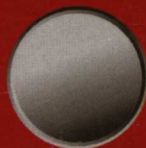




21世纪高等院校计算机科学与技术系列教材

主编 郝 谦 孙英华



C 程序设计

C CHENGXU SHEJI



北京邮电大学出版社
www.buptpress.com

C 程 序 设 计

主 编 郝 谦 孙英华
副主编 曾 辉 曾纯清 钟敏娟

北 京 邮 电 大 学 出 版 社

· 北 京 ·

内 容 简 介

本书共 13 章,第 1 章介绍 C 语言及程序设计的基本概念,重点是将读者引入到 C 语言及程序设计的世界中;第 2 章重点介绍数据类型和运算符,这是初学程序设计的重点之一;第 3 章至第 5 章分别介绍程序设计的 3 种结构和程序设计的方法,学习程序设计就必须熟练掌握这些内容;第 6 章介绍 C 语言的数组概念;第 7 章介绍 C 语言的函数概念;第 8 章介绍 C 语言的预处理命令;第 9 章为 C 语言的精华和难点——指针;第 10 章讨论结构体和共用体;第 11 章介绍了 C 语言的位运算;第 12 章介绍 C 语言的文件操作;第 13 章介绍了 C 语言家族的其他开发平台及 C 语言的发展。

本书以大量的案例来推动概念的讲解,使得难点和重点分解到不同的示例中,深入浅出、避虚就实,使读者能循序渐进,逐步了解,不但可以熟练掌握 C 语言,更重要的是可以掌握程序设计的基本思想和基本方法。

本书适用于本科、专科等各种层次的学校教学或计算机培训教材,也可作为程序设计爱好者的自学及参考书。

本书源代码与课件随书的配套光盘一并发行。

图书在版编目(CIP)数据

C 程序设计/郝谦,孙英华主编. —北京:北京邮电大学出版社,2005

ISBN 7-5635-1076-1

I. C... II. ①郝...②孙... III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2005)第 096738 号

出 版 者:北京邮电大学出版社(北京市海淀区西土城路 10 号) 邮编:100876

发行部电话:(010)62282185 62283578(传真)

电子信箱:publish@bupt.edu.cn

经 销:各地新华书店

印 刷:北京源海印刷有限责任公司

开 本:787 mm×1 092 mm 1/16

印 张:17

字 数:420 千字

印 数:1—3 000 册

版 次:2005 年 8 月第 1 版 2005 年 8 月第 1 次印刷

ISBN 7-5635-1076-1/TP·187

定价:28.00 元

·如有印装质量问题,请与北京邮电大学出版社发行部联系·

21 世纪高等院校计算机科学与技术系列教材

编 委 会

主 任：金怡濂

委 员：(按姓氏笔划排名)

王士同 王明严 刘 弘

朱其亮 何炎祥 汪厚祥

金 海 徐 涛 潘振宽

序

计算机科学技术是科学性 with 工程性并重的一门学科。它的迅猛发展除了源于微电子学等相关学科的发展外,更主要源于其应用需求的广泛性不断增长,它已渗透到人类社会的各个领域,成为经济发展的倍增器,科学文化与社会进步的催化剂。计算机与通信的融合和全球联网,更显示出它无可限量的发展前景。任何一个领域的发展都离不开计算机已成为无可否认的事实。应用是计算机科学技术发展的动力、源泉和归宿,而计算机科学技术又不断为应用提供先进的方法、设备与环境。

近年来,计算机科学技术的发展不仅极大地促进了整个科学技术的发展,而且明显地推进了经济信息化和社会信息化的进程。计算机科学技术对一个国家在政治、经济、科技、文化、国防等方面的催化作用和强化作用都具有难以估量的意义。计算机知识与能力已成为 21 世纪人才素质的基本要求之一,因此,计算机科学技术的教育在世界各国都备受重视,我国政府和教育部门对计算机科学技术的教育及人才培养也非常重视。为了适应社会发展对计算机科学技术人才的强烈要求,各高校均在着力培养基础扎实、知识面广、综合素质高、实践能力强、富有创新精神,且具有较强的科学技术运用、推广、转化能力的高层次人才。

