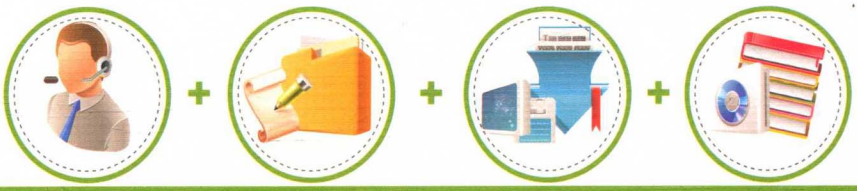
全国高等院校应用型**创新规划**教材 计算机系列

C语言
程序设计实例教程

卢守东 编著



- 以基础理论—实用技术—实训为主线
- 按照教与学的实际需要取材谋篇
- 精心设置了“小型案例实训”，旨在培养学生的实践能力
- 配备丰富的免费教学资源——电子教案、习题答案

清华大学出版社

全国高等院校应用型创新规划教材·计算机系列

C语言程序设计实例教程

卢守东　编著

清华大学出版社
北京

内 容 简 介

本书以 Visual C++ 6.0 为开发工具，介绍 C 语言程序设计的有关技术与相关应用以及结构化程序设计的基本思想与方法，内容包括 C 语言概述、编程基础、控制结构、数组、函数、指针、构造类型、文件操作、类型定义与编译预处理、应用系统(程序)设计与实现，并附有大量的各类习题以及相应的实验指导。全书遵循程序设计与案例教学的基本思想，以应用为导向，以实用为原则，以能力提升为目标，以典型代码、经典实例、完整案例为依托，既利于程序设计基本思想与方法以及 C 语言编程技术的掌握，又利于计算思维与问题求解能力的培养。

本书内容全面，实例翔实，案例丰富，编排合理，循序渐进，语言流畅，通俗易懂，准确严谨，解析到位，注重算法设计与应用开发能力的培养，既可作为各高校本科或高职高专计算机、电子商务、信息管理与信息系统及相关专业高级语言程序设计、程序设计基础、C 语言程序设计等课程的教材或教学参考用书，也可作为 C 语言程序设计人员的技术参考书以及初学者的自学教程。

图书在版编目(CIP)数据

C 语言程序设计实例教程/卢守东编著. —北京：清华大学出版社，2017
(全国高等院校应用型创新规划教材 • 计算机系列)
ISBN 978-7-302-47947-5

Ⅰ. ①C…　Ⅱ. ①卢…　Ⅲ. ①C 语言—程序设计—高等学校—教材　Ⅳ. ①TP312.8

中国版本图书馆 CIP 数据核字(2017)第 205973 号

责任编辑：孟　攀
封面设计：杨玉兰
责任校对：李玉茹
责任印制：宋　林
出版发行：清华大学出版社
网　　址：http://www.tup.com.cn, http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn
质量反馈：010-62772015, zhiliang@tup.tsinghua.edu.cn
课件下载：http://www.tup.com.cn, 010-62791865
印 装 者：三河市金元印装有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　张：**22.5　　**字　数：**530 千字
版　　次：2017 年 9 月第 1 版　　**印　次：**2017 年 9 月第 1 次印刷
印　　数：1～2000
定　　价：49.80 元

产品编号：074332-01

前　言

C 语言是目前国内外广泛使用的一种计算机高级语言，也是当今诸多高校普遍开设的第一门程序设计教学语言。作为一种面向过程的结构化程序设计语言，C 语言对于结构化程序设计基本思想与方法的学习来说是极其有利的。正因如此，各高校计算机、电子商务、信息管理与信息系统及相关专业程序设计方面的教学大多选用 C 语言作为入门语言，以利于学生切实掌握程序设计的基本方法与技术，有效提升学生的编程技能以及分析解决实际问题的能力，并为后续有关语言的学习与实际应用的开发奠定良好的基础。

