21世纪

高等院校规划教材

主　编　任正云

C语言程序设计

中国水利水电出版社
www.waterpub.com.cn

21世纪高等院校规划教材

C语言程序设计

主　编　任正云

中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书遵照 C 语言标准，全面、系统、深入浅出地阐述了 C 语言程序设计的基本概念、语法和语义，介绍了用 C 语言进行程序设计的基本方法和技巧。内容包括数据类型和表达式、流程控制、算法分析、函数与程序结构等。本书概念准确，结构合理，层次清晰，实例丰富，选材独到，语言通俗易懂。每章末都配有习题可供不同层次的读者练习。

本书是一本准确、全面反映标准 C 语言的教材，还配有《C 语言程序设计上机指导、题解、实验、课程设计与等级考试上机题》一书。阅读和使用本教材，不需要读者具备高级语言程序设计的基础。

本教材既可供高等院校计算机和非计算机专业本、专科或培训班教学使用，也是广大科技工作者和编程爱好者的一本很好的参考书。

本书配有免费电子教案，读者可以从中国水利水电出版社网站以及万水书苑下载，网址为：http://www.waterpub.com.cn/softdown/和 http://www.wsbookshow.com。

图书在版编目（CIP）数据

C语言程序设计 / 任正云主编. -- 北京 : 中国水利水电出版社, 2009.9

21世纪高等院校规划教材
ISBN 978-7-5084-6826-6

Ⅰ. ①C… Ⅱ. ①任… 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170703号

策划编辑：杨庆川　责任编辑：李　炎　封面设计：李　佳

书　名	21 世纪高等院校规划教材 **C 语言程序设计**
作　者	主　编　任正云
出版发行	中国水利水电出版社 （北京市海淀区玉渊潭南路 1 号 D 座　100038） 网址：www.waterpub.com.cn E-mail：mchannel@263.net（万水） sales@waterpub.com.cn 电话：（010）68367658（营销中心）、82562819（万水）
经　售	全国各地新华书店和相关出版物销售网点
排　版	北京万水电子信息有限公司
印　刷	北京市天竺颖华印刷厂
规　格	184mm×260mm　16 开本　18.25 印张　445 千字
版　次	2009 年 9 月第 1 版　2009 年 9 月第 1 次印刷
印　数	0001—4000 册
定　价	29.00 元

序

随着计算机科学与技术的飞速发展，计算机的应用已经渗透到国民经济与人们生活的各个角落，正在日益改变着传统的人类工作方式和生活方式。在我国高等教育逐步实现大众化后，越来越多的高等院校会面向国民经济发展的第一线，为行业、企业培养各级各类高级应用型专门人才。为了大力推广计算机应用技术，更好地适应当前我国高等教育的跨跃式发展，满足我国高等院校从精英教育向大众化教育的转变，符合社会对高等院校应用型人才培养的各类要求，我们成立了“21 世纪高等院校规划教材编委会”，在明确了高等院校应用型人才培养模式、培养目标、教学内容和课程体系的框架下，组织编写了本套“21 世纪高等院校规划教材”。

众所周知，教材建设作为保证和提高教学质量的重要支柱及基础，作为体现教学内容和教学方法的知识载体，在当前培养应用型人才中的作用是显而易见的。探索和建设适应新世纪我国高等院校应用型人才培养体系需要的配套教材已经成为当前我国高等院校教学改革和教材建设工作面临的紧迫任务。因此，编委会经过大量的前期调研和策划，在广泛了解各高等院校的教学现状、市场需求，探讨课程设置、研究课程体系的基础上，组织一批具备较高的学术水平、丰富的教学经验、较强的工程实践能力的学术带头人、科研人员和主要从事该课程教学的骨干教师编写出一批有特色、适用性强的计算机类公共基础课、技术基础课、专业及应用技术课的教材以及相应的教学辅导书，以满足目前高等院校应用型人才培养的需要。本套教材消化和吸收了多年来已有的应用型人才培养的探索与实践成果，紧密结合经济全球化时代高等院校应用型人才培养工作的实际需要，努力实践，大胆创新。教材编写采用整体规划、分步实施、滚动立项的方式，分期分批地启动编写计划，编写大纲的确定以及教材风格的定位均经过编委会多次认真讨论，以确保该套教材的高质量和实用性。