由北京邮电大学出版社联合北京邮电大学、武汉大学、华中理工大学及山东、江苏等多所高校的计算机专业教学负责人组成的“21 世纪高等院校计算机科学与技术系列教材编委会”按照《中国计算机科学与技术学科教程 2002》的要求组织编写的系列教材,体现了近年计算机学科的新理论、新技术。内容涵盖计算机专业学生所应掌握的相关知识,并根据目前计算机科学技术的发展趋势与实际应用相结合,能够满足目前高校计算机专业教学的需要,也可作为计算机专业人员的自学参考材料。

本系列教材作者均为多年从事教学、科研的一线教师,有着丰富的教学和科研实践经验,所编写的这套教材具有结构严谨、内容丰富、理论与实际结合紧密的特点,是他们的教学经验和科研成果的结晶。

计算机科学技术日新月异,所以教材也要不断推陈出新,我希望本系列教材能为我国高校计算机专业教育做出新的贡献。

中国工程院院士

金怡廉

前 言

计算机应用能力是新世纪人才不可缺少的基本素质,而程序设计是理工科学生重要的技术基础。C语言是目前广泛使用的一种程序设计语言,是国内外学习计算机程序设计方法的首选语言之一。

教材是人才培养过程中重要的工具,是知识传授和能力培养的基础。本书以大量的案例来推动概念的讲解,真正从程序设计实际能力培养出发,力求按照人的认知规律,由浅入深、循序渐进、避虚就实、理论与实践有机结合、融会贯通、知识传授与能力培养并重,使读者不但可以熟练掌握C语言的使用,还能培养其程序设计的基本思想和基本方法。

本书内容丰富、注重实践、案例广泛、图文并茂、深入浅出。全书共分13章和7个附录,第1章介绍C语言及程序设计的基本概念,重点是将读者引入到C语言及程序设计的世界中。第2章重点介绍数据类型和运算符,这些概念不但是学习C语言的基础,也是其他语言平台的基础,是初学程序设计的重点之一。第3章至第5章分别介绍程序设计的3种结构和程序设计的方法,这些是学习程序设计必须熟练掌握的内容。第6章介绍C语言的数组概念。第7章介绍C语言的函数概念。第8章介绍C语言的预处理命令。第9章为C语言的精华和难点——指针。第10章讨论结构体和共用体。第11章介绍了C语言的位运算。第12章为C语言的文件操作。第13章介绍了C语言家族的其他开发平台及C语言的发展。

为了方便读者,同时编写了7个附录,供读者在使用时查询。

程序设计是一个实践性很强的技能,实践是学习程序设计的必备环节,本书将所有的示例代码单独列出,供读者参考对照。

本书的讲授课时为70左右,建议其中理论与实践课时数之比为1:1。如果有可能,建议教师增加课程设计环节,在本书配套的《习题解答与上机指导》中会给出部分课程设计的题目。



本书由郝谦、孙英华主编,曾辉、曾纯清、钟敏娟任副主编;郝谦、曾纯清主审了本书及课件,程山英审核了部分章节和部分课件。参加编写并制作课件的还有姚群、程琳、邓茹仁、程山英、李宏芳、范俨、李希等同志,在本书编写过程中,得到了北京邮电大学出版社的大力帮助,在此一并表示感谢。

由于水平及时间原因,本书难免有不足或错误之处,恳请读者批评指正。

本书适用于本科、专科等各层次的学校教学及培训班使用,也可作为自学用书和软件开发的参考资料。

编者

2005年7月



目 录

第 1 章 C 语言概述

1.1 C 语言概况	1
1.1.1 C 语言的发展过程	1
1.1.2 C 语言的特点	1
1.2 简单的 C 程序	2
1.3 程序设计	5
1.4 模块化和结构化程序设计	5
1.4.1 模块化设计	5
1.4.2 结构化程序设计	6
1.4.3 C 语言程序设计的基本结构	7
1.4.4 C 语言函数的结构	8
1.4.5 C 语句的介绍	9
习题 1	10

