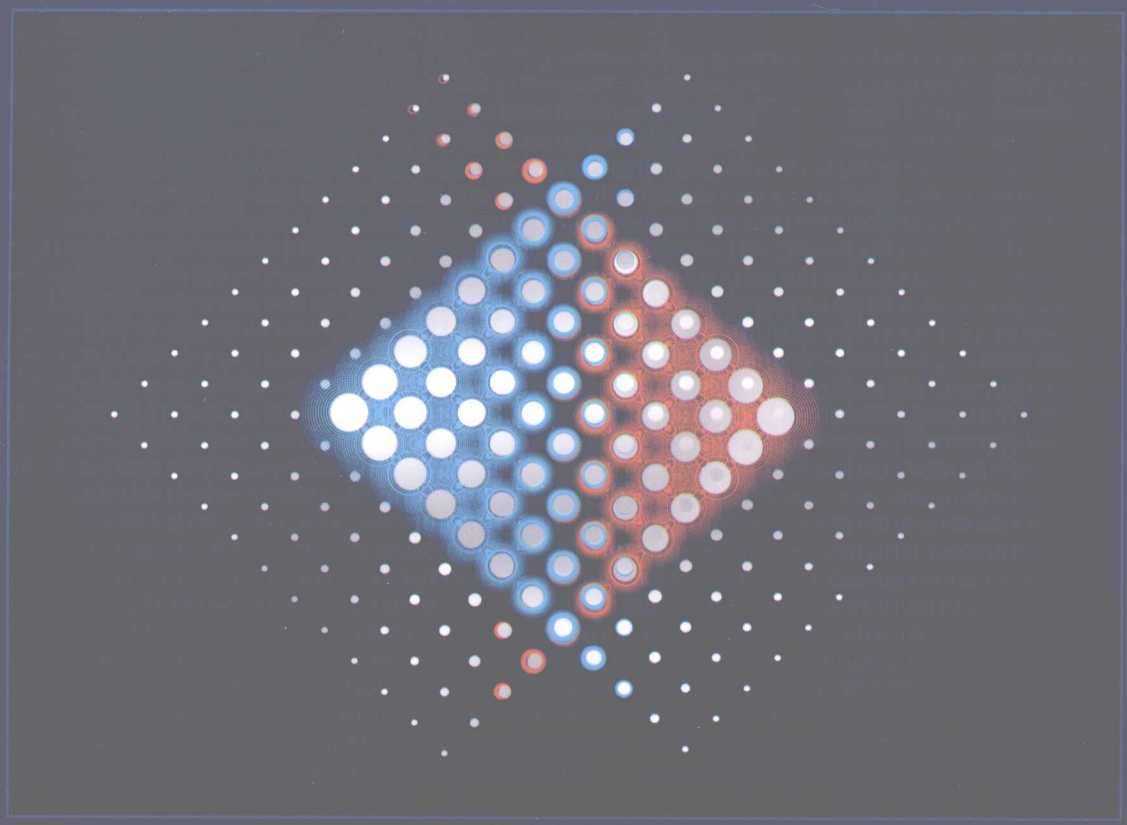
普通高等教育“十一五”国家级规划教材

新编计算机类本科规划教材

C程序设计语言

魏东平　朱连章　于广斌　编著



電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

普通高等教育“十一五”国家级规划教材
新编计算机类本科规划教材

C 程序设计语言

魏东平　朱连章　于广斌　编著

電子工業出版社
Publishing House of Electronics Industry
北京·BEIJING

内 容 简 介

本书是普通高等教育“十一五”国家级规划教材，从实用性、适应性和先进性出发，以培养读者C语言程序设计的能力为目标，结合大量实例，较全面地介绍C语言的基本概念和程序设计的基本方法。全书共分13章，主要内容包括：C语言基础，顺序、选择和循环程序设计，数组，指针，字符串，函数，自定义数据类型，文件操作，位操作等。本书配套《C 程序设计语言实验与习题指导》，并提供配套电子课件、习题解答和程序源代码。

