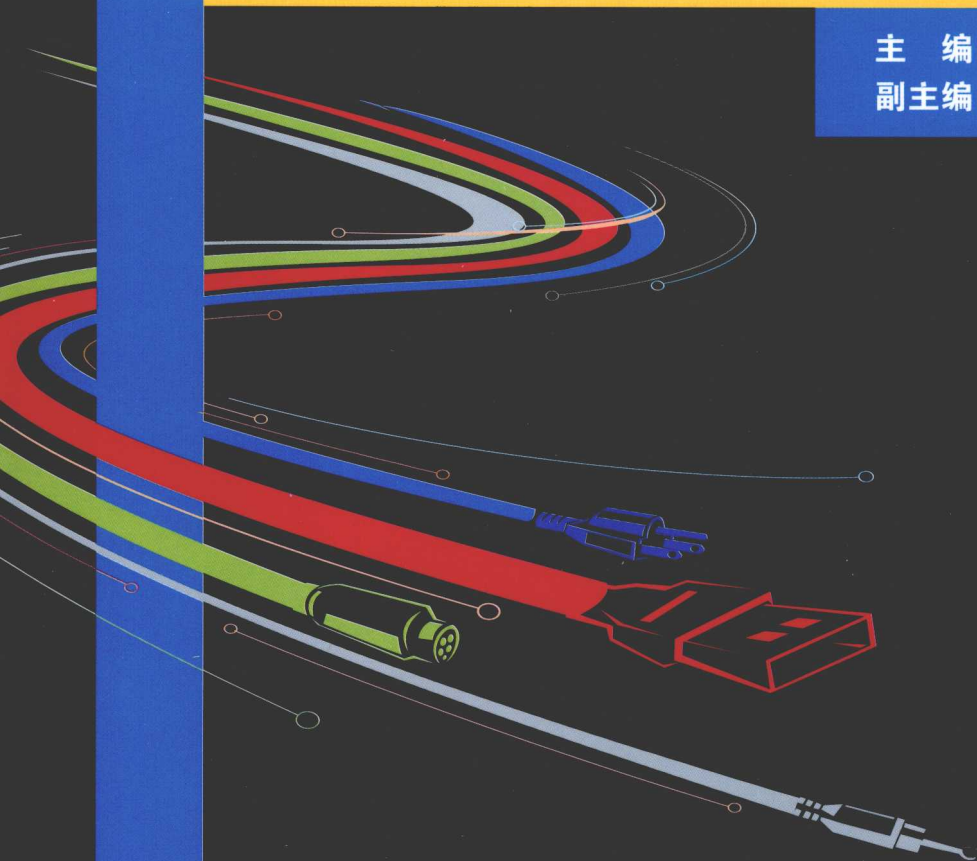


大学计算机基础与应用系列立体化教材

C程序设计教程

（面向经管类）

主 编 李 俊
副主编 于会萍



中国人民大学出版社

大学计算机基础与应用系列立体化教材

C程序设计教程

（面向经管类）

主 编 李 俊
副主编 于会萍

中国人民大学出版社
· 北 京 ·

图书在版编目 (CIP) 数据

C 程序设计教程 (面向经管类) /李俊主编.
北京: 中国人民大学出版社, 2009
(大学计算机基础与应用系列立体化教材)
ISBN 978-7-300-11593-1

- I. ①C...
- II. ①李...
- III. ①C 语言-程序设计-高等学校-教材
- IV. ①TP312

中国版本图书馆 CIP 数据核字 (2009) 第 231054 号

大学计算机基础与应用系列立体化教材
C 程序设计教程 (面向经管类)
主 编 李 俊
副主编 于会萍
C Chengxu Sheji Jiaocheng

出版发行	中国人民大学出版社		
社 址	北京中关村大街 31 号	邮政编码	100080
电 话	010-62511242 (总编室)		010-62511398 (质管部)
	010-82501766 (邮购部)		010-62514148 (门市部)
	010-62515195 (发行公司)		010-62515275 (盗版举报)
网 址	http://www.crup.com.cn		
	http://www.ttrnet.com(人大教研网)		
经 销	新华书店		
印 刷	北京鑫丰华彩印有限公司	版 次	2010 年 1 月第 1 版
规 格	185 mm×260 mm 16 开本	印 次	2010 年 1 月第 1 次印刷
印 张	22.25 插页 1	定 价	35.00 元
字 数	496 000		

