



附光盘
CD-ROM

「教程」实验「练习」

C语言程序设计 教程、实验与练习

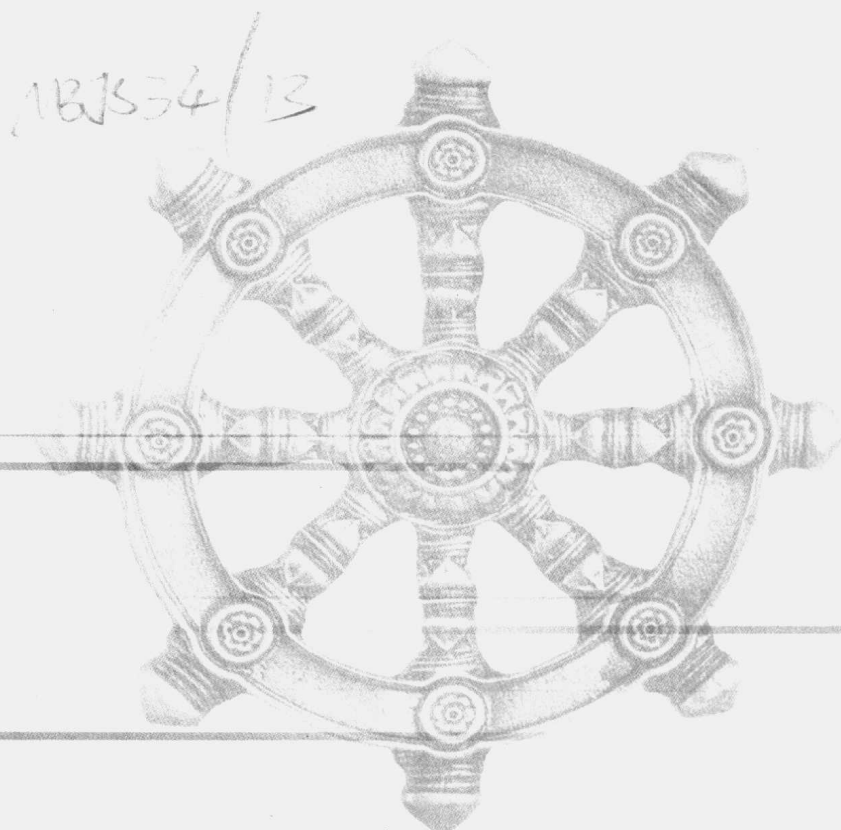
「杨开城 张志坤 编著」

人民邮电出版社
POSTS & TELECOMMUNICATIONS PRESS

教程
实验
练习

C语言程序设计 教程、实验与练习

杨开城 张志坤 编著



人民邮电出版社

图书在版编目(CIP)数据

C 语言程序设计教程、实验与练习/杨开城, 张志坤编著. —北京: 人民邮电出版社, 2002.8
新世纪高等学校计算机系列教材

ISBN 7-115-10478-6

I. C... II. ①杨... ②张... III. C 语言—程序设计—高等学校—教材 IV.TP312

中国版本图书馆 CIP 数据核字 (2002) 第 054125 号

内 容 提 要

本书结合了大量的应用实例和图表, 全面深入地介绍了有关 C 语言编程的基础知识, 包括 C 语言程序的基本构成, C 语言的基本数据类型和构造数据类型 (数组、指针、结构体、共用体、位域和枚举) 的使用方法, 各种运算符的用法, 程序控制语句的用法, 预处理命令的用法, 常见标准库函数 (格式化输入输出、屏幕定位彩色输出、键盘读取、字符串操作、文件操作、内存分配与释放等) 的用法等。为了培养读者的学习兴趣, 在每章的实验指导部分, 针对本章节讲述的内容, 设计了一些有趣的游戏类或界面类实验题目。在每章内, 还提供了丰富的精心设计的练习, 并在章末给出了答案。

本书适用于计算机专业的本科生、研究生、大专生、专升本的学生使用, 也可以作为非计算机专业的学生的教学参考书。

C 语言程序设计教程、实验与练习

- ◆ 编 著 杨开城 张志坤
责任编辑 王文娟
- ◆ 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>
读者热线 010-67180876
北京汉魂图文设计有限公司制作
北京鸿佳印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 28.75
字数: 697 千字 2002 年 8 月第 1 版
印数: 1-6 000 册 2002 年 8 月北京第 1 次印刷

