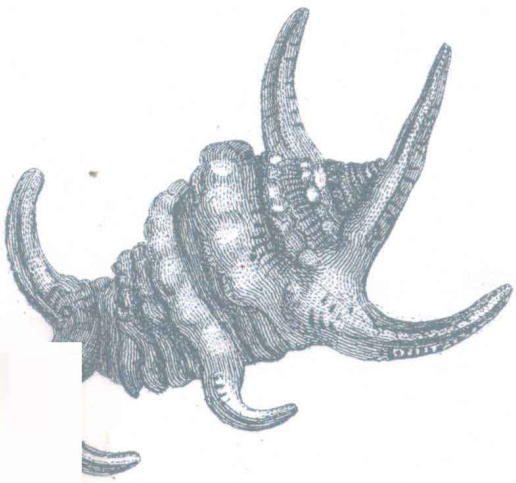
工业和信息化普通高等教育“十二五”规划教材立项项目
21世纪高等学校计算机规划教材
21st Century University Planned Textbooks of Computer Science

C语言程序设计——从入门到进阶

Programming in The C Language ——From Foundation to Advancement

巨同升 主编
孙福振 薛磊江 贾凌 副主编



高校系列

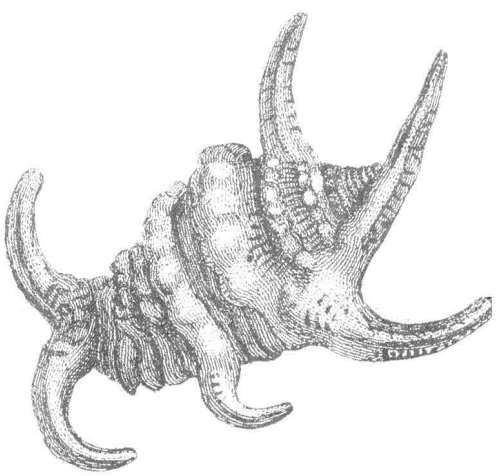
人民邮电出版社
POSTS & TELECOM PRESS

工业
21
21st Cc
教育“十二五”规划教材立项项目
划教材
ater Science

C语言程序设计——从入门到进阶

Programming in The C Language ——From Foundation to Advancement

巨同升 主编
孙福振 薛磊江 贾凌 副主编



高校系列

人民邮电出版社
北京

图书在版编目（CIP）数据

C语言程序设计 ：从入门到进阶 / 巨同升主编. --
北京 ：人民邮电出版社，2011.12
21世纪高等学校计算机规划教材
ISBN 978-7-115-27216-4

Ⅰ. ①C… Ⅱ. ①巨… 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(2011)第275626号

内 容 提 要

本书在内容编排上，采用“应用驱动知识”的方式，即根据每一章应用目标的需求，合理地安排每一个知识主题的引入点，从而将C语言中枯燥难懂的语法知识分解到全书各章并融入到丰富的实例中。本书在编程方法的讲解上，采用“逐步构造法”，即通过问题分析、算法设计、程序原型等环节一步一步地构造出完整的程序，从而加深读者对编程方法的理解和掌握。

本书主要内容包括引论、基本数据与运算、顺序结构程序设计、选择结构程序设计、循环结构程序设计、数组、指针、字符串处理、函数、函数的进一步讨论、编译预处理命令、结构体与共用体、位运算、文件等。本书以Visual C++ 6.0运行环境为基础进行介绍，符合当前软件的发展趋势，便于读者学习。

本书内容编排顺畅合理，编程方法讲解新颖独特，特别适合初学者自学。本书可作为高等院校各专业学生学习C语言程序设计的教材和参考书。

工业和信息化普通高等教育“十二五”规划教材立项项目

21世纪高等学校计算机规划教材

C语言程序设计——从入门到进阶

◆ 主　　编　巨同升
副 主 编　孙福振　薛磊江　贾　凌
责任编辑　董　楠

◆ 人民邮电出版社出版发行　　北京市崇文区夕照寺街14号
邮编　100061　　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：14.75　　2011年12月第1版
字数：381千字　　2011年12月河北第1次印刷

