

计 算 机 科 学 丛 书

原书第5版

C语言参考手册

(美) Samuel P. Harbison III Guy L. Steele Jr. 著 邱仲潘 等译

A Reference Manual
FIFTH EDITION



Samuel P. Harbison III • Guy L. Steele Jr.

C: A Reference Manual
Fifth Edition



机械工业出版社
China Machine Press

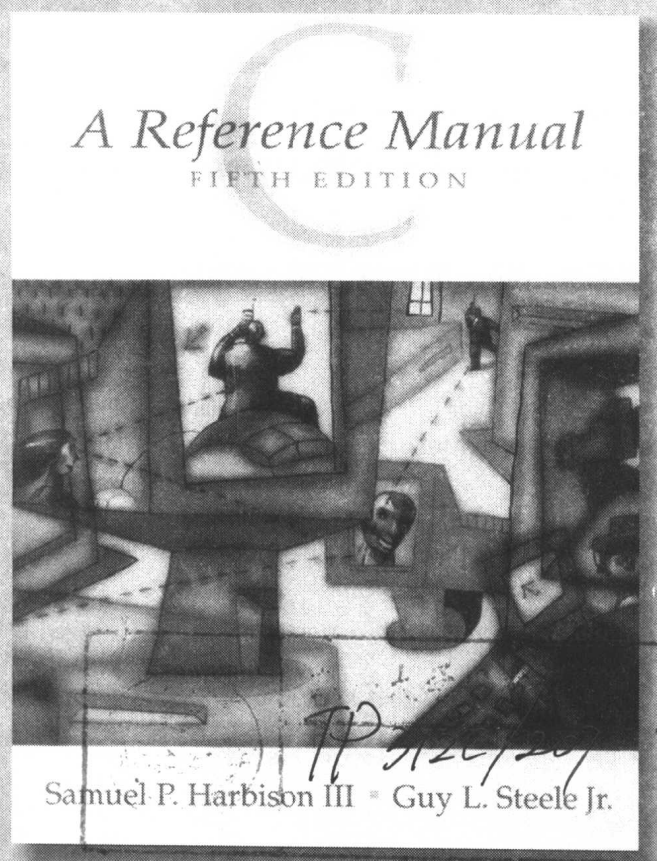
PEARSON
Prentice
Hall

计 算 机 科 学 丛 书

原书第5版

C语言参考手册

(美) Samuel P. Harbison III Guy L. Steele Jr. 著 邱仲潘 等译



C: A Reference Manual
Fifth Edition



机械工业出版社
China Machine Press



0769006 — 10

本书是经典C语言参考手册的最新版,在强调正确性、可移植性和可维护性的基础上,对C语言的具体细节、运行库以及C语言编程风格做了完整、准确的描述。

本书涵盖了传统C语言、C89、C95、C99等所有C语言版本的实现,同时讨论了C++与C语言兼容的部分。全书自上而下介绍了C语言的词法结构、预处理器、声明、类型表达式、语句、函数和运行库,是所有C语言编程人员必备的参考书。

Simplified Chinese edition copyright © 2003 by Pearson Education Asia Limited and China Machine Press .

Original English language title: *C: A Reference Manual , Fifth Edition* (ISBN: 0-13-089592-X) by Samuel P. Harbison III, Guy L. Steele Jr. , Copyright © 2002. All rights reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Prentice-Hall, Inc.

This edition is authorized for sale only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong and Macau).

本书封面贴有Pearson Education培生教育出版集团激光防伪标签,无标签者不得销售。
版权所有,侵权必究。

本书版权登记号:图字:01-2002-3650

图书在版编目(CIP)数据

C语言参考手册(原书第5版)/(美)哈比逊(Harbison, S. P. III),斯蒂尔(Steele, G. L. Jr.)著;邱仲潘等译.-北京:机械工业出版社,2003.8

(计算机科学丛书)

书名原文:C: A Reference Manual, Fifth Edition

ISBN 7-111-12219-4

I. C… II. ①哈… ②斯… ③邱… III. C语言-程序设计-技术手册 IV. TP312-62

中国版本图书馆CIP数据核字(2003)第039007号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑:庞燕

北京瑞德印刷有限公司印刷·新华书店北京发行所发行

2003年8月第1版第1次印刷

787mm×1092mm 1/16·25.75印张

印数:0 001-5 000册

定价:39.00元

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

出版者的话

文艺复兴以降，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域中取得了垄断性的优势；也正是这样的传统，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅擘划了研究的范畴，还揭橥了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短、从业人员较少的现状下，美国等发达国家在其计算机科学发展的几十年间积淀的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章图文信息有限公司较早意识到“出版要为教育服务”。自1998年开始，华章公司就将工作重点放在了遴选、移译国外优秀教材上。经过几年的不懈努力，我们与Prentice Hall, Addison-Wesley, McGraw-Hill, Morgan Kaufmann等世界著名出版公司建立了良好的合作关系，从它们现有的数百种教材中甄选出Tanenbaum, Stroustrup, Kernighan, Jim Gray等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及收藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力襄助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专诚为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍，为进一步推广与发展打下了坚实的基础。

