21世纪高等教育计算机规划教材

案例式C语言程序设计教程

许薇 武青海 李丹 主编
单继芳 薄小永 副主编

21st Century University Planned Textbooks of Computer Science

中国工信出版集团
人民邮电出版社
POSTS & TELECOM PRESS

21世纪高等教育计算机规划教材

案例式C语言程序设计教程

许薇 武青海 李丹 主编
单继芳 薄小永 副主编

21st Century University
Planned Textbooks of Computer Science

人民邮电出版社
北京

图书在版编目（CIP）数据

案例式C语言程序设计教程 / 许薇，武青海，李丹主编. -- 北京 : 人民邮电出版社，2015.12
21世纪高等教育计算机规划教材
ISBN 978-7-115-41447-2

Ⅰ. ①案… Ⅱ. ①许… ②武… ③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(2016)第017410号

内 容 提 要

本书涵盖了教育部考试中心制定的《全国计算机考试二级考试大纲》中有关C语言程序设计的知识点，内容主要包括Visual C++ 6.0基础知识、C语言的各种数据类型和运算符、表达式、语句结构、函数、指针、数组、结构体及共用体、文件等。

本书选材新颖，内容丰富，理论联系实际、深入浅出、循序渐进，注重培养读者的程序设计能力，以便读者养成良好的程序设计习惯。

为了配合本书的学习，作者还编写了与本书配套的《案例式C语言程序设计教程实验指导》一书，可供读者学习时参考使用。

本书可作为高等院校计算机程序设计的入门教材，也可作为全国计算机等级考试及各类培训班的培训教材，还可作为软件开发人员的自学参考书。

◆ 主　　编　许　薇　武青海　李　丹
副 主 编　单继芳　薄小永
责任编辑　武恩玉
责任印制　沈　蓉　彭志环
◆ 人民邮电出版社出版发行　　北京市丰台区成寿寺路11号
邮编　100164　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
三河市潮河印业有限公司印刷
◆ 开本：787×1092　1/16
印张：20.75　　2015年12月第1版
字数：548千字　　2015年12月河北第1次印刷

定价：49.80元

读者服务热线：(010)81055256　印装质量热线：(010)81055316
反盗版热线：(010)81055315

前　言

随着计算机科学技术的发展，程序设计已经成为很多计算机爱好者的研究方向。C 语言适合程序设计者作为入门语言。它是一种理想的结构化程序设计语言，具有功能丰富、使用灵活方便、应用面广等特点。C 语言课程在高等院校专业课中能起到桥梁的作用。

《案例式 C 语言程序设计教程》具有理论和实践并重的特征，因此学这门课程不仅要掌握理论基础知识，而且还得具备运用理论知识处理实际问题的技能。在学习过程中涉及的知识点和语法规则比较多，使大多数学生感觉到学习有些困难，所以，在编写的过程中，编者按照认知规律进行编写，对每章知识点的讲解从案例出发，通过给出的案例，让读者对应用场景有初步认识，然后逐步引出相关知识点。本书在编写的过程中紧扣教育部考试中心 C 语言考试大纲的要求，将本书分为 13 章。

第 1 章为 C 语言概述。主要内容包括：C 语言的产生与发展、C 语言的特点、编制简单的 C 语言程序、Visual C++6.0 简介、算法及算法表示等。

第 2 章介绍 C 程序设计的基本知识。主要内容包括：C 语言的数据类型，标识符、常量与变量，算术运算符和算术表达式，关系运算和逻辑运算表达式，赋值运算符和赋值表达式，逗号运算符和逗号表达式，自加、自减运算符等。