ISBN 7-115-10478-6/TP · 2994

定价: 42.00 元

本书如有印装质量问题, 请与本社联系 电话: (010)67129223

序 言

一、编写初衷

在撰写本书之前，我们正在从事“C 语言程序设计”这门课的教学改革研究。在研究过程中，我们发现目前现有的大部分教材不能完全满足我们教学改革的需要。这主要表现在：

1. 内容传递的逻辑顺序存在问题。C 语言是具有良好的结构的知识体系，但语法要素之间的联系又存在一定的结构不良的特性。因此，要想科学地安排哪些知识单元先学、哪些知识单元后学，必须将线性的学习顺序和螺旋的学习顺序恰如其分地结合起来。目前有些教材中，存在着在前面章节中使用后面章节才学习的概念和术语，而且不给出详细的解释这样的问题，导致学习者学习起来有一定的障碍。

2. 个别原理讲述模糊甚至错误。最明显的是 C 函数参数的值传递原则。有些教材认为指针型参数能够做到“改变形参可以改变实参”，这是一个严重的错误。这种说法给读者理解指针参数的真正含义带来很大的障碍。

3. 只强调讲述 C 语言语法，不重视编程能力培养。目前大多数版本的教材只罗列了 C 语言的语法和应用实例，这本身只是帮助学习者理解 C 语言的概念和规则，不等于能力培养。C 语言的运用能力不是指在纸上写程序的能力，而是指在编程环境中编写、调试程序的能力，尤其是调试能力的培养，一直被很多 C 语言教材所忽略。

4. 利用数学问题作为编程内容，降低了学习者的学习兴趣。目前大多数版本的教材都将数学问题作为编程练习的主要内容，这不仅对学生的数学能力有一定要求，也不是大多数学生工作后实际编程所面临的问题类型。并且实践经验告诉我们，将数学问题作为主要的编程练习，会降低学习者对 C 语言的兴趣。

5. 只提供规则使用的正例，而不提供规则使用的反例。C 语言中的知识类型主要是概念和规则，规则的教学应提供应用实例，并且正例和反例（即错误的用法）都应该提供。C 语言学习的主要障碍之一是对学习者头脑中错误的语言规则应用模式的克服，提供 C 语言规则的应用反例，会帮助学习者能够事先鉴别出常见的错误应用模式。

6. 只有学科内容，没有对学生有帮助的学习建议。目前大多数教材都只关注学科内容的传递，并不关心学生应如何更快速有效地学习 C 语言。

事实表明，高等学校计算机专业的“C 语言程序设计”课程急需一项将现有教材有效成分按照现代教学设计观念进行整合的工作。这也是我们撰写这本教材的原因。现代教育技术观认为，一部好的教材应该是一个完整的学科能力培养方案，而不仅仅是知识单元的堆砌。我们试图在新的教材中克服上面列举的不足，但由于水平有限，本教材肯定存在缺点和不足，热切期望得到同行、专家和读者的批评指正。

二、本书的特色

长期以来，人们都认为，C 语言的学习就是 C 语言语法的学习。但实际上，本课程的教学目标不仅仅是学习语法，还包含着库函数的学习。C 语言包含有上百个库函数，在一个学期内，将这些函数都学会是不可能的。精心选择学习哪些库函数是非常必要的，选择的依据是 C 语言实验的需要。因此，实验设计应该是本课程的主要内容之一，也是一本 C 语言教材不可缺少的部分。

本教材将 C 语言的学习分为 10 章，每一章都指定了教学目标。大部分章内的练习都被切分成更小的逻辑单元穿插在章节内容之间，目的是让学习者及时巩固所学知识。

本书提供的练习题进行了难度区分，不带星号的难度最低，带一个星号的难度要高一点，带两个星号的难度最高。每一章的最后都提供了大部分练习题的答案。学习者可以通过题目难度信息和自己练习的情况大致推断自己 C 语言“纸上谈兵”的水平。

如果说做练习是纸上谈兵，那么做实验就相当于野地实战了。每一章都有实验的具体任务和实验指导。为了培养学生的编程能力，前面几章重点介绍 Borland C 编辑器的使用和 Borland C 调试程序的手段，并设计了相应的实验内容来训练调试技巧。