版权所有 侵权必究 印装差错 负责调换

内容简介

本书由浅入深、循序渐进地介绍了C语言程序设计的思路和方法，并通过趣味性强精彩案例介绍了C语言中的知识点，如猜数游戏、验证哥德巴赫猜想、文件加密等，从而提高读者学习的兴趣，培养读者的自主学习能力和独立思考能力。全书共分为11章，系统地介绍了C语言开发环境、数据类型、运算符、表达式、流程控制、模块化程序设计方法和文件系统的基本操作。

本书内容翔实、案例新颖、结构清晰、趣味性强，以丰富有趣的案例驱动知识点教学。本书可作为高等院校计算机程序设计类课程的教材，也可作为计算机程序设计培训教材和各种计算机等级考试的参考教材。

总序

随着计算机与互联网应用的普及、信息技术的发展及中小学对信息技术基础课程的普遍开设,针对大学计算机基础与应用教育的方向和重点,我们认为应该研究新的教育与教学模式,使得计算机基础与应用课程摆脱传统的“课堂上课+课后上机”这种简单、低效的教学方式,逐步转向以实践性教学和互动式教学为手段,利用现代化的计算机实现辅助教学、管理与考核,同时提供包括教材、教辅、教案、习题、实验、网络资源在内的丰富的立体化教学资源 and 实时或在线答疑系统,使得学生乐于学习、易于学习、学有所成效、学有所用,同时减轻教师备课、授课、布置作业与考核、阅卷的工作量,提高教学效率。这是我们建设这套“大学计算机基础与应用系列立体化教材”的初衷。

根据大学非计算机专业学生的社会需求和教育部对计算机基础与应用教育的指导意见,中国人民大学从2005年开始对计算机公共课进行大规模改革,包括增设课程、改革教学方式和考核方式、进行教材建设等多个方面的内容。在最新的《中国人民大学本科生计算机教学指导纲要(2008年版)》中,将与计算机教育有关的内容分为三个层次。第一层次为“计算机应用基础”课程,第二层次为“计算机应用类”课程(包含约10门课程),第三层次纳入专业基础课或专业课教学范畴,形成1+X+Y的计算机基础与应用教育格局。其中,第一层次的“计算机应用基础”课程和第二层次的“计算机应用类”课程,作为分类分层教学中的核心课程,走在教学改革的前列,同时结合中国人民大学计算机教学改革中开展的其他项目,已经形成了教材(部分课程)、教案、教学网站、教学系统、作业系统、考试系统、答疑系统等多层次、立体化的教学资源。同时,部分项目获得了学校、北京市、全国各级教学成果奖励和立项。

为了巩固我们的计算机基础与应用教学改革成果并使其进一步深化,我们认为有必要系统地建立一套更合理的教材,同时将前述各项立体化、多层次的教学资源整合到一起。为此,我们组织中国人民大学、中央财经大学、天津财经大学、河北大学、东华大学、华北电力大学等多所院校中从事计算机基础与应用课程教学的一线骨干教师,共同建设“大学计算机基础与应用系列立体化教材”项目。

本项目对中国人民大学及合作院校的计算机公共课教学改革和课程建设起着非常关键的作用,得到了各校领导和相关部门的大力支持。该项目将在原来的应用教学的基础上,更进一步地加强实践性教学、实验和考核环节,让学生真正地做到学以致用,与信息技术的发展同步成长。

本系列教材覆盖了“计算机应用基础”(第一层次)和“计算机应用类”(第二层次)的十余门课程,包括:

- 大学计算机应用基础

- Internet 应用教程
- 多媒体技术与应用
- 网站设计与开发
- 数据库技术与应用
- 管理信息系统
- Excel 在经济管理中的应用
- 统计数据分析基础教程
- 信息检索与应用
- C 程序设计教程
- 电子商务基础与应用

每门课程均编写了教材和配套的习题与实验指导。

随着信息化技术的发展,许多新的应用不断涌现,同时数字化的网络教学手段也在发展和成熟。我们将为此项目全面、系统地构建立体化的课程与教学资源体系,以方便学生学习、教师备课、师生交流。具体措施如下:

- 教材建设:在教材中减少纯概念性理论的内容,加强案例和实验指导的分量;增加关于最新的信息技术应用的内容并将其系统化,增加互联网和多媒体应用方面的内容;密切跟踪和反映信息技术的新应用,使学生学到的知识马上就可以使用,充分体现“应用”的特点。