ISBN 978-7-115-27216-4

定价：26.50元

读者服务热线：(010)67170985　印装质量热线：(010)67129223
反盗版热线：(010)67171154
广告经营许可证：京崇工商广字第0021号

怎样学好C语言（代序）

C语言作为一门专业型的语言，具有功能强大、效率高、实用性强等特点。但是C语言作为学习程序设计的普及型入门语言，却存在着诸多不足之处，如C语言的语法过于灵活，C语言的指针功能过于强大等。凡此种种，往往给初学者造成了极大的困惑，甚至严重打击了初学者的自信心。

其实，我们只要回归到学习C语言的终极目标上去，这些困惑均可一一破解。学习C语言的目的是学会编写程序解决实际问题，而不是单纯为了熟练地通晓C语言的语法和各种所谓的编程技巧。

因此，学习C语言应该注意以下几个方面。

（1）C语言的学习包括语法和编程两条主线，这两条主线应该相互协调、同步进展。比如，为了学习顺序结构程序设计，只需先学习顺序结构程序设计所必需的语法即可；而对于顺序结构程序设计不是必不可少的语法，则可以延后再学，如自增自减运算符的用法，完全可以延后到循环结构程序设计中再学习。

（2）语法是为编程服务的，脱离了编程的语法是毫无意义的。比如，有些在编程实践中极少用到的语法，就可以暂不作探讨。如printf函数中各种格式符的详细用法，就不必一下子全部掌握，而只需掌握编程实践中常用的几种用法即可。

（3）学习中要遵循由初阶到高阶、循序渐进的原则。在C语言中，顺序结构、选择结构、循环结构属于初阶应用；数组、函数属于中阶应用；指针、结构体、文件属于高阶应用。学习者应该在熟练地掌握了初阶应用之后，再学习中阶应用；在熟练地掌握了中阶应用之后，再学习高阶应用。有些抽象难懂的高阶应用，如指针的高级应用、链表的应用等，就应该在学习者达到了一定的编程水平之后，再来探讨。

（4）学习编程的制胜法宝是上机调试程序。编程并不完全是一种书面上的推导演算，程序最终要由计算机来执行。只有通过上机调试，才能发现问题、解决问题。因此要不厌其烦地上机调试程序。最初先从最简单的例题入手，在能够比较顺利地调试简单例题之后，再来调试自己编写的程序。

在调试程序时，要逐步地学会借助于出错信息和输出中间结果，来定位出错位置。因为有时候一个错误会导致多条出错信息，因此不必等找出所有错误之后再重新编译程序。

（5）学习C语言的目的是编程解决实际问题。因此在具有了一定的编程能力之后，应当着手将编程与实际应用结合起来。若能够利用编程解决工作生活中的实际问题，那么你的编程能力就达到了一个新的境界。

编　者

2011年10月

前 言

C 语言是目前世界上使用最广泛的高级程序设计语言，广泛地应用于系统程序设计、数值计算、自动控制等诸多领域。

C 语言的产生颇为有趣。C 语言实际上是 UNIX 操作系统的一个副产品。在 1972 年，美国贝尔实验室的 D.M.Ritchie 为了开发 UNIX 操作系统，而专门设计出了一种新的语言——C 语言。由于 C 语言具有强大的功能、很高的编程效率，兼具高级语言的直观性与低级语言的硬件访问能力，因而很快从贝尔实验室走入了广大程序员的编程世界。

由于 D.M.Ritchie 设计 C 语言的初衷是用于开发 UNIX 操作系统，因此 C 语言称得上是一门专业型的语言。这使得 C 语言在具有强大的功能、很高的编程效率的同时，也在一定程度上存在语法晦涩难懂、不便于初学者掌握的不足之处。

因此，C 语言似乎不太适合作为程序设计初学者的入门语言。不过在现代人的效率观念驱使下，仍有许多学校将 C 语言选作初学者的入门语言。

其实，这种做法也未尝不可。只不过在教学中应思考如何采取有效的应对措施，使初学者避开那些晦涩难懂的语法，从 C 语言中最基本、最实用的编程方法入手，力争使学生尽快地学会程序设计的基本方法，进而达到应用编程解决实际问题的境界。