为了增加学习者的兴趣，本教材选择了一些有趣的游戏软件和界面丰富的软件，作为推荐的实验内容。但为了降低实验的编程难度，对于稍难一点的实验，本书所附的光盘提供了部分实现，并且实验指导中有详细的编程提示。

一般情况下，每一章都有多个推荐的实验。同练习题的设置相同，不带星号的实验题目是最简单的题目，带一个星号的题目是稍难一点的，带两个星号的题目是难度最高的。在实验内容的选择方面，本教材表现出了一定的连续性，同一个实验具有不同难度等级的版本，做出了低等级的版本后，可以继续编写同一个软件的高等级版本。

在每个章节的讲解方面，本教材尽量采用图表的方式解释概念、规则和程序运行结果。这样可以调用读者右脑积极参与知识的学习，降低教材的阅读难度。

为了配合本教材的使用，随教材发行了一张光盘。这张光盘的主要内容包括：

- 各章的例子程序和实验演示程序；
- 各章的电子化的自主练习题，双击“C 语言课后习题.exe”就可以进入自我练习。

总体来说，本教材具有以下几方面的特性：

- 将 C 语言知识的学习和编程能力的培养有机地结合起来；
- 对内容的逻辑顺序进行了合理安排；
- 大量使用图表来阐述知识内容；
- 每一章都为学习者提供了学习建议；
- 教材中整合了实验设计；
- 实验具有难度等级差别；
- 同一个实验具有难度等级不同的版本供学生选择；

- 选择游戏软件或界面丰富的软件作为推荐的实验题目；
- 为实验提供了丰富的编程提示。

三、适用的读者

本教材适用于计算机专业的本科生、研究生以及计算机爱好者使用。对于软件学院的学生，本书深入浅出的讲解方式，也很适合非计算机专业的学生迅速、深入地掌握 C 语言的精髓。本书读者至少应学习过类似于“计算机基础”这样的前导课程。

四、写给学习者的话

学习任何一种知识都会经历两个阶段：意义获取和能力生成。意义获取就是通常所说的“理解”；能力生成就是指将知识转化为通过应用知识解决实际问题的能力的过程。理解是将知识转化成能力的基础。C 语言的学习也是一样。

学习者首先要理解教材中讲解的内容，然后必须将自己理解的知识付诸实践。建议读者按照下面的顺序组织自己的学习活动：

1. 阅读教材和听教师讲解教材，特别要认真研读教材中的例子程序。对于前几章的例子程序，学习者最好亲手输入到 Borland C 中检验。后面章节中的例子程序可以直接从附盘中获取。
2. 完成每一章的学习建议中的内容。
3. 完成各章节的练习。
4. 在理解例子基础上，改写例子程序，可以将例子程序改得复杂一些，也可以将例子改得简炼一些。
5. 重新编写例子程序。从例子的功能出发，重新编写整个例子程序。然后将自己的程序与例子程序进行对比。
6. 根据自己的能力水平，选择合适实验程序。在上机之前，最好先在纸上写出程序，然后再输入到计算机中。
7. 复习一章的内容时，列出这一章主要内容的提纲或示意图。

五、出版说明

本书的编者为这本教材的撰写花了很多心血，杨开城先生设计了所有的实验题目，实现了光盘发行的实验程序，撰写了所有章节的大部分内容。张志坤先生录入了所有的练习题目，并制作了光盘上的电子练习题。为了使我们的劳动成果能够出版发行，人民邮电出版社的王文娟女士和各位领导为我们提供了很大的帮助，在这里我们衷心感谢他们对我们的信任以及为我们提供的机会和支持。