第 3 章介绍顺序结构。主要内容包括：C 语言的基本语句、格式输入/输出函数、字符数据的输入/输出函数、程序举例等。

第 4 章介绍选择结构。主要内容包括：if 语句、switch 语句、条件表达式构成的选择结构、程序举例。

第 5 章介绍循环结构。主要内容包括：while 循环结构、do-while 循环结构和 for 循环结构等。

第 6 章介绍函数。主要内容包括：函数定义和返回值、函数的调用、函数的嵌套调用和函数的递归调用、局部变量和全局变量等。

第 7 章介绍数组。主要内容包括：一维数组的定义和一维数组元素的引用、函数之间对数组和数组元素的引用、二维数组的定义和二维数组元素的引用、字符数组等。

第 8 章介绍地址和指针。主要内容包括：地址和指针的概念、指针变量、指向函数的指针、对指针变量的操作、函数之间地址值的传递等。

第 9 章介绍编译预处理和动态存储分配。主要内容包括：编译预处理、动态存储分配等。

第 10 章介绍结构体、共用体和枚举。主要内容包括：概述、结构体数组的定义及初始化、指向结构体类型变量的指针、用指针处理链表、共用体、枚举类型等。

第 11 章介绍位运算。主要内容包括：位运算符的基本概念法、位运算符的运算功能举例等。

第 12 章介绍文件。主要内容包括：C 语言文件的概念、文件访问的步骤、文件的打开与关闭、非标准文件的读写和文件定位函数等。

第 13 章介绍程序的综合设计。主要内容包括：程序举例、综合设计。

本书具有如下特点。

（1）内容丰富。涵盖教育部考试中心 C 语言考试大纲的所有知识点，实例与知识点结合恰当，习题安排合理。

（2）图文并茂。在讲解 Visual C++6.0 的知识点过程中配有丰富的图解说明，语言通俗、流畅，有很强的实用性和可操作性。

（3）配有 PPT 和实验指导。电子教案，用 PowerPoint 制作，方便教师在上课时根据需要进行修改。为配合读者学习，编者编写了《案例式 C 语言程序设计实验教程》一书作为本书的配套参考书，供读者复习和检查学习效果时使用。

本书是编者根据多年从事 C 语言及计算机专业相关课程的教学实践，在多次编写讲义、教材的基础上编写而成的，内容充实，循序渐进，书中所有例题都在 Visual C++6.0 环境中上机调试通过。

本书编写分工为：许薇编写第 1 章、第 2 章、第 3 章；薄小永编写第 4 章、第 5 章；单继芳编写第 6 章、第 12 章及附录 D～附录 E；武青海编写第 7 章、第 8 章、第 9 章及附录 A～附录 C；李丹编写第 10 章、第 11 章、第 13 章。全书由许薇、武青海两位老师进行统稿。本书在编写过程中，参考了国内许多正式和非正式出版的相关著作，在此向这些作者们致以衷心的谢意！

希望本书的出版能够给读者学好程序设计课程带来启迪和帮助。本书的出版也是一部应用型转型的教材，限于编者水平，书中难免存在疏漏和不足之处，恳请广大读者批评指正。

编者

2015 年 11 月

目 录

第1章 C语言概述

学习目标

通过本章的学习，读者能够了解C语言的产生和发展，并对C语言的特点有初步的认识。通过引入案例，初步了解使用Visual C++ 6.0开发环境运行C程序的具体过程，了解与C语言程序设计密切相关的算法的概念、算法的特性以及算法的表示。

学习要求

- 了解C语言的产生和发展。
- 了解C语言的特点。
- 了解Visual C++ 6.0开发环境及运行简单的C语言程序的方法。
- 了解C语言算法的概念及表示方法。

1.1 C语言的产生与发展

C语言是近年来应用比较广泛的一种计算机语言。它功能丰富，表达能力强，使用灵活方便，应用面广，目标程序效率高，而且可以在不同的操作系统下执行，可移植性好，既具有高级语言的特点，又具有低级语言的许多特点。所以，C语言通常适合编写系统软件。现在比较流行的嵌入式系统通常也用C语言来编写。由于C语言的特点，现在C语言已经不仅仅限于计算机专业工作者使用，更多的非计算机专业出身的人也都非常喜爱和使用这门语言。

