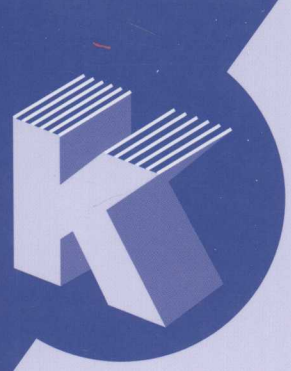


高等学校计算机基础课程教材

Programming with C Language

C语言程序设计

主　编　方娇莉
副主编　潘晟旻

高等教育出版社
HIGHER EDUCATION PRESS

高等学校计算机基础课程教材

C 语言程序设计

C Yuyan Chengxu Sheji

主　编　方娇莉
副主编　潘晟旻

高等教育出版社·北京
HIGHER EDUCATION PRESS　BEIJING

内容简介

本教材以提高学生计算思维能力为导向，以培养学生编程能力为目标，通过多元算法、程序重构等方法引导学生理解计算的实现机制、构筑计算思维模式下的实践与创新能力。

全书共分13章。第1章介绍C语言的入门知识，第2～4章介绍顺序结构、选择结构和循环结构，第5章介绍数组，第6章介绍函数，第7章介绍用户自定义数据类型，第8章介绍指针，第9章介绍文件，第10章介绍位运算，第11章介绍面向对象的程序设计方法，第12章介绍与程序设计有关的基础知识，第13章介绍C语言程序设计编程综合应用。

本书适合作为高等学校计算机基础课程教材及相关专业程序设计课程的参考教材，也可作为全国计算机等级考试（二级C）的参考用书。

图书在版编目(CIP)数据

C语言程序设计/方娇莉主编. --北京:高等教育出版社,2013.3

ISBN 978-7-04-036922-9

Ⅰ. ①C… Ⅱ. ①方… 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(2013)第022343号

策划编辑 何新权　责任编辑 何新权　封面设计 赵 阳　版式设计 于 婕
插图绘制 尹文军　责任校对 刘丽娴　责任印制 刘思涵

出版发行 高等教育出版社
社　　址 北京市西城区德外大街4号
邮政编码 100120
印　　刷 山东鸿杰印务集团有限公司
开　　本 787mm×1092mm 1/16
印　　张 20.25
字　　数 490千字
购书热线 010-58581118
咨询电话 400-810-0598
网　　址 http://www.hep.edu.cn
http://www.hep.com.cn
网上订购 http://www.landraco.com
http://www.landraco.com.cn
版　　次 2013年3月第1版
印　　次 2013年3月第1次印刷
定　　价 35.80元

物料号 36922-00

前　言

C语言与通信卫星、移动电话等一并被誉为贝尔实验室改变世界的十大发明，从诞生至今近半个世纪一直受到全世界程序员的青睐。C语言结构简单、数据类型丰富、运算灵活方便，用它编写的程序具有速度快、效率高、代码紧凑、可移植性好等优点，能够有效地用来编制各种系统软件和应用软件。从Internet核心开发到驱动“好奇号”火星车，C语言遍及全球、跨越星际。因其在操作系统、工业控制、人工智能、数据处理等诸多领域的广泛应用，“C语言程序设计”已成为计算机专业和非计算机专业首选的计算机语言课程。

本教材以国际标准C语言(ANSI C)的知识和结构为基本内容，以普及率甚高的Visual C++ 6.0为编译系统，全面系统地介绍了C语言及其程序设计方法，所有例题均通过调试。

针对初学者的特点，本书对内容进行了精心安排，用读者容易理解的方法去组织教材、介绍知识。为了方便读者学习，本教材还配套由楼静主编的《C语言程序设计习题与上机实践》，让读者加深和巩固所学知识。为方便教学，本书有配套的PPT课件资源，可发电子邮件到fangjiaoli@163.com索取。在使用本书过程中，如有任何问题或建议可以与作者联系。

本教材具有如下特点：

1. 内容安排合理，各章衔接紧密。以C语言的最基本内容为主线，深入浅出，通俗易懂，难点分散，使初学者能够很好地掌握课程的基本内容。

2. 各章均包含了C语言语法知识之外的成熟的程序设计思想、程序设计理念和程序设计方法，可引导读者少走弯路，尽快掌握运用C语言进行程序设计的思想精髓。

3. 设置的典型例题有较好的示范作用；精选的习题有利于读者掌握知识点和考点；丰富的实训内容有利于培养读者调试程序的能力，进而全面掌握C语言的知识。

本书共分13章，分3个层次：第1～4章是基础层次，第5～9章是提高层次，第10～13章是深入层次。其中，基础层次和提高层次的理论教学大约需要32学时，深入层次则视学生的接受情况和具体学时数而定。