随着学科建设的初步完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都步入一个新的阶段。为此，华章公司将加大引进教材的力度，在“华章教育”的总规划之下出版三个系列的计算机教材：除“计算机科学丛书”之外，对影印版的教材，则单独开辟出“经典原版书库”；同时，引进全美通行的教学辅导书“Schaum's Outlines”系列组成“全美经典学习指导系列”。为了保证这三套丛书的权威性，同时也为了更好地为学校和老师们的服务，华章公司聘请了中国科学院、北京大学、清华大学、国防科技大学、复旦大学、上海交通大学、南京大学、浙江大学、中国科技大学、哈尔滨工业大学、西安交通大学、中国人民大学、北京航空航天大学、北京邮电大学、中山大学、解放军理工大学、郑州大学、湖北工学院、中国国家信息安全测评认证中心等国内重点大学和科研机构在计算机的各个领域的著名学者组成“专

家指导委员会”，为我们提供选题意见和出版监督。

这三套丛书是响应教育部提出的使用外版教材的号召，为国内高校的计算机及相关专业的教学度身订造的。其中许多教材均已为M. I. T., Stanford, U.C. Berkeley, C. M. U. 等世界名牌大学所采用。不仅涵盖了程序设计、数据结构、操作系统、计算机体系结构、数据库、编译原理、软件工程、图形学、通信与网络、离散数学等国内大学计算机专业普遍开设的核心课程，而且各具特色——有的出自语言设计者之手、有的历经三十年而不衰、有的已被全世界的几百所高校采用。在这些圆熟通博的名师大作的指引之下，读者必将在计算机科学的宫殿中由登堂而入室。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证，但我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。教材的出版只是我们的后续服务的起点。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方式如下：

电子邮件：hzedu@hzbook.com

联系电话：(010) 68995264

联系地址：北京市西城区百万庄南街1号

邮政编码：100037

专家指导委员会

(按姓氏笔画顺序)

尤晋元
石教英
张立昂
邵维忠
周克定
郑国梁
高传善
裘宗燕

王 珊
吕 建
李伟琴
陆丽娜
周傲英
施伯乐
梅 宏
戴 葵

冯博琴
孙玉芳
李师贤
陆鑫达
孟小峰
钟玉琢
程 旭

史忠植
吴世忠
李建中
陈向群
岳丽华
唐世渭
程时端

史美林
吴时霖
杨冬青
周伯生
范 明
袁崇义
谢希仁

前言

本书作为一本C语言参考手册，对C语言的基本概念和运行库提供了完整和准确的描述，同时还强调了以正确性、可移植性和可维护性为根本出发点的良好的编程风格。

我们希望读者已经了解基本编程概念，并且很多读者已经可以用C语言熟练编程。为了保持参考手册的格式，我们自下而上介绍C语言的词法结构、预处理器、声明、类型、表达式、语句、函数和运行库。书中包括许多交叉引用，使读者可以从任何地方入手。

第5版完整地介绍了最新的国际C语言标准ISO/IEC 9899:1999(C99)，明确指出语言本身和库函数中哪些特性是C99新增的，指出C99与原有C89标准的不同之处。这是目前惟一一本适用于所有主流C语言版本的参考书：包括传统C语言、1989年C标准、1995年对C89的修改与补充以及当前的C99标准。本书还介绍了标准C语言和标准C++的原始C语言子集。尽管C99中有许多新的信息，但我们没有对本书的章节组织做很大的修改，这样就可以使熟悉旧版的读者能够顺利找到所要的材料。

本书最初源于我们在Tartan公司的工作——为从微机到大型机的一系列计算机开发C语言编译器系列。我们要求编译器文档齐全，提供精确和有用的错误诊断信息并能产生有效的目标代码。一个经过某一编译器正确编译的C语言程序应能在硬件差别允许的前提下，在所有其他编译器中正确编译。

1984年，尽管C语言已经非常普及，但还没有一本书能够非常精确地介绍C语言，以便指导我们设计新的编译器。同样，当时的文档对编程人员和客户也不够精确，人们希望利用编译器比采用当时已经习惯的方法可以更彻底地分析C语言程序。本书特别注意影响程序清晰度、目标代码有效性和不同环境中程序移植性的语言特性。