从学习者的角度来说，要注意抓住 C 语言学习的要害所在——编程方法，而不要沉溺于 C 语言语法的细节之中。因为学习 C 语言的目的是学会编写程序解决实际问题，而不是单纯为了通晓 C 语言的语法和各种所谓的编程技巧。

针对上述问题，本书作者在教学内容的编排上，采用了“应用驱动知识”的方式，即根据每一章应用目标的需求，合理地安排每一个知识主题的引入点，从而将 C 语言中枯燥难懂的语法知识分解到全书各章并融入到丰富的实例中，本书在编程方法的讲解上，采用“逐步构造法”，即通过问题分析、算法设计、程序原型等环节一步一步地构造出完整的程序，从而加深读者对编程方法的理解和掌握。

本书第 1 章、第 12 章由贾凌、周洁编写，第 2 章、第 3 章、第 9 章和附录 A 由巨同升编写，第 4 章、第 13 章由于潇、高馨编写，第 5 章、第 11 章由冷淑霞、王立香编写，第 6 章由薛磊江编写，第 7 章、第 10 章、第 14 章由孙福振、刘冬霞编写，第 8 章由薛磊江、孙福振编写。全书由巨同升统筹并定稿。

在本书编写过程中得到了山东理工大学计算机科学与技术学院广大同仁的大力支持与帮助，在此表示感谢。

由于编者水平所限，书中难免存在不足之处，请广大专家和读者批评指正。

编 者

2011 年 10 月

试读结束：需要全本请在线购买： www.ertongbook.com

目 录

第1章 引论

计算机的出现为我们的工作、生活带来了全新的体验。我们利用计算机解决各种问题，一般是基于各种软件的支持来实现的。例如，使用字处理软件编辑文档，使用媒体播放器观看视频等。各类软件已经成为我们工作、生活中的一个重要角色。在软件的设计开发过程中，一项非常重要的工作就是编制计算机程序。本章将简要介绍程序设计的一般步骤，以及C语言的基本知识，希望读者以此为起点，步入程序设计的精彩世界。

1.1 程序与程序设计语言

现实生活中，我们面对任何一项较复杂的任务时，都需要通过一系列的操作过程来实现。这些按照一定顺序安排，进行操作的任务，通常称为程序。比如，以下为一个从自助取款机提取现金的程序：

① 插入银行卡；

② 输入密码；

③ 确定取款金额；

④ 取款；

⑤ 退卡。

在这个简单的过程中，我们按一定动作顺序实现了对银行卡、现金等对象的各种操作。

对于计算机而言，程序是指完成某一特定任务的一组指令序列。编写计算机程序所采用的语言形式称为程序设计语言。随着计算机科学技术和计算机应用水平的不断提高，程序设计语言由低级到高级逐步发展起来，目前已经有上千种计算机程序设计语言出现，通常被分为低级语言和高级语言两大类。

面向机器的计算机语言称为低级语言，机器语言和汇编语言都是低级语言。机器语言是计算机能够直接识别和执行的计算机语言。机器语言中的每条指令都是二进制指令，因此，机器语言占用内存少，执行速度快，但是编写程序难度大，修改调试不方便，而且，机器语言随不同计算机而异，通用性差。汇编语言是符号化的机器语言，用一些形象化的符号代替机器语言的二进制指令，比机器语言容易理解和使用。但是，与机器语言相同，随机而异，通用性差。

为了克服低级语言给程序员带来的困难，20世纪50年代，出现了高级语言。这是一种与具体计算机硬件无关，表达方式接近人类自然语言和数学公式的程序设计语言，便于理解和维护，极大地提高了程序设计的效率和可靠性。