本书第1章由普运伟编写，第2、4章由方娇莉编写，第3、8章由楼静编写，第5章由郭玲编写，第6、10章由潘晟旻编写，第7章由马晓静编写，第9章由王凌编写，第11章由耿植林编写，第12章由黎志编写，第13章由付湘琼编写。本书由方娇莉任主编并负责统稿，潘晟旻任副主编。

本书在编写过程中得到了昆明理工大学教务处、部门领导、教学团队和许多同行的大力支持和帮助，在此表示感谢！同时，对本书所参考的文献、资料的作者表示感谢！

衷心感谢高等教育出版社的编辑，感谢他们为本书顺利出版而付出的辛勤汗水。

作为本书的读者，您的批评、建议和评价是最为宝贵的，我们殷切地期望您对本书提出中肯的意见，以便本书今后的完善。

编　者

2013年1月

目　录

第1章 认识C语言

计算机是现代信息处理的重要工具，而所有的信息处理任务都是通过软件操控硬件的方式来实现的。各式各样的软件极大地丰富了计算机的功能，不断拓展其应用领域，帮助人们解决了众多生产、生活和学习中的实际问题，甚至帮助我们完成了许多手工方式难以完成的工作和任务。

C语言作为当今最为流行的程序设计语言之一，被广泛应用于各行各业的程序设计中。本章首先简要介绍计算机语言及程序设计技术的发展情况，然后重点介绍C程序的基本结构、操作过程、C语言基本语法元素及C语言的编程规范。

1.1 计算机语言及程序的发展

软件的主体是程序，程序是指挥计算机进行各种信息处理任务的一组有序的指令。或者说，程序是能实现特定功能的一组指令序列。计算机程序一般采用某种计算机语言并根据特定的方法和步骤编写而成。

1.1.1 计算机语言的历史

计算机语言是人类和计算机进行相互“交流”的语言。自计算机诞生以来，人们已先后发展了数以千计的计算机语言。通常，根据这些语言和计算机的交互方式和特点，可将其分为机器语言、汇编语言和高级语言三种。

1. 机器语言

计算机只能识别由0和1组成的机器代码。不同的计算机具有不同的指令集合，亦即具有自身可以理解的机器语言。机器语言是和特定硬件设备相关，指示计算机一次完成一个最基本操作的二进制数字串。可见，机器语言具有计算机能够直接识别并执行的优点，但和具体的机器设备相关。为一台IBM PC机编写的机器语言程序，无法移植到一台Apple MAC机器上，甚至不能在另一台不同型号的IBM PC机上执行。若有变量A，将其加上5后再存储回去，某台计算机的机器语言程序可能如下：

```
00100100 01000000
00000010 00000101
00110000 01000000
```

可见，机器语言编写的程序较难理解，极易出错，非专业程序员难以掌握。

2. 汇编语言

汇编语言采用类英语单词助记符来编写程序，使人们摆脱了枯燥、繁琐、艰涩的0、1编程方式。这大大降低了编程的难度，程序的可读性也得到明显提高。但由于计算机并不能直接识别程序中所用的助记符，因此用汇编语言编写的程序（称为汇编源程序）必须通过一个被称为“汇编程序”的中间翻译工具进行转换，得到计算机能够直接执行的机器语言程序，这个过程通常称为“汇编”。值得注意的是，汇编语言的助记符实质上是对特定CPU内部指令的一种简略记法，因此和机器语言一样，汇编语言也是和硬件平台相关的。因此，人们通常将机器语言和汇编语言统称为低级语言。相对机器语言而言，汇编语言使得程序较为清晰、可读。例如，下面的汇编源程序片段同样将变量A的值加上5后再存储起来：

```
LOAD    A
ADD     5
STORE   A
```

3. 高级语言

汇编语言编写的程序较为清晰，但编程时依然采用计算机执行任务时所采取的详细步骤，这使得要完成较为复杂的任务仍需编写较多的代码。为了进一步提高程序的开发效率，人们开发出了类似人类自然语言（主要指英语）的高级语言编程方式。对于“将变量A的值加上5后再存储回去”这样一个问题，很多高级语言程序采用下面的语句：

```
A=A+5;
```