本书结合教学实践与经验，遵循程序设计与案例教学的基本思想，以 Visual C++ 6.0 为工具，以应用为导向，以实用为原则，以能力提升为目标，以典型代码、经典实例、完整案例为依托，按照由浅入深、循序渐进的方式，精心设计，合理安排，全面介绍了 C 语言程序设计的有关技术与相关应用以及结构化程序设计的基本思想与基本方法。全书实例翔实，案例丰富，编排合理，循序渐进，结构清晰，内容主要包括 C 语言概述、编程基础、控制结构、数组、函数、指针、构造类型、文件操作、类型定义与编译预处理、应用系统(程序)设计与实现等。各章均有“本章要点”“学习目标”与“本章小结”，既便于抓住重点、明确目标，也利于温故知新、总结提高。书中的诸多内容亦设有相应的“说明”“注意”“提示”等知识点，以便于读者的理解与提高，并为其带来“原来如此”“豁然开朗”的美妙感觉。此外，各章均安排有大量的各类习题，以利于读者的及时检测、理解掌握。书末还附有全面的实验指导，便于读者的上机实践、练习提高。

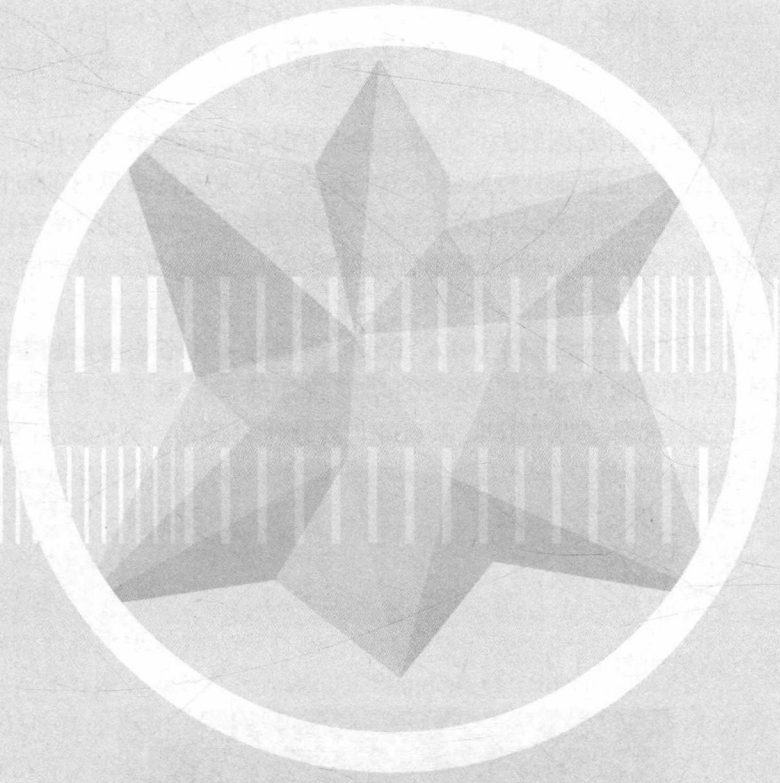
本书内容全面，紧扣基础，面向应用，解析到位，语言流畅，通俗易懂，准确严谨，颇具特色，集系统性、条理性于一身，融实用性、技巧性于一体，注重算法设计与应用开发能力的培养，可充分满足课程教学的实际需要，适合各个层面、各种水平的读者，既可作为各高校本科或高职高专计算机、电子商务、信息管理与信息系统及相关专业高级语言程序设计、程序设计基础、C 语言程序设计等课程的教材或教学参考书，也可作为 C 语言程序设计人员的技术参考书以及初学者的自学教程。

本书的写作与出版，得到了清华大学出版社的大力支持与帮助，在此表示衷心感谢。在紧张的写作过程中，自始至终也得到了家人、同事的理解与支持，在此一并深表谢意。

由于作者经验不足、水平有限，且时间较为仓促，书中不妥之处在所难免，恳请广大读者多加指正、不吝赐教，并将宝贵的意见或建议反馈至作者的电子邮箱 Lsd21cn@21cn.com。