本书可作为高等学校计算机与信息技术课程的基础教材，也可供相关领域的工程技术人员学习、参考。

图书在版编目（CIP）数据

C程序设计语言 / 魏东平，朱连章，于广斌编著.—北京：电子工业出版社，2009.2
（新编计算机类本科规划教材）
ISBN 978-7-121-08141-5

Ⅰ. C… Ⅱ. ①魏…②朱…③于… Ⅲ. C程序—程序设计—高等学校—教材 Ⅳ. TP312

中国版本图书馆CIP数据核字（2009）第008421号

责任编辑：王羽佳
印　　刷：
装　　订：北京京师印务有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路173信箱　邮编：100036
开　　本：787×1092　1/16　印张：17.5　字数：448千字
印　　次：2009年2月第1次印刷
印　　数：5000册　　定价：29.90元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至zlts@phei.com.cn，盗版侵权举报请发邮件至dbqq@phei.com.cn。

服务热线：（010）88258888。

前　言

21 世纪，人类社会步入了高速发展的信息时代，掌握信息技术已经成为每一个人最基本的需求。信息技术的核心是计算机技术，计算机应用技能的培养离不开计算机教育。随着科教兴国战略的实施和社会信息化进程的加快，我国高等教育事业的发展驶入了快车道，计算机教育改革也日益受到更加广泛的重视，许多高等学校都把计算机教育“四年不断线”列为教育改革的方向。而计算机技术的核心是程序设计，计算机教育就是围绕程序设计展开的。程序设计的过程贯穿了阅读、判断、分析、思考、抽象、综合、工具、环境等多项技能，对计算机技能的培养至关重要。

理论研究与教学实践都已表明，大学的第一门程序设计课程必须从程序设计领域最基本、最重要的问题出发，也就是要求学生掌握最基本的概念、最基本的思考问题方式和可能使用的技术。

什么是程序设计的基本概念呢？一般来说，这些概念应该包括数据及其表示、变量的类型和值、基本命令（语句）、流程控制结构、子程序（函数与过程）抽象、循环、接口（界面）与实现的分离与相互关联、复杂数据的组织、程序的复杂性及其控制（程序组织）等。程序设计课程应该围绕这些基本概念展开，帮助学生掌握这些概念，并基于这些概念，在使用某种程序设计语言解决实际问题的过程中学习程序设计。

当然，程序设计课程的重点不是单纯地讲授程序设计语言的理论知识，而是以某种程序设计语言为工具，讲授程序设计的基本思想、方法和技术，让学生掌握用编程工具解决实际问题的能力。

对于大多数学生而言，学习程序设计语言就如同学习外语，掌握基本的语言要素（如语法、词法等）就已经比较困难了，还要灵活地使用语言，也就是听、说、读、写，当然更加困难。面对这一问题，许多学者结合教育理论和教学经验提出了多种不同的应对方法，最有影响的是案例教学法，即把程序设计的基本概念由浅入深地融入若干程序“案例”中，通过分析、设计、总结，让学生在不断尝试“编程”的同时，学习程序设计的基本知识，理解程序设计的基本思想，掌握程序设计的基本方法。案例教学突出了实践的重要性，强调让学生在编写程序的实践中逐渐增加成就感、培养学习兴趣，通过形象思维逐步加深理解、巩固知识。

目前，选择 C 语言作为第一门程序设计语言是最普通的做法，这得益于 C 语言所具有的自由的书写格式、良好的表达能力、丰富的数据结构、结构化的程序特征等优势。C 语言具有与汇编语言一样的效率，便于与硬件技术的融合；具有与 C++、Java 等相似的风格，便于用户进一步学习。这些都促成了 C 语言在计算机教学中的特殊地位。

但由于 C 语言涉及的概念较多，语法规则比较繁杂，特别是 C 语言的数据类型、输入和输出等都普遍具有低级语言的特征，与计算机系统的关系密切，对于缺乏计算机基础知识的初学者来说，容易引起混乱。这也是造成 C 语言“难学”的主要原因之一。国内很多学者都对 C 语言的教学进行了研究，并在此基础上编写了许多很有价值的教材和辅导材料，取得了可喜的成绩。

本教材是作者在总结了十几年的教学经验后编写的，融合了许多对于 C 语言教学的认识

和思考。在内容选择和结构组织上，试图体现以培养程序设计能力为核心，以C语言基础知识、算法基本概念和程序基本结构为重点的教学理念。教材中引入了大量应用实例，侧重实例分析，避免了过多地罗列C语言的语法规则。在实例选取上，既考虑了实例的典型性，又考虑了实用性和趣味性，实例分析的重点也放在了程序设计的思想方法上，力求做到引人入胜，不断增加读者的编程兴趣。当然，作为教材，本书也特别注意了内容选取的科学性和组织的有效性。