显然，用高级语言编写的程序更为简洁，可读性更好。但和汇编语言源程序一样，计算机并不能直接识别高级语言源程序，必须通过中间翻译工具处理后才能转换成计算机可识别的二进制代码，即机器指令。

对于高级语言源程序，通常采用“解释”和“编译”两种方式将其转换为机器语言程序。其中，解释方式是一种“边翻译、边执行”的过程，类似日常生活中的“口译”，是一种对源程序进行逐条解释为机器指令并立即执行的方式，最终不产生任何目标程序。而编译方式是一种“整体翻译”的过程，类似日常生活中的“笔译”，该方式通过编译程序将高级语言源程序转换为等价的用机器语言表示的目标程序，然后再执行。一般而言，解释方式占用内存少，易于查错和移植程序，但执行速度较慢；而编译方式执行速度较快，但每次修改源程序后必须要重新进行编译处理。C语言是一种功能强大的编译型高级程序设计语言。

1.1.2 C语言的特点

C语言是由贝尔实验室的Dennis Ritchie于1972年开发出来的，当时他正与B语言的创建者Ken Thompson为一台DEC PDP-11计算机设计UNIX操作系统。正因为UNIX的巨大成功，C语言作为一种实用、高效的开发语言迅速流行起来。尽管时至今日，一些开发人员认为C语言已经过时，转而使用更为强大的C++、C#、Java、PHP等语言，但C语言却凭借自身的简洁高效、广泛的硬件平台支持等众多优势依然成为许多实际开发项目的首选语言。况且，当今的许多主流程序设计语言大多承袭了C语言的很多特点。因此，学好C语言及其相关的编程知识，是进一步学习其他高级程序设计语言的基础，也是叩开编程开发之门的主要通路。下面先来看看C语言的几个显著特点。

1. 高效性和快速性

C 语言是在实际编程过程中开发出来的一种实用语言，具有高效、快速的特点。用 C 语言编写的代码十分紧凑，执行速度较快。C 语言程序甚至可以达到汇编语言的执行效率和最大内存利用率。

2. 功能强大且兼顾灵活性

C 语言把高级语言的基本结构和低级语言的实用性紧密结合起来，这使得 C 语言不仅适于编写各种应用软件，还适于编写类似 Unix 这样的系统软件，以及诸如 Fortran、Perl、Pascal、Lisp 和 Basic 等多种其他高级语言的编译器和解释器，实现对硬件的直接访问和操作。同时，C 语言的灵活性使得它在处理许多任务时表达方式极其简洁。

3. 结构化和模块化特性

C 语言是一种结构化的程序设计语言。结构化的程序设计方式使得程序层次清晰，逻辑性强，便于阅读和调试。同时，C 语言采用“函数”方式进行模块化设计，这不仅使编写的程序更可靠、易懂、易于维护，而且增强了代码重用的可行性，为编写复杂程序奠定了坚实的基础。

4. 可移植性

C 语言是一种与硬件较少关联的高级语言，且 C 编译器在不同计算机系统上广泛使用，这使得用 C 语言编写的程序只需作很少改动或不作任何改动便可以在各种不同的计算机上运行。当然，针对特定操作系统或硬件设备编写的程序代码，通常是不能移植的。

1.1.3 程序设计技术的发展

正如前述，程序是完成特定功能的一组指令序列，这组指令根据既定的程序逻辑控制计算机的运行。实际上，人类用程序指挥计算机工作的思想早于世界上第一台计算机 100 余年便已产生。1843 年，英国著名诗人拜伦的女儿艾达（Ada Lovelace）为巴贝奇的分析机编写了世界上第一个计算机程序，并提出“变量”、“算法”和“程序流程”等概念。也正因为如此，Ada 被公认为世界上第一位计算机程序员，以至于后来美国国防部基于 Pascal 语言开发的一种结构化和多任务的语言被命名为 Ada 语言。

自计算机诞生以来，人们使用的编程工具经历了从低级语言到高级语言的演变，程序设计方法也从最初的面向计算机的程序设计逐渐发展为面向过程和面向对象的程序设计。

在计算机刚被发明的最初十年，计算机还是非常昂贵的奢侈品，各种硬件设备性能极为低下，程序员采用机器语言或汇编语言为计算机设计程序，必须考虑计算机实现特定任务所必须采取的操作步骤，即以计算机的工作方式来思考和组织程序代码，同时还要以提高执行效率、少占内存作为程序设计的目标。