教材编委会分析研究了应用型人才与研究型人才在培养目标、课程体系和内容编排上的区别，分别提出了 3 个层面上的要求：在专业基础类课程层面上，既要保持学科体系的完整性，使学生打下较为扎实的专业基础，为后续课程的学习做好铺垫，更要突出应用特色，理论联系实际，并与工程实践相结合，适当压缩过多过深的公式推导与原理性分析，兼顾考研学生的需要，以原理和公式结论的应用为突破口，注重它们的应用环境和方法；在程序设计类课程层面上，把握程序设计方法和思路，注重程序设计实践训练，引入典型的程序设计案例，将程序设计类课程的学习融入案例的研究和解决过程中，以学生实际编程解决问题的能力为突破口，注重程序设计算法的实现；在专业技术应用层面上，积极引入工程案例，以培养学生解决工程实际问题的能力为突破口，加大实践教学内容的比重，增加新技术、新知识、新工艺的内容。

本套规划教材的编写原则是：

在编写中重视基础，循序渐进，内容精炼，重点突出，融入学科方法论内容和科学理念，反映计算机技术发展要求，倡导理论联系实际和科学的思想方法，体现一级学科知识组织的层次结构。主要表现在：以计算机学科的科学体系为依托，明确目标定位，分类组织实施，兼容互补；理论与实践并重，强调理论与实践相结合，突出学科发展特点，体现学科发展的内在规律；教材内容循序渐进，保证学术深度，减少知识重复，前后相互呼应，内容编排合理，整体

结构完整；采取自顶向下设计方法，内涵发展优先，突出学科方法论，强调知识体系可扩展的原则。

本套规划教材的主要特点是：

（1）面向应用型高等院校，在保证学科体系完整的基础上不过度强调理论的深度和难度，注重应用型人才的专业技能和工程实用技术的培养。在课程体系方面打破传统的研究型人才培养体系，根据社会经济发展对行业、企业的工程技术需要，建立新的课程体系，并在教材中反映出来。

（2）教材的理论知识包括了高等院校学生必须具备的科学、工程、技术等方面的要求，知识点不要求大而全，但一定要讲透，使学生真正掌握。同时注重理论知识与实践相结合，使学生通过实践深化对理论的理解，学会并掌握理论方法的实际运用。

（3）在教材中加大能力训练部分的比重，使学生比较熟练地应用计算机知识和技术解决实际问题，既注重培养学生分析问题的能力，也注重培养学生思考问题、解决问题的能力。

（4）教材采用“任务驱动”的编写方式，以实际问题引出相关原理和概念，在讲述实例的过程中将本章的知识点融入，通过分析归纳，介绍解决工程实际问题的思想和方法，然后进行概括总结，使教材内容层次清晰，脉络分明，可读性、可操作性强。同时，引入案例教学和启发式教学方法，便于激发学习兴趣。

（5）教材在内容编排上，力求由浅入深，循序渐进，举一反三，突出重点，通俗易懂。采用模块化结构，兼顾不同层次的需求，在具体授课时可根据各校的教学计划在内容上适当加以取舍。此外还注重了配套教材的编写，如课程学习辅导、实验指导、综合实训、课程设计指导等，注重多媒体的教学方式以及配套课件的制作。

（6）大部分教材配有电子教案，以使教材向多元化、多媒体化发展，满足广大教师进行多媒体教学的需要。电子教案用 PowerPoint 制作，教师可根据授课情况任意修改。相关教案的具体情况请到中国水利水电出版社网站 www.waterpub.com.cn 下载。此外还提供相关教材中所有程序的源代码，方便教师直接切换到系统环境中教学，提高教学效果。

总之，本套规划教材凝聚了众多长期在教学、科研一线工作的教师及科研人员的教学科研经验和智慧，内容新颖，结构完整，概念清晰，深入浅出，通俗易懂，可读性、可操作性和实用性强。本套规划教材适用于应用型高等院校各专业，也可作为本科院校举办的应用技术专业的课程教材，此外还可作为职业技术学院和民办高校、成人教育的教材以及从事工程应用的技术人员的自学参考资料。

我们感谢该套规划教材的各位作者为教材的出版所做出的贡献，也感谢中国水利水电出版社为选题、立项、编审所做出的努力。我们相信，随着我国高等教育的不断发展和高校教学改革的不断深入，具有示范性并适应应用型人才培养的精品课程教材必将进一步促进我国高等院校教学质量的提高。