编　者

目录



第 1 章

C 语言概述

本章要点：

C 语言简介；C 语言的基本语法；C 语言程序的基本结构；C 语言程序的编辑与运行。

学习目标：

了解 C 语言的概况；熟悉 C 语言的基本语法与 C 语言程序的基本结构；掌握 C 语言输入函数、输出函数以及赋值语句的基本用法；掌握 Visual C++ 6.0 的基本用法以及 C 语言程序的编辑、编译、连接与运行方法。

1.1 C 语言简介

C 语言是什么？C 语言是国际上广泛流行的一种计算机高级语言，也是一种结构化程序设计语言。实际上，C 语言既具有高级语言的特点，又兼有低级语言的特性，因此是一种“中级语言”。C 语言功能强大，简洁高效，灵活易用，既可用于编写应用软件，又可用于编写系统软件，因此自问世以来便迅速得到推广，至今仍广为使用，可谓是长盛不衰。

C 语言是何样子？通过经典的“Hello, World!”程序，可在初次接触中揭开程序设计语言的神秘面纱。“Hello, World!”程序的功能较为简单，就是在屏幕上显示“Hello, World!”信息。用 C 语言编写的“Hello, World!”程序十分简洁，其代码如下：

```
#include <stdio.h>
void main()
{
printf("Hello, World!\n");
}
```

该程序的运行结果如图 1.1 所示。

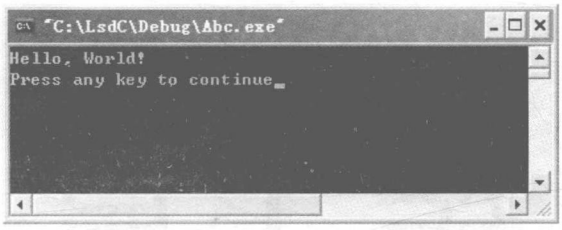


图 1.1 “Hello, World!”程序的运行结果

说明： 在 Visual C++ 6.0 中运行 C 语言程序时，程序执行完毕后会自动在最后提示“Press any key to continue”，意为“按任意键继续”。此时，用户只需按一下键盘上的任何一个按键(如 Enter 键)，即可关闭运行结果窗口，并返回程序的编辑状态。

【C 语言的发展简史】

C 语言最早的原型是 ALGOL 60 语言。1963 年，剑桥大学将 ALGOL 60 语言发展为 CPL(Combined Programming Language)语言。1967 年，剑桥大学的 Matin Richards 将 CPL 语言简化为 BCPL (Basic Combined Programming Language)语言。1970 年，美国 AT&T 贝尔实验室(Bell Labs)的 Ken Thompson(肯·汤普逊)对 BCPL 语言进行了修改，取名为 B 语言，意思是“煮干 CPL 以提取其精华(Boiling CPL down to its basic good features)”，同时用 B 语言编写了首个 UNIX 系统。1973 年，贝尔实验室的 Dennis Ritchie(丹尼斯·里奇)又将 B 语言“煮”了一下，在 BCPL 与 B 语言的基础上设计出一种新的语言，并取 BCPL 中的第二个字母将其命名为 C 语言，同时用 C 语言扩展了 UNIX 系统。随后，UNIX 的内核

(Kernel)与应用程序均用 C 语言重新进行改写，C 语言也因此成为在 UNIX 环境下使用最为广泛的主流编程语言。

1977 年，Dennis Ritchie 发表了不依赖于具体计算机系统的《可移植的 C 语言编译程序》。1978 年，Dennis Ritchie 与 Brian Kernighan 合作出版了经典著作 *The C Programming Language*(《C 语言程序设计》)，并在书末的参考指南(Reference Manual)中给出 C 语言的完整定义，成为当时 C 语言事实上的标准，通常称之为 K&R C。此后，C 语言被先后移植到各种大、中、小、微型计算机上，并得到了广泛的支持，在软件开发中几乎一统天下。