Web站点

欢迎读者访问本书的Web站点**CAReferenceManual.com**，其中包括了书中示例的代码、更深入的讨论、澄清的问题以及更多C语言资源的链接。

致谢

在第5版准备过程中，特别感谢原NCITS J11主席Rex Jaeschke、芬兰赫尔辛基的Antoine Trux以及爱迪生设计集团创始人Steve Adamczyk的帮助。

对于本书的以前版本给予过帮助的人员包括Jeffrey Esakov、Alan J.Filipski、Frank J. Wagner、Debra Martin、P. J. Plauger以及Steve Vinoski。其他提供过帮助的人员包括Aurelio Bignoli、Steve Clamage、Arthur Evans、Jr.,Roy J. Fuller、Morris M. Kessan、George V.Reilly、Mark Lan、Mike Hewett、Charles Fischer、Kevin Rodgers、Tom Gibb、David Lim、Stavros Macrakis、Steve Vegdahl、Christopher Vickery、Peter van der Linden和Dave Wilson。还要感谢Michael Angus、Mady Bauer、Larry Breed、Sue Broughton、Alex Czajkowski、Robert Firth、David Gaffney、Steve Gorman、Dennis Hamilton、Chris Hanna、Ken Harrenstien、Rex Jaeschke、

Don Lindsay、Tom MacDonald、Peter Nelson、Joe Newcomer、Keyin Nolish、David Notkin、Peter Plamondon、Roger Ray、Larry Rosler、David Spencer以及Barbara Steele。

本书最初的一些示例程序参考了下列著作中的算法：

- Beeler, Michael, Gosper, R. William, and Schroepel, Richard, *HAKMEM*, AI Memo 239 (Massachusetts Institute of Technology Artificial Intelligence Laboratory, 1972年2月);
- Bentley, Jon Louis, *Writing Efficient Programs*(Prentice-Hall, 1982);
- Bentley, Jon Louis, "Programming Pearls" (monthly column appearing in *Communications of the ACM* beginning August 1983);
- Kernighan, Brian W., and Ritchie, Dennis M., *The C Programming Language*(Prentice-Hall, 1978);
- Knuth, Donald E., *The Art of Computer Programming Volumes 1-3*(Addison-Wesley, 1968, 1969, 1973, 1981);
- Sedgewick, Robert, *Algorithms*(Addison-Wesley, 1983).

感谢这些作者的灵感。

还有一点需要说明，Guy Steele先生工作繁忙，无法在新版本中参与更多工作。虽然书中仍然处处体现出他对C语言精辟的分析，但新版本中的任何新问题，都不能再由他负责。

《C语言参考手册》一书已经超过17岁，感谢所有读者多年来对她的关爱。

Sam Harbison

于宾夕法尼亚州匹兹堡市

harbison@CReferenceManual.com

参加本书翻译的人员有：周阳生、刘文红、邹能东、彭振庆、黄志坚、李耀平、郭王旋。刘文琼、邱冬全、邱燕等完成了本书的录入工作，刘云昌、刘联昌对手写稿与打印稿进行了校对。

目 录

出版者的话
专家指导委员会
前言

第一部分 C语言

第1章 简介	1
1.1 C语言的演变	1
1.1.1 传统C语言	1
1.1.2 标准C语言(1989)	1
1.1.3 标准C语言(1995)	2
1.1.4 标准C语言(1999)	2
1.1.5 标准C++语言	2
1.1.6 本书内容	3
1.2 使用C语言的何种方言	3
1.3 C语言编程概述	3
1.4 符合性	4
1.5 语法符号	5
第2章 词法元素	6
2.1 字符集	6
2.1.1 执行字符集	7
2.1.2 空白符与行终结符	7
2.1.3 字符编码方式	8
2.1.4 三字符组	8
2.1.5 多字节字符与宽字符	9
2.2 注释	10
2.3 记号	11
2.4 运算符与分隔符	12
2.5 标识符	12
2.6 关键字	14
2.7 常量	15
2.7.1 整型常量	15
2.7.2 浮点型常量	18
2.7.3 字符型常量	20

2.7.4 字符串型常量	21
2.7.5 转义符	23
2.7.6 字符转义符	24
2.7.7 数字转义符	25
2.8 C++兼容性	25
2.8.1 字符集	25
2.8.2 注释语句	26
2.8.3 运算符	26
2.8.4 标识符与关键字	26
2.8.5 字符型常量	26
2.9 字符集、指令集和编码方式	26
2.10 练习	27
第3章 C语言预处理器	29
3.1 预处理器命令	29
3.2 预处理器词法规则	29
3.3 定义和替换	30
3.3.1 对象式宏定义	31
3.3.2 函数式宏定义	31
3.3.3 重新扫描宏表达式	33
3.3.4 预定义宏	34
3.3.5 取消宏定义与重新定义宏	36
3.3.6 宏扩展中的优先级错误	36
3.3.7 宏参数的副作用	37
3.3.8 将记号转换为字符串	37
3.3.9 宏扩展中的记号合并	38
3.3.10 宏中的可变参数表	39
3.3.11 其他问题	39
3.4 文件包含	40
3.5 条件编译	41
3.5.1 #if、#else与#endif命令	41
3.5.2 #elif命令	42
3.5.3 #ifdef与#ifndef命令	43
3.5.4 条件命令中的常量表达式	44