本书作者电子信箱：yangkc@www.elec.bnu.edu.cn

本书责任编辑电子信箱：wangwenjuan@ptpress.com.cn

欢迎广大读者和我们交流。

目 录

第 1 章 C 语言程序设计基础	1
1.1 计算机的工作原理简介	2
1.2 C 语言的发展历史及其基本特性	3
1.3 C 程序的基本结构	4
1.4 编制 C 语言程序的基本步骤	8
1.5 计算机内部数值的表示及运算	9
1.5.1 二进制、八进制和十六进制的概念	9
1.5.2 二进制数的算术运算	10
1.5.3 二进制、八进制、十六进制和十进制数之间的转换	11
1.5.4 二进制数的位运算	13
1.5.5 计算机中的数据单位	13
◆练习	14
1.6 实验指导	15
1.6.1 Borland C++3.1 的安装	15
1.6.2 Borland C++3.1 编辑器的使用	17
1.7 练习答案	22
第 2 章 基本数据类型、运算符和表达式	23
2.1 C 语言的基本数据类型	25
2.1.1 常量与变量	25
2.1.2 整型数据	25
2.1.3 实型数据	30
2.1.4 字符型数据和字符串常量	32
2.1.5 数据的简单输出	34
◆练习 2-1	36
2.2 C 语言的运算符和表达式	39
2.2.1 赋值运算符和赋值表达式	40
2.2.2 强制类型转换符	40
2.2.3 算术运算符、算术表达式	42
2.2.4 自增自减运算符和负号运算符	44
2.2.5 算术运算中的数据类型转换规则	45
2.2.6 位运算符、位运算表达式	46
2.2.7 逗号运算符和逗号表达式	47

2.2.8 sizeof 运算符和复合赋值运算符	48
2.3 运算符的优先级和结合性	48
2.4 小结及常见错误列举	49
◆练习 2-2	53
2.5 实验指导	58
2.5.1 完整的编程操作	58
2.5.2 修改错误程序	59
2.6 练习答案	61
第 3 章 基本输入输出和顺序程序设计	63
3.1 C 程序中的数据输出	64
3.1.1 整数的输出	65
3.1.2 实数的输出	68
3.1.3 字符和字符串的输出	69
3.1.4 格式化输出小结	71
3.1.5 屏幕定位及彩色输出	72
◆练习 3-1	78
3.2 C 程序中的数据输入	82
3.2.1 数据的格式化输入	82
3.2.2 字符数据的非格式化输入	86
3.3 顺序程序设计举例	89
3.4 小结及常见错误列举	92
◆练习 3-2	94
3.5 实验指导	96
3.5.1 学会调试程序	96
3.5.2 程序改错	99
3.5.3 编程实验	99
3.6 练习答案	100
第 4 章 程序控制	103
4.1 三种程序控制结构	104
4.2 C 语句小结	106
4.3 关系运算符、逻辑运算符和条件运算符	108
4.3.1 关系运算符和关系表达式	108
4.3.2 逻辑运算符和逻辑表达式	109
4.3.3 条件运算符和条件表达式	110
◆练习 4-1	111
4.4 选择结构的程序设计	116
4.4.1 if 语句	116

4.4.2	switch 语句	122
◆练习 4-2		127
4.5	循环结构的程序设计	133
4.5.1	while 语句	133
4.5.2	do-while 语句	138
4.5.3	for 语句	139
◆练习 4-3		141
4.6	混合控制结构的程序设计	146
4.7	小结及常见错误列举	152
◆练习 4-4		157
4.8	实验指导	159
4.8.1	修改程序	159
4.8.2	编写程序	161
4.9	练习答案	165
第 5 章	数组	171
5.1	一维数组的定义与引用	172
◆练习 5-1		179
5.2	二维数组的定义与引用	183
◆练习 5-2		187
5.3	字符串与数组	191
5.3.1	字符串的本质	191
5.3.2	字符及字符串操作	192
5.3.3	字符串数组	197
◆练习 5-3		200
5.4	小结及常见错误列举	204
◆练习 5-4		208
5.5	实验指导	210
5.5.1	修改程序	210
5.5.2	新编程序	211
5.6	练习答案	218
第 6 章	函数	225
6.1	函数的定义和调用	226
6.1.1	不带参数没有返回值的函数	226
6.1.2	不带参数有返回值的函数	228
6.1.3	带参数没有返回值的函数	230
6.1.4	带参数有返回值的函数	232
◆练习 6-1		234