随着 C 语言在各个领域的推广与应用，一些新的特性不断被各种编译器实现并添加进来。作为当时被认为最接近标准的版本，K&R C 与实际使用的 C 语言已存在较大的差别。本着“为 C 语言程序建立一个清楚、无歧义的标准，从而系统地阐述对公认的现有的 C 语言的定义，提高用户程序的可移植性”的目的，美国国家标准学会(American National Standards Institute，ANSI)根据 C 语言问世以来各种版本对 C 语言的发展与扩充，于 1983 年制定了第一个 C 语言标准草案，即 83 ANSI C。1987 年，ANSI 又制定了另一个 C 语言标准草案，即 87 ANSI C。1989 年，草案被 ANSI 正式通过，成为一个新的 C 语言标准——ANSI X3.159-1989，简称 C89。随后，根据 C89 进行了更新的 *The C Programming Language* 第二版开始出版发行。

1990 年，国际标准化组织(International Standard Organized，ISO)批准 C89 为 C 语言的国际标准——ISO/IEC 9899:1990，简称 C90。除标准文档在印刷编排上的某些细节不同外，ISO C(C90)与 ANSI C(C89)在技术上是完全一样的。1995 年，ISO 对 C90 进行了一些修订与扩充，称为 C95。1999 年，ISO 又对 C 语言标准进行了修订与扩充，在基本保留原来的 C 语言特征的基础上，增加了一些面向对象的特性，称之为 ISO/IEC 9899:1999，简称 C99。2000 年 3 月，ANSI 采用 C99 作为 C 语言的新标准。

C99 是迄今为止关于 C 语言的最新、最权威的标准，但各软件厂商所提供的 C 语言编译系统大多数都是以 C89/C90 为基础进行开发的，尚未实现对 C99 的完整支持。此外，不同版本的 C 语言编译系统所实现的语言功能与语法规则也略有差异。因此，在具体应用时，应了解所用 C 语言编译系统的特点。目前，常用的 C 语言编译系统与集成开发环境主要有 Turbo C、Visual C++等。

【C 语言的主要特点】

C 语言的主要特点可归为以下几点：

(1) 简洁紧凑、灵活方便。C 语言共有 32 个关键字、9 种控制语句，程序书写较为自由。

(2) 运算符丰富。C 语言共有 34 个运算符，运算类型极其丰富。灵活使用各种运算符可实现在其他高级语言中难以实现的有关运算。

(3) 数据类型多。C 语言的数据类型十分丰富，包括整型、实型、字符型、数组类型、指针类型、结构体类型、联合体类型、枚举类型等，可用于实现各种复杂数据的处理。

(4) 模块化设计。C 语言是一种结构化的程序设计语言，具有顺序、选择、循环三种

基本控制结构语句。在 C 语言中，函数是组成程序的最小模块，可分别实现特定的功能。按模块组织程序，易于实现程序的结构化。

(5) 执行效率高。C 语言程序的目标代码体积小、质量高，运行速度快(一般只比汇编程序生成的目标代码效率低 10%～20%)。实际上，C 语言是一种“中级语言”，既具有高级语言的特点，又兼有低级语言的特性，可像汇编语言一样对位、字节和地址(而这三者正是计算机最基本的工作单元)进行操作，因此常用于编写系统软件。

(6) 可移植性好。C 语言编译系统几乎可以在各种类型的计算机和各种操作系统上运行。在一种类型的计算机上编写的 C 程序基本上不做修改就能在其他类型的计算机和操作系统上运行。

(7) 绘图功能强。C 语言有强大的图形功能，既可用于二维、三维图形设计与动画设计，也可用于计算机辅助设计。

随着学习的深入，对于 C 语言的特点，大家将有更深刻的体会。

1.2 C 语言的基本语法

C 语言与汉语、英语等自然语言一样，也具有一定的语法结构与构成规则。C 语言的基本成分包括字符、单词、语句、函数等，由字符可以构成单词，由单词可以构成语句，由语句可以构成函数，由函数可以构成程序。