第 2 章 数据类型、运算符与表达式

2.1 C 语言常用符号	11
2.1.1 字符集	11
2.1.2 C 语言的关键字和特定字	11
2.1.3 用户自定义字	12
2.2 数据类型	12
2.2.1 C 语言的数据类型	12
2.3 常量与变量	13
2.3.1 常量和符号常量	14
2.3.2 变量	14
2.4 整 型	15
2.4.1 整型常量	15
2.4.2 整型变量	16
2.5 实型数据	20
2.5.1 实型常量的表示方法	20
2.5.2 实型变量	21





2.6 字符型数据	23
2.6.1 字符常量	23
2.6.2 转义字符	23
2.6.3 字符变量	25
2.6.4 字符数据在内存中的存储形式及使用方法	25
2.6.5 字符串常量	26
2.7 变量赋初值	26
2.8 常用运算符与表达式	27
2.8.1 算术运算符	27
2.8.2 赋值运算符和赋值表达式	31
2.8.3 比较运算符	32
2.8.4 逻辑运算符	32
2.8.5 条件运算符和条件表达式	33
2.8.6 逗号运算符和逗号表达式	33
2.8.7 其他运算符	34
2.8.8 复合运算及优先级	34
2.8.9 类型转换运算	35
习题 2	36

第 3 章 顺序结构程序设计

3.1 赋值语句	38
3.2 基本输入与输出操作	38
3.2.1 格式的输入和输出操作	38
3.2.2 标准字符的输入和输出操作	45
3.3 顺序结构程序设计	47
3.3.1 顺序结构程序设计	47
3.3.2 算法	47
3.3.3 程序举例	47
习题 3	50

第 4 章 选择结构程序设计

4.1 问题	51
4.2 选择结构	51
4.2.1 if 形式	51
4.2.2 程序实例	52
4.2.3 if...else 形式	53
4.2.4 程序实例	53



4.2.5 else if 形式	55
4.2.6 程序实例	56
4.2.7 switch 语句及算法	57
4.3 选择结构的嵌套与综合运用	58
习题 4	62

第 5 章 循环结构程序设计

5.1 问题	63
5.2 while 语句	63
5.2.1 while 语句	64
5.2.2 程序举例	65
5.2.3 注意的问题	65
5.3 do··while 语句	66
5.3.1 do··while 语句	66
5.3.2 程序举例	67
5.3.3 注意的问题	68
5.4 for 语句	68
5.4.1 for 语句	68
5.4.2 程序举例	71
5.4.3 注意的问题	71
5.5 break 与 continue 语句	72
5.6 goto 语句	74
5.7 循环结构的嵌套及综合应用举例	75
习题 5	77

第 6 章 数 组

6.1 问题的提出	79
6.2 一维数组的定义和引用	79
6.2.1 一维数组的定义	79
6.2.2 一位数组元素的引用	80
6.2.3 一维数组的初始化	81
6.2.4 程序举例	81
6.3 二维数组的定义和引用	84
6.3.1 二维数组的定义	84
6.3.2 二维数组的引用	84
6.3.3 二维数组的初始化	85
6.3.4 程序举例	85





6.4 字符数组	87
6.4.1 字符数组的定义	87
6.4.2 字符数组的初始化	87
6.4.3 字符串和字符串结束标志	88
6.4.4 字符数组的输入和输出	88
6.4.5 字符串常用函数	90
6.4.6 字符数组举例	92
习题 6	92

第 7 章 函 数

7.1 问题	94
7.2 函数概述	94
7.3 函数的定义	96
7.3.1 函数的定义形式	96
7.3.2 函数的参数	100
7.3.3 函数的调用及声明	103
7.3.4 函数的返回与返回值	105
7.4 函数的参数传递方式	107
7.5 函数的嵌套与递归	110
7.5.1 函数的嵌套	110
7.5.2 递归	112
7.5.3 递归问题求解方法	113
7.6 变量的作用域	118
7.6.1 局部变量	118
7.6.2 全局变量	119
7.6.3 自动变量	121
7.6.4 静态变量	121
7.6.5 外部变量	124
7.6.6 寄存器变量	126
7.7 内部函数和外部函数	127
7.7.1 内部函数	127
7.7.2 外部函数	127
习题 7	129