6.2 变量的作用域和存储类别	240
6.3 函数的嵌套调用和递归调用	248
6.4 函数的作用域	252
◆练习 6-2	253
6.5 利用工程管理多个源程序文件	256
6.6 小结及常见错误列举	259
6.7 实验指导	264
6.7.1 图形输出	264
6.7.2 编制程序	267
6.8 练习答案	271
第 7 章 指针	281
7.1 指针变量的定义与引用	282
7.2 指针的运算	285
◆练习 7-1	286
7.3 指针与数组	287
7.4 指针与字符串	292
7.5 指针与内存的动态分配	297
◆练习 7-2	302
7.6 指针与数组作为函数的参数	307
7.7 带参数的 main 函数	313
7.8 指针作为函数的返回值	315
7.9 函数指针的定义与引用	316
◆练习 7-3	319
7.10 小结及常见错误列举	325
7.11 实验指导	329
7.12 练习答案	332
第 8 章 预处理命令	341
8.1 预处理命令简介	342
8.2 宏定义	342
8.2.1 不带参数的宏定义	342
8.2.2 带参数的宏定义	344
8.3 文件包含	346
8.4 条件编译	347
◆练习	350
8.5 练习答案	352
第 9 章 复杂数据类型	353

9.1 复杂数据类型概述	354
9.2 结构体	354
9.2.1 结构体类型的定义	354
9.2.2 结构体变量的定义和引用	355
9.2.3 简化结构体类型名	359
9.2.4 结构体数组	360
◆练习 9-1	362
9.3 线性链表	368
9.4 共用体	372
9.4.1 共用体类型的定义	372
9.4.2 共用体变量的定义和引用	373
9.5 位域	375
9.6 枚举类型变量的定义和引用	378
9.7 复杂数据类型作为函数参数和返回值	378
◆练习 9-2	381
9.8 小结及常见错误列举	383
9.9 实验指导	386
9.10 练习答案	388
第 10 章 文件操作	399
10.1 文件操作概述	400
10.2 文件指针	400
10.3 文件的打开、读写和关闭	401
10.3.1 文件的打开与关闭	401
10.3.2 文件的读写	402
10.4 文件的定位读写	410
10.5 小结及常见错误列举	412
◆练习	413
10.6 实验指导	417
10.7 练习答案	417
附录一 常见问题解答	419
附录二 常见库函数（按函数名排序）	423
附录三 常见库函数（按函数类别排序）	429
附录四 C 语言中常用的专业术语对译	435
附录五 C 语言中的关键字（保留字）	441

附录六 运算符和结合性	443
附录七 ASCII 码表	445

第 1 章

C 语言程序设计基础

教学目标

- ◆ 了解计算机的工作原理;
- ◆ 理解程序、内存、输入输出设备、CPU、源程序、标识符、函数、参数、主函数、变量、函数调用、程序注释、数制等概念;
- ◆ 知道 C 程序的基本构成;
- ◆ 掌握十进制、二进制、八进制和十六进制之间的相互转换规则和各种进制数的算术运算规则;
- ◆ 掌握上机步骤、Borland C++的启动方法以及 Borland C++编辑器的使用。

学习建议

- ◆ 如果读者从未接触计算机,对计算机的基本原理不了解,会影响 C 语言的学习。为了更容易记住计算机的基本工作原理,建议读者阅读完本章第 1 节后,根据自己对计算机的理解,独自画出计算机的工作原理示意图。
- ◆ 请读者阅读完本章后,列出所有的有关 C 程序的术语,并对照第 3 节内容看看漏掉了哪些术语。
- ◆ Borland C++的安装和编辑器的使用一定要实际操作一遍,“实验指导”中列出了许多有关编辑器的键盘命令,建议读者尽量多地学会使用,这会使你的 C 语言实验效率倍增。

1.1 计算机的工作原理简介

计算机是一种进行计算和逻辑判断的设备，但处理计算和逻辑判断任务的速度是人所不能比的。无论计算机进行计算还是进行逻辑判断，它都要处理数据。这些数据是从外部输入的。当计算机对这些数据进行计算或逻辑判断等处理后，还要将处理完的数据输出。因此，简单地说，计算机就是一个数据的输入—处理—输出的设备。图 1-1 所展示的计算机外围结构中，扫描仪、鼠标、键盘都是负责输入数据的，被称为输入设备；打印机、显示器都是负责输出数据的，被称为输出设备；集线器、调制解调器既可以输入数据，也可以输出数据，它们可以将单个计算机与计算机网络相连接。