C语言不是最早的计算机语言，它有着一定的发展历程。Fortran语言是历史上的第一门计算机高级语言，它主要用于科学计算。随着Fortran语言的出现和使用，越来越多的计算机专家和工程师们对高级语言的研究、设计和使用产生了浓厚的兴趣。诞生于20世纪60年代的Algol 60是一门结构良好、逻辑严谨、简明易学的算法语言，但由于它应用范围较窄，离硬件远，不适合用来编写系统软件，所以没有得到很好的推广。1967年，英国剑桥大学的Matin Richards在Algol 60的基础上推出了比较接近于硬件的基本组合程序设计语言（Basic Combined Programming Language，BCPL），1970年，美国贝尔实验室的Ken Thompson以BCPL为基础设计出简单而又很接近硬件的B语言，并用B语言写出UNIX操作系统初版。20世纪70年代初期，美国出现的Pascal语言是第一门反映结构化程序设计的高级程序设计语言，得到了较广泛的推广，最初计算机专业的人员都把Pascal语言作为计算机专业的入门语言。几乎在Pascal语言诞生的同时，C语

言在美国著名的贝尔实验室中酝酿并诞生了。1973 年，贝尔实验室 Dennis M. Ritchie 在 B 语言的基础上设计出了 C 语言，它既保留了 B 语言简洁和接近硬件的特点，又克服了它过于简单、无数据类型的缺点。后来，Ken Thompson 和 Dennis M. Ritchie 合作，进一步改写 UNIX 操作系统。与 Fortran、Algol 和 Pascal 语言不同，C 语言诞生之时并没有什么研制报告和语言报告，而是在设计 UNIX 操作系统时不断得到更新和完善。1978 年，C 语言已经在大、中、小及微型计算机上广泛使用。

1978 年，Brian W. Kernighan 和 Dennis M. Ritchie 合作编著了 The C Programming Language，建立了所谓 C 语言的 K&R 标准，这本书成为后来 C 语言的基础，称为标准 C。后来，美国国家标准学会（American National Standard Institute，ANSI）又推出新的标准 C，它比原来的标准 C 又有较大的发展。

C 语言从产生到现在，自身也在不断地发展。20 世纪 80 年代中期，出现了面向对象程序设计的概念，贝尔实验室的 B. Stroustrup 博士借鉴了类的概念，将面向对象的语言成功引入 C 语言中，设计出了 C++语言。C++语言赢得了广大程序员的喜爱，不同的机器、不同的操作系统几乎都支持 C++语言。同时，C++语言也得到了国际标准化组织（ISO）的认可，为此，国际标准化组织对 C/C++语言实现标准化。C/C++语言对新语言的形成影响力也较大。20 世纪 90 年代中期以来，随着 Internet 的日益普及，用于开发 Internet 的 Java 语言成为人们掌握的热门语言。事实上，Java 语言与 C++语言极为相似，2002 年，微软公司正式推出了 C#语言，该语言与 C/C++也有密切的联系，它已成为.NET 环境的重要编程语言。

随着信息化、智能化、网络化的发展以及嵌入式系统技术的发展，C 语言的地位也会越来越高，C 语言将在云计算、物联网、移动互联、智能家居等未来信息技术中发挥重要作用。

1.2 C 语言的特点

C 语言是一种通用的、面向工程的程序设计语言，它之所以能够存在和发展，并具有强有力的生命力，与其不同于其他语言的特点有关。

C 语言的主要特点如下。

1. 语言简洁，使用方便灵活

用 C 语言编写的源程序往往比用其他语言编写的源程序短，使得程序输入工作量减少。C 语言一共有 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 语言的运算符丰富，共有 34 种，它把括号、赋值、强制类型等都作为运算符处理，从而使 C 的运算类型极其丰富，表达式类型多样化；数据结构丰富，具有现代化语言的数据结构。