3.5.5 defined运算符	45	4.5.5 复合声明符	70
3.6 显式的行编号	45	4.6 初始化语句	71
3.7 杂注指令	46	4.6.1 整数变量初始化语句	72
3.7.1 标准杂注	46	4.6.2 浮点数变量初始化语句	73
3.7.2 标准杂注的位置	46	4.6.3 指针变量初始化语句	73
3.7.3 _Pragma运算符	47	4.6.4 数组类型变量初始化语句	74
3.8 错误指令	47	4.6.5 枚举类型变量初始化语句	76
3.9 C++兼容性	47	4.6.6 结构类型变量初始化语句	76
3.10 练习	48	4.6.7 联合变量初始化语句	77
第4章 声明	50	4.6.8 省略花括号	77
4.1 声明组织	50	4.6.9 指定初值	77
4.2 术语	51	4.7 隐式声明	79
4.2.1 作用域	51	4.8 外部名称	79
4.2.2 有效性	52	4.8.1 初始化语句模型	79
4.2.3 向前引用	52	4.8.2 省略存储类模型	79
4.2.4 名称重载	53	4.8.3 公用模型	80
4.2.5 重复声明	54	4.8.4 混合公用模型	80
4.2.6 重复有效性	54	4.8.5 总结与建议	80
4.2.7 生存期	55	4.8.6 未引用的外部声明	81
4.2.8 初值	55	4.9 C++兼容性	81
4.2.9 外部名称	56	4.9.1 作用域	81
4.2.10 编译名称	57	4.9.2 标志名称与typedef名称	81
4.3 存储类说明符与函数说明符	57	4.9.3 类型的存储类说明符	82
4.3.1 默认存储类说明符	58	4.9.4 const类型限定符	82
4.3.2 存储类说明符举例	58	4.9.5 初始化语句	82
4.3.3 函数说明符	59	4.9.6 隐式声明	82
4.4 类型说明符与限定符	59	4.9.7 定义声明与引用声明	82
4.4.1 默认类型说明符	60	4.9.8 函数连接	83
4.4.2 忽略声明符	61	4.9.9 无参函数	83
4.4.3 类型限定符	61	4.10 练习	83
4.4.4 const类型限定符	62	第5章 类型	85
4.4.5 volatile类型限定符与序列点	63	5.1 整数类型	86
4.4.6 restrict类型限定符	65	5.1.1 带符号整数类型	86
4.5 声明符	66	5.1.2 无符号整数类型	89
4.5.1 简单声明符	66	5.1.3 字符类型	90
4.5.2 指针声明符	67	5.1.4 扩展整数类型	91
4.5.3 数组声明符	67	5.1.5 布尔类型	91
4.5.4 函数声明符	69	5.2 浮点数类型	92

5.3 指针类型	95	5.11.7 源文件之间的兼容性	124
5.3.1 通用指针	96	5.12 类型名与抽象声明符	125
5.3.2 null指针与无效指针	97	5.13 C++兼容性	126
5.3.3 指针注意事项	98	5.13.1 枚举类型	126
5.4 数组类型	98	5.13.2 typedef名称	126
5.4.1 数组与指针	98	5.13.3 类型兼容性	127
5.4.2 多维数组	99	5.14 练习	127
5.4.3 数组边界	100	第6章 转换与表示	129
5.4.4 运算	100	6.1 表示	129
5.4.5 变长数组	101	6.1.1 存储单元与数据长度	129
5.5 枚举类型	102	6.1.2 字节顺序	130
5.6 结构类型	104	6.1.3 对齐限制	131
5.6.1 结构类型引用	106	6.1.4 指针长度	132
5.6.2 结构运算	107	6.1.5 寻址模型	132
5.6.3 成员	107	6.1.6 类型表示	134
5.6.4 结构成员存储布局	108	6.2 转换	134
5.6.5 位字段	109	6.2.1 表示方法改变	134
5.6.6 移植性问题	111	6.2.2 普通转换	134
5.6.7 结构长度	112	6.2.3 转换成整数类型	135
5.6.8 灵活数组成员	112	6.2.4 转换成浮点数类型	136
5.7 联合类型	113	6.2.5 转换成结构类型或联合类型	136
5.7.1 联合成员存储布局	114	6.2.6 转换成枚举类型	136
5.7.2 联合类型长度	114	6.2.7 转换成指针类型	137
5.7.3 使用联合类型	115	6.2.8 转换成数组类型或函数类型	137
5.7.4 误用联合类型	116	6.2.9 转换成void类型	137
5.8 函数类型	117	6.3 普通转换	138
5.9 void类型	119	6.3.1 类型转换	138
5.10 typedef名称	119	6.3.2 赋值转换	138
5.10.1 函数类型的typedef名称	121	6.3.3 普通一元转换	139
5.10.2 重定义typedef名称	121	6.3.4 普通二元转换	141
5.10.3 实现注意事项	122	6.3.5 默认函数参数转换	142
5.11 类型兼容性	122	6.3.6 其他函数转换	143
5.11.1 一致类型	122	6.4 C++兼容性	143
5.11.2 枚举兼容性	123	6.5 练习	143
5.11.3 数组兼容性	123	第7章 表达式	145
5.11.4 函数兼容性	124	7.1 对象、左值与指定符	145
5.11.5 结构和联合兼容性	124	7.2 表达式与优先级	146
5.11.6 指针兼容性	124	7.2.1 运算符优先级与结合律	146