进入 20 世纪 60 年代，计算机硬件性能有了很大提高，应用领域不断拓展，人们对软件的需求越来越复杂，面向过程的程序设计方法随之应运而生。面向过程的程序设计方法以程序的可读性、清晰性和可维护性为目标，采用结构化和模块化设计思想，易于通过自顶向下和逐步求精的规划解决各种复杂问题。本书介绍的 C 语言便是一种典型的面向过程的结构化程序设计语言。

20 世纪 80 年代后期，软件的规模更加扩大，大型软件尤其是 GUI 界面软件的开发变得越来越困难，这直接导致面向对象的程序设计方法的流行。面向对象的程序设计方法采用客观世界

的描述方式，以类和对象作为程序设计的基础，将数据和操作紧密连接在一起，通过继承、多态等特性，大大降低了程序开发的复杂性，提高了软件开发的可重用性和开发效率。C语言也通过扩展对面向对象机制的支持，衍生、发展出C++语言，详见第11章。

到这里，有必要提醒读者注意的是，不要误认为面向对象的程序设计方法一定比面向过程的程序设计方法优越。实际上，这两种方法各适用于不同的场合，互为补充。面向对象的程序设计方法被广泛运用于大型软件系统和GUI程序的设计，而面向过程的结构化程序设计在一些小型控制系统和嵌入式开发中具有无可比拟的优越性。另外，即使在面向对象设计中，具体到一个功能模块（常称为方法）的编写，也处处体现出结构化程序设计的思想。这或许也是我们必须学好C语言的重要目的之一。

1.1.4 算法及其表示

程序的设计要遵循一定的步骤。算法就是指为解决一个具体问题而采取的方法和步骤的集合。在日常的学习和生活中，算法的例子随处可见。求解一道数学应用题，要有清晰的思路和明确的步骤；使用《新华字典》查询一个生字，也要讲究是采用拼音检字法还是部首检字法，甚或是直接查询法等，这些都是算法的实例。

如果要编制程序让计算机帮助人们解决一个实际的问题，同样需要考虑解决问题的算法。不难想象，一个好的算法可以很快得到待解问题的答案，一个不好的算法却要耗费更多的时间和资源，而一个错误的算法则根本无法得到正确的结果。可见，正如著名计算机科学家Niklaus Wirth所言：算法是程序设计的灵魂，程序设计的关键在于算法。

1. 算法的基本特征

使用计算机程序求解问题，要有一个明确的起点，确定的步骤序列，程序终止执行时要能给出最后的结果。因此，一个算法应具备如下5个重要特征。

① 输入：0或多个输入，其中0输入表示算法本身已给出初始条件。

② 输出：1个或多个输出，没有输出的算法是毫无意义的。

③ 有穷性：算法的执行步骤是有限的，且每一步骤的执行时间是可容忍的。

④ 确定性：算法的每一步骤具有确切的含义，不允许出现歧义。

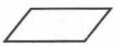
⑤ 可行性：算法的每一步操作都可以通过已经实现的基本运算执行有限次数来实现。

2. 算法的表示

在算法的设计过程中，有必要将算法的主要步骤清晰地记录和表示出来，这不仅有利于编程者之间相互交流所设计算法的主要思路，而且有利于算法后期的改进和优化。常见的算法表示方法有伪代码、流程图、N-S图和PAD图等，本书主要采用流程图来表示算法。

流程图是一种采用程序框、流程线及简要文字说明来表示算法的有效方法。其中，程序框图用于表示算法中的具体步骤，流程线表示算法的执行顺序。流程图中常用的符号如表1-1所示。

表 1-1 流程图中常用的符号

符号	名称	功 能
	起止框	表示算法的起始和结束,有时为了简化流程图也可省略
	输入/输出框	表示算法的输入和输出的信息
	判断框	判断条件是否成立,成立时在出口处标明“是”或“Y”;不成立时标明“否”或“N”
	处理框	赋值、计算。算法中处理数据需要的算式、公式等分别写在不同的用以处理数据的处理框内
	流程线	连接程序框,带有控制方向