本书的前5章是基础部分，介绍C语言的基础知识和程序的基本结构。第6～12章是重点部分，侧重程序设计实践。其中，第6～8章的核心是数据的表示和处理，首先介绍了数组和指针的概念和用法，然后在第8章集中介绍字符串的处理方法，体现了C语言的优势和特点。第9～11章以函数为核心，通过程序组织、模块化设计、数据组织等的学习，培养学生应用程序设计解决实际问题的能力。第12章介绍文件的操作。出于对部分理工科专业的特殊需要的考虑，第13章介绍了C语言的位运算。

本书配套《C程序设计语言实验与习题指导》，并提供配套多媒体电子课件、习题解答和程序源代码，以利于教师备课和学生自学。请登录华信教育资源网（http://www.huaxin.edu.cn或http://www.hxedu.com.cn）注册下载。

全书共分为13章，第1～4章由于广斌老师编写，第5～8章由魏东平老师编写，第9～13章由朱连章老师编写，全书由魏东平老师统稿。中国石油大学的李宗民教授在百忙之中对全书进行了审阅。在本书编写期间，中国石油大学的葛元康、李克文、孙东海等老师提出了很多宝贵的意见和建议，电子工业出版社的王羽佳编辑在成书过程中也做了大量工作，在此一并表示感谢。

本书中的不足之处，竭诚希望得到广大读者和同行的批评指正。

作　者

目　录

第 1 章　C 语言概述

学习目标

通过本章学习，应该掌握：

- C 语言程序的基本特征
- C 语言程序的运行环境

1.1　程序设计与程序设计语言

1.1.1　计算机与程序设计

半个世纪以来，计算机技术无论作为科学学科，还是作为现代产业，都已从一颗幼苗成长为枝繁叶茂的参天大树。回顾其发展历程，计算机也许是人类 20 世纪带给 21 世纪的最有价值的礼物，是人类文明历史上最伟大的发明之一，现在估计它对人类生活将会产生多么大的影响也许还为时尚早。目前，计算机可以在怎样的程度上延长或代替大脑的活动，计算机可以在何种程度上被广泛而深入地应用于各个领域，谁也不能指出一个“到顶”不再发展的时间。不过现在可以指出的是，使计算机具有如此影响力的根本原因是，计算机不是一个一次性的直接服务产品，它为人类服务是有条件的，这个条件就是程序和程序设计。

那么，对计算机而言，程序是什么呢？人们要让计算机解决一个问题时，需要把解决这个问题的步骤通过一条条指令的形式告诉计算机。一般，把人们事先准备好的、用来指挥计算机工作的描述工作步骤的指令序列称为程序，把程序员设计编写程序的过程称为程序设计。用来编写程序的语言称为程序设计语言。没有程序和程序设计，计算机就是一堆废物，也就是说，程序（软件）是计算机的必要组成部分。

计算机首先要求人们不断地在程序设计上付出大量的创造性劳动，然后才能享受到它的服务。计算机好像是唯命是从的仆人，严格地按照程序规定的步骤完成任务。为计算机编写程序是一项非常复杂和具有挑战性的工作。也可以说，自计算机问世的半个世纪以来，人们都是在研究设计各种各样的程序，使计算机完成各种各样的任务。程序设计是一项永无止境、极其困难复杂而又富有魅力和创造乐趣的工作，每年都吸引着数以十万计的优秀人才投入其中，促使计算机产业和计算学科取得了日新月异的发展。

1.1.2　程序设计语言的发展

程序设计的任务就是用程序设计语言编写程序，然后交给计算机去执行。在计算机应用的最初的十几年中，大多数计算机程序都是用机器语言编写的。这种“语言”虽然十分简单，机器可以“看”懂，但对于程序员来说却很不方便。于是，相继出现了汇编语言和高级语言。

1. 机器语言

在20世纪50年代，计算机专家们直接使用计算机能识别的指令系统（称为机器语言）来编写程序。由于机器语言是由二进制代码组成的，人们编写或阅读程序都十分困难，又容易出错，而且不同机器的指令系统也不同，很难进行交流（在一种计算机上调试通过的程序不能到另外一种计算机上运行）。机器语言程序虽然执行效率很高，但花费在程序设计和调试程序上的时间太多，这样就降低了整个解决问题的效率。