7.2.2 溢出和其他算术异常	147	7.12 求值顺序	182
7.3 主表达式	148	7.13 放弃值	183
7.3.1 名称	148	7.14 优化内存访问	184
7.3.2 字面值	149	7.15 C++兼容性	185
7.3.3 括号表达式	149	7.16 练习	185
7.4 后缀表达式	149	第8章 语句	187
7.4.1 下标表达式	150	8.1 语句的一般语法规则	187
7.4.2 成员选择	151	8.2 表达式语句	188
7.4.3 函数调用	153	8.3 标号语句	188
7.4.4 后缀自增运算符与后缀自减运算符	154	8.4 复合语句	189
7.4.5 复合字面值	155	8.5 条件语句	190
7.5 一元表达式	156	8.5.1 多路条件语句	191
7.5.1 类型转换	157	8.5.2 悬而未决的else问题	191
7.5.2 sizeof运算符	157	8.6 迭代语句	192
7.5.3 一元负号运算符与一元正号运算符	158	8.6.1 while语句	192
7.5.4 逻辑非运算符	159	8.6.2 do语句	193
7.5.5 按位反运算符	159	8.6.3 for语句	194
7.5.6 地址运算符	160	8.6.4 for语句应用	195
7.5.7 间接访问运算符	160	8.6.5 多个控制变量	197
7.5.8 前缀自增运算符与前缀自减运算符	161	8.7 switch语句	198
7.6 二元运算符表达式	162	8.8 break语句与continue语句	200
7.6.1 乘法运算符	162	8.9 return语句	202
7.6.2 加法运算符	164	8.10 goto语句	203
7.6.3 移位运算符	165	8.11 null语句	203
7.6.4 关系运算符	167	8.12 C++兼容性	204
7.6.5 判等运算符	168	8.12.1 复合语句	204
7.6.6 按位运算符	169	8.12.2 循环中的声明	204
7.6.7 整数集合示例	170	8.13 练习	204
7.7 逻辑运算符表达式	174	第9章 函数	206
7.8 条件表达式	175	9.1 函数定义	206
7.9 赋值表达式	176	9.2 函数原型	208
7.9.1 简单赋值	177	9.2.1 何时存在原型	209
7.9.2 复合赋值	178	9.2.2 混合原型声明与非原型声明	210
7.10 顺序表达式	179	9.2.3 正确使用原型	211
7.11 常量表达式	180	9.2.4 原型与调用规则	211
7.11.1 预处理器常量表达式	180	9.2.5 与标准C语言和传统C语言的兼容性	212
7.11.2 整型常量表达式	181	9.3 正式参数声明	213
7.11.3 初始化程序常量表达式	181	9.4 调整参数类型	215

9.5 参数传递规则	216
9.6 参数一致性	216
9.7 函数返回类型	217
9.8 返回类型一致性	218
9.9 主程序	218
9.10 内联函数	219
9.11 C++兼容性	221
9.11.1 原型	221
9.11.2 参数类型声明与返回类型声明	221
9.11.3 返回类型一致性	221
9.11.4 main函数	221
9.11.5 内联	221
9.12 练习	221

第二部分 C语言库

第10章 库简介	223
10.1 标准C语言函数	223
10.2 C++兼容性	224
10.3 库头文件与名称	225
10.3.1 assert.h	225
10.3.2 complex.h	226
10.3.3 ctype.h	226
10.3.4 errno.h	226
10.3.5 fenv.h	226
10.3.6 float.h	226
10.3.7 inttypes.h	227
10.3.8 iso646.h	227
10.3.9 limits.h	227
10.3.10 locale.h	227
10.3.11 math.h	227
10.3.12 setjmp.h	229
10.3.13 signal.h	229
10.3.14 stdarg.h	229
10.3.15 stdbool.h	229
10.3.16 stddef.h	229
10.3.17 stdint.h	229
10.3.18 stdio.h	229
10.3.19 stdlib.h	230