我们期待广大读者对本套规划教材提出宝贵意见，以便进一步修订，使该套规划教材不断完善。

21 世纪高等院校规划教材编委会

2004 年 8 月

前　言

随着计算机技术的飞速发展，社会对人们的计算机应用水平和开发水平的要求日益提高，作为计算机基础课程的程序设计语言也应顺应形势的发展进行改革，程序设计课程是提高学生综合开发能力的重要环节，但选择何种语言环境进行教学，是一个值得高度重视的问题。

编程语言的选择，应该考虑三个方面的问题：首先，要有利于培养学生程序设计的能力；其次，要注意和当代最新编程技术的衔接；第三，语言本身的应用潜力。C 语言作为结构化的程序设计语言，同其他编程语言相比，在以上三个方面都有突出优势。C 语言既适合开发系统程序，又适合开发应用程序。既具有高级语言功能强大、使用灵活的特点，又具有汇编语言适合于编写底层应用程序的实用性的功能。对计算机专业的学生而言，C 语言可以作为学习面向对象 C++程序设计的基础课程；而对非计算机专业的学生而言，C 语言本身就具有广泛的应用价值，各领域现在用 C 语言编程已经很普遍，C 语言已经成为一种必不可少的编程工具。因此，选择 C 语言作为计算机程序设计的基础课程，是适应当前形势发展的需要。

C 语言程序设计是一门实践性很强的课程，在理解概念的基础之上，既要动手编程，又要上机实践，练就过硬的程序设计能力是我们培养学生的目标。学习程序设计时，一定要活学活用，要举一反三。

本教材融入了编者多年的教学经验，充分考虑到初学者的能力、认知水平、知识结构等因素，遵循循序渐进、由浅入深的原则，文字叙述简明扼要、通俗易懂，理论阐述简明科学，选例经典适用，分析透彻浅显，使读者在把握要点时感到具体生动，而不抽象枯涩。

本书由任正云任主编，严永松、李敏、王晓雨、赖玲、刘珊艳、王娅芬、胡玉荣、张牧、肖衡、陈万华、李俊梅和田雯参加了部分章节的编写和审稿工作，全书由任正云统稿，所有的程序由张牧、胡玉荣负责调试，在编写过程中得到了田原、李素若副教授和相关专家的指导，在此一并表示衷心的感谢。

由于编写水平有限，本书难免有错误之处，恳请广大读者不吝赐教。

编　者

2009 年 6 月

目　　录

第 1 章　程序设计基础

1.1　程序设计及程序设计语言

程序是以某种语言为工具编制出来的指令序列，它表达了人的思想。计算机程序是用计算机程序设计语言所要求的规范书写出来的一系列指令，它表达了程序员要求计算机执行的操作。对于计算机来说，一组机器指令就是程序，它是按计算机硬件设计规范的要求，编制出来的指令序列。对于使用计算机的人来说，程序员用某高级语言编写的语句序列也是程序。程序通常以文件的形式保存起来。所以，源文件、源程序和源代码都是程序。通俗的讲，程序是能被机器识别并执行的一系列的指令代码，这些指令代码是用程序设计语言来描述的。程序设计语言是人与计算机对话的工具。程序设计需要在一定的语言和环境下进行。

1.1.1　程序设计语言

程序设计语言可以分为低级语言和高级语言两大类。

1. 低级语言

低级语言又叫面向机器的语言，它是特定的计算机系统所固有的语言，又可分为机器语言和符号语言（汇编语言）两类。

机器语言就是计算机能够直接识别并执行的指令集合。由于计算机只能识别“0”和“1”两种状态，所以机器语言指令都是二进制指令。例如某种型号的计算机用 10000000 表示“进行一次加法”，用 10010000 表示“进行一次减法”。很明显，这种由 0 和 1 组成的指令难学、难记、难阅读、难修改，给用户带来很大的不便。而且机器语言随机器不同而不相同，因此移植性很差。程序员不仅要考虑解题的思路，还要熟悉机器的内部结构，并且还要“手工”地进行存储器分配，这种编程劳动强度大，给计算机的普及和推广造成了极大的障碍。