2. 汇编语言

人们为了解决二进制代码编程带来的困难，采用了助忆码和符号地址来代替机器语言中的二进制指令代码和指令地址，然后通过一个叫做汇编程序的翻译程序翻译成机器语言程序，再让计算机执行。这种采用助忆码和符号地址的语言称为汇编语言，汇编程序的翻译方式称为汇编。用汇编语言编写的程序的执行效率与机器语言程序一样高，且其可读性提高了，因此，到了20世纪60年代，用机器语言编程已经比较少了。但是，汇编语言也是面向机器的语言，同样不利于交流。

3. 高级语言

随着计算机各种技术的发展，程序设计语言也在不断发展，为了脱离机器和便于非计算机专业人员的使用，人们开始用一种比较接近于自然语言（主要指英语）和数学语言的语言来编写程序，这样的语言称为高级语言。高级语言克服了面向机器语言的缺点，使得程序易读、易维护、易交流。高级语言发展很快，如今已达数百种之多，常用的高级语言如下。

① FORTRAN语言：诞生于20世纪50年代中期，是第一个算法语言，适合于科学和工程计算。

② BASIC语言：诞生于20世纪60年代中后期，简单易学，适合初学者学习。

③ Pascal语言：诞生于20世纪70年代初，是一门结构化程序设计语言，适合于教学、科学计算、数据处理和系统软件开发等。

④ C语言：诞生于20世纪70年代初，20世纪80年代开始风靡全世界，适合于系统软件、数值计算、数据处理等。

⑤ Java语言：诞生于20世纪90年代，是一种新型的跨平台分布式程序设计语言，具有简单、安全、稳定、可移植性强等特点，在网络环境中得到了广泛的应用。

一般地，人们把用高级语言或汇编语言编写的程序称为源程序。像用汇编语言编写的源程序必须经过汇编过程一样，用高级语言编写的源程序同样需要经过一个类似的过程翻译成计算机能识别的二进制代码程序，才能让计算机执行。翻译成二进制代码的方式通常有两种：编译方式和解释方式。目前，大多数常用的高级语言都采用编译方式。

编译方式是将源程序全部翻译成功能等价的机器语言程序（一般称为目标程序），然后经过连接程序将用户的目标程序与系统配置好的一些通用程序连接装配在一起，形成可执行程序，最后让计算机执行。编译方式经编译和连接得到的可执行程序可以重复执行无数次，其效率很高。

解释方式是通过解释程序逐句翻译、执行的，不产生目标代码程序，每次执行都要重新翻译，所以对那些重复执行的程序效率较低。

1.1.3　程序设计方法

程序设计技术的发展，是一个逐步提高的过程，是一个与实际应用需要互相制约和促进的螺旋式的发展进程。程序设计技术的不断进步，导致了计算机应用水平的飞跃；反之，计算机应用领域的扩展、软件规模和复杂度的提高又促进了程序设计技术的升级。在这个过程中，主要出现了结构化程序设计（SP，Structured Programming）方法和面向对象程序设计（OOP，Object-Oriented Programming）方法。

1．结构化程序设计

高级语言的开发和使用，使计算机的应用进入了一个新时期。20 世纪 60 年代出现的“软件危机”和关于“goto 语句应从高级语言中去掉”的讨论，促使人们重视对程序设计方法学的研究。随着一些规模大、复杂度高、使用周期长，以及投入人力、物力较多的大型程序设计任务的提出，程序设计的目标把可靠性、可维护性的要求放在了比高效率更重要的位置上，并促使人们研究程序设计的方法和风格，最终形成了结构化程序设计的基本思想。

结构化程序设计是一种设计程序的技术，主要采用自顶向下、逐步求精的设计方法和单入口、单出口的控制结构。自顶向下、逐步求精的设计方法符合人们解决复杂问题的普遍规则，可以提高软件开发的成功率。由此开发出的程序具有清晰的层次结构，易于阅读理解和修改调试扩充。结构化程序设计只包括顺序、选择和循环 3 种基本控制结构，并建议程序员不用或少用 goto 语句。