用程序设计语言编写正确有效的程序，解决特定问题的过程称为程序设计。对于一般的问题，

程序设计通常可以分为以下几个步骤。

（1）分析问题，建立数学模型

首先，对问题进行充分的分析，确定问题是什么，然后，针对要解决的问题，找出已知的数据和条件，确定需要使用的数据量，建立各种数据量之间的关系，即建立解决问题的数学公式或数学模型。很多情况下，需要利用已有的基本数学模型做进一步的分析、推理来构造出解决问题的模型。数学模型的建立，在很大程度上决定了程序的正确性和复杂度。

（2）确定算法

算法，是解决问题的方法和步骤。根据建立的数学模型，确定合适的算法加以实现。算法的好坏，直接影响到一个程序的质量。

（3）编程

根据确定的算法，利用程序设计语言按照算法顺序和程序设计语言的规则，编写出具体的程序代码。

（4）调试程序

程序编写完毕后，必须对程序进行调试，尽可能地发现程序中的错误和缺陷，加以纠正，分析运行结果。

1.2 C 语言的发展及特点

1.2.1 C 语言的发展

C 语言是一门通用的模块化的编程语言，被广泛应用于操作系统和应用软件的开发。由于其高效、灵活、功能强、可移植性好，所以深受开发人员的青睐。自从 C 语言诞生，其发展和应用的程度是目前其他任何一种语言所无法比拟的。

1972 年，美国贝尔实验室的 D.M.Ritchie 在 Ken Thompson 发明的 B 语言的基础上设计了 C 语言。C 语言的诞生和 UNIX 操作系统的开发密不可分，原先的 UNIX 操作系统都是用汇编语言编写的，1973 年 Ken Thompson 和 D.M.Ritchie 将 UNIX 操作系统的核心用 C 语言改写，即 UNIX 第五版，从此以后，C 语言成为编写操作系统的主要语言。

1978 年美国贝尔实验室正式发表了 C 语言。同时，以 UNIX 第七版中的 C 编译程序为基础，Brain W.Kernighan 和 Dennis M.Ritchie 合著了《The C Programming Language》一书。书中描述的 C 语言成为后来广泛使用的 C 语言版本的基础，被称为标准 C。

C 语言被广泛应用，衍生了 C 语言的很多不同版本。1983 年美国国家标准化协会(ANSI)根据各种 C 语言版本对 C 语言的扩充和发展，制定了 ANSI C 标准。1987 年，美国国家标准化协会又颁布了新标准，称为 87 ANSI C。

1990 年国际标准化组织 ISO 接受了 87 ANSI C 作为 ISO C 的标准，这是目前使用最广泛的一个标准版本，几乎所有的开发工具都支持此标准。最新的 C 语言标准是在 1999 年颁布并在 2000 年 3 月被 ANSI 采用的 C99。

C 语言同时具有高级语言和汇编语言的优点，很多编程语言都是在 C 语言的基础上发展起来的，比如 C++、C#、Java、Javascript 等。也正因为 C 语言的影响力，先掌握 C 语言，再学其他编程语言，大多能触类旁通，很快掌握。

试读结束：需要全本请在线购买： www.ertongbook.com

目前，在微机上广泛使用的C语言版本有Microsoft C、Turbo C、Visual C/C++等。本教材以Visual C++ 6.0作为实践环境。

1.2.2 C语言的特点

C语言之所以具有强大的生命力，成为最受欢迎的语言之一，主要在于C语言与其他语言相比，有其非常显著的优点，主要有以下几点。

（1）语言简洁，使用灵活

C语言一共只有32个关键字，9种控制语句，程序书写自由，主要用小写字母表示，压缩了一切不必要的成分。

（2）运算符丰富

C语言共有34种运算符，表达式类型多样化，可以实现其他高级语言难以实现的运算。

（3）数据结构类型丰富

C语言具有各种各样的数据类型，有整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型和枚举类型等。它们能用来实现各种复杂的数据结构，使C语言具有很强的数据处理能力。