符号语言是从机器语言发展演变而来的，它用一些“助记符号”来代替那些冗长的二进制指令。例如用 ADD 表示加法，SUB 表示减法，等等。显然，这种表示形式要比机器语言容易理解和使用。但是计算机并不能直接执行符号语言程序，必须先把它们逐条翻译成机器指令，然后交给计算机执行。这个翻译工作叫“汇编”，是由一种专门的汇编程序来实现的，因此符号语言又叫汇编语言。虽然用汇编语言来编制程序，使编程的效率和程序的可读性提高了不少，但汇编语言始终是一种和机器语言非常接近的语言，它的书写格式在很大程度上取决于特定计算机上的机器指令，因此，它仍然是一种低级语言，对于人们的抽象思维和交流十分不便。

2. 高级语言

高级语言是类似于人类自然语言和数学描述语言的程序设计语言，分为面向过程的程序设计语言和面向对象的程序设计语言，如 C 语言、Pascal 语言、FoxBase、Visual C++、Visual Basic 等。

（1）面向过程的程序设计语言。汇编语言和机器语言都是面向机器的，随机器而异，1954 年出现的 FORTRAN 语言以及随后相继出现的其他计算机语言，使人们开始摆脱进行程序设

计时必须先熟悉机器的桎梏，而把精力集中在解题的思路和方法上，使程序设计语言开始与解题方法相结合。其中一种就是把解题过程看作数据被加工的过程。基于这种方法的程序设计语言称为面向过程的程序设计语言，C 语言就是一种面向过程的程序设计语言。下面是一个计算圆面积的 C 语言程序设计的片段：

```
main()                          /*主函数*/
{                               /*程序开始*/
  float r;                      /*定义一个实型变量 r，表示圆的半径 */
  float s;                      /*定义面积变量 s*/
  s=3.14*r*r;                   /*把计算的面积赋给 s*/
  printf("%f ", s);             /*输出面积 s 的值*/
}                               /*程序结束*/
```

这个程序是很好理解的，其中计算面积的语句与我们习惯的数学式子没有什么根本的区别（“/*”与“*/”之间的内容称为注释，目的是为了阅读的方便，让程序更容易被理解）。显然使用高级语言编程可以较大地降低编程过程中的劳动强度，提高编程效率，高级语言的诞生是计算机发展史上的一个里程碑。它使人们能摆脱具体机器指令系统的束缚，用接近人们习惯的语言来构思解题过程，从而大大提高了编程效率，使人们能够编制出规模越来越大的程序，以满足日益广泛而深入的应用需求。

（2）面向对象的程序设计语言。面向对象的程序设计是一种结构模拟方法，它把现实世界看成是由许多对象（object）所组成，对象之间通过相互发送和接收消息进行联系。消息激发对象本身的运动，形成对象状态的变化。从程序结构的角度，每个对象都是一个数据和方法的封装体——抽象数据类型。

从分类学的观点来看，客观世界中的对象都可以分类。也就是说，所有的对象都属于特定的类（class），或者说每个对象都是类的一个实例。因而面向对象的程序设计的一个关键是定义“类”，并由“类”生成“对象”。

面向对象的程序比面向过程的程序更清晰易懂，更适宜编写更大规模的程序，正在成为当代程序设计的主流。面向对象的程序设计语言有 Java、Visual Basic、Visual Basic.NET 等，由 C 派生出的 C++语言也属于面向对象的程序设计语言，它是一种多范型程序设计语言，不仅可以利用它编写面向对象的程序，还可以用它编写面向过程的程序。

1.1.2 C 语言的发展过程

C 语言是目前世界上流行最广泛的高级程序设计语言。C 语言的发展过程可粗略地分为三个阶段：1970 年至 1973 年为诞生阶段，之后至 1988 年为发展阶段，1988 年以后为成熟阶段。

1. C 语言的诞生

C 语言是为写 UNIX 操作系统而诞生的。1970 年美国 AT&T 公司贝尔实验室的 Ken Thompson 为实现 UNIX 操作系统提出了一种仅供自己使用的工作语言，由于该工作语言是基于 1967 年由英国剑桥大学的 Martin Richards 提出的 BCPL 语言设计的，因而被作者命名为 B 语言，B 取自 BCPL 的第一个字母。B 语言被用于在 PDP-7 计算机上实现了第一个 UNIX 操作系统。1972 年贝尔实验室的 Dennis M.Ritchie 又在 B 语言基础上系统地引入了各种数据类型，从而使 B 语言的数据结构类型化。1973 年 K.Tompson 和 D.M.Ritchie 用 C 语言重写了 UNIX 操作系统，推出 UNIX v5，1975 年又推出 UNIX v6。此时的 C 语言是附属于 UNIX 操作系统的。