2. 结构化程序设计

具有结构化的控制语句，以函数作为程序设计的基本单位，具有自定义函数的功能，便于实现程序模块化；语法限制不太严格，程序设计自由度较大。

3. 既具有高级语言的功能，又具有低级语言的许多功能

C 语言能直接访问物理地址和端口，并且能够进行位操作，因此能实现汇编语言的大部分功能，可以直接对硬件进行操作，可用来编写系统软件。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”，意味着它兼有高级和低级语言的特点。

4. 用途广泛

C 语言的应用几乎遍及了程序设计的各个领域，如科学计算、系统程序设计、字处理软件和电子表格软件的开发、信息管理、计算机辅助设计、图形图像处理、数据采集、实时控制、嵌入式系统开发、网络通信、Internet 应用和人工智能等。

5. 可移植性好

若程序员在书写程序时严格遵循 ANSI C 标准，则其代码可不做修改，即可用于各种型号的计算机和各种操作系统，因此，C 语言具有良好的可移植性。

以上只介绍了 C 语言最容易理解的一般特点，至于 C 语言内部的其他特点将结合后续各章的内容进行具体介绍。由于 C 语言的可移植性好和硬件控制能力高，表达和运算能力强，许多软件都用 C 语言来编写，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　编制简单的 C 语言程序

1.3.1　简单的程序设计

在计算机尚未诞生之前，就有了“程序”的概念，所谓程序就是事情进行的先后次序。那么什么是“计算机程序”？计算机程序是指为了让计算机完成一项任务，而在计算机中存放的一系列计算机可以识别的指令。

程序可以简单，也可以复杂。简单的程序就几条指令，而复杂的程序有成千上万条指令。程序的规模越大，内容越复杂，所需要的程序指令就越多，程序的结构也越复杂。人们把完成某种任务而编写的一系列指令（或程序）交由计算机去执行，这个过程叫作程序设计。程序设计要求编写程序的人首先对需要完成的任务有一个比较清晰的认识，然后按照计算机可以识别的方式来组织这些指令以形成程序，最后将描述这个任务的程序交由计算机去执行，从而完成这个任务。由于任务的复杂性和多样性，使得程序设计不可能一次就达到要求，需要在程序的设计过程中不断地修改完善，最终满足任务的需求，这个过程叫作程序的调试和测试。

人类的语言是由语法和词汇构成的，同样，计算机语言也是由语法和词汇构成的。所谓语法就是规则的集合，规定什么是允许的，什么是不允许的，什么是正确的，什么是错误的；词汇也就是符号，它是语言的构成要素。计算机所能识别的语言只有一种，这就是机器语言。机器语言是由 0 和 1 组成的指令序列。由于人们对二进制数据书写和理解都存在一定的困难，因而产生了多种高级语言，这些高级语言比较接近人们日常使用的自然语言，但又都有一定的语法规则，才能让计算机识别和执行。懂得了一定的语法规则，我们就可以设计自己想设计的程序，让计算机为人类做事，计算机语言是人与计算机进行对话的工具。

程序设计实际就是人们用计算机语言来描述问题的解决方案，所以想让计算机帮我们解决问题，同样需要完成以下 4 个步骤：分析问题，寻找解决问题的方法和步骤，用程序语言来描述解决过程，最后让计算机来执行这个过程以完成任务。如何完成，需要程序员编写程序告诉计算机应该做什么、怎么做，这就需要计算机语言的支持。

每种语言都有语法规则，C 语言也不例外，在这里先介绍几个 C 语言的小程序，然后从中分析编制 C 语言程序的过程以及语法规则。

【例 1-1】从键盘上输入两个整数，求两个数的和，并将结果输出。

```
#include <stdio.h>
main()                                  /*主函数*/
{                                       /*函数体*/
  int  a , b , sum ;                    /*变量声明*/
  printf("Enter two numbers:") ;        /*输出*/
  scanf("%d %d", &a , &b ) ;
  sum=a+b ;
  printf("sum=%d",sum);
}
```