第 8 章 编译预处理

8.1 宏定义	132
8.1.1 带有参数的宏定义	132



8.1.2 不带参数的宏定义	136
8.2 文件包含	138
8.3 条件编译	139
习题8	141

第9章 指针

9.1 指针的概念	143
9.2 变量的指针和指向变量的指针变量	144
9.2.1 定义一个指针变量	144
9.2.2 指针变量的引用	144
9.2.3 存储器申请与释放	147
9.3 指针和函数	148
9.3.1 指针变量作为函数参数	148
9.3.2 返回指针值的函数	152
9.3.3 指向函数的指针变量	153
9.4 指针与数组	155
9.4.1 指向数组元素的指针	155
9.4.2 通过指针引用数组元素	156
9.4.3 用数组名作函数参数	158
9.4.4 多维数组与指针	165
9.5 字符串与指针	168
9.5.1 字符串的表现形式	168
9.5.2 字符串指针作函数参数	171
9.5.3 使用字符串指针变量与字符数组的区别	172
9.6 指针数组和指向指针的指针	172
9.6.1 指针数组的概念	172
9.6.2 指针数组与二维数组指针变量的区别	174
9.6.3 指向指针的指针	174
习题9	176

第10章 结构体与共用体

10.1 结构体	177
10.1.1 结构体的定义	177
10.1.2 结构体变量	178
10.1.3 结构体变量的引用	179
10.1.4 结构体数组与结构指针	180
10.1.5 结构体变量作为函数的参数	183





10.2 链表	183
10.2.1 链表及其实现	183
10.2.2 内存分配函数和回收函数	184
10.2.3 用指针和结构体构成链表	185
10.2.4 对链表的删除操作	188
10.2.5 对链表的插入操作	190
10.3 共用体	192
10.3.1 共用体类型及共用体变量的定义	192
10.3.2 共用体变量的引用	193
10.3.3 共用体类型数据在内存中的存储	194
10.4 枚举类型	194
10.4.1 枚举类型的定义	195
10.4.2 枚举类型变量的定义和使用	195
10.5 typedef 的应用	196
10.5.1 用 typedef 定义基本类型	196
10.5.2 用 typedef 定义构造类型	196
习题 10	196

第 11 章 位运算

11.1 位运算的含义和应用	198
11.2 简单的位运算	199
习题 11	204

第 12 章 文件

12.1 文件概述	205
12.1.1 文件的概念	205
12.1.2 文件的分类	205
12.1.3 文件类型指针	205
12.1.4 数据文件	206
12.1.5 缓冲文件系统	207
12.1.6 C 语言的设备文件	207
12.2 文件的处理过程	208
12.3 文件的打开与关闭	208
12.3.1 文件的打开	208
12.3.2 文件的关闭	210
12.4 文件的读写操作	211
12.4.1 文件的字符输入/输出函数	211



12.4.2 文件的字符串输入/输出函数	212
12.4.3 文件的格式化输入输出函数	213
12.5 文件的定位与随机读写	214
12.5.1 文件指针定位函数	215
12.5.2 文件随机读写函数	217
12.6 文件状态检查函数	219
12.6.1 文件状态检测函数 feof()	219
12.6.2 报告文件操作错误状态函数 ferror()	219
12.6.3 清除错误标志函数 clearer()	220
习题 12	221
 第 13 章 C 语言的发展现状	
13.1 面向对象及可视化	223
13.1.1 面向对象的一些概念	223
13.1.2 可视化	224
13.2 C++、VC++	224
13.2.1 几个 C++ 的平台	224
13.2.2 VC++	225
13.3 Java 及 C#	226
13.3.1 Java	226
13.3.2 C#	227
 附录	
附录 1 ASCII 码表	228
附录 2 C 语言中的关键字及用途	229
附录 3 运算符、运算符优先级及结合性一览表	230
附录 4 预处理命令的使用	231
附录 5 C 语言编译错误信息	235
附录 6 Turbo C 集成环境下调试 C 程序的方法	239
附录 7 VC++ 集成环境下调试标准 C 程序的方法	247
附录 8 常用库函数介绍	250
 参考文献	 256