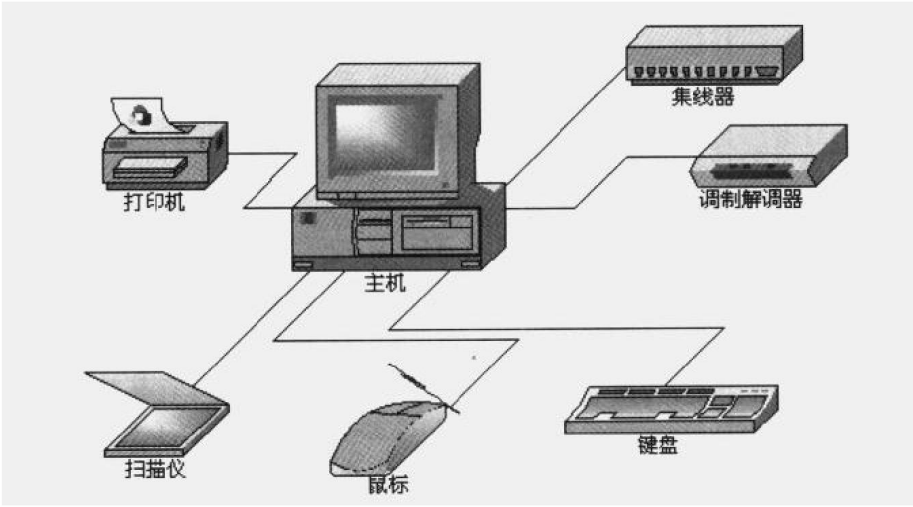


图 1-1 计算机设备示意图

计算机对数据的处理依靠两个系统来完成：硬件系统和软件系统。硬件系统是计算机执行输入、输出操作以及对数据进行处理的硬件设备和单元。图 1-1 中所列出的各种设备都是硬件。但计算机的硬件是在软件的指挥下工作的，没有了软件，计算机的硬件设备将无法工作。软件就是指挥硬件进行具体操作的指令集，这些指令集是以程序为单元进行组织的。图 1-2 是计算机内部逻辑结构图。输入输出接口负责将数据输入到内存中以及将内存中的数据输出，内存是一种计算机内部存储数据的硬件设备，由多个存储单元构成。每个存储单元都能存放一个数据。这些存储单元是连续存放的，并且被编了号，这个号就是内存单元的地址。计算机的其他硬件单元（比如 CPU）将依据内存地址在内存中的特定单元存取数据。计算机所要处理的数据必须事先位于内存中才能被处理。中央处理单元 CPU 负责管理其他硬件单元的工作，算术逻辑单元 ALU 负责完成计算和逻辑判断的工作。CPU 通过控制总线向其他硬件单元传递控制指令，通过地址总线来选择要读取或写入内存单元。计算机中的所有数据都通过数据总线进行交换。

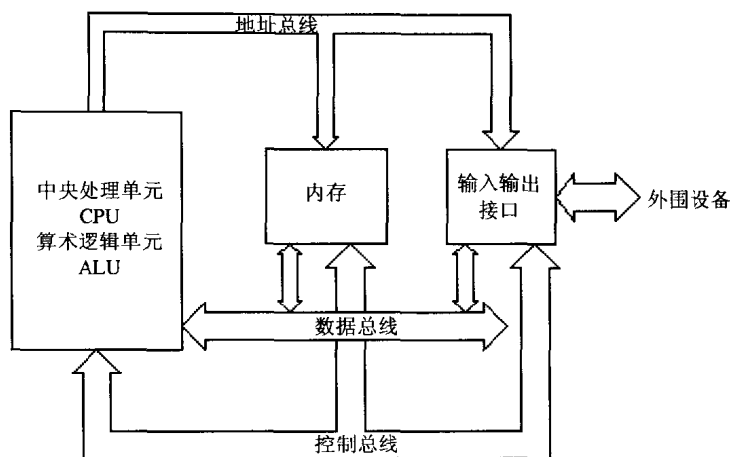


图 1-2 计算机的逻辑结构

软件系统是各种程序的集合，一个程序基本上对应一个可执行的文件。软件系统中，有一类程序专门与计算机硬件直接打交道，负责管理计算机硬件资源，这类程序被称为操作系统，属于系统软件。而其他程序必须在操作系统的基础上运行，这类程序被称为应用软件。初学 C 语言的读者主要任务是学习如何编写应用软件。