字符是 C 语言最基本的元素，C 语言程序其实就是由一系列的字符构成的。除了字符常量、字符串常量与注释以外，C 语言所支持的字符集主要由以下几类字符构成。

(1) 字母：包括小写字母 a~z、大写字母 A~Z，共 52 个。

(2) 数字：0~9，共 10 个。

(3) 空白符：包括空格符、制表符、换行符等。

(4) 运算符：包括加(+)、减(-)、乘(*)、除(/)、取模(%)、赋值(=)、大于(>)、小于(<)等。

(5) 标点符号：包括逗号(,)、分号(;)、单引号(')、双引号(")、左花括号({)、右花括号(})等。

(6) 特殊字符：包括下划线(_)、反斜线(\)等。

说明： C 语言所支持的字符集可参见 ASCII 码表。ASCII 是 American Standard Code for Information Interchange 的缩写，意为美国信息交换标准码。在 C 语言中，空格符、制表符、换行符等统称为空白符。空白符只在字符常量与字符串常量中有效，在其他地方出现时只是起到间隔的作用，编译程序对其忽略不计。不过，在程序中适当的地方使用空白符，将增强程序的清晰性和可读性。

提示： 在字符常量、字符串常量与注释中，还可以使用汉字或其他可表示的图形符号等，不受语法限制。

字符按照一定的规则可构成各种单词，而单词则是构成语句的最小单位。C 语言中的单词主要分为两种，即关键字与标识符。

语句是组成程序的基本单位，具有独立的程序功能。在 C 语言中，每条语句均以分号

(;)结尾。从形式上看，C 语言的语句可分为 3 种，即简单语句、复合语句与空语句。

简单语句即单独的一条语句。如：

```
x=1;
y=x+1;
```

复合语句是指用花括号“{}”括起来的一组语句。如：

```
{
x=1;
y=x+1;
}
```

说明： 复合语句被视为一个整体，通常用在条件语句或循环语句中。

空语句是指只有一个分号的语句。实际上，空语句是不执行任何操作的。

说明： 程序中的语句数量是没有限制的。另外，C 语言程序的书写格式非常自由，既允许在一行内写几条语句，也允许一条语句分为几行书写，但每条语句都必须以分号结束。不过，为使程序清晰易读，通常每行只书写一条语句，并按缩进格式书写为阶梯形状。具体地说，就是第 1 层次的语句靠左对齐，第 2 层次的语句比第 1 层次的语句缩进若干空格(一般为 2 个或 4 个空格)，第 3 层次的语句比第 2 层次的语句再缩进若干空格……

【实例 1-1】在屏幕上显示“世界，您好！”。

程序代码：

```
#include <stdio.h>
void main()
{
printf("世界，您好！\n");
}
```

运行结果如图 1.2 所示。

图 1.2　实例 1-1 程序的运行结果

程序解析：

(1) 本程序类似于经典的“Hello, World!”程序，功能就是在屏幕上输出“世界，您好！”信息。

(2) 第 1 行为#include 编译预处理命令，用于将标准输入输出头文件 stdio.h 包含至本

程序中。在文件名 stdio.h 中，stdio 是 standard input & output 的缩写，而后缀 h 则为 head 的首字母。

(3) 第 2 行中的 main()为 C 语言程序的主函数，其后由一对花括号({})括起来的语句则为该函数的函数体。main 前面的 void 为空类型，表示该函数没有返回值。

提示： 每个 C 语言程序都必须有一个 main()函数(即主函数)。

(4) 第 4 行为 printf 语句，用于输出字符串“世界，您好！\n”，其中的“\n”是以转义序列表示的换行符(输出一个换行符相当于执行一个回车换行操作，即将光标定位到下一行开始处)。在此，字符串的输出是通过调用 C 编译系统所提供的标准库函数 printf()来实现的。在调用该函数时，应先对其进行声明，而该函数的声明是在 stdio.h 头文件中完成的，因此在程序的开头就通过#include 命令包含 stdio.h 头文件。

提示： 在 C 语言中，每条语句都必须以分号(;)结束。此外，字符串常量必须使用双引号(“”)括起来。

【编程要点：库函数与头文件】

在 C 语言程序中，为实现有关功能，通常要调用相应的库函数。库函数实际上是由 C 语言编译系统提供的一系列功能函数。为便于使用，各库函数已按其功能进行分类，并将其声明写到相应的以“.h”为扩展名的头文件中。例如，输入/输出类库函数的头文件为 stdio.h，数学类库函数的头文件为 math.h，字符串类库函数的头文件为 string.h。