2. C 语言的发展

为了使 UNIX 操作系统能够在别的机器上得到推广，1977 年 C 语言的作者发表了不依赖于具体机器系统的 C 语言编译文本“可移植 C 语言编译程序”，从而推动了 UNIX 操作系统在各种机器上的实现以及 UNIX 操作系统的不断发展。1978 年以后相继推出了 UNIX v7，UNIX systemV。UNIX 操作系统的巨大成功和广泛使用，使人们普遍注意到 C 语言的突出优点，从而又促进了 C 语言的迅速推广。同时，C 语言也伴随着 UNIX 操作系统的发展而不断发展。1978 年 Brian W.Kernighan 和 D.M.Ritchie 以 UNIX v7 中编译程序为基础写了影响深远的名著《The C Programming Language》，这本书上介绍的 C 语言是以后各种 C 语言版本的基础，被称为传统 C 语言。1978 年以后，C 语言先后移植到各种大型机、中型机、小型机及微型机上。目前，C 语言成为世界上使用最广泛的高级程序设计语言，且不依赖于 UNIX 操作系统而独立存在。

3. C 语言的成熟

1978 年以后，C 语言的不断发展产生了各种 C 语言版本，不同的 C 语言版本对传统 C 语言都有所扩充和发展。1983 年，美国国家标准协会（ANSI）综合了各版本对 C 的扩充和发展，制定了新标准，称为 ANSI C。Kernighan 和 D.M.Ritchie 按 ANSI C 标准重写了他们的经典著作，于 1990 年正式发表了国际标准化组织（ISO）公布的 C 语言标准。C 语言标准的制定标志着 C 语言的成熟，1988 年以后推出的各种 C 语言版本与 ANSI C 是相容的。

1.1.3　C 语言的标准

C 语言的灵活性、丰富性和可移植性很快得到了普遍的认可，接着适合于各种操作系统（UNIX、MS-DOS、CP/M-80/86 等）和不同机型（字长为 8bit～32bit）的 C 语言编译系统相继出现。1982 年，美国国家标准学会（American National Standards Institute，ANSI）语言标准化委员会开始着手进行 C 语言的标准化工作，并于 1983 年公布了第一个 C 语言标准草案（'83 ANSI C）。1989 年，ANSI 又发布了一个完整的 C 语言标准——ANSI X3.159-1989，通常称为“ANSI C”，简称“C89”，1990 年，国际标准化组织 ISO/JEC JTC1/SC22/WG14 采纳了 C89，做了少量修改后，以国际标准 ISO/IEC 9899：1990 发布，通常称其为“C90”，它同 C89 基本上相同。

1995 年，WG14 对 C89 做了两处技术修订和一个扩充，人们将其称为“C89 增补 1”或“C95”，同时，WG14 开始着手对 C 标准全面修订，并于 1999 年获得通过，形成了正式的 C 语言标准，命名为 ISO/IEC 9899：1999，简称“C99”。

鉴于目前各厂商提供的所有 C 编译器都未实现 C99 所建议的功能。为了便于读者运行 C 程序，本书所介绍的程序都是符合 ANSI C 标准并能在大多数 C 编译器上运行和通过的程序。

1.2　C 语言的特点

一种语言之所以能存在和发展，并具有生命力，总是有其不同于（或优于）其他语言的特点。C 语言的主要特点如下：

（1）语言简洁、紧凑，使用方便、灵活。C 语言共有 32 个关键字，9 种控制语句，程序书写形式自由，主要用小写字母表示，压缩了一切不必要的成分。C 程序比其他许多高级语言

简练，源程序短，因此输入程序时工作量少。

（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，从而使其运算类型极其丰富，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，具有现代化语言的各种数据结构。C 语言的数据类型有整型、浮点型、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结构（如链表、树、栈等）的运算，尤其是指针类型数据，使用起来比 Pascal 更为灵活、多样。

（4）具有结构化的控制语句（如 if...else 语句、while 语句、do...while 语句、switch 语句、for 语句）。用函数作为程序的模块单位，便于实现程序的模块化。C 语言是良好的结构化语言，符合现代编程风格的要求。