综上所述，我们可以这样理解计算机系统：

- 计算机系统是一个输入—处理—输出数据的系统；
- 计算机系统由软件系统和硬件系统组成，软件指挥硬件完成工作；
- 软件系统是程序的集合；
- 一个程序对应一个可执行文件，是计算机指令的集合；
- 程序是人类控制计算机的手段；
- 程序是计算机进行数据处理的指令集。

1.2 C 语言的发展历史及其基本特性

为了使人类能方便地控制计算机的运行，人们发明了各种计算机程序的编制工具，计算机语言就是其中的一种。利用计算机语言编制程序的人被称为程序员。最先，程序员只能利用机器语言（即直接的机器指令）来编制程序。这样的程序难懂、易出错，只能局限于少量科学家阅读和使用。为了降低程序编制和维护的难度，人们又发明了汇编语言，利用特定的助记符来帮助程序员记忆机器指令。但利用汇编语言编制的程序通常不能是大规模的，在软件的开发和维护方面，汇编语言仍表现出严重地不适应性。由于机器语言和汇编语言都是直接面向机器的，与人们使用的自然语言有很大的区别，一般称为低级语言。

随着计算机语言的发展，人们发明了各种能直接表达计算式和逻辑式的语言，比如，Pascal、Basic、C/C++、COBOL 等，这些语言被称为高级语言。C 语言是其中比较成功的一种语种。

C 语言是从 BCPL 语言和 B 语言演化而来的。1967 年，Martin Richards 开发出了 BCPL

(Basic Compound Programming Language) 语言, 其目的是为了编写操作系统和编译器。1970 年 Ken Thompson 在 BCPL 语言的基础上开发出了 B 语言, 并在 DEC PDP-7 计算机上利用 B 语言实现了第一个 UNIX 操作系统。

C 语言是贝尔实验室的 Dennis Ritchie 在 B 语言的基础上开发出来的, 并逐渐被用于 UNIX 操作系统的系统软件和应用软件的开发。事实上, 当今许多重要的操作系统和应用软件都是利用 C/C++ 语言来实现的。

随着 C 语言在各种计算机上的迅速推广, 出现了许多不同的 C 语言版本。这些版本之间通常是不兼容的。这种情况给程序员开发跨平台的程序带来了极大的障碍。为此, 1988 年, 美国国家标准化协会制定了 C 语言的标准, 这个标准称为 ANSI 标准或 ANSI C。本书的内容将基于 ANSI C 进行展开, 但实验设计基于 Borland C++ 3.1 环境。

C 语言之所以能够生存和发展到今天, 完全依赖于不同于其他语种的独特优势。简单地说, C 语言突出的特性包括:

- 语言简洁、紧凑, 使用方便、灵活。C 语言一共只有 30 多个关键字, 9 种控制语句。由于语法限制不严, 在实现强大功能的同时, 程序书写极具美感。
- 语言表述功能强大。C 语言中的运算符丰富, 既有简单数据类型, 又可以定义复杂数据类型, 还允许直接访问内存的物理地址, 进行位操作和强制类型转换。
- 程序的可移植性好。在一种计算机上编写的程序, 无需修改或经过很少的修改, 就可以在其他类型的计算机上运行。
- C 语言提供了预处理器, 程序可以利用宏指令提高可读性和可移植性。
- C 语言允许递归调用, 在解决递归问题上具有独特优势。

1.3 C 程序的基本结构

这一节的主要任务是了解 C 程序的基本结构, 即 C 程序是什么样子的。一个 C 程序可以是非常简单的, 也可以是特别复杂的, 这取决于程序所要实现的功能。我们先来认识一个最简单的 C 程序——程序 1-1, 如图 1-3 所示。