（4）控制流程结构化

C语言具有结构化控制语句，并用函数作为程序模块，以实现程序的模块化。

（5）允许直接访问物理地址

C语言能实现汇编语言的大部分功能，可以直接对硬件进行操作。它既有高级语言的特点，又能完成低级语言的许多功能，因此有人把它称为中级语言。

（6）生成目标代码质量高，程序执行效率高

试验表明，C语言编写的程序，代码效率只比汇编语言的代码效率低10%～20%，而优于其他高级程序设计语言。C语言程序速度快，可读性好，易于调试和修改。

（7）可移植性好

C语言程序本身不依赖于具体的硬件系统和软件环境，从而便于在不同的计算机间和各种操作系统中实现程序的移植。

C语言具有如上所述诸多优点，因此具有旺盛的生命力，是描述系统软件和应用软件比较理想的工具。但也有不足之处，应在编程时引起注意，比如C语言语法限制不严格、灵活性大，对程序员的要求相对较高。

1.3 C程序的构成

为了对C语言有一个感性的认识，下面我们来预览一个简单的C语言程序。

【例1.1】 已知做匀速直线运动物体的速度为20米/秒，运动时间为10秒，编程序求其位移。

问题分析：

这是一个很简单的物理问题，我们可以轻而易举地求得问题的答案。不过，这里需要我们完成的是，编写一个程序让计算机通过执行这个程序求得问题的答案。

如何编写这个程序呢？别着急，有人已经替我们编写好了这个程序，让我们来看一下。

```
#include <stdio.h>
```

```
main()
{
int v,t,s;
v=20;
t=10;
s=v*t;
printf("%d",s);
}
```

这就是一个简单而完整的 C 语言程序。我们来分析一下它的结构组成。

C 语言是一种函数型语言，每个 C 语言程序都是由若干个函数组成的。即函数是 C 语言程序的基本构成单位。

该程序由一个函数构成。其中的 main()是函数首部（也称为函数头），而 main 是它的函数名。

以一对花括号括起来的部分是函数的主体部分，称为函数体。函数体是由若干条语句构成的。

下面分析一下各条语句的功能。

```
int v,t,s;
```

这条语句定义了 3 个变量 v、t、s，分别用来存储 3 个物理量的值。一般来说，有几个物理量就定义几个变量。其中的 int 表示这 3 个变量是用来存储整数的变量。

```
v=20;t=10;
```

这两条语句用来分别将一个数存入到一个变量中。也就是将两个已知量的值告诉计算机。

```
s=v*t;
```

这条语句用来将未知量与已知量的关系告诉计算机，即通过已知量 v、t 求得未知量 s 的值。

```
printf("%d",s);
```

这条语句用来在显示器上输出变量 s 的结果。其中的"%d"表示以整数格式输出。

通过以上例子，我们可以总结出简单 C 语言程序的一般构成。

（1）定义变量。变量的作用是存储各个物理量的值。一般来说，有几个物理量就需要定义几个变量。

（2）输入已知量的值。也就是将已知量的值告诉计算机。

（3）将已知量与未知量的关系告诉计算机，求得未知量的值。

（4）输出求得的未知量的值。

【例 1.2】 已知做匀加速直线运动物体的初速度为 90 米/秒，经过 10 秒之后的速度为 60 米/秒，编程序求其加速度。

源程序：

```
#include <stdio.h>
main()
{
int v0,vt,t,a;
v0=90;
vt=60;
t=10
a=(v0-vt)/t;
printf("%d",a);
}
```

综合上面的例子，我们来看一下 C 语言程序最基本的构成规则。

（1）一个 C 语言程序是由若干个函数组成的。其中必须有一个主函数（main 函数）。

可见函数是 C 语言程序的基本构成单位（C 语言中的函数相当于其他语言中的子程序）。

（2）一个 C 语言程序总是从 main 函数开始执行的，而不论 main 函数位于其他函数之前或其

他函数之后。

（3）一个函数由函数首部和函数体两部分组成。最简单的函数首部，是由函数名以及其后的一对圆括号组成，如 main()。而函数体是由括在一对花括号中的若干条语句组成的。