- 教辅建设:针对教材内容,精心编制习题与实验指导。每门课程均安排大量针对性很强的实验,充分体现课程的实践性特点。

- 教学视频:针对主要教学要点,我们将逐步录制教学操作视频,使得学生的学习和复习更为方便。

- 电子教案:我们为教师提供电子教案,针对不同专业和不同的课时安排提出合理化的教学备课建议。

- 教学网站:纸质课本容量有限,更多更全面的教学内容可以从我们的教学网站上查阅。同时,新的知识、技巧和经验不断涌现,我们亦将它们及时地更新到教学网站上。

- 教学辅助系统:针对采用本教材的院校,我们开发了教学辅助系统。通过该系统,可以完成课程的教学、作业、实验、测试、答疑、考试等工作,极大地减轻教师的工作量,方便学生的学习和测试,同时网络的交流环境使师生交流答疑更为便利。(对本教学辅助系统有兴趣的院校,可联系 yxd@yxd.cn 了解详情。)

- 自学自测系统:针对个人读者,可以通过我们提供的自学自测系统来了解自己学习的情况,调整学习进度和重点。

- 在线交流与答疑系统:及时为学生答疑解惑,全方位地为学生(读者)服务。

相信本套教材和教学管理系统不仅对参与编写的院校的计算机基础与应用教学改革起到促进作用,而且对全国其他高校的计算机教学工作也具有参考和借鉴意义。

杨小平

2009 年 6 月

前言

C语言是目前国际上广泛流行的一种结构化的程序设计语言，它具有高级语言和低级语言的功能，提供类型丰富、使用灵活的基本运算和数据类型，具有较高的可移植性。C语言不仅适合于开发系统软件，而且也是开发应用软件和进行大规模科学计算的常用程序设计语言。

本书由浅入深、循序渐进地介绍了C语言程序设计的思路和方法。全书共分为11章，系统地介绍了C语言开发环境、数据类型、运算符、表达式、流程控制、模块化程序设计方法和文件系统的基本操作。

本书内容翔实，语言简明扼要、重点突出，案例新颖、趣味性强、结构清晰、可操作性强。

本书具有如下主要特点：

1. 案例新颖、趣味性强

教材中的每个案例都由作者精心设计，趣味性较强。通过这些案例，不仅可以提高读者学习的兴趣，而且也可以使读者对所学知识点达到举一反三的效果，从而使得读者更深刻地理解所学的知识。

2. 通过综合案例融合知识点

很多C语言教材都是独立地介绍C语言的知识点，这样就会造成读者无法将C语言的各个知识点融为一个整体。为了解决这个问题，本书不仅针对每个知识点都有精彩的案例，每一章还都设计了综合案例，这些综合案例将这一章的知识点以及前面的知识点综合起来，使读者能够直观地将这些知识点融为一体。

3. 利于提高读者分析问题和独立思考问题的能力

读者在学习的过程中，经常会遇到这样的问题：教材的例子能看懂，教师讲的内容也能听明白，就是遇到问题时，自己无从下手。为了解决这个问题，作者在编写每个案例时，都先对案例进行分析，以便培养和提高读者分析问题的能力；然后编写程序，并在程序中给出大量的注释；最后在案例的后面设计了一些思考题，以便培养和提高读者独立思考问题的能力。

4. 内容安排循序渐进、由易到难

本书内容安排循序渐进、由易到难，全书共分为11章。第1章介绍了C语言程序设计的基本知识和开发环境的使用。第2章介绍了C语言的数据类型、运算符与表达式。第3、4、5章介绍了C语言的控制结构。第6章介绍了函数和模块化程序设计的思想。第7、8章介绍了数组和指针的应用。第9章介绍了自定义类型，即结构体、共

用体和枚举类型。第 10 章介绍了位运算。第 11 章介绍了 C 语言的文件操作。

本书作者具有多年的 C 语言程序设计和相关专业课程的教学经验。本书由李俊主编并进行总体设计，于会萍编写第 1 章至第 5 章；李俊编写第 6 章至第 11 章，并负责全书的统稿。