（5）语法限制不太严格，程序设计自由度大。例如对数组下标越界不做检查，由程序编写者自己保证程序的正确。对变量的类型使用比较灵活，例如整型数据与字符型数据可以通用。一般的高级语言语法检查比较严，能检查出几乎所有的语法错误。而 C 语言允许程序编写者有较大的自由度，因此放宽了语法检查。程序员应当仔细检查程序，保证其正确，而不要过分依赖 C 编译程序去查错。“限制”与“灵活”是一对矛盾。限制严格，就失去灵活性；而强调灵活，就必然放松限制。一个不熟练的编程人员，编一个正确的 C 程序可能会比编写其他高级语言的程序难一些。也就是说，对用 C 语言编程的人，要求对程序设计更熟练一些。

（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 语言能进行位（bit）操作，能按地址访问字节，可以实现汇编语言的大部分功能，可以直接对硬件进行操作。因此，C 语言既具有高级语言的特点，又具有低级语言的许多功能，既可用来编写系统软件，又可用来编写应用软件。C 语言的这种双重性，使它既是成功的系统描述语言，又是通用的程序设计语言。有人把 C 语言称为“高级语言中的低级语言”或“中级语言中的高级语言”，意思是说 C 语言兼有高级语言和低级语言的特点，但一般仍习惯将 C 语言称为高级语言，因为 C 程序也要通过编译、连接才能得到可执行的目标程序，这和其他高级语言是相同的。

（7）生成目标代码质量高，程序执行效率高。一般只比汇编程序生成的目标代码效率低 10%～20%。

（8）程序可移植性好（与汇编语言比）。

上面只介绍了 C 语言的最容易理解的一般特点，至于 C 语言应用内部的其他特点将结合以后各章的内容作介绍。由于 C 语言的这些优点，使 C 语言应用面很广。许多大的软件都用 C 语言编写，这主要是由于 C 的可移植性好和硬件控制能力高，表达和运算能力强。许多以前只能用汇编语言处理的问题，现在可以改用 C 语言来处理了。

1.3 C 程序结构

1.3.1 简单的 C 程序介绍

【例 1.1】 C 程序输出。

```
main()
{
```

```
    printf("This is a C program.\n");
}
```

输出结果是：This is a C program.

说明：

（1）main 表示“主函数”。每个 C 语言程序都必须有一个 main 函数，它是每一个 C 语言程序的执行起始点（入口点）。main()表示“主函数” main 的函数头。

（2）用{}括起来的是“主函数” main 的函数体。main 函数中的所有操作（或语句）都在这一对{}之间，也就是说 main 函数的所有操作都在 main 函数体中。

（3）“主函数” main 中只有一条语句，它是 C 语言的库函数（见附录 B），功能是用于程序的输出（显示在屏幕上），本例用于将一个字符串"This is a C program.\n"的内容输出。即在屏幕上显示：

```
This is a C program.
```

\n 的意思是回车换行。

（4）注意：每条语句用“;”号结束。

【例 1.2】设计一个程序，计算两数之和并输出结果。

```
main()
{
    int a,b,sum;            /* 这是定义三个整型变量 a,b,sum*/
    a=123;b=456;            /* 给 a,b 赋初值，使得 a 的值为 123，b 的值为 456 */
    sum=a+b;                /* 把 a、b 两数相加（结果为 579），并把结果赋给 sum */
    printf("sum=%d\n",sum);     /* 输出 sum 的值 */
}
```

运行结果为：

```
sum=579
```

说明：

（1）程序包含一个 main 函数作为程序执行的起点。{}之间为 main 函数的函数体，main 函数的所有操作均在 main 函数体中。

（2）“/*”和“*/”括起来的部分是一段注释，注释只是为了改善程序的可读性，在编译、运行时不起作用（事实上编译时会跳过注释，目标代码中不会包含注释）。注释可以放在程序任何位置，并允许占用多行，只是需要注意“/*”、“*/”匹配，一般不要嵌套注释。

（3）int a,b,sum;是变量声明，声明了三个具有整数类型的变量 a，b，sum。C 语言的变量必须先声明再使用。

（4）a=123;b=456;是两条赋值语句。将整数 123 赋给整型变量 a，将整数 456 赋给整型变量 b。a，b 两个变量的初始值分别为 123，456。注意这是两条赋值语句，每条语句均用“;”结束。也可以将两条语句写成两行，即：

```
a=123;
b=456;
```