2．面向对象程序设计

结构化程序设计对于确定的问题来说，无疑是一种很好的方法，但对于千变万化的世界，特别是当今信息化的世界，一个成为产品的程序很难适应形势的变化，因为在程序的设计阶段问题是确定的，其数据结构、数据模型也可以确定，所以设计出的程序使用、操作没有问题，但当表示问题的数据结构发生变化时，程序就可能会出现问题，而维护起来也相当困难。原因就是当初的设计是面向数据进行自顶向下设计的，要修改就需要全部修改。本来客观世界就是由许许多多、各种各样的对象组成的，每个对象都有各自的内部状态（属性）和运动规律（行为方式），不同对象的相互作用和联系构成了各种各样不同的系统，构成了我们所面对的客观世界。如果按照客观世界对象的特点进行程序设计，就会适应形势的发展，这就是自 20 世纪 80 年代开始流行，现已广泛应用的面向对象的程序设计方法。

面向对象的程序设计方法吸取了结构化程序设计的基本思想和主要优点，将数据与对数据的操作放在一起作为一个相互依存、不可分割的整体来处理，这个整体称为对象。对象的内部状态（属性）用数据表示，运动规律用所谓的方法（也称成员函数）表示。对象是一个整体，所以我们说把数据隐藏在这个整体中了。对象之间的相互作用通过所谓的消息与对象的对外接口（成员函数的调用形式）实现。这个系统通过不断地向对象发送消息而使对象从初始状态到达终止状态，从而实现了问题的求解。

如果问题的数据结构发生了变化，只要描述问题的对象的对外接口不变，所有涉及这些数据结构的地方都不需要修改，而只需修改对象中的方法就行了。

对象是各种各样的，但总有相似性。例如，一匹白马和一匹黑马是两个对象，但它们都

具有马的特性，只是颜色有差异而已。将对象进行抽象、划分便得到了类。类描述了属于该类型的所有对象的性质，包括外部特征和内部实现，实际上是抽象数据类型的具体实现。反之，对象是类的具体实例。

20世纪90年代的程序设计方式，由于有了面向对象程序设计技术的支持，发生了质的变化，未来的软件开发将向着可重用部件的组装方式发展，而可重用部件的存在形式就是类及其派生系统。

1.2　C语言的产生与发展

C语言是在20世纪70年代初问世的。1978年，美国电话电报公司（AT&T）的贝尔实验室正式发表了C语言。同时，由Brian W. Kernighan和Dennis M. Ritchie合著了著名的《The C Programming Language》，通常简称为《K & R》，也称为《K & R》标准。此书介绍的C语言成为后来被广泛使用的C语言版本的基础，被称为标准C。但是，在《K & R》中并没有定义一个完整的标准C语言，因而出现了许多不同的C语言版本。后来，根据这些版本对C语言的扩充和发展，美国国家标准协会（ANSI，American National Standards Institute）重新制定了新的标准，并于1983年发表，通常称为ANSI C。1988年，按ANSI C标准又重写了《The C Programming Language》一书。很多C语言教材都是以ANSI C为基准编写的。目前，广泛流行的各种C语言版本的编译系统的基本内容是相同的，只是在个别地方有所不同。

早期的C语言主要用于UNIX操作系统。由于C语言的强大功能和各方面的优点逐渐被人们所认识，到了20世纪80年代，C语言开始进入其他操作系统，并很快在各类大、中、小和微型计算机上得到了广泛的使用。特别是微型计算机上的C语言的普及，反过来又极大地推动了C语言的发展。

C语言的产生和发展与UNIX操作系统是密不可分的，可以说，C语言就是为编写UNIX操作系统而设计并加以实现的。UNIX操作系统的源代码有90%以上是用C语言编写的，它的流行应归功于C语言。

其实，C语言并不是孤立产生的，它是在B（BCPL的第一个字母）语言的基础上发展起来的，而B语言又是在A（ALGOL）语言基础上发展而来的。1960年出现的ALGOL 60与硬件相差甚远，不宜用来编写系统程序。1963年，英国的剑桥大学推出了比较接近于硬件的CPL（Combined Programming Language）语言，该语言规模较大，不宜实现。1967年，剑桥大学的Martin Richards对CPL进行简化推出了BCPL（Basic Combined Programming Language）语言。1970年，美国贝尔实验室又在BCPL语言基础上进一步简化推出了B语言，但B语言的功能太简单。于是，1972年，贝尔实验室的D. M. Ritchie又在B语言的基础上推出了C（BCPL的第二个字母）语言。