由于作者的水平有限，书中错误和不妥之处在所难免，敬请读者批评指正。

李 俊

2009 年 9 月

目 录

CONTENTS

第 1 章 C 语言程序设计概述	1
1.1 C 语言的发展及特点	1
1.1.1 C 语言的发展	1
1.1.2 C 语言的特点	2
1.2 C 语言程序的基本结构	3
1.3 C 语言字符集、标识符与关键字	4
1.3.1 C 语言字符集	4
1.3.2 C 语言标识符与关键字	5
1.4 C 语言程序的开发环境	6
1.4.1 C 语言程序开发过程	6
1.4.2 TurboC 2.0 介绍	6
1.4.3 Microsoft Visual C++6.0 集成开发环境	9
本章小结	13
习题	14
第 2 章 数据类型、运算符与表达式	15
2.1 C 语言的数据类型	15
2.1.1 数据类型概述	15
2.1.2 整数类型	16
2.1.3 实数类型	16
2.1.4 字符类型	17
2.2 常量与变量	17

2.2.1 常量	17
2.2.2 变量	20
2.3 运算符和表达式	21
2.3.1 算术运算符和算术表达式	22
2.3.2 赋值运算符和赋值表达式	23
2.3.3 关系运算符和关系表达式	25
2.3.4 逻辑运算符与逻辑表达式	27
2.3.5 条件运算符、逗号运算符和求字节运算符	29
2.4 运算符的优先级	31
2.5 数据类型转换	32
2.5.1 自动类型转换	32
2.5.2 强制类型转换	33
本章小结	33
习题	34
第3章 顺序结构	35
3.1 算法	35
3.1.1 算法的概念	35
3.1.2 算法的组成要素	36
3.1.3 算法的描述	37
3.2 C语言基本语句	39
3.3 数据的输入与输出	42
3.3.1 字符数据的输入输出	42
3.3.2 格式化输出函数 printf	44
3.3.3 格式化输入函数 scanf	48
3.4 综合案例	52
本章小结	53
习题	54
第4章 选择结构	56
4.1 if语句	56
4.1.1 单分支 if语句	56
4.1.2 双分支 if语句	58
4.1.3 多分支 if语句	59
4.1.4 if语句的嵌套	61
4.2 条件运算符	63
4.3 switch语句	64
4.4 综合案例	67

本章小结	70
习题	70
第 5 章 循环结构	73
5.1 循环结构算法	73
5.2 while 语句	75
5.3 do-while 语句	76
5.4 for 语句	78
5.5 break 语句和 continue 语句	81
5.5.1 break 语句	81
5.5.2 continue 语句	82
5.6 循环结构的嵌套	83
5.7 综合案例	85
本章小结	89
习题	89
第 6 章 函数与编译预处理	92
6.1 模块化设计	92
6.2 函数的定义与调用	93
6.2.1 标准库函数	93
6.2.2 函数的定义	94
6.2.3 函数的调用	96
6.2.4 参数的传递	99
6.3 函数的递归调用	101
6.4 变量的存储类型、作用域	105
6.4.1 变量的作用域	105
6.4.2 变量的存储类型	108
6.5 编译预处理	114
6.5.1 宏定义	114
6.5.2 文件包含	119
6.5.3 条件编译	119
6.6 综合案例	121
本章小结	124
习题	125
第 7 章 数组	127
7.1 概述	127
7.2 一维数组	128

7.2.1	一维数组的定义	128
7.2.2	一维数组的初始化	129
7.2.3	一维数组的引用	130
7.2.4	一维数组应用	132
7.3	二维数组	136
7.3.1	二维数组的定义	136
7.3.2	二维数组的初始化	136
7.3.3	二维数组的引用	138
7.3.4	二维数组应用	140
7.4	字符数组与字符串	142
7.4.1	字符数组的定义与初始化	142
7.4.2	字符串的概念及存储	143
7.4.3	字符数组的输入输出	144
7.4.4	字符串处理函数	146
7.4.5	字符数组应用	149
7.5	数组作为函数参数	151
7.6	综合案例	154
	本章小结	159
	习题	160
第8章	指针	162
8.1	指针与指针变量	162
8.1.1	指针的概念	162
8.1.2	指针变量的定义与初始化	163
8.1.3	指针运算	165
8.2	指针与数组	169
8.2.1	一维数组的指针表示法	169
8.2.2	二维数组的指针表示法	173
8.3	指针与字符串	177
8.3.1	字符串的指针表示方法	177
8.3.2	字符串数组与指针数组	181
8.4	指针与函数	183
8.4.1	指针作为函数的形参	183
8.4.2	指针型函数	190
8.4.3	指向函数的指针	192
8.5	命令行参数	195
8.6	指向指针的指针变量	196
8.7	综合案例	197