程序的运行结果如下。

```
Enter two numbers: 5  8
sum =13
```

本程序的作用是从键盘上任意输入两个整数，求两个数之和。其中#include <stdio.h>是编译预处理命令，在 C 语言中不能直接进行输入、输出操作，所以要用到“scanf( )、printf( )”两个函数来完成数据的输入与输出，这两个函数包含在 stdio.h 文件中。本教材所选用的环境为 VC++6.0，因此在程序的开头要加上此句，如用 Turbo c 编写程序，只用到 scanf( )、printf( )两个函数而没有其他函数，#include <stdio.h>语句可以省略。main( )是主函数，每一个 C 语言程序必须有且只能有一个主函数。一个 C 语言程序的执行总是从主函数开始，需要注意的是 main 后的“( )”是不能省略的。/*……*/表示注释部分，为便于理解，通常用来解释程序，可以用汉字，也可以用英文，对程序的编译和执行并不产生影响。“{……}”是函数体部分，此部分又包括说明部分和执行部分两大部分，int 是 C 语言的关键字，表示整数类型；“%d”是输入/输出的“格式字符串”，用来指定输入、输出时的数据类型和格式；“%d”表示“十进制整数类型”。“;”是一条语句结束的标志。“&a，&b”其中的“&”表示的是“取地址”运算符，将从键盘上输入的两个整数分别放在 a、b 两个存储单元中，至于两个单元确切的位置我们不需要知道，在声明 a、b 为整型变量的时候，计算机就在内存中为 a、b 开辟了两个存储单元，当从键盘输入数据的时候，计算机自动找到这两个存储单元，把数据放入存储单元中。printf( )函数表示的是在屏幕上输出，scanf( )函数表示从键盘输入。

【例 1-2】从键盘上输入两个数，输出它们的最大数。

```
#include <stdio.h>
main()
{
  int  a , b , max ;
  printf("Enter two numbers:") ;
  scanf("%d %d", &a , &b ) ;
  if (a>b)
    max=a;
  else
    max=b;
printf("max=%d",max);
}
```

程序的运行结果如下。

```
Enter two numbers: 100  200
max=200
```

从本例中可以看出，在声明变量的时候，用 max 来表示存储两个数中的最大数的变量，max 在英文中的意思是“最大值/最大”。“if”英文中是“如果”的意思，在这里是判断 a、b 的大小。因为 a、b 都有可能是最大的，所以要进行判断。如果 a>b，那么 a 是最大的，否则 b 是最大的。

程序的其他部分与上例相同，在此不再加以说明。

通过以上两个例子，我们可以清楚地看到 C 语言的结构和各个组成部分的作用。

1. C 语言程序是由函数构成的

一个 C 语言源程序包含一个 main( )函数，也可以包含若干个其他函数（此部分内容将在第 6 章中介绍）。因此，函数是 C 语言程序设计的基本单位。

2. 函数的组成

（1）函数的首部。函数的首部即函数的第 1 行，如“main( )”，它包括函数名、函数类型、函数参数名和参数类型。一个函数名后面必须跟一对圆括弧，函数类型、函数参数名和参数类型可以省略，但圆括弧不能省略。