由此可见 C 语言程序的书写可以相当随意，但是为了保证容易阅读要遵循一定的规范。

（5）sum=a+b;是将 a，b 两变量内容相加，然后将结果赋值给整型变量 sum。此时 sum 的内容为 579。

（6）printf("sum=%d\n",sum);是调用库函数输出 sum 的结果。%d 为格式控制，表示 sum 的值以十进制整数形式输出。

程序运行后，输出（显示）：sum=579。

【例 1.3】输入两个整数，计算两者较大的数，并输出。

```
main()                              /* 主函数 */
{                                   /* main 函数体开始 */
    int a,b,c;                      /*定义三个整型变量 a,b,c*/
    int max(int x,int y);           /* max()函数的声明*/
    scanf("%d,%d",&a,&b);           /*调用库函数 scanf()给 a,b 赋值*/
    c=max(a,b);                     /*用 a，b 做实参调用 max()，将调用结果赋给 c*/
    printf("max=%d",c);
}                                   /* main 函数体结束 */
int max(int x,int y)                /*定义函数 max()*/
{                                   /* max 函数体开始 */
    int z;                          /*定义变量 z */
        if(x>y)
        z=x;
    else
        z=y;
        return z;                   /* 将 z 值返回到调用处 */
}                                   /*  max()函数体结束*/
```

说明：

（1）本程序包括两个函数。其中主函数 main 仍然是整个程序执行的起点。函数 max 计算两数中较大的数。

（2）主函数 main 调用 scanf 函数获得两个整数，赋给两个变量 a，b，然后调用函数 max 获得两个数字中较大的值，并赋给变量 c。最后输出变量 c 的值（结果）。

（3）程序第 9 行 int max(int x,int y)是函数 max 的函数头，函数 max 的函数头表明此函数获得两个整数，返回一个整数。

（4）函数 max 同样也用{ }将函数体括起来。max 的函数体是函数 max 的具体实现。从参数表获得数据，处理后得到结果 z，然后将 z 返回给调用处。

（5）本例还表明函数除了调用库函数外，还可以调用用户自己定义、编制的函数。

1.3.2 C 程序结构

综合上述三个例子，我们对 C 语言程序的基本组成和形式（程序结构）有了一个初步了解。

（1）C 程序由函数构成，C 语言是函数式的语言，函数是 C 程序的基本单位（以例 1.3 说明）。

1）一个 C 源程序至少包含一个 main 函数，也可以包含一个 main 函数和若干个其他函数。函数是 C 程序的基本单位。

2）被调用的函数可以是系统提供的库函数，也可以是用户根据需要自己编写设计的函数。C 语言是函数式的语言，程序的全部工作都是由各个函数完成。编写 C 程序就是编写一个个函数。

3）C 函数库非常丰富，ANSI C 提供 100 多个库函数，Turbo C 提供 300 多个库函数。

（2）main 函数（主函数）是每个程序执行的起始点（以例 1.3 说明）。一个 C 程序总是从 main 函数开始执行，而不论 main 函数在程序中的位置。可以将 main 函数放在整个程序的

最前面，也可以放在整个程序的最后，或者放在其他函数之间。

（3）一个函数由函数首部和函数体两部分组成（以例 1.3 的 max 函数说明）。

1）函数首部：一个函数的第一行。

```
返回值类型 函数名([函数参数类型 1 函数参数名 1][,…,函数参数类型 2，函数参数名 2])
int max(int x,int y)
```

注意：函数可以没有参数，但是后面的一对括号不能省略。

2）函数体：函数首部后面用一对{ }括起来的部分。如果函数体内有多个{ }，最外层是函数体的范围。函数体一般包括声明、执行两部分。

```
{
    [声明部分]：定义本函数所使用的变量。
    [执行部分]：由若干条语句组成命令序列（可以在其中调用其他函数）。
}
```

（4）C 程序书写格式自由。

1）一行可以写几个语句，一个语句也可以写在多行上。

2）C 程序没有行号，也没有 FORTRAN、COBOL 那样严格规定书写格式（语句必须从某一列开始）。

3）每条语句的最后必须有一个分号“;”表示语句的结束。

（5）可以使用“/*”、“*/”对 C 程序中的任何部分作注释。注释可以提高程序可读性，使用注释是编程人员的良好习惯。

1）编写好的程序往往需要修改、完善。事实上很多人会发现自己编写的程序在经历了一些时间以后，由于缺乏必要的文档、必要的注释，最后连自己都很难再读懂。需要花费大量时间重新思考、理解原来的程序，这浪费了大量的时间。如果一开始编程就对程序进行注释，虽然刚开始麻烦一些，但日后可以节省大量的时间。