本章小结	203
习题	203
第 9 章 结构体、共用体与枚举类型	205
9.1 结构体类型的定义	205
9.2 结构体变量	207
9.2.1 结构体变量的定义	207
9.2.2 结构体变量的使用	209
9.2.3 结构体变量的初始化	210
9.3 结构体数组	212
9.3.1 结构体数组的定义	212
9.3.2 结构体数组的初始化	213
9.3.3 结构体数组的使用	214
9.4 结构体类型指针	215
9.4.1 指向结构体变量的指针	215
9.4.2 指向结构体数组的指针	218
9.5 结构体与函数	220
9.5.1 结构体变量作为函数参数	220
9.5.2 结构体指针变量作为函数参数	221
9.5.3 函数的返回值为结构体类型	222
9.6 链表	223
9.6.1 链表概述	223
9.6.2 内存管理函数	225
9.6.3 链表的基本操作	227
9.7 共用体类型	234
9.7.1 共用体类型与共用体变量	234
9.7.2 共用体变量的使用	235
9.8 枚举类型	238
9.9 类型定义	242
9.10 综合案例	243
本章小结	248
习题	249
第 10 章 位运算	250
10.1 位操作运算符和位运算	250
10.2 位段	253
10.3 综合案例	255
本章小结	259

习题	259
第 11 章 文件	260
11.1 文件概述	260
11.2 文件的打开与关闭	262
11.2.1 文件指针	262
11.2.2 文件的打开	263
11.2.3 文件的关闭	265
11.3 文件读写函数	266
11.3.1 读写文件字符函数	266
11.3.2 读写文件字符串函数	268
11.3.3 格式化读写	271
11.3.4 块读写	272
11.4 文件定位与随机读写	275
11.5 文件检测函数	278
11.6 综合案例	279
本章小结	283
习题	284
 附录 A C 语言关键字	285
附录 B ASCII 码表	287
附录 C C 语言运算符优先级和结合方向	289
附录 D C 语言库函数	291
附录 E PC I/O 端口的功能	320
附录 F C 语言常见的出错信息	324
附录 G C 语言常用算法	328

C 语言程序设计概述

计算机自从 20 世纪 40 年代诞生以来，无论从硬件方面还是软件方面，都有了极大的发展。硬件的发展使得计算机的运算速度得到了快速的提升。软件的发展丰富了计算机的功能。计算机各个硬件工作的协调是由软件来实现的，而软件都是依靠程序员利用程序设计语言来编写出来的。

在众多的程序设计语言中，C 语言作为一种高级程序设计语言，具备方便性、灵活性和通用性等特点。同时，它还为程序员提供了直接操作硬件的功能，具备低级语言的特点，适合各种类型软件的开发。因此，C 语言是深受程序设计人员欢迎的编程语言。

本章主要介绍 C 语言程序设计的发展及其特点，描述了 C 语言的基本结构、字符集以及 C 语言程序的开发环境等内容。

1.1 C 语言的发展及特点

1.1.1 C 语言的发展

C 语言是国际上广泛流行的、很有发展前途的计算机高级语言，它是一种编译型语言，它的发展是一个逐渐充实和完善的过程。

C 语言是在 B 语言的基础上发展起来的，它的根源可以追溯到 ALGOL60。1960 年出现的 ALGOL60 是一种面向问题的高级语言，它离硬件比较远，不宜用来编写系统程序。1963 年，英国的剑桥大学推出了 CPL (Combined Programming Language) 语言。CPL 语言在 ALGOL60 的基础上更接近硬件，但规模比较大，难以实现。1967 年，英国剑桥大学的马丁·理查斯 (Martin Richards) 对 CPL 语言做了简化，推出了

BCPL (Basic Combined Programming Language) 语言。1970 年, 美国贝尔实验室的汤普森 (Ken Thompson) 以 BCPL 语言为基础, 对其又做了进一步简化, 设计出了很简单的而且很接近硬件的 B 语言 (取 BCPL 的第一个字母), 并用 B 语言编写了第一个 UNIX 操作系统, 在 PDP-7 上实现。1971 年在 PDP-11/20 上实现了 B 语言, 并编写了 UNIX 操作系统。但 B 语言过于简单, 功能有限。1972—1973 年间, 贝尔实验室的里奇 (D. M. Ritchie) 在 B 语言的基础上设计出了 C 语言 (取 BCPL 的第二个字母)。C 语言既保持了 BCPL 和 B 语言的优点 (精练, 接近硬件), 又克服了它们的缺点 (过于简单, 数据无类型等)。最初的 C 语言只是为描述和实现 UNIX 操作系统提供一种工作语言而设计的。1973 年, 汤普森和里奇两人合作把 UNIX 操作系统的 90% 以上用 C 语言改写, 即 UNIX 第 5 版。原来的 UNIX 操作系统是 1969 年由汤普森和里奇开发成功的, 是用汇编语言写的。