C语言为UNIX系统而设计，又由于UNIX系统的日益广泛使用而迅速得到推广，到20世纪80年代，C语言已风靡世界。

1.3　C语言的特点

C语言之所以风靡世界，是因为它有许多不同于其他语言的特点。在这些特点中，大部分是优点（与其他语言比），也有一少部分，对初学者来说，如果运用不当则会适得其反。

① C 语言简洁、紧凑，使用方便、灵活，库函数丰富、实用。ANSI C 一共只有 32 个关键字和 9 种控制语句，压缩了一切不必要的成分，用简单、规整的方法就可以构造出相当复杂的结构。

② C 语言程序书写格式较自由，降低了格式要求，从而降低了程序员的劳动强度。

③ C 语言的数据类型丰富，同时具有现代化程序设计语言普遍配置的多种数据结构。C 语言具有整型、实型、字符型、数组、指针、结构体、共用体和枚举类型等，能用来实现各种复杂的数据结构（如链表、树等）。特别是与地址密切相关的指针，使用起来比其他高级语言更加灵活多样。但指针对于初学者来说也是较难理解的。

④ C 语言的运算符十分丰富，包括算术运算符、关系运算符、逻辑运算符、位运算符、指针运算符、地址运算符等，还把括号、逗号、赋值号、强制类型转换等都作为运算符处理，连同复合赋值运算符共有 44 种运算符。灵活使用这些运算符可以把非常复杂的运算逻辑用最简单的方式表达出来，使得程序更加简洁。

⑤ 具有编写结构良好的程序所需要的各种控制流结构。C 语言具有实现顺序、分支和循环 3 种基本结构的控制语句。而且，C 语言的函数结构非常便于采用自顶向下、逐步求精的结构化程序设计方法将整个系统划分成若干功能相对独立的模块。C 语言是结构化程序设计的理想语言。

⑥ C 语言的目标代码质量高，执行效率高。一般，C 程序只比汇编程序生成的代码效率低 10%～20%，是各类高级语言中最快的。

⑦ 可以直接对硬件进行操作。C 语言允许直接访问物理地址，能实现汇编语言所能实现的大部分功能，如地址处理、二进制数位运算及指定用寄存器存放变量等。在硬件技术领域，C 语言有着非常广泛的应用基础，是最主要的程序设计语言。

由于 C 语言既具有高级语言的特点，又具有低级语言的特点，因此，也有人把它称为中级语言，特别适合作为编写系统程序和各种应用软件的工具。

⑧ 可移植性好。虽然 C 语言具有低级语言的功能，但与汇编语言相比，它不依赖于机器硬件，在硬件结构不同的各种型号的计算机之间不做修改或稍做修改即可实现程序的移植。

⑨ 语法限制不太严格，程序设计的自由度大。例如，对数组下标越界不做检查，由程序员自己保证其正确性；对变量类型的使用比较灵活，整型和字符型数据在一定范围内可以通用；等等。大多数高级语言编译程序对语法检查都比较严格，但 C 语言放宽了语法检查，允许程序员有较大的自由度，这就要求程序员必须具有一定的编程和调试程序的素质。

1.4　C 语言程序简介

学习 C 语言的目的就是根据实际问题设计 C 语言程序，那么，C 语言程序是怎样构成的呢？为了了解 C 语言程序的构成，下面先来看几个例子。

【例 1.1】　在屏幕上显示字符串“This is a C program.”。

程序如下：

```
1   #include <stdio.h>          /* 文件包含*/
2   void main()                 /* main 是主函数的函数名，表示这是一个主函数*/
3   {                           /* 函数体开始*/
```

```
4       printf("This is a C program.\n");
5   }                          /* 函数体结束*/
```

这是一个只由 main 函数构成的C语言程序，运行后，将在计算机屏幕上显示如下结果：

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

为了更好地对C语言程序进行说明，本书将在程序的每行前都加上行号。注意，C语言程序事实上是没有行号的。

该程序的第1行是文件包含的预处理命令，将头文件“stdio.h”的内容包含到该程序中，由于该头文件中包含了输出函数的信息，因此在该程序中就可以调用输出函数了。第2行是主函数 main 的说明部分，void 表示函数类型，说明该函数没有返回值，main 是函数名。需要注意的是，在任何一个 C 语言程序中必须有 main 函数，而且只能有一个。第3～5行是 main 函数的函数体，函数体以第3行的左花括号“{”开始，以第5行的右花括号“}”结束。main 函数的函数体只有一条语句，即第 4 行，它是一个函数调用语句，调用了标准库函数 printf，用于在屏幕上输出一个字符串“This is a C program.\n”，其中的“\n”是一个特殊的控制符——换行符，表示在输出最后一个字符“.”后换行，相当于按了回车键。