第 1 章 C 语言概述

C 语言系列是一种常用的程序设计语言,它具有丰富的运算符和表达式、先进的控件结构和数据结构。C 语言具有表达能力强,编译目标文件质量高,语言简单灵活,容易移植及容易实现等优点。

1.1 C 语言概况

1.1.1 C 语言的发展过程

C 语言是美国贝尔实验室的 Dennis Ritchie 在 1972 设计的,当时,他正和 Ken Thompson 合作设计 UNIX 操作系统。用的是汇编语言,Thompson 为了摆脱汇编语言的困扰,在 1970 年决定开发一种高级语言,以便能更有效地描述 UNIX,他以 BCPL 语言为基础开发了一种新的语言——B 语言。但 B 语言有一定的缺陷,因而未能流行。为了改进,从 1971 年开始,Dennis Ritchie 用了一年左右的时间,在 B 语言的基础上改进,形成了 C 语言。

1978 年,Brian W. Kernighan、Ken Thompson、Dennis Ritchie 3 人合作,写了一本著名的书——《The C Programming Language》,该书介绍的 C 语言被称为标准 C。

1988 年,美国国家标准化协会(ANSI)公布了标准 ANSI C。1989 年,国际标准化组织(ISO)也采用了 ANSI C 标准,称为 ANSI/ISO Standard C。1994 年,ISO 修订了标准,称之为 ISO C。

C 语言由于其自身的优点,先后被移植到各种计算机平台上,得到了广泛的使用,出现了很多的编译系统版本。比较著名的有 MS C、Turbo C、Quick C、Borland C 等。

在 C 的基础上,又发展了 C++、Visual C++、Java、C# 等,这将在第 13 章简要介绍。

1.1.2 C 语言的特点

简单地讲,C 语言的优点是简洁、灵活、表达能力强、产生的目标代码质量高、可移植性好;具体讲,C 语言有以下几个特点:

(1) C 语言同时具备了高级语言和低级语言的特征。高级语言应该具备的优点 C 语言都具备;同时 C 语言还提供了某些接近于汇编程序的功能,如地址处理、位运算以及指定用寄存器存放变量等低级语言的功能。因此,有人称 C 语言为中级语言。

(2) C 语言是结构化程序设计语言,具有结构化程序设计所要求的控制语句,如条件语句、循环语句等。

(3) C 语言支持模块化程序设计。C 语言的程序是由函数构成的,每个函数可以单独



编写和调试。因此,遇到大量程序,程序员们可以分别编写不同的模块,这使得开发、调试和管理工作变得简单、方便,并且可以实现软件重用。

(4) C 语言具有丰富的数据。C 语言支持各种高级语言普遍使用的基本数据类型,并允许用基本数据类型构造复杂数据类型。

(5) C 语言简练、紧凑、使用方便、灵活,关键字少。C 语言一共只有 34 个关键字,9 个控制语句。程序编写形式自由,支持大小写敏感。使得程序编写者具有很大的发挥空间。

(6) C 语言运算符丰富、功能强大。C 语言一共有 34 种运算符,将括号、赋值、强制类型转换都以运算符形式出现,从而使 C 语言的表现能力和处理能力极强,更容易实现算法。

(7) 数据类型丰富。C 语言的数据类型有基本数据类型和构造数据类型,能够方便地实现各种复杂的数据结构。

(8) C 语言有大量的标准化的库函数。这些库函数不仅包括了各种数学计算的函数,还有用于输入输出的库函数以及系统,给程序员编写程序带来了极大的方便。

(9) C 语言生成代码质量高,有良好的可移植性,应用性广泛。

由于 C 语言具有以上的特点,所以在编写 C 程序时要注意以下两点:

(1) C 语言语法灵活,限制不十分严格,所以对 C 程序进行调试相对困难,尤其是初学者。

(2) 由于 C 语言可以直接访问物理地址,所以它具有一定的攻击能力,如果设置错误有可能导致系统的崩溃,因此使用 C 语言时要特别的注意。

1.2 简单的 C 程序

下面给出几个示例,在这几个示例中不要求大家对语句等了解得很清楚,只要大家对 C 程序有个感性认识。

例 1.1 显示“My first C program!”

```
#include "stdio.h"
main()
{
    printf("My first C program! \n");
}
```