10.3.20 string.h	230
10.3.21 tgmath.h	230
10.3.22 time.h	231
10.2.23 wchar.h	231
10.3.24 wctype.h	231
第11章 标准语言补充	232
11.1 NULL、ptrdiff_t、size_t、offsetof	232
11.2 EDOM、ERANGE、EILSEQ、 errno、strerror、perror	233
11.3 bool、false、true	234
11.4 va_list、va_start、va_arg、va_end	235
11.5 标准C语言运算符宏	238
第12章 字符处理函数	239
12.1 isalnum、isalpha、isctrl、iswalnum、 iswalphabet、iswcntrl	239
12.2 iscsym、iscsymf	241
12.3 isdigit、isodigit、isxdigit、iswdigit、 iswxdigit	241
12.4 isgraph、isprint、ispunct、iswgraph、 iswprint、iswpunct	241
12.5 islower、isupper、iswlower、iswupper	242
12.6 isblank、isspace、iswhite、iswspace	243
12.7 toascii	243
12.8 toint	244
12.9 tolower、toupper、tolower、toupper	244
12.10 wctype_t、wctype、iswctype	245
12.11 wctrans_t、wctrans	246
第13章 字符串处理函数	247
13.1 strcat、strncat、wcscat、wcsncat	247
13.2 strcmp、strncmp、wcscmp、wcsncmp	248
13.3 strcpy、strncpy、wcscpy、wcsncpy	249
13.4 strlen、wcslen	249
13.5 strchr、strchr、wcschr、wcschr	250
13.6 strspn、strcspn、strpbrk、strpbrk、 wcspn、wcscspn、wcpbrk	251
13.7 strstr、strtok、wcsstr、wcstok	252
13.8 strtod、strtof、strtold、strtoll、 strtoul、strtoull	253

13.9	atof、atoi、atol、atoll	253	15.13	fread、fwrite	286
13.10	strcoll、strxfrm、wcscoll、wcsxfrm	253	15.14	feof、ferror、clearerr	287
第14章	内存函数	255	15.15	remove、rename	287
14.1	memchr、wmemchr	255	15.16	tmpfile、tmpnam、mktemp	288
14.2	memcmp、wmemcmp	255	第16章	通用函数	289
14.3	memcpy、memccpy、memmove、 wmemcpy、wmemmove	256	16.1	malloc、calloc、mimalloc、clalloc、 free、cfree	289
14.4	memset、wmemset	257	16.2	rand、srand、RAND_MAX	291
第15章	输入/输出函数	258	16.3	atof、atoi、atol、atoll	291
15.1	FILE、EOF、wchar_t、wint_t、WEOF	259	16.4	strtod、strtof、strtold、strtoll、 strtoul、strtoull	292
15.2	fopen、fclose、fflush、freopen、fwide	259	16.5	abort、atexit、exit、_Exit、EXIT_ FAILURE、EXIT_SUCCESS	294
15.2.1	文件访问方式	261	16.6	getenv	295
15.2.2	文件定向	262	16.7	system	295
15.3	setbuf、setvbuf	262	16.8	bsearch、qsort	296
15.4	stdin、stdout、stderr	263	16.9	abs、labs、llabs、div、ldiv、lldiv	298
15.5	fseek、ftell、rewind、fgetpos、fsetpos	264	16.10	mblen、mbtowc、wctomb	299
15.5.1	fseek与ftell	264	16.10.1	编码方式与转换状态	299
15.5.2	fgetpos与fsetpos	265	16.10.2	长度函数	300
15.6	fgetc、fgetwc、getc、getwc、getchar、 getwchar、ungetc、ungetwc	266	16.10.3	转换成宽字符	300
15.7	fgets、fgetws、gets	267	16.10.4	转换宽字符	300
15.8	fscanf、fwscanf、scanf、wscanf、 sscanf、swscanf	267	16.11	mbstowcs、wcstombs	300
15.8.1	控制字符串	268	16.11.1	转换成宽字符串	301
15.8.2	转换说明	269	16.11.2	转换宽字符串	302
15.9	fputc、fputwc、putc、putwc、 putchar、putwchar	273	第17章	数学函数	302
15.10	fputs、fputws、puts	274	17.1	abs、labs、llabs、div、ldiv、lldiv	302
15.11	fprintf、printf、sprintf、snprintf、 fwprintf、wprintf、swprintf	274	17.2	fabs	302
15.11.1	输出格式	275	17.3	ceil、floor、lrint、llrint、lround、 llround、nearbyint、round、rint、trunc	303
15.11.2	转换说明	276	17.4	fmod、remainder、remquo	304
15.11.3	转换标志	276	17.5	frexp、ldexp、modf、scalbn	304
15.11.4	最小字段宽度	277	17.6	exp、exp2、expm1、ilogb、 log、log10、log1p、log2、logb	305
15.11.5	精度说明	277	17.7	cbirt、fma、hypot、pow、sqrt	306
15.11.6	长度说明	277	17.8	rand、srand、RAND_MAX	307
15.11.7	转换操作	278	17.9	cos、sin、tan、cosh、sinh、tanh	307
15.12	v[x]printf、v[x]scanf	285	17.10	acos、asin、atan、atan2、	