（2）函数体。即函数首部下面的大括弧“{……}”内的部分。如果一个函数内有多个大括弧，则最外层的一对为函数体的范围。

函数体一般包括声明部分和执行部分。

① 声明部分：在这部分中定义所用到的变量，如“int　a , b , max ; ”。

② 执行部分：这部分由若干条语句构成，这是整个函数所要完成的功能。当然，在某些情况下也可以没有声明部分，甚至可以既没有声明部分，也没有执行部分。

如：

```
aa( )
{    }
```

就是一个既没有声明部分，又没有执行部分的一个用户自定义函数，我们称为空语句，它是合法的。

（3）一个 C 语言程序总是从 main( )函数开始执行的，而不论 main( )函数在整个程序中的位置。

（4）C 语言程序书写格式自由。一行可以写几个语句，一个语句也可以分写在多行上。C 语言程序没有行号，也不像某些语言那样严格规定书写格式。

（5）每个语句和数据定义的最后必须跟有一个分号。分号是 C 语句的必要组成部分，分号不可以少，即使是程序中最后一个语句也应包含分号。

（6）C 语言本身没有输入、输出语句。输入和输出的操作是由库函数来完成的，如 printf( )、scanf( )等。C 语言对输入、输出实行“函数化”。由于输入、输出操作涉及具体的计算机设备，把输入、输出操作放在函数中处理，就可以使 C 语言本身的规模较小，编译程序简单，很容易在各种机器上实现，程序具有可移植性。当然，不同的计算机系统需要对函数库中的函数进行不同的处理。不同计算机系统除了提供函数库中的标准函数外，还要按照硬件的情况提供一些专门的函数。因此，不同计算机系统中所提供的函数个数和功能可能有所不同。

（7）可以用/*……*/对 C 语言程序中的任何部分进行注释。一个好的、有使用价值的源程序都应当加上必要的注释，以增加程序的可读性。

3. 编译预处理命令

C 语言程序的开头一般可以看到有些程序行以“#”符号开头，这些程序行就是编译预处理命令。C 语言提供 3 类编译预处理命令：文件包含、宏定义、条件编译（第 9 章介绍），如例题中的“#include <stdio.h>”。其中，stdio.h 是一个头文件，也称标准输入、输出头文件。由于程序中要用到 printf( )、scanf( )两个函数，而这两个函数的定义和说明，系统已经存放在头文件 stdio.h 中，因此要包含该头文件。所谓包含，就是把头文件代码引入程序中，由于这个工作是在编译程序前

完成的，所以称为编译预处理命令。

1.3.2 C 程序的编辑、编译和连接

C 语言是一种高级程序设计语言，通常把这种用高级语言来编写的程序称为源程序。用 C 语言编写的源程序，它的扩展名为.c。通常用 C 语言书写的程序是不能直接运行的，它必须生成与之对应的可执行程序，也就是我们通常说的要经过编译和连接后才能执行。

具体过程如下。

（1）编辑源程序，完成后将源程序以扩展名为.c 保存。

（2）对源程序进行编译。源程序通过编译程序（也称编译器）编译以后生成二进制的目标文件，通常其扩展名为.obj，此二进制代码不能运行。若源程序有错，必须予以修改，然后重新编译。

（3）对编译生成的目标文件进行连接。C 语言程序中可能包含几个目标文件和库文件（扩展名为.lib），它们都是二进制文件，应把它们连接起来，生成一个完整的可执行文件，这个连接工作由专门的软件——连接程序（又称连接器）来完成。连接过程中，可能出现未定义的函数等错误，为此，必须修改源程序，重新编译和连接。

（4）执行生成的可执行文件。连接后，系统生成可执行程序文件，文件的扩展名为.exe，所以也称为 exe 文件。源程序通过编译和连接后产生一个可执行文件，这个过程一般来说不可能一次成功，必须修改源程序，重新编译和连接，有一个反复调试的过程。如果执行后能得到正确结果，则整个编辑、编译、连接、运行过程顺利结束。

从编辑源程序开始一直到产生一个可执行文件，一般都在一个集成开发环境中完成，这个开发环境集编辑、编译和连接于一体，在编辑与连接过程中，根据是否有出错信息来决定是否要重新编辑或修改源程序。

C 语言的运行环境版本较多，本书将采用 Visual 作为开发环境。Visual++6.0 不仅功能强大，而且字符串和注释可用中文，但该环境对初学者学习有一定的难度，还希望读者及时掌握。

1.4 Visual C++ 6.0 简介

1.4.1 Visual C++ 6.0 简介

Visual C++ 6.0（简称 VC++ 6.0）是开发运行于 Microsoft Windows NT 环境下的 Win32 应用程序，由于它为软件开发人员提供了完整的编辑、编译和调试工具，而成为可视化编程工具中最重要的成员之一。