例如,要输出两个数中的较大者,其算法流程图如图 1-1 所示[注]。

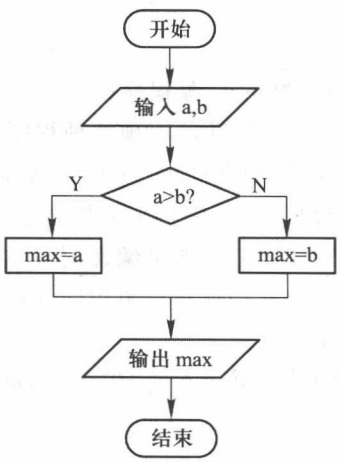


图 1-1 输出两个数中较大者的算法流程图

1.2 C 程序的操作过程

C 语言是一种编译型语言,从创建 C 程序到运行出最终结果,一般要经历 4 个主要的阶段:编辑、编译、链接和运行,图 1-2 给出了 C 程序的基本操作过程。下面以 Windows 平台下的 Microsoft Visual C++ 6.0 为例,具体介绍每个步骤的含义和操作方法。

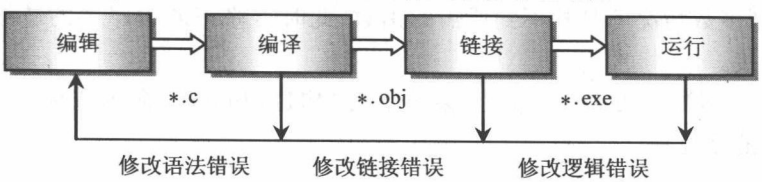


图 1-2 C 程序的基本操作过程

注:为与程序中的表现形式一致,本书中的变量及数学公式一律采用正体。

1.2.1 编辑

实际上,在正式编写程序代码之前,一般还需要对程序所要实现的目标进行分析,然后大致绘出程序算法的流程图,这对于大型程序和复杂系统的设计尤为重要。编辑C程序的过程,是指将程序算法用C语言进行实现,输入并保存为扩展名为.c的C源程序。这个过程一般可采用文本编辑器或专用的C编辑器完成。当然,以VC++ 6.0为代表的集成开发环境(IDE)提供了更大的方便,可实现图1-2所示的所有C程序操作过程,且可方便地对程序进行调试和测试。

要提醒读者注意的是,VC++ 6.0采用"工程"对程序项目进行管理。一个工程可以只包含一个.c文件,也可包含多个.c文件(但此时只有一个文件中含有main函数,有关main函数的介绍见1.3节)。因此,一般情况下,在编写一个C程序时,要先新建一个工程(如:文件→新建→工程→Win32 Console Application→输入工程名和保存位置),然后再创建C源程序(如:文件→新建→文件→C++ Source File→输入文件名和扩展名),最后进行保存。

1.2.2 编译

编译过程就是调用编译器将编写好的C源程序(*.c)翻译为本机目标代码文件(*.obj)的过程。在此过程中,C编译系统会首先自动调用预处理程序对C源程序进行一定的处理,这些处理通常包括把其他文件包含到要编译的文件中,或者使用程序文本替换专门的符号。在编译阶段,如果所编写的C源程序没有任何错误,则将生成扩展名为.obj的目标文件,该文件是用本机机器语言表示的代码,但还不能直接运行。如果编译时发生错误,则不会生成目标文件,此时编译器会给出错误提示信息,必须根据这些信息对源程序的语法错误进行修改,然后再重新进行编译,直到程序成功被编译通过为止。

在VC++ 6.0环境中,可采用"编译"菜单下的"编译(Compile)"命令或标准工具栏上的"编译"按钮对程序进行编译处理。

1.2.3 链接

链接过程是将目标代码文件(*.obj)和程序中使用到的其他代码连接在一起,以形成一个能在本地计算机上执行的文件(*.exe),该过程由链接器(Linker)负责完成。一般来说,C程序总会引用到标准库函数或其他地方的函数。其中,C标准库函数是C语言提供的一个大型函数库(参见附录C),包含许多有用的函数模块,这些函数能够满足编程人员大多数的要求,且其编写是严格和高效的;其他地方的函数主要指编程者自己或他人创建的函数,以满足一些特定场合的特殊要求。在链接阶段,如果程序被正常连接,则系统将会生成和源文件同名的可执行文件;如果链接失败,则多数情况下是找不到所需的函数,此时需修正源程序中的链接错误,然后再重新进行编译和链接。

