

高职高专“十一五”规划教材



C 语言程序设计

姚宏坤 左红岩 主编



中国电力出版社

www.infopower.com.cn

高职高专“十一五”规划教材

C 语言程序设计

姚宏坤 左红岩 主编



中国电力出版社
www.infopower.com.cn

“十一五”规划教材

内容提要

本书针对高职高专学生的认知特点,由具有多年C语言程序设计教学经验的一线教师根据实践教学经验 and 体会编写而成的。书中使用“示范实例—模仿实例—思考与验证—常见错误列举—实训”的体例。每章首先给出学习目标,明确对知识、技能掌握程度的要求,然后选用大量具有实用性、趣味性及为学生所熟悉的问题,作为实例与实训任务,便于教师组织互动式教学。

本书适合作为高等职业院校、高等专科学校以及成人高校的教材,也适合广大计算机爱好者及参加培训、考级的人员作为自学和参考用书。

图书在版编目(CIP)数据

C语言程序设计 / 姚宏坤, 左红岩主编. —北京: 中国电力出版社, 2008

高职高专“十一五”规划教材

ISBN 978-7-5083-6581-7

I. C… II. ①姚… ②左… III. C语言—程序设计—高等学校: 技术学校—教材 IV. TP312

中国版本图书馆CIP数据核字(2008)第012237号

丛 书 名: 高职高专“十一五”规划教材

书 名: C语言程序设计

出版发行: 中国电力出版社

地 址: 北京市三里河路6号

邮政编码: 100044

电 话: (010) 68362602

传 真: (010) 68316497, 88383619

服务电话: (010) 58383411

传 真: (010) 58383267

E-mail: infopower@cepp.com.cn

印 刷: 航远印刷有限公司

开本尺寸: 185mm×233mm

印 张: 17.25 字 数: 388千字

书 号: ISBN 978-7-5083-6581-7

版 次: 2008年2月北京第1版

印 次: 2008年2月第1次印刷

印 数: 0001—4000册

定 价: 26.00元

敬告读者

本书封面贴有防伪标签,加热后中心图案消失
本书如有印装质量问题,我社发行部负责退换

版 权 专 有 翻 印 必 究

前 言

C 语言是国内外广泛使用的一种计算机语言,是计算机应用人员,特别是硬件产品开发和底层程序开发人员应该掌握的程序设计工具之一。C 语言以其功能丰富、表达能力强、使用灵活方便、应用面广、目标程序效率高、可移植性好而成为软件开发中的主流。

由于大多数操作系统都采用 C 语言开发,如 Windows、UNIX、Linux、Mac OS 等;还有很多新型的语言都是由 C 语言衍生的,如 C++、Java、C#等。可以这样说,掌握了 C 语言,就等于掌握了很多门语言,经过简单的学习,就可以用这些新型的语言去开发程序。因此,学习和使用 C 语言已经成为广大计算机应用人员和学生的迫切需求。鉴于此,我们在多年从事教学及实践应用的基础上总结经验,参考国内外有关资料编写了本书。

全书包括以下内容:第 1 章程序设计概述;第 2 章数据类型与数据的输入输出;第 3 章表达式与语句;第 4 章控制结构;第 5 章数组;第 6 章函数;第 7 章指针;第 8 章结构;第 9 章文件;附录。

本书采用了目前流行的“任务驱动”模式,使用“示范实例—模仿实例—思考与验证—常见错误列举—实训”的体例编写。每章首先给出学习目标,明确对知识、技能掌握程度的要求,以求学生能熟练掌握基础知识和基本技能。同时,教师可利用书中大量例题组织互动式教学。其中,“示范实例”可作为教师授课时的选用例题,每个实例均从问题分析到算法表示,从程序代码编写到运行结果分析逐一列出;“模仿实例”可作为学生上课时的课堂练习或课后习题;“思考与验证”是对一些技巧性强、容易模糊的知识要点,采取启发和引导的形式,使学生把理性思考和动手验证结合起来;“常见错误列举”帮助学生克服头脑中错误的语言规则应用模式,通过提供 C 语言规则的应用反例,使学生能够事先鉴别出常见的错误应用模式;而“实训”中的习题可作为实训课的选取内容,以期提高学生程序编写和调试能力。