acosh、asinh、atanh	307	21.5 指针长度类型与最大长度类型	335
17.11 fdim、fmax、fmin	308	21.6 ptrdiff_t、size_t、wchar_t、wint_t 与sig_atomic_t的范围	336
17.12 通用类型宏	309	21.7 imaxabs、imaxdiv、imaxdiv_t	336
17.13 erf、erfc、lgamma、tgamma	312	21.8 strtoumax、strtoimax	337
17.14 fpclassify、isfinite、isinf、 isnan、isnormal、signbit	312	21.9 wcstoumax、wcstoumax	337
17.15 copysign、nan、nextafter、nexttoward	313	第22章 浮点数环境	338
17.16 isgreater、isgreaterequal、isless、 islessequal、islessgreater、isunordered	314	22.1 概述	338
第18章 时间和日期函数	315	22.2 浮点数环境	338
18.1 clock、clock_t、CLOCKS_PER_SEC、 times	315	22.3 浮点数异常	339
18.2 time、time_t	316	22.4 浮点数舍入方式	340
18.3 asctime、ctime	316	第23章 复数算术	341
18.4 gmtime、localtime、mktime	317	23.1 复数库规则	341
18.5 difftime	318	23.2 complex、_Complex_I、imaginary、 _Imaginary_I、I	342
18.6 strptime、wcsftime	319	23.3 CX_LIMITED_RANGE	341
第19章 控制函数	322	23.4 cacos、casin、catan、ccos、csin、ctan	341
19.1 assert、NDEBUG	322	23.5 cacosh、casinh、catanh、ccosh、 csinh、ctanh	342
19.2 system、exec	322	23.6 cexp、clog、cabs、cpow、csqrt	343
19.3 exit、abort	322	23.7 carg、cimag、creal、conj、cproj	344
19.4 setjmp、longjmp、jmp_buf	323	第24章 宽字符与多字节函数	346
19.5 atexit	324	24.1 基本类型和宏	346
19.6 signal、raise、gsignal、ssignal、psignal	324	24.2 多字节字符与宽字符的转换	346
19.7 sleep、alarm	326	24.3 宽字符串与多字节字符串的转换	347
第20章 区域设置函数	327	24.4 转换成算术类型	348
20.1 setlocale	327	24.5 输入与输出函数	349
20.2 localeconv	328	24.6 字符串函数	349
第21章 扩展整数类型	331	24.7 日期与时间转换	350
21.1 一般规则	331	24.8 宽字符分类函数与映射函数	350
21.1.1 类型种类	331		
21.1.2 全部定义或全部不定义	331		
21.1.3 最小限制与最大限制	331		
21.1.4 PRI...与SCN...格式字符串宏	332		
21.2 准确长度类型	333		
21.3 最小长度类型	333		
21.4 快速长度类型	334		

第三部分 附 录

附录A ASCII字符集	353
附录B C语言语法	354
附录C 练习答案	367
索引	374

第一部分 C 语言

第1章 简介

20世纪70年代初, Dennis Ritchie在贝尔实验室设计了C语言, 它的前身可以追溯到1960年的ALGOL 60语言, 1963年剑桥的CPL语言, 1967年Martin Richard的BCPL语言, 以及1970年贝尔实验室Ken Thompson的B语言。尽管C语言是一种通用编程语言, 但通常用于系统编程。特别值得一提的是, 著名的UNIX操作系统最初就是用C语言写成的。

C语言的普及有许多原因。它是个小巧、高效而强大的编程语言, 具有丰富的运行库, 而且不使用很多的隐藏机制就可以对计算机进行精确控制。经过10多年的标准化之后, 编程人员已经习惯了C语言。一般来说, 很容易用C语言编写可以在不同国家用不同语言的不同计算系统之间移植的程序。而且, 现有的大量遗留C语言代码正在被修改和扩展。