随着 UNIX 操作系统的日益广泛使用, C 语言也迅速得到了推广。1978 年以后, C 语言先后移植到大、中、小、微型机上, 而且此时的 C 语言出现了不同的版本, 并以柯奈汉 (Brain W. Kernighan) 和里奇合著的名著《C 程序设计语言》(*The C Programming Language*) 作为 C 语言的标准。1983 年, 美国国家标准协会 (ANSI) 又制定了新的标准, 称为 ANSI C。现在的 C 语言已风靡全世界, 成为世界上使用最广泛的几种计算机语言之一。

1.1.2 C 语言的特点

C 语言之所以能被推广并被广泛使用, 概括地说主要有如下特点:

1. 简洁紧凑、灵活方便

C 语言一共只有 32 个关键字, 9 种控制语句, 程序书写自由, 主要用小写字母表示。它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。

2. 运算符丰富

C 语言的运算符包含的范围很广泛, 共有 34 种运算符。C 语言把括号、赋值、强制类型转换等都作为运算符处理, 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。

3. 数据结构丰富

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、共用体类型等, 能用来实现各种复杂的数据类型的运算, 并引入了指针概念, 使程序效率更高。另外, C 语言具有强大的图形功能, 支持多种显示器和驱动器, 且计算功能、逻辑判断功能强大。

4. C 语言是结构化语言

结构化语言的显著特点是代码及数据的分隔化, 即程序的各个部分除了必要的信息交流外彼此独立。这种结构化方式可使程序层次清晰, 便于使用、维护以及调试。C 语言的主要结构成分是函数, 这些函数可方便地调用, 并具有多种循环、条件语句控制程序流向, 从而使程序完全结构化。

5. C 语言语法限制不太严格, 程序设计自由度高

虽然 C 语言也是强类型语言, 但它的语法比较灵活, 允许程序编写者有较大的自由度。

6. C语言允许直接访问物理地址,可以直接对硬件进行操作

C语言既具有高级语言的功能,又具有低级语言的许多功能,能够像汇编语言一样对位、字节和地址进行操作,而这三者是计算机最基本的工作单元,因此C语言可以用来写系统软件。

7. C语言程序生成代码质量高,程序执行效率高

C语言程序的生成代码一般只比汇编程序生成的目标代码效率低10%~20%。

8. C语言适用范围大,可移植性好

C语言程序具有较高的可移植性。可移植性指的是,可以把为某种计算机编写的软件运行在另一种机器或操作系统上,如在DOS下写的程序,能够方便地在Windows2000下运行,这个程序就是一个可移植的程序。C语言不包含依赖硬件的输入/输出机制,其输入/输出功能是由独立于C语言的库函数来实现。这样就使C语言程序本身不依赖于硬件系统,也便于在不同的机器和系统间移植。

1.2 C语言程序的基本结构

任何一种程序设计语言都具有特定的语法规则和规定的表达方法。一个程序只有严格按照程序设计语言规定的语法和表达方式编写,才能保证编写的程序在计算机中能正确地执行,同时也便于阅读和理解。为了了解C语言的基本程序结构,我们先介绍一个简单的C程序。

【例1—1】 已知两个整数5和7,求这两个数的乘积,并将结果显示出来。

```
#include<stdio.h>      /* 标准输入输出头文件 */
main()                  /* 主函数 */
{
    void OutStar();      /* 声明函数 */
    int a,b,c;           /* 定义3个整型变量 */
    a=5;b=7;             /* 变量赋值 */
    c=a*b;               /* 算术运算并赋值 */
    OutStar();           /* 调用 OutStar 函数 */
    printf("c = %d\n",c); /* 输出结果 */
    OutStar();           /* 调用 OutStar 函数 */
}

void OutStar()          /* 定义 OutStar 函数,void 指定该函数不返回值 */
{
    printf("\n ***** \n");
}
```

从上面的程序可以看出以下几点。