VC++ 6.0 是建立在全面封装了 Win32 API（Application Programming Interface）的 MFC（Microsoft Foundation Class Library）基础类库上的，从而有效地缩短了 Windows 应用程序的开发周期。Windows 操作系统本身大部分是使用 C/C++语言编写的，而 Visual ++ 6.0 正是使用了 C/C++语言的 Win32 应用程序的开发，因此，使用 Visual ++ 6.0 来进行 Windows 应用程序的开发有着得天独厚的优势。

1. VC++ 6.0 的版本

VC++ 6.0 是 Microsoft 公司于 1998 年 6 月最新推出的 VC++编译器，它包括 3 个版本。

（1）学习版：除了代码优化、剖析程序（一种分析程序运行时行为的开发工具）和到 MPC 库静态链接外，VC++ 6.0 学习版还提供了专业版的所有功能。学习版的价格要比专业版低很多，这是为了让使用 VC++ 6.0 来学习 C++语言的个人也可以负担得起。但不能使用 VC++ 6.0 学习版来公开发布软件，其授权协议明确禁止这种做法。

（2）专业版：VC++ 6.0 可用来开发 Win32 应用程序、服务和控件，在这些应用程序、服务和控件中可使用由操作系统提供的图形用户界面或控制 API。

（3）企业版：可用来开发和调试为 Internet 或企业内网设计的客户服务器应用程序，在该版本中还包括开发和调试 SQL 数据库应用程序和简化小组开发的开发工具。

2. VC++ 6.0 集成开发环境

VC++ 6.0 是 Microsoft 公司提供的在 Windows 环境下进行应用程序开发的 C/C++编译器。相比其他的编程工具而言，VC++ 6.0 在提供可视化的编程方法的同时，也适用于编写直接对系统进行底层操作的程序。随 VC++ 6.0 一起提供的 Microsoft 基础类库（Microsoft Foundation Class Library，MFC）对 Windows 9x/NT/2000 等所用的 Win32 应用程序接口（Win32 Application Programming Interface）进行了彻底封装，使得 Windows 9x/NT 应用程序的开发可以使用完全面向对象的方法来进行，从而大大缩短了应用程序的开发周期，降低了开发成本，也使得 Windows 程序员从大量的复杂劳动中解脱出来，并没有因为获得这种方便而牺牲应用程序的高效性和简洁性。

VC++ 6.0 是 Microsoft 公司出品的基于 Windows 的 C/C++开发工具，它是 Microsoft Visual Studio 套装软件的一个有机组成部分，在以前版本的基础上又增加了许多新特性。VC++ 6.0 在以前版本的 Visual 工作平台基础上，做了进一步的发展，从而更好地体现了可视化编程的特点。VC++ 6.0 软件包含了许多单独的组件，以及各种各样为开发 Microsoft Windows 下的 C/C++ 程序而设计的工具，它还包含了一个名为 Developer Studio 的开发环境。

为了学习和使用 VC++ 6.0，需要一台运行 Windows 95/98 或 Windows NT 的计算机作为工作平台，要求有足够的内存和其他资源以支持各种工具。在绝大多数情况下，计算机的最低配置应该是 Pentium 166MHz、64MB 内存，至少 1GB 的硬盘空间。

（1）新特性。Microsoft Developer Studio 用于 Visual J++1.1、Visual C++6.0、Visual InterDev 和 MSDN。新增 Developer Studio 包括以下新特性。

① 自动化和宏。可以使用 Visual Basic 脚本来自动操纵例行的和重复的任务。可以将 Visual Studio 及其组件当作对象来操纵，还可以使用 Developer Studio 查看 Internet 上的 World Wide Web 页。

② Class View。使用文件夹来组织 C++和 Java 中的类，包括使用 MFC、ATL 创建或自定义的新类。

③ 可定制的工具条和菜单。工具条和菜单可按用户的要求自己定制。

④ 调试器。可连接到正在运行的程序并对其进行调试，还可以使用宏语言来自动操作调试器。

⑤ 项目工作区和文件。可以在一个工作空间中包括多个不同类型的工程，工作空间文件使用扩展名.dsw 来代替过去的扩展名.mdp，工程文件使用扩展名.dsp 来代替过去的扩展名.mak。

