量著述，是 *Exploring the Unix System* 和 *Unix Shell Programming* 等书的著者或合著者。

有贡献的作者，第 4 版

Dean Miller 是一位作家和编辑，在出版和特许消费者产品领域都拥有超过 20 年的经验。他是 *Sams Teach Yourself C in One Hour a Day* 和 *Sams Teach Yourself Beginning Programming in 24 Hours* 最近几个版本的合著者。

致谢

衷心地感谢下列各位在我准备本书各个版本时提供的帮助：Douglas McCormick、Jim Scharf、Henry Tabickman、Dick Fritz、Steve Levy、Tony Ianinno 和 Ken Brown。还要感谢纽约大学的 Henry Mullish 在写作方面给我的诸多教益，感谢他带我走进出版领域。

在 Pearson 的工作人员中，我要感谢 Mark Taber 和我的责任编辑 Mandie Frank，还要感谢我的文字编辑 Charlotte Kughen 和我的技术编辑 Siddhartha Singh。最后，在 Pearson 还有许多参与本项目但没有与我直接接触的工作人员，在此向他们所有人表示感谢。

速览

前言	1
1 基础知识	5
2 编译和运行第一个程序	13
3 变量、数据类型和算术表达式	23
4 程序循环	45
5 做出决策	69
6 使用数组	101
7 使用函数	127
8 使用结构体	175
9 字符串	205
10 指针	247
11 位运算	295
12 预处理器	317
13 用枚举数据类型、类型定义和数据类型转换扩展数据类型	341
14 处理大型程序	353
15 C 语言中的输入与输出操作	369
16 其他及高级特性	397
17 调试程序	415

18 面向对象程序设计	437
附录 A C 语言概要	453
附录 B 标准 C 语言库	497
附录 C 用 gcc 编译程序	525
附录 D 常见编程错误	527
附录 E 资源	533

目录

前言	1
1 基础知识	5
程序设计	5
高级语言	6
操作系统	7
编译程序	7
集成开发环境	10
语言解释器	11
2 编译和运行第一个程序	13
编译程序	14
运行程序	14
理解第一个程序	15
显示变量的值	17
注释	19
练习	21
3 变量、数据类型和算术表达式	23
理解数据类型和常量	23
整数类型 int	24
浮点数值类型 float	25
扩展精度类型 double	26
单字符类型 char	26
布尔数据类型 _Bool	27

类型说明符：long、long long、short、unsigned 和 signed...	28
使用变量	31
使用算术表达式	33
整数算术及一元减运算符	36
运算与赋值相结合：赋值运算符	42
_complex 和_imaginary 类型	43
练习	43
4 程序循环	45
三角数	45
for 语句	46
关系运算符	48
输出的对齐	52
程序输入	53
嵌套 for 循环	55
for 循环的变体	57
while 语句	59
do 语句	63
break 语句	65
continue 语句	65
练习	66
5 做出决策	69
if 语句	69
if-else 构造	73
复合关系测试	76
嵌套 if 语句	79
else if 构造	81
switch 语句	88
布尔变量	91

条件运算符	96
练习	97
6 使用数组	101
定义数组	102
以数组元素为计数器	106
生成斐波那契数	109
用数组生成质数	111
初始化数组	113
字符数组	115
使用数组的基数变换	116
const 限定符	118
多维数组	120
变长数组	122
练习	124
7 使用函数	127
定义函数	127
参数和局部变量	131
函数原型声明	132
自动局部变量	132
返回函数结果	134
函数调用	139
声明返回类型和参数类型	142
检查函数参数	144
自上而下的程序设计	146
函数与数组	146
赋值运算符	151
数组排序	152
多维数组	155

全局变量	161
自动变量与静态变量	165
递归函数	168
练习	171
8 使用结构体	175
结构体的基础知识	175
用于存储日期的结构体	176
在表达式中使用结构体	178
函数与结构体	181
用于存储时间的结构体	187
初始化结构体	190
复合字面量	191
结构体的数组	192
包含结构体的结构体	196
包含数组的结构体	198
结构体变体	202
练习	203
9 字符串	205
复习字符串的基础知识	205
字符数组	206
变长字符串	209
字符串的初始化与显示	211
判断两个字符串是否相等	214
输入字符串	216
单字符输入	219
空字符串	224
转义字符	228
再论常量字符串	230

字符串、结构体和数组	231
一种更好的搜索方法	235
字符操作	240
练习	243
10 指针	247
指针和间接性	247
定义指针变量	248
在表达式中使用指针	252
使用指针和结构体	253
包含指针的结构体	256
链表	258
关键字 const 和指针	266
指针和函数	267
指针和数组	273
稍许离题：程序优化	277
是数组还是指针？	278
指向字符串的指针	280
常量字符串和指针	282
递增和递减运算符回顾	283
指针操作	287
指向函数的指针	288
指针和内存地址	289
练习	291
11 位运算	295
位的基础知识	295
位运算符	297
按位与运算符	297
按位或运算符	300