（4）每条语句的末尾必须有一个分号。分号是 C 程序语句必不可少的组成部分。

（5）C 程序书写格式自由。既可以将几条语句写在同一行上，也可以将一条语句写在几行上。因为可以用分号来区分不同的语句。

（6）为了增强程序的可读性，可以对 C 程序中的任意部分作注释说明。注释信息必须写在/*和*/之间。它们对程序的执行不产生任何影响。

1.4　C 程序的运行

编写出的 C 程序需要在一定硬件和软件环境下进行编辑、编译、连接、运行，直到得到正确的运行结果。下面简单介绍 Visual C++ 6.0 环境下 C 程序的上机步骤。

1. 编辑源程序

打开 Visual C++ 6.0 的主窗口，如图 1.1 所示；主窗口的左侧是项目工作区，右侧是编辑窗口。

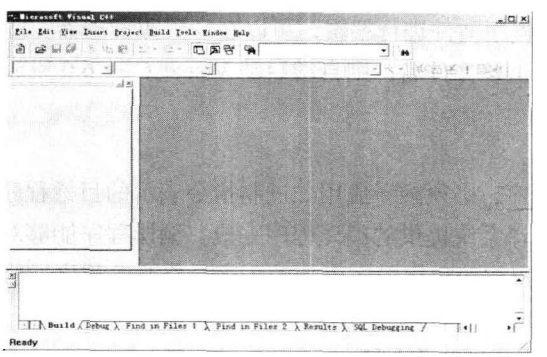


图 1.1　Visual C++ 6.0 主窗口

在主菜单栏中单击“File（文件）”，在其下拉菜单中选择“New（新建）”命令，弹出“New（新建）”对话框。单击该对话框上方的“File（文件）”选项卡，选择“C++ Source File（建立新的 C 源程序文件）”。然后，在该对话框的“File（文件名）”文本框中输入文件名（如 EX1_1.c），在“Location（路径）”文本框中输入源程序文件的存储路径（如 E:\C 程序），如图 1.2 所示。

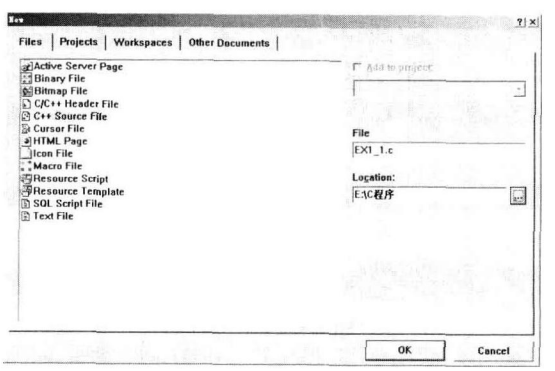


图 1.2　新建对话框

单击 OK 按钮，将返回到 Visual C++ 6.0 的主窗口，就可以在编辑窗口中编辑源程序了。例如输入例 1.1 中的 C 程序，如图 1.3 所示。

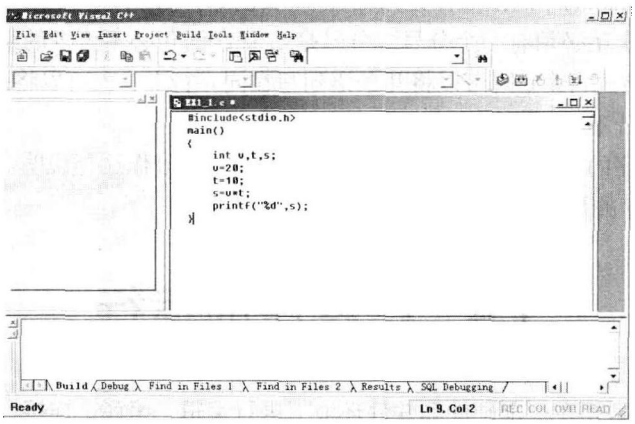


图 1.3　输入源程序后的主窗口

新的源程序输入并检查无误后，可以将它保存在文件中。单击主菜单栏的“File（文件）”，选择“Save（保存）”命令。

如果需要对已有的源程序文件进行编辑，可以在“资源管理器”或“我的电脑”中找到已有的 C 源程序文件，然后双击该文件名，则自动启动 Visual C++ 6.0 集成环境，源程序文件的内容显示在编辑窗口中。