⑥ 改进的资源编辑器。在 VC++ 6.0 中，可以使用 WizarBad 将程序同可视化元素联系起来。使用快捷键、二进制、对话框和字符串编辑器对 ASCII 字符串、十六进制字符串、控件 ID 和标签以及指定字符串的 Find 命令一次修改多项。

⑦ 改进的文本编辑器。可以使用正确的句法着色设置来显示无扩展名的文件，可以定制选定页边距的颜色更好地区分同一源代码窗口的控件和文本区域，Find in Files 命令支持两个单独的窗格。

⑧ 改进的 WizardBar。可用于 Visual J++程序的编写。在 VC++ 6.0 中，集成环境有了更大的用处，在 Visual Studio 中，不仅可以进行 VC++程序的编写，而且也可在其中编写 Visual J++程序。

⑨ 便捷的类库提示。相信使用过 Visual Basic（简称 VB）的读者一定会注意到在 VB 中便捷的函数提示。在 VC++ 6.0 中，输入时在线的丰富提示也实现了。这使得程序员从记忆众多的函数参数中解脱出来，从而极大地提高了编程效率。

⑩ 上下文相关的 What's This 帮助。上下文帮助信息，为初学者提供了学习的平台，初学者可以通过帮助更好地了解 VC++ 6.0 各工具的使用及菜单中各子菜单的作用。

（2）用户界面。从 Visual Studio 的光盘中运行 VC++ 6.0 安装程序（Setup.exe），安装完成之后，就可以从【开始】菜单中运行 VC++ 6.0，如图 1-1 所示。通常，VC++ 6.0 的外观如图 1-2 所示。

图 1-1　启动 VC++ 6.0 界面

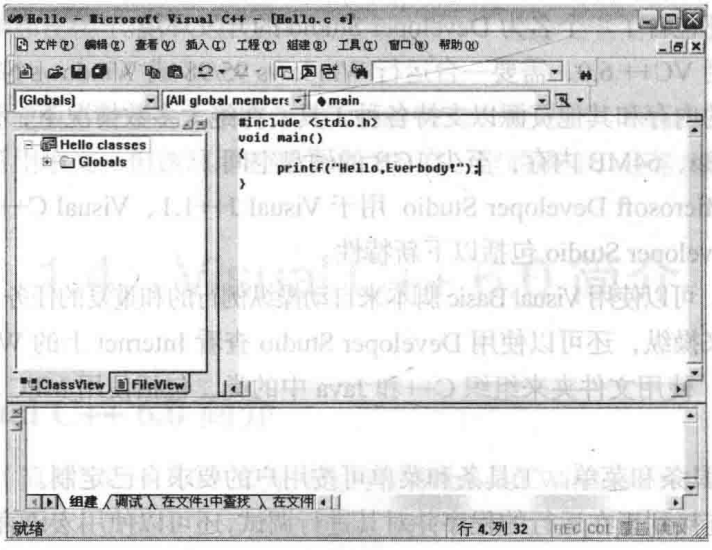

图 1-2　VC++ 6.0 的环境界面

VC++ 6.0 带有一个预先定义好的工具栏集，单击便可以访问它们。如果需要更多的工具按钮，可以自己设计和定制工具栏来增大工具栏集。

图 1-2 所示为“Standard”和“Wizardbar”工具栏定位在 VC++ 6.0 窗口的顶端时的界面。工具栏的安排可以在窗口的四边任意位置，可以在屏幕四周移动工具栏，通过拖曳边框来调整它们的矩形大小，也可以使任何工具栏可见或隐藏。例如，默认设置中，“Debug”工具栏只在调试过程中才可见。当鼠标停留在工具栏按钮的上面时，按钮会凸起，主窗口中的状态栏显示了该按钮