头文件(又称标题文件)包含与库函数有关的变量定义、宏定义以及对库函数的声明等。为调用库函数，在程序的开头要将相应的头文件包含进来。为此，应使用 include 预编译命令。该命令有以下两种语法格式：

```
#include <头文件名>
#include "头文件名"
```

include 命令的功能是包含指定的头文件，以便在程序中调用在头文件中声明的库函数。其中，第一种格式为尖括号形式，编译系统在编译程序时从存放 C 语言库函数头文件的子目录中查找所要包含的头文件，这称为标准方式；第二种格式为双引号形式，编译系统在编译程序时先从当前目录中查找所要包含的头文件，找不到时再按标准方式进行查找。

提示： 若要调用库函数，则应包含系统所提供的头文件。此时，应使用尖括号形式，以提高效率。如果要包含的头文件不是系统提供的头文件，而是用户自己编写的文件(这种文件一般存放在当前目录中)，那么应使用双引号形式，否则会找不到所需要的文件。若要包含的文件不在当前目录中，则可在双引号中写上文件的路径，如#include "C:\LsdCommon\abc.h"。

注意： 在 C 语言中，头文件名是不区分大小写的。

【实例 1-2】求任意一个圆的面积。

程序代码：

```
#include <stdio.h>  //包含头文件 stdio.h
void main()  //主函数
{
    float r,s;  //定义两个浮点型变量，r 表示半径，s 表示面积
    printf("r=");  //提示输入半径
    scanf("%f",&r);  //输入半径
    s=3.14159*r*r;  //计算面积
    printf("s=%f\n",s);  //输出面积
}
```

运行结果如图 1.3 所示。其中，第 1 行中的 1.5 为输入，即圆的半径。

图 1.3　实例 1-2 程序的运行结果

程序解析：

(1) 本程序具备了数据的输入与输出功能，且通用性较强，可由用户任意输入圆的半径，然后据此计算其面积并输出。

(2) 程序中添加了注释。注释是程序中的说明性文字，其作用在于增强程序的可读性，是不会被编译与执行的。在本程序中，各行的注释内容均以双斜杠(//)开始，至行末结束。

(3) 程序中须存放圆的半径与面积，因此要定义两个相应的变量，即表示半径的 r 与表示面积的 s。考虑到半径与面积不一定是整数，因此将变量的类型定义为 float 型，即浮点型(又称单精度型)。

提示： *在 C 语言中，使用变量前必须先进行定义，否则会出现编译错误。*

(4) 在本程序中，半径的输入通过调用 scanf()函数来实现，面积的输出通过调用 printf()函数来实现。与 printf()函数一样，scanf()函数也是 C 编译系统所提供的一个标准库函数，其声明也包含在 stdio.h 头文件中。

(5) 在 scanf 语句中，指定将输入的半径赋给变量 r。圆括号内的第一部分为格式控制字符串(用于指定输入数据的类型与格式)，在此只包含一个转换说明符“%f”，表示要输入一个单精度实数。圆括号内的第二部分为输入项目列表，在此只有一个输入项目&r，表示变量 r 的地址(&r 中的“&”为取址运算符)。

(6) 在 printf 语句中，圆括号内的第一部分也是格式控制字符串(用于指定输出数据的类型与格式)，在此为“s=%f\n”。圆括号内的第二部分则为输出项目列表，在此只有一个输出项目，即变量 s。在格式控制字符串“s=%f\n”中，“s=”为普通字符，按原样输出；“%f”为转换说明符，表示要输出一个单精度实数(小数部分为 6 位)，其值来自于相应的输出项目，在此为变量 s；“\n”为换行符，表示回车换行。

提示： scanf()函数规定，输入项目均为地址量，在相应变量名的前面均应加上“&”。实际上，系统是根据地址将用户所输入的数据存放至相应变量在内存中所占用的存储单元的。