2）一个实际的系统往往是多人合作开发，程序文档、注释是其中重要的交流工具。

（6）C 语言本身不提供输入/输出语句，输入/输出的操作是通过调用库函数（scanf，printf）完成。输入/输出操作涉及具体计算机硬件，把输入/输出操作放在函数中处理，可以简化 C 语言和 C 的编译系统，便于 C 语言在各种计算机上实现。不同的计算机系统需要对函数库中的函数作不同的处理，以便实现同样或类似的功能。

不同的计算机系统除了提供函数库中的标准函数外，还按照硬件的情况提供一些专门的函数。因此不同计算机系统提供的函数数量、功能会有一定差异。

1.4　源程序的编辑、编译、连接与运行

用高级语言编写的程序称为“源程序”，通常简称为程序。高级语言程序也必须被转换为机器语言程序才能被机器理解和执行，完成这种转换任务的系统软件称为编译程序。相应的转换过程通常称为编译。

高级语言是面向解题过程的，语言本身与具体的机器系统无关，因而用高级语言编写的应用程序可移植性好。编译程序是一种语言的具体实现，编译程序与具体机器系统有关，通常称为语言的一个版本。同一语言的不同版本不完全相同，在使用一种具体的高级语言及其编译程序开发软件时，必须参考与编译程序配套的有关资料。本书阐述的内容遵从 ANSI C 标准，

对于大多数C编译程序具有通用性。为了使上机环境尽量简单，书中所有的例题均在Turbo C 2.0上通过。

C语言采用编译方式将源程序转换为二进制的目标代码。编写好一个C程序到完成运行一般经过以下几个步骤。

1. 编辑

所谓编辑，包括以下内容：①将源程序逐个字符输入到计算机内存；②修改源程序；③将修改好的源程序保存在磁盘文件中。编辑的对象是源程序，它是以ASCII代码的形式输入和存储的，不能被计算机执行。目前使用较多的编辑软件有：UNIX下的编辑程序ed、vi等，MS-DOS下的EDLIN、Wordstar，Windows下的Write、Word等字处理软件。关于编辑软件的使用方法请参阅《C语言程序设计上机指导题解、实验、课程设计与等级考试上机题》一书或其他相关手册。

2. 编译

编译就是将已编辑好的源程序（已存储在磁盘文件中）翻译成二进制的目标代码。在编译时，还要对源程序进行语法检查，如发现有错，则在屏幕上显示出错信息，此时应重新进入编辑状态，对源程序进行修改后再重新编译，直到通过编译为止。编译后得到的二进制代码在UNIX下是后缀为.o的文件，在MS-DOS下是后缀为.obj的文件。应当指出，经编译后得到的二进制代码还不能直接执行，因为第一个模块往往是单独编译的，必须把经过编译的各个模块的目标代码与系统提供的标准模块（如C语言中的标准函数库）连接后才能运行。

3. 连接

将各模块的二进制目标代码与系统标准模块经连接处理后，得到具有绝对地址的可执行文件，它是计算机能直接执行的文件。在UNIX下它以.out为后缀（例如，f.out），在MS-DOS下以.exe为后缀（例如，f.exe）。

4. 执行

执行一个经过编译和连接的可执行的目标文件。只有在操作系统的支持和管理下才能执行它。图1-1用以表示编辑、编译、连接、运行的过程。

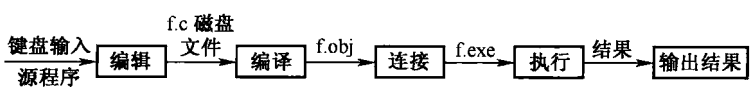

图1-1 程序执行步骤

其中文件取名为f，后缀按MS-DOS的规则表示。

近来“集成化”的工具环境已将编辑、编译、连接、调试工具集于一身（例如Turbo C），用户可以方便地在窗口状态下进行编辑、编译、连接、调试、运行的全过程。

5. C编译系统简介

由于C语言的优点及其应用的普及性，因而产生了不少的版本和编译系统。在微机上使用的C编译系统有Turbo C、Borland C、Microsoft C、Quick C等，它们在具体的实现上略有差别。另外，还可以使用可视化的编译工具Visual C++。前面所列举的几种C编译系统都是在DOS环境下工作的，而Visual C++是在Windows环境下工作的。

有关编译环境的介绍和使用方法请参阅本书配套教材《C语言程序设计上机指导题解、实