按位异或运算符	301
二进制求补运算符	301
左移运算符	304
右移运算符	304
移位函数	305
循环移位	307
位段	309
练习	314
12 预处理器	317
#define 语句	317
程序可扩展性	322
程序可移植性	323
更高级的定义类型	324
#运算符	330
##运算符	331
#include 语句	332
系统包含文件	335
条件编译	335
#ifdef、#endif、#else 和#ifndef 语句	336
#if 和#elif 预处理器语句	338
#undef 语句	339
练习	339
13 用枚举数据类型、类型定义和数据类型转换扩展数据类型	341
枚举数据类型	341
typedef 语句	345
数据类型转换	348
符号扩展	350
参数转换	350

练习	351
14 处理大型程序	353
将程序分为多个文件	353
由命令行编译多个源文件	354
模块之间的通信	356
外部变量	357
静态、动态变量与函数	359
有效使用头文件	362
其他处理大型程序的实用工具	364
make 实用工具	364
cvs 实用工具	366
Unix 实用工具: ar、grep、sed 等	366
15 C 语言中的输入与输出操作	369
字符 I/O: getchar() 和 putchar()	369
格式化 I/O: printf() 和 scanf()	370
printf() 函数	370
scanf() 函数	377
文件输入与输出操作	382
将 I/O 重定向到文件	382
文件尾	385
处理文件的特殊函数	386
fopen 函数	387
getc() 和 putc() 函数	389
fclose() 函数	389
feof 函数	391
fprintf() 和 fscanf() 函数	392
fgets() 和 fputs() 函数	392
stdin、stdout 和 stderr	393

exit() 函数.....	394
重命名和删除文件.....	395
练习.....	396
16 其他及高级特性.....	397
其他语言语句.....	397
goto 语句.....	397
空语句.....	398
使用联合体.....	399
逗号运算符.....	402
类型限定符.....	403
register 限定符.....	403
volatile 限定符.....	403
restrict 限定符.....	404
命令行参数.....	404
动态内存分配.....	408
calloc() 和 malloc() 函数.....	409
sizeof 运算符.....	410
free 函数.....	413
练习.....	414
17 调试程序.....	415
用预处理器调试.....	415
用 gdb 调试程序.....	422
处理变量.....	425
源文件显示.....	426
控制程序的执行.....	427
获得栈轨迹.....	432
调试函数, 设置数组和结构体.....	432
用 gdb 命令获取帮助.....	433

杂项	435
18 面向对象程序设计	437
到底什么是对象?	437
实例与方法	438
编写一个处理分数的 C 程序	440
定义一个处理分数的 Objective-C 类	441
定义一个处理分数的 C++ 类	446
定义一个处理分数的 C# 类	448
附录 A C 语言概要	453
1.0 双字符组和标识符	453
2.0 注释	454
3.0 常量	455
4.0 数据类型和声明	458
5.0 表达式	469
6.0 存储类和作用域	482
7.0 函数	483
8.0 语句	486
9.0 预处理器	490
附录 B 标准 C 语言库	497
标准头文件	497
字符串函数	500
内存函数	502
字符函数	503
I/O 函数	503
内存中的格式转换函数	509
字符串转换为数字	510
动态内存分配函数	512
数学函数	513

复数运算	519
通用函数	521
附录 C 用 gcc 编译程序	525
通用命令格式	525
命令行选项	526
附录 D 常见编程错误	527
附录 E 资源	533
C 程序设计语言	533
C 编译器和集成开发环境	534
其他	535

前 言

C 程序设计语言是由 AT&T 贝尔实验室的 Dennis Ritchie 在 20 世纪 70 年代设计开发的。但一直到 20 世纪 70 年代晚期，这一程序设计语言才开始得到广泛普及和支持。这是因为，在此之前的 C 编译器无法轻松应用于贝尔实验室之外的商业用途。最初，C 语言普及程度得以提高的部分原因是 Unix 操作系统以相同甚至更快的速度得到普及。这一操作系统也是在贝尔实验室开发的，它以 C 语言作为“标准”程序设计语言。事实上，这个操作系统本身就有超过 90% 的部分是用 C 语言编写的！

IBM PC 及类似计算机的巨大成功很快使 MS-DOS 成为最流行的 C 语言环境。随着 C 语言在不同操作系统上变得日益普及，越来越多的厂商加入了这一潮流，开始推销自己的 C 编译器。其中绝大多数的 C 语言版本都基于第一本 C 语言程序设计教科书中的一个附录，这本书就是 Brian Kernighan 和 Dennis Ritchie 的《C 程序设计语言》(*The C Programming Language*)。遗憾的是，这个附录中并没有给出一个完整的、无歧义的 C 语言定义，也就是说，这门语言的某些方面要留待供应商自行解读。

20 世纪 80 年代早期，出现了对 C 语言的定义进行标准化的需求。美国国家标准委员会 (ANSI) 作为处理此类事务的一家组织，其在 1983 年因此成立了 ANSI C 委员会 (称为 X3J11)，负责实现 C 语言的标准化。1989 年，该委员会的工作获得批准。1990 年，C 语言的第一个正式 ANSI 标准定义发布。

因为 C 语言在世界各地都有应用，所以国际标准组织 (ISO) 很快参与了进来。他们采用了上述标准，称之为 ISO/IEC 989:1990。在此之后，又对 C 语言进行了其他修改。最近的标准是 2011 年采用的，称为 ANSI C11 或 ISO/IEC 9899:2011。本书便基于 C 语言的这一版本。