书中的全部例题均以目前最为广泛使用的 Visual C++为开发平台,并通过验证。

本书的读者对象主要是高职高专工科各专业的学生。

全书由姚宏坤、左红岩编写，其中第1~4章由姚宏坤执笔，第5~9章由左红岩执笔，全书由姚宏坤负责统稿。

本书配有相应的电子教案及例题库，需要的老师可与编者联系，联系方式：
yhk1968@126.com。

限于编者水平，书中难免存在疏漏和不足之处，敬请广大师生及读者批评指正。

编者

2007年12月

目 录

前 言	1
第 1 章 程序设计概述	1
1.1 程序设计入门	1
1.2 创建和运行一个 C 语言程序	4
1.3 程序设计的基本步骤	10
1.4 Visual C++ 集成开发环境	11
1.5 实训	14
第 2 章 数据类型与数据的输入输出	16
2.1 C 语言的数据类型	16
2.2 数据在 C 程序中的表示	18
2.3 数据的输入输出	22
2.4 常见错误列举	31
2.5 实训	33
第 3 章 表达式与语句	36
3.1 C 语言的表达式	36
3.2 C 语言的语句	43
3.3 编写简单 C 语言程序	44
3.4 程序执行过程的跟踪	48
3.5 常见错误列举	53
3.6 实训	54
第 4 章 控制结构	55
4.1 C 语言的程序控制结构	55
4.2 选择结构程序设计	56
4.3 常见错误列举	77
4.4 实训	78
4.5 循环结构程序设计	79
4.6 常见错误列举	110
4.7 实训	111
4.8 使用断点调试程序	112
第 5 章 数组	118
5.1 一维数组	118

5.2	二维数组	134
5.3	字符数组与字符串	142
5.4	常见错误列举	161
5.5	实训	162
第 6 章	函数	164
6.1	函数的基本概念	164
6.2	函数的定义与调用	167
6.3	变量作用域和存储类别	175
6.4	函数应用举例	179
6.5	函数的跟踪调试	192
6.6	常见错误列举	201
6.7	实训	202
第 7 章	指针	203
7.1	指针变量的定义	203
7.2	指针与数组	206
7.3	指针与字符串	210
7.4	指针与内存的动态分配	219
7.5	常见错误列举	224
7.6	实训	226
第 8 章	结构	227
8.1	结构变量	227
8.2	结构数组	233
8.3	结构应用举例	234
8.4	常见错误列举	244
8.5	实训	245
第 9 章	文件	246
9.1	C 文件概述	246
9.2	文件的读写	249
9.3	常见错误列举	259
9.4	实训	259
附录 A	常用字符与 ASCII 代码对照表	261
附录 B	C 语言关键字	263
附录 C	C 语言常见库函数	264
参考文献		268

第 1 章 程序设计概述

学习目标:

通过本章学习,你将能够

- ☒ 了解算法和计算机程序的概念
- ☒ 初步掌握算法的自然语言表达方式
- ☒ 了解简单 C 程序结构
- ☒ 了解程序设计的基本步骤
- ☒ 掌握 Visual C++ 6.0 开发环境下程序的编辑、编译、连接和执行操作

1.1 程序设计入门

1.1.1 程序和算法

“程序”一词来自于生活,通常指完成某些事务的一种既定方式和过程。可以将程序看成对一系列动作执行过程的描述。日常生活中可以找到许多“程序”实例。例如去银行取钱的行为可以描述为:

- (1) 带上存折去银行;
- (2) 填写取款单;
- (3) 将存折和取款单递给银行职员;
- (4) 银行职员办理取款事宜;
- (5) 拿到钱;
- (6) 离开银行。

日常生活中程序性活动的情况与计算机里的程序执行很相似,这一情况可以帮助我们理解计算机的执行方式。

人们使用计算机,就是要利用计算机处理各种不同的问题。不要忘记计算机是机器,需要人们告诉它们工作的内容和完成工作的方法。为使计算机能按照人的指挥工作,计算机提供了一套指令,其中每一种指令对应着计算机能执行的一个基本动作。为让计算机完成某项任务而编写的逐条执行的指令序列就称为程序。在解决数学问题时,程序就是解决数学问题的步骤。例如求两数之和的解决步骤如下:

- (1) 获得要计算的数;

(2) 求出两数之和;

(3) 显示计算结果。

为了让计算机能够准确无误地完成任任务,人们就必须事先对各类问题进行分析,确定解决问题的具体方法和步骤,再编制好一组让计算机执行的指令,交给计算机,让计算机按人们指定的步骤有效地工作。这些具体的方法和步骤,其实就是解决一个问题的算法。由此可见,程序设计的关键之一就是设计解题的方法与步骤,即算法。算法可以有许许多多不同的形式来表达,像上面那样用(1)、(2)、(3)逐条列出,是一种用自然语言形式描述的算法。这种形式能够让人理解,而程序是能够让计算机理解和执行的,因此,前者往往不那么精确,语法、格式可以比较自由,后者则必须符合一套严格的语法规则,我们在学习和实践中必须充分重视,直到熟悉并掌握它。

【示范实例 1-1】用自然语言表达求解一位学生 3 门课程的考试成绩和平均分的算法。

(1) 获得要计算的 3 个数;

(2) 求出 3 个数之和;

(3) 把和除以 3;

(4) 显示和及平均分。

【示范实例 1-2】用自然语言表达求解圆的面积和周长的算法。

(1) 获得圆的半径 r ;

(2) 求出圆的面积 $s = \pi r^2$;

(3) 求出圆的周长 $l = 2\pi r$;

(4) 显示圆的面积和周长。

这个实例采用了代数符号来表示数据和运算,使叙述变得简洁、精确。

【模仿实例 1-3】用自然语言表达求解一元二次方程的算法。

(1) 获得一元二次方程 $ax^2 + bx + c = 0$ 的 3 个系数 a 、 b 、 c ;

(2) 计算 $b^2 - 4ac$, 得到中间结果 d ;

(3) 计算 d 的算术平方根 s ;

(4) 分别计算 $(-b+s)/(2a)$ 和 $(-b-s)/(2a)$;

(5) 显示一元二次方程的两个根。

1.1.2 一个简单 C 语言程序

下面通过一个简单的程序实例,来体会一下 C 程序。

```
/*
```

```
源文件名: hello.c
```

```
功能: 显示 Hello World!
```

```
*/
```

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

```
void main()
```

```
{  
    printf("Hello World!\n"); //在显示器上输出 Hello World!  
}
```

程序执行后，在屏幕上显示如下内容：

Hello World!