在VC++ 6.0环境中,可采用"编译"菜单下的"构建(Build)"命令或标准工具栏上"构建"按钮对程序进行链接处理。

1.2.4 运行

运行过程是将链接得到的可执行文件(*.exe)加载入内存,由CPU进行执行处理,以测试

程序运行结果的过程。在此阶段,如果程序编写正确、无误,算法设计恰当、合理,则可得到想要的运行结果。相反,如果程序虽然能够运行,但执行结果却不正确,这多数是设计逻辑有误造成的。或者说,是因为算法设计有问题而导致的。这时,必须重新审视所设计的算法,分析程序执行的逻辑,以发现导致错误的原因,进而修改源程序中的运行逻辑错误,然后再重新进行编译、链接和运行测试过程。

在 VC++ 6.0 环境中,可采用“编译”菜单下的“执行(Execute)”命令或标准工具栏上“执行”按钮运行程序。同时,可以利用调试工具栏上的按钮对程序进行调试运行,以检查程序是否含有逻辑错误。

1.3 C程序的基本结构

通过前面的学习,我们已对程序设计的一些相关知识、C 语言的特点以及 C 程序的基本操作有了一定的了解,本节将通过几个简单的实例带你轻松迈入 C 语言的编程世界,并通过它们了解 C 程序的基本结构,熟悉 C 程序的操作过程。

【例 1.1】在计算机屏幕上输出一行文本“Welcome to C!”。

首先,在 VC++ 6.0 中新建一个工程(取名为 test1 或其他名称),然后在该工程中新建一个源程序(取名为 first.c 或其他文件名,但扩展名必须为 .c),之后在打开的编辑器窗口中输入如下的程序代码:

```
/*我的第一个C语言程序-设计时间:2012-12-10 */
#include "stdio.h"
void main( ){
        printf("Welcome to C! \n");        /*调用库函数 printf( )在屏幕上输出信息 */
}
```

程序经编译、链接处理后,运行该程序,将在屏幕上显示输出结果为:

```
Welcome to C!
```

说明:

(1) C 语言是一种由函数组成的语言,每个函数就是一个功能模块。在本例中,定义了一个函数 main( ),并在其中调用 C 标准库函数 printf( )在显示器上输出信息。

(2) main 函数又称为主函数,它是 C 程序的运行入口,任何 C 程序均有且仅有一个这样的 main 函数。main 之后的“( )”表示这是一个不带参数的函数,也即运行程序时不需要给 main 函数提供任何参数。main 之前的 void 其含义为“空,没有”,表示调用该函数不会向调用者返回任何值(此时的调用者为计算机系统自身)。

(3) 任何函数都以左花括号“{”表示函数的开始,以右花括号“}”表示函数的结束,两者之间的部分称为函数体,它一般由一组 C 语句序列组成,以完成具体的功能。由于本例十分简单,因此 main 函数的函数体只包含一条调用 printf( )函数的语句。

(4) 每条 C 语句以分号“;”结束。在 C 语言中,同一行上可以写一条或多条语句,一条语句也可以写在多行。但为了程序的清晰性,除非必要时,建议不要这样做。

(5) 本例中 printf( )函数的作用是在显示器上输出双引号里的内容。但仔细观察不难发现,输出结果为 Welcome to C!,而不是预期的 Welcome to C! \n,这是因为\n 并非普通的符号(称为转义字符,详见 1.5.3 节),它具有特殊的含义,表示输出时换行。也就是说,当输出时遇到\n,之后的内容将被输出到下一行上。另外,由于 printf( )函数被定义在 stdio.h 文件中,为了让系统知道 printf 到底是什么,必须在文件开头部分使用文件包含命令#include "stdio.h"。这样在编译时,编译器就会预先将 stdio.h 文件里的内容包含到 first.c 中,然后再进行编译。stdio.h 称为标准输入输出头文件,里面定义了一些常用的系统 I/O 函数。在 C 语言中,类似的扩展名为 .h 的文件被称为头文件,常用于组织 C 标准函数库中的函数。

(6) 程序的第 1、2 行以及调用 printf( )函数语句后的/* …… */之间的内容均为程序注释,主要用于增加程序的可读性。在编译时,编译器会忽略所有注释语句,但养成良好的注释习惯有助于程序使用者阅读和理解程序。C 语言的标准注释方法以"/*"开始,直到遇到"*/"结束,适用于单行和多行注释。在 VC++ 6.0 环境中,也可使用 C++语言的注释风格,以"//"进行单行注释,如调用 printf 函数的语句可注释为:

```
printf("Welcome to C! \n");    //调用库函数 printf( )在屏幕上输出信息
```