这是一个简单的 C 程序。尽管简单,但已经充分说明了 C 程序的基本组成。该程序包括了三大部分,即注释、预处理命令及函数定义。

说明:

(1) 程序开始部分用了注释:夹在“/*”、“*/”中的字符是用于说明程序的功能或目的,是为了告诉读程序的人员相关的信息或提示,使程序易于理解和提示作用,这些注释在程序运行时自动跳过。一个好的程序注释非常重要,应重视使用注释。一个好的程序员必须习惯为程序写出详细而必要的注释,养成良好的编程习惯。注释可以在程序的开头、结尾或程序中的任意一行位置;可以占一行,也可以多行。



(2) 以“#”开头的是预处理命令:这些命令是在编译系统编译代码之前需要由预处理程序处理的语句,有的书称它为文件包含命令。这里是表示将文件 `stdio.h` 包含到程序中来作为程序的一部分。被包含的文件通常是由系统提供的,其扩展名为“.h”,因此也称之为头文件或首部文件。如果使用了系统提供的库函数,一般应在文件的开始用 `#include` 命令,将被调用的库函数信息包含到本文件中。本例中的 `#include "stdio.h"` 是因为调用了标准输入输出库中的 `printf`。需要说明的是:C语言规定如果只对 `scanf` 和 `printf` 这两个函数调用,可以省去对其头文件的包含命令。以后的例题中经常会出现这种情况,请加以注意。`#include`语句后可以用双引号(如“`stdio.h`”),也可用尖括号(如`<stdio.h>`),两者的作用略有不同,这点将在第8章说明。

(3) `main` 函数:函数是C语言的基本组成部分(命令组成函数,函数组成程序);函数是一个单独的程序模块,完成指定的功能;每一个C程序,无论大小,都是由一个或几个函数组成,但一定有且只有一个名为 `main` 的函数。`main` 函数称之为主函数,对于其他的定义函数,一般而言,可以任意命名(但还是有一定的要求,具体将在以后的章节讲解函数的命名)。任何一个C程序都从 `main` 函数开始运行。在实际运用中,`main` 函数常常用于调用其他函数来协助完成工作,被调用的函数可以是系统函数库提供的,也可以是程序编写者自己定义编写的。

(4) 花括号:函数一定要有函数名和函数体。用一对花括号括起来的函数语句构成函数体。

(5) 函数调用:当调用一个函数时,先要给出这个函数的名字,然后考虑与被调用函数的数据交换。在函数之间进行数据交换的一种方法是让调用函数向被调用函数提供一串叫做实参的值。被调用的函数名后面用一对圆括号把这一串实参(实参表)括起来。本例中,`main` 函数不要求任何实参,故用 `()` 表示;同时还调用了用于打印的格式化输出库函数 `printf`。`printf` 是系统提供的一个用于打印的格式化输出库函数,在本例中,它用于在屏幕上输出“My first C program!”(不包括双引号)。用双引号括起来的字符序列叫字符串。本例是以字符串作为 `printf` 函数的实参,其中的“\n”表示输出后下一个输出点将在下一行的开始,也就是说“\n”是换行符。这个函数的其他说明在后面的章节中逐一介绍。

(6) 分号“;”:C程序中的命令由语句和分号构成,分号表示一个命令的结束。

(7) C程序的书写可以采用自由格式,但一个好的程序员要有好的书写习惯,如本例花括号中的语言的书写格式是向右缩进了两个字符,这样一个函数的结构在视觉上非常清楚。另外C程序中的关键字及特定字要用小写字母(本例中 `main` 是关键字,`include` 是特定字)。

例 1.2 计算: $s = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5}$ 。

```
#include "stdio.h"
main()
{
    /* 定义整型变量 */
    int sum;
    /* 计算累加和,将计算结果赋给变量 */
    sum = 1 + 1 / 2 + 1 / 3 + 1 / 4 + 1 / 5;
```