从20世纪90年代末期开始, 虽然C语言慢慢被“大哥”C++取代, 但它仍然有许多忠实追随者, 在不需要C++特性或不接受C++的开销的场合, C语言仍然非常流行。

C语言经受了时间的检验, 仍不失为一种熟练的编程人员用来迅速有效地工作的编程语言。几百万行的代码充分证明了这种语言的优势。

1.1 C语言的演变

1984年我们编写本书第1版时, C语言已经被广泛使用, 但还没有关于这个语言的正式标准和精确描述。事实上的标准是当时正在使用的C语言编译器。1989年, C语言建立了国际标准, 1994年作了修订, 1999年又作了一次重大修改。

仅仅改变语言的定义并不能自动改变全世界已有的几百万行C语言代码。我们努力使本书与时俱进, 这样编程人员在遇到各种C语言方言时都能以本书作为参考手册。

3

1.1.1 传统C语言

最初的C语言描述出现在Brian Kernighan与Dennis Ritchie (通常合称为“K&R”)的著作《The C Programming Language》第1版 (Prentice-Hall, 1978)。此书出版后, 这个语言不断有细小的演变, 增加或删除了一些特性。我们把20世纪80年代初公认的C语言定义称为传统C语言, 是标准化之前的方言。当然, 各个C语言提供商也对传统C语言进行了各种扩展。

1.1.2 标准C语言 (1989)

1982年, 美国国家标准协会 (ANSI) 认识到标准化将有助于C语言在商业化编程中的普及, 因此成立了一个委员会来为C语言及其运行库制定标准。这个委员会, 即X3J11 (现为NCITS J11) 的主席是Jim Brodie, 它制定了一个标准并在1989年被正式采用, 即美国国家标准X3.159-1989或称作“ANSI C”。

考虑到编程活动是国际化的, 因此完成ANSI C语言之后, 成立了一个国际标准化组织

ISO/IEC JTC1/SC22/WG14, 在P.J.Plauger的领导下, 只作了少量编辑性修改, 即把ANSI标准变成国际标准ISO/IEC 9899:1990。此后, ISO/IEC标准被ANSI采用, 人们把这个公共标准称为“标准C语言”。由于这个标准后来又有了改变, 因此我们称其为“标准C语言(1989)”或简称“C89”。

传统C语言到C89的改变包括:

- 增加了真正的标准库
- 新的预处理器命令与特性
- 函数原型允许在函数声明中指定参数类型
- 一些新关键字, 包括**const**、**volatile**与**signed**
- 宽字符、宽字符串和多字节字符
- 对约定规则、声明和类型检查的许多小改动与澄清

1.1.3 标准C语言(1995)

作为对C语言标准的正常维护工作, WG14对C89作了两处技术修订(缺陷修复)和一个补充(扩展)。总的来看, 尤其是通过增加新的库函数, 以上这些工作对C语言标准进行了相对合适的修改, 得到的结果我们称之为“C89增补1”或“C95”。C95对C89所作的改变包括:

- 3个新的标准库头文件**iso646.h**、**wctype.h**与**wchar.h**
- 几个新的记号和宏, 用于替换一些国家的字符集中没有的运算符和标点符号
- **printf/scanf**系列函数的一些新格式代码
- 大量新函数和一些类型与常量, 用于多字节字符和宽字符

1.1.4 标准C语言(1999)

ISO/IEC标准都需要经常进行审查和更新。1995年, WG14开始对C语言标准作更大的修订, 最终于1999年完成并获批准。新标准ISO/IEC 9899:1999或“C99”取代原有的标准(及所有修订与补充), 成为正式标准C语言。提供商根据新标准更新各自的C语言函数库和编译器。

C99在C89/C95语言和库函数中增加了许多新特性, 包括:

- 复数运算
- 扩展整数类型, 包括长标准类型
- 变长数组
- 布尔类型
- 对非英语字符集更好的支持
- 对浮点数类型更好的支持, 包括所有类型的数学函数
- C++风格的注释(//)

C99的改动比C95更大, 包括语言的改变和函数库的扩展。C99标准文档比C89文档大得多。但是, 改变还是“本着C语言精神”进行的, 语言的基本性质没有改变。

1.1.5 标准C++语言

C++是20世纪80年代初由AT & T贝尔实验室的Bjarne Stroustrup设计的, 现在其地位已大大超过C语言, 成为主流编程语言。大多数C语言实现实际上是C/C++实现, 编程人员可以自己选择要使用哪一种语言。C++本身也已经标准化为ISO/IEC 14882:1998或称“标准C++语言”。C++在C语言之上作了许多改进, 适合开发大型应用程序, 包括对类型检查的改进和面向对象编程的支持。但是, C++也是最复杂的编程语言之一, 对于粗心的编程人员来说, 会陷入很多陷阱。