- 程序 1-1 的运行结果是在计算机屏幕上显示这样一行文本: **Welcome to C!**。
- 程序的第一行是注释信息。C 程序中, 注释是程序员为了增加程序的可读性和易懂性, 人为增加的说明性信息。主要用来说明程序的功能、用途、符号的含义以及程序的逻辑等。注释由 “/*” 开始, 由 “*/” 结束。在 “/*” 和 “*/” 中间放置注释的内容。注释信息不影响程序的功能。
- 从 main 开始到最后的 “}” 之间称为函数, 确切地说, 是函数定义, main 是函数名。每一个函数都有自己的名字, 并且不能与其他函数同名。
- C 程序中, 函数代表一小段可重复执行的计算机指令, 执行一个函数称为函数调用。函数可以带参数, 也可以不带参数。
- main 的后面跟着空的 (), 表明 main 函数没有参数。main () 后面的 {} 中是函数的具体内容, 称为函数体。
- main 函数中只有一行, 这一行是一个函数调用。printf 也是一个函数名, 因为它后面

跟着 ()。这一行 printf 函数的功能是将文本 “Welcome to C!” 显示在计算机的屏幕上。文本 “Welcome to C!” 被放在了一对双引号中间了。被双引号括起来的文本称为字符串。

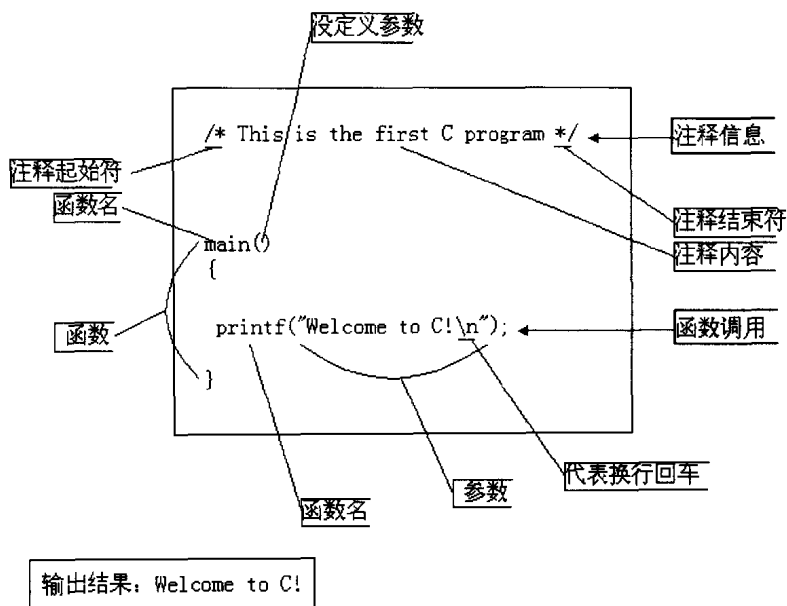


图 1-3 程序 1-1

➤ 在 `printf ("Welcome to C!");` 中, 字符串 “Welcome to C!” 被称为函数 `printf` 的参数。

程序 1-1 非常简单, 它所体现的概念主要是注释、函数、函数调用和参数。下面我们来认识一个稍微复杂一点的程序——程序 1-2, 如图 1-4 所示。

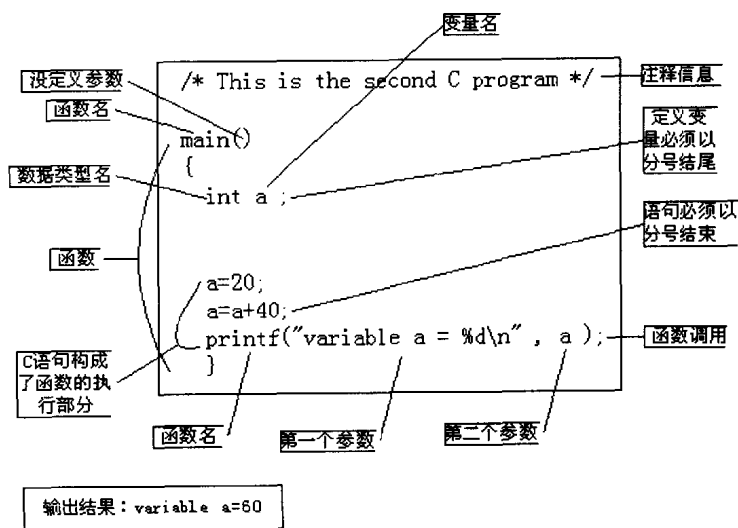


图 1-4 程序 1-2

程序 1-2 与程序 1-1 相比, 多了三行文本, `printf` 函数的调用也复杂了一些。