想一想:要输出以下的结果,该示例应该如何修改?

```
******************
* Welcome to C! *
******************
```

【例 1.2】输出长方形的面积。

问题解析:长方形的面积等于长和宽的乘积。假定长方形的长和宽分别为 a 和 b,则面积 area=a×b,再调用 printf( )函数就可解决问题。

程序代码如下:

```
/*计算长方形的面积 */
#include "stdio.h"
void main( ){
    float a, b, area;          //定义 3 个实型变量
    a=2.5; b=3.0;              //这里是两条赋值语句写在同一行上
    area=a*b;                  //C 语言中的乘号用 * 表示
    printf("该长方形的面积为:%f\n", area);     /*要输出一个实数,格式符为%f */
}
```

该程序的运行结果为:

```
该长方形的面积为:7.500000
```

说明:

(1) main 函数的第 1 条语句为变量声明语句,定义了 a、b、area 三个单精度实型(float)变量。变量是用来存储数据的内存空间,声明变量的含义是指通知 C 编译系统为所定义的变量分配合适大小的存储空间。每个变量的存储空间的大小主要由变量的类型和 C 编译环境决定。变量的类型可以是字符型(char)、整型(int)、单精度实型(float)和双精度实型(double)等,它们在 VC++ 6.0 环境下的存储空间大小分别为 1、4、4 和 8 个字节。

(2) 系统为变量分配相应的存储空间时,变量的值一般为随机数(和此时分配到的具体存

储单元有关)。之后,便可通过赋值操作对变量的值进行修改。例如,a=2.5; b=3.0;两条赋值语句可将变量 a 和 b 中存放的数据分别修改为 2.5 和 3.0,area=a * b;语句将把 a 和 b 的乘积赋值给变量 area。

(3) 本例中的 printf( )函数比例 1.1 复杂,双引号中的"该长方形的面积为:"会原样输出到屏幕上,之后遇到格式符号"%f",表示这里要输出一个实型值,这个值存储在双引号后的 area 变量中。最后,仍然采用转义符号"\n"进行换行。

【例 1.3】从键盘任意输入两个整数,并输出它们中较大的那个数。

问题解析:例 1.2 中变量 a 和 b 的值是在编程时通过赋值语句给定的,这导致程序的灵活性较差。通过系统提供的 scanf( )函数,可实现程序运行时动态地从键盘输入数据。另外,1.1.4 节已给出判断两个数中较大者的算法流程图,为了程序的清晰性和代码复用,可根据该算法流程编制一个单独的功能模块 findMax,然后在 main 函数中进行调用。

程序代码如下:

```
#include "stdio.h"
int findMax(int a, int b){                    //定义寻找两个数中较大者的函数
  int max;
  if(a>b)                                     //该 if 语句得到较大值,存放在 max 中
    max=a;
  else
    max=b;
  return max;                                 //通过 return 语句返回 max
}
void main( ){
  int x, y, t;
  printf("Please input two integers(x, y): "); //输入操作提示
  scanf("%d, %d", &x, &y);                    //从键盘输入数据
  t= findMax(x, y);                           //调用 findMax( )函数寻找最大的数
  printf("The maximum is:%d\n", t);
}
```

该程序的运行结果为:

```
Please input two integers (x, y):18,30        //本书运行时的输入内容用斜体、下画线标识
The maximum is:30
```

说明:

(1) 本例充分体现了 C 语言的模块化编程理念,将寻找两个数中较大者的功能定义为 findMax函数,供 main 函数调用。由于 main 函数是 C 程序的运行起点,尽管 findMax 函数的定义在前,程序仍然从 main 函数的第一条语句开始执行。

(2) scanf( )函数可实现程序运行时从键盘输入数据,大大提高了程序的灵活性。输入数据时,一定要按照 scanf 双引号中的格式进行输入。对于本例来说,双引号中指定的格式为"%d,%d",表明需要输入两个整数(每个%d 代表一个整数),且中间以逗号分隔。scanf 函数中的 &x,&y 称为输入地址列表,表示要将输入的两个整数分别存放在变量 x 和 y 的相应地址空间中。其中,& 符号为取地址运算符。