2. 编译源程序

用 C 语言编写的源程序，必须翻译成用二进制指令表示的目标程序，才能被计算机识别，这个过程称为编译，由 C 编译系统提供的编译程序完成。编译程序能够对源程序进行语法检查，当发现错误时，将错误信息提供给程序员，以帮助程序员修改源程序，修改后的源程序必须重新进行编译。编译通过后，生成扩展名为“.OBJ”的目标程序。

对例 1.1 源程序进行编译的操作如下。

单击主菜单栏中的“Build(构建)”，在其下拉菜单中选择“Compile EX1_1.c（编译 EX1_1.c）”命令，如图 1.4 所示。

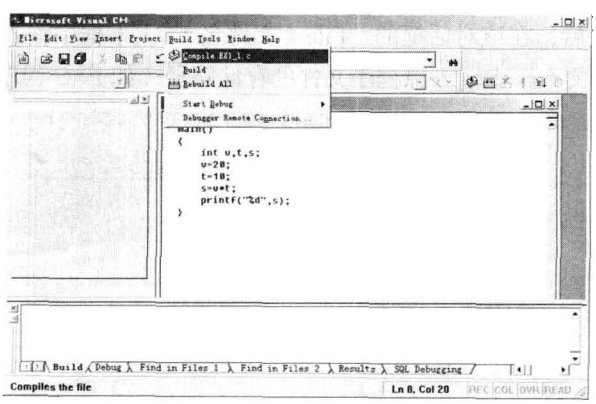


图 1.4　编译程序

单击编译命令后，屏幕出现如图 1.5 所示的提示信息，意思是：此编译命令要求一个活动的

项目工作区，你是否同意建立一个默认的项目工作区？单击“是（Y）”，表示同意建立默认项目工作区。

屏幕如果继续出现“将改动保存到 E:\C 程序\EX1_1.c”，单击“是（Y）”。

编译过程检查源程序有无语法错误，如果有错，错误的类型和位置将显示在屏幕下方的调试信息窗口。编译结束，产生 EX1_1.obj 目标文件，如图 1.6 所示。

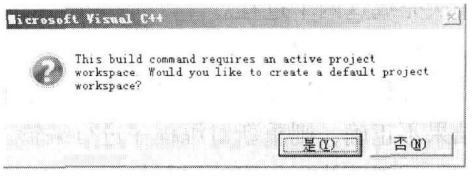


图 1.5　提示创建默认的项目工作区

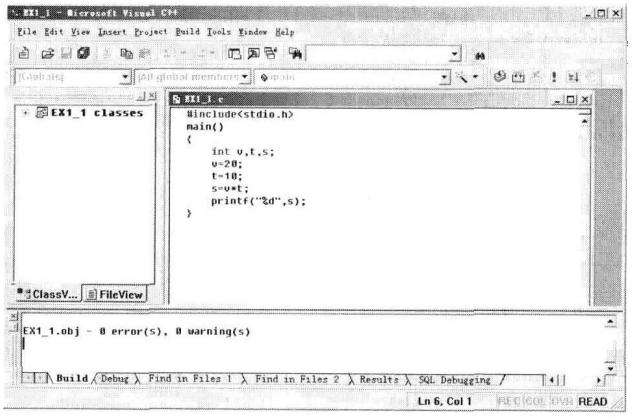


图 1.6　编译结束

3. 连接

在得到了目标程序后，需要由系统提供的连接程序将目标程序、库函数或其他目标程序进行连接，最后生成一个可执行文件。在连接阶段，如果发现有连接错误，在编辑修改以后，一定要重新编译和连接。