【例 1.2】　求整数 10、20 的和。

程序如下：

```
1   #include <stdio.h>
2   void main()
3   {
4       int first,second,sum;          /* 定义变量*/
5       first=10;                      /* 给变量赋值*/
6       second=20;
7       sum=first+second;      /*求和，将 first 和 second 的值相加后赋给 sum*/
8       printf("sum=%d\n",sum);        /* 输出 sum 的值*/
9   }
```

这还是一个只有 main 函数的 C 语言程序，运行结果如下：

```
sum=30
```

该程序的第4行定义了3个整型变量，即在内存中开辟了3个用于存放整型数据的存储单元。第5行和第6行都是赋值语句，分别将整数 10 和 20 赋给整型变量 first 和 second，即将 10 和 20 分别存入分配给 first 和 second 的内存单元中。第7行也是赋值语句，取整型变量 first 和 second 的值（即从分配给 first 和 second 的内存单元中分别取出 10 和 20）相加后，把和赋给整型变量 sum。第8行调用 printf 函数，输出 sum 的值。

【例 1.3】　找出任意两个整数中较大的数。

程序如下：

```
1   #include <stdio.h>
2   int max(int x,int y)               /* 定义 max 函数 */
3   {   return(x>y?x:y);               /* 求出两数中的较大数并返回 */
```

```
4   }                                   /* max 函数结束 */
5   void main()
6   {   int num1,num2,m;
7       printf("Input the first integer number: ");   /* 提示输入第 1 个整数 */
8       scanf("%d",&num1);              /* 从键盘上输入第 1 个整数 */
9       printf("Input the second integer number: "); /* 提示输入第 2 个整数 */
10      scanf("%d",&num2);              /* 输入第 2 个整数 */
11      m=max(num1,num2);               /* 调用 max，计算两个数中的较大数 */
12      printf("max=%d\n",m);           /* 输出结果 */
13  }
```

这是一个由 main 函数和 max 函数构成的 C 语言程序，运行情况如下：

```
Input the first integer number: 6 ↙①
Input the second integer number: 9 ↙
max=9
```

该程序包括两个函数，第 2～4 行构成 max 函数，第 5～13 行构成 main 函数。第 2 行是 max 函数的说明部分，包括函数类型（int）、函数名（max）、形式参数及其定义（int x, int y）。第 3 行是 max 函数的函数体，是一条 return 语句，用于返回给调用函数一个整型值，该值是条件表达式“x>y ? x : y”的结果，即如果 x>y，则返回 x 的值，否则，返回 y 的值。第 7 行是输入提示，用于提示用户输入程序运行所需的数据，请读者注意体会和使用。第 8 行是函数调用语句，调用了标准库函数 scanf（该函数和 printf 函数是系统提供的标准输入和输出函数），用于从键盘上输入一个整型数据赋给整型变量 num1。第 9 行和第 10 行分别与第 7 行和第 8 行类似。第 11 行是一条赋值语句，赋值号“=”的右边调用了 max 函数，将 max 函数的返回值赋给赋值号“=”左边的整型变量 m。第 12 行是函数调用语句，调用了标准库函数 printf，输出 m 的值。

这 3 个程序中出现了数字、字母及一些符号，它们都属于 C 语言的字符集。程序中的“int”、“return”等是 C 语言中固定的具有特殊意义的词，称为关键字。“first”、“second”、“sum”、“num1”、“num2”等是用户给变量起的名字，称为标识符。“>”、“+”、“=”等是 C 语言的运算符。“,”、“;”、“"”、“{”、“}”等是 C 语言的分隔符。

1. 字符集

字符是组成语言的最基本的元素。在 C 语言程序中，只能使用 C 语言字符集中的字符，不能使用字符集以外的字符。

C 语言的字符集（采用 ANSI 字符集）主要包括英文字母、阿拉伯数字和特殊字符 3 类：

① 大、小写英文字母（52 个）：A、B、C、…、Z、a、b、c、…、z

② 阿拉伯数字（10 个）：0、1、2、…、9

③ 特殊字符（29 个）：= + - * / % # ^ & ~ ( ) _ [ ] { } : ; , ' " ? . < > | ! 空格

① 带下划线的是输入部分，“↙”代表回车。