尝试一下：

如果还要计算圆的周长，那么该如何修改该程序呢？

【编程要点：关键字与标识符】

关键字是 C 语言中具有特定意义和用途的字符序列，如 void(空类型)、int(整型)、float(浮点型)等。关键字由C语言系统内部定义，又称保留字。

标识符是 C 语言程序中用于命名有关对象的字符序列，如变量名 r、s、name、sex、age 等。与关键字不同，标识符是由用户根据需要自行定义的。在 C 语言中，一个合法的标识符由字母、数字和下划线(不能全为下划线)组成，且第一个字符必须是字母或下划线，最大长度不能超过 32 个字符。

注意： 标识符不能与关键字同名。此外，标识符是区分大小写的。例如，name 与 Name 是两个不同的标识符。

说明： C 语言标准 ANSI C89/ISO C90 所规定的关键字共有 32 个(均为小写形式)，分别为：

auto	break	case	char	const	continue	default
do	double	else	enum	extern	float	for
goto	if	int	long	register	return	short
signed	sizeof	static	struct	switch	typedef	union
unsigned	void	volatile	while			

1999 年 12 月，ISO 推出了 C99 标准。该标准新增了 5 个 C 语言关键字，分别为：

inline　restrict　_Bool　_Complex　_Imaginary

2011 年 12 月，ISO 又发布了 C 语言的新标准 C11。该标准新增了 7 个 C 语言关键字，分别为：

_Alignas　_Alignof　_Atomic　_Static_assert　_Noreturn
_Thread_local　_Generic

【编程要点：注释】

注释是程序中的一些说明性的文字，既可以添加到某条语句的末尾，也可以独占一行或分为几行。在C语言程序中，注释有两种格式：

```
//注释内容
/*注释内容*/
```

其中，第一种格式以“//”开始，一般用于单行注释；第二种格式以“/*”开始、以“*/”结束，通常用于多行注释。

【实例 1-3】求两个整数的和。

程序代码：

```
#include <stdio.h>
void main()  /*主函数*/
{
    int sum(int x,int y);  /* 对被调用函数 sum 进行声明 */
    int a,b,c;  /*定义变量 a、b、c */
    scanf("%d%d",&a,&b);  /*输入变量 a 和 b 的值*/
    c=sum(a,b);  /*调用 sum 函数，将返回的值赋给 c */
    printf("sum=%d\n",c);  /*输出 c 的值*/
}
int sum(int x,int y)  //子函数 sum，返回值为 x 与 y 的和
{
    int z;  //定义变量 z
    z=x+y;  //计算 x 与 y 的和，并赋给 z
    return (z);  //返回 z 的值
}
```

运行结果如图 1.4 所示。其中，第 1 行为输入，第 2 行为输出。所输入的两个整数为 10、20，二者以空格分隔(也可以用制表符或回车符来分隔)。

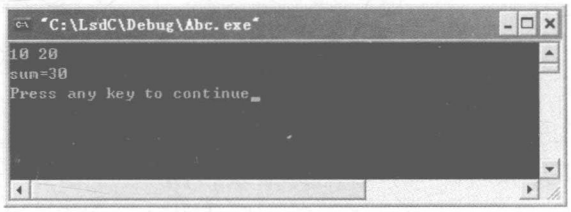


图 1.4　实例 1-3 程序的运行结果

程序解析：

(1) 除主函数 main()外，程序中还定义了一个子函数 sum()。该子函数具有一个整型的返回值，也就是形式参数 x 与 y 的和。

(2) 在主函数中，通过语句“c=sum(a,b);”实现对子函数 sum()的调用，并将其返回值(即 a 与 b 的和)赋给变量 c。在调用时，实际参数 a、b 的值分别传递给子函数 sum()的形式参数 x、y。

(3) 由于子函数 sum()的定义出现在其调用语句“c=sum(a,b);”之后，因此应先对其进行声明，语句为“int sum(int x,int y);”。若将子函数 sum()的定义置于主函数之前，则无须对其进行声明即可直接调用。

尝试一下：

如果要计算两个整数的积，那么该如何修改该程序呢？

【编程要点：函数】

函数是 C 语言程序的基本模块，由函数头与函数体两部分构成，其基本格式为：

[数据类型] 函数名([形式参数])