对编译后的程序进行连接的操作如下。

单击主菜单栏中的“Build(构建)”，在其下拉菜单中选择“Build EX1_1.exe(构建 EX1_1.exe)”命令，如图 1.7 所示。

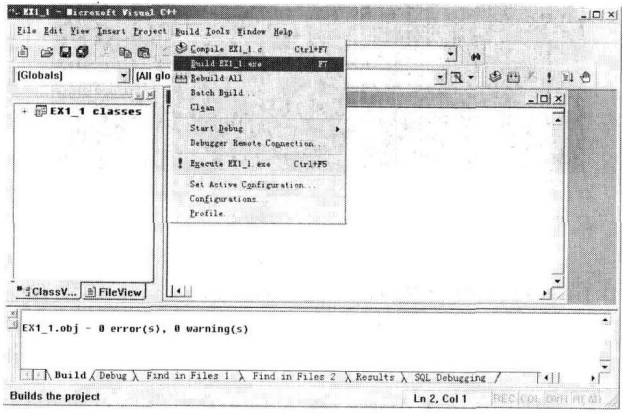


图 1.7　执行构建命令

成功完成连接后，生成一个可执行文件 EX1_1.exe。

以上是分别进行了程序的编译和连接，用户也可以通过选择“Build”菜单下的“Build”命令

一次完成这两个过程。

4. 运行程序

源程序经过编辑、编译、连接，生成可执行文件之后，即可运行以获取处理结果。如果运行结果不正确，则重新对源程序进行编辑、编译、连接、运行的过程。

运行可执行程序的过程如下。

单击主菜单栏中的“Build（构建）”，在其下拉菜单中选择“！ Execute EX1_1.exe（执行 EX1_1.exe）”命令，如图 1.8 所示。

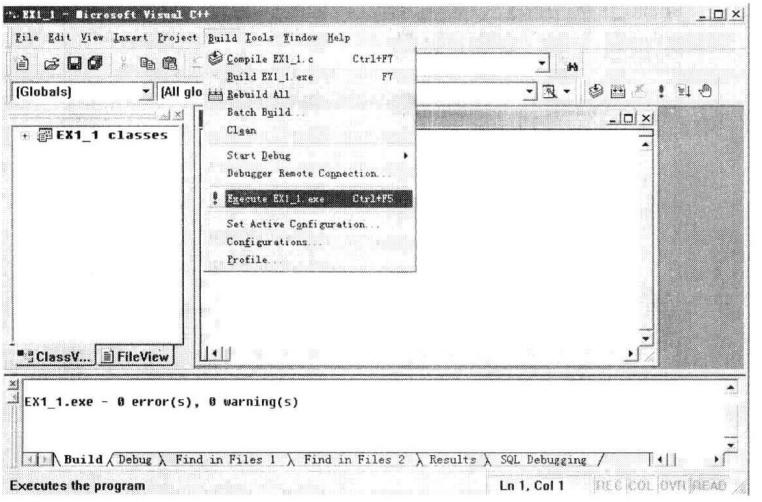


图 1.8 运行程序

程序运行后，屏幕切换到输出结果的窗口，显示出运行结果，如图 1.9 所示。运行例 1.1 可以看到结果“200”，按下任意一键后，输出窗口消失，返回到 Visual C++ 6.0 的主窗口。

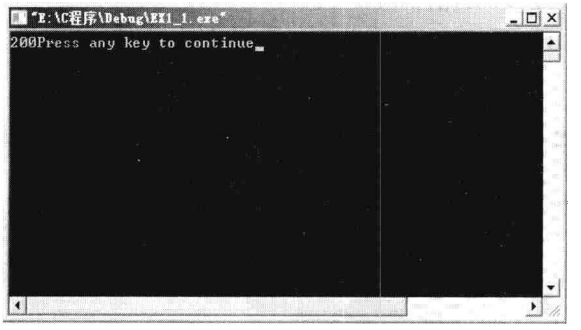


图 1.9 运行结果

在一个程序中存在错误是在所难免的，往往需要经过若干次编辑、编译、连接、运行的反复过程，才能获得正确的运行结果。程序中最主要的三种错误是语法错误、逻辑错误和运行错误。

语法错误能够在编译过程中被检查出来，并给出错误提示信息供程序员参考。而逻辑错误和运行错误则不会被编译程序和连接程序所发现。这两类错误通常是由于算法设计不合理、数据使用不合理等非语法问题造成的，主要表现为实际的运行结果与期望不相符，通常需要依靠程序员的经验来查找和改正。

试读结束：需要全本请在线购买： www.ertongbook.com