下面是对这个程序的一些解释，读者目前只需在表面上了解，暂且不必深究，随着以后的学习，会逐渐加深理解。

(1) “/*……*/”表示注释部分，只是给人看的，目的是帮助有关人员了解、管理程序。计算机在接受程序时，忽略此内容。注释分为行注释和块注释，行注释用“//”表示，它的范围只到本行结束，不允许跨行。块注释用“/*……*/”表示。两种注释均可以加在程序中的任何位置。为了便于理解，我们用汉字表示注释，当然也可以用英语或汉语拼音作注释。

(2) “#include”：以#开始的语句称为预处理命令语句。因为在编译器开始工作之前，先对这些命令进行预处理，然后将预处理命令的结果和源程序一起再进行通常的编译处理，以得到目标代码。并不是每个C程序都必须有该语句，但是，如果程序有该语句，就必须将它放在程序的开始处，位于任何其他语句之前。

(3) “<stdio.h>”：以.h为扩展名的文件被称为头文件，文件名写在“<”和“>”之间或一对“”之间，它可以是C程序中现成的标准库文件，也可以是自定义的库文件。标准库文件定义任何程序内可以使用的函数，使得开发人员可以更轻松地执行日常任务。stdio.h文件中包含了有关输入输出语句的函数。

(4) “void main(){……}”：是程序的主体。main()表示主函数，main是它的函数名。由一组大括号{}括起来的是函数体。

(5) “printf(“Hello World!\n”);”：在屏幕上产生一行输出“Hello World!”，并换行(\n)。函数主体中的每个语句都以分号结束。

1.1.3 C程序的结构

C程序的一般结构可以用下面的示意图1-1表示。

位于开头的注释区是对整个程序的说明，其他各部分中都可以包含注释，对程序局部内容加以说明。

在程序区之前是声明区，它通常包含若干个行预处理命令，以“#include”开始的预处理命令是最常用的。

程序区中包含着函数，一个程序中一定有且仅有一个主函数，另外根据需要，还可以有若干个其他函数。每个函数都有函数首部和函数体，有自己的函数名。主函数的函数名一定是main。

函数体以“{”开始，以“}”结束，其中包含变量声明和若干语句（执行部分）。一个

语句可以按一定规则分成多行，而一行也可以写多个语句，每个语句以分号结束。

名称				
注释区			/* 源文件名: hello.c 功能: 显示 Hello World! */	
声明区			#include <stdio.h>	
程序区	主函数	函数首部		void main()
		函数体	{	{
			声明部分	
			执行部分	printf(“Hello World!\n”);
	其他函数		}	}

图 1-1 C 语言程序的结构

在 C 语言中，同一字母的大小写被当作两个不同字母，例如 A 和 a 不能混为一谈。

程序编写好后，需要以文件形式保存在磁盘上，以便长期保存和修改。以“.c”为文件名后缀的文件就保存着 C 程序。用任何文本编辑工具打开这种文件，都可以查看、修改程序的内容。上面程序注释区中的“源文件名: hello.c”就是说明文件名的。

1.2 创建和运行一个 C 语言程序

我们以上一节介绍的程序为例，动手经历一下 C 程序设计的实际过程。众所周知，支持 C 语言的编译器有很多种，如 Turbo C、Visual C++、Borland C 等。目前，Visual C++ 已是被广泛应用的程序开发平台，越来越多的学校采用它作为 C/C++ 程序设计课程的教学平台。根据这一实际需求，下面的步骤是以 Microsoft Visual C++ 6.0（简称 VC++）为软件环境进行操作。

在此过程中，Visual C++ 将把我们的程序称为“工程”。每个工程会建立一个文件夹，工程中包含一组文件，这组文件可以具有不同的扩展名，其中部分文件由 Visual C++ 自动创建，这组文件组合在一起形成一个完整的应用程序。创建工程涉及的过程和步骤如下：

(1) 打开 Microsoft Visual C++ 工作界面，如图 1-2 所示。

(2) 使用 Microsoft Visual C++ 不仅可以创建控制台应用程序，也可以创建 Windows 应用程序，在此选择创建一个控制台应用程序。选择“文件”→“新建”→选择“工程”选项卡→单击“Win32 Console Application”选项，如图 1-3 所示。

(3) 单击“确定”按钮，显示对话框如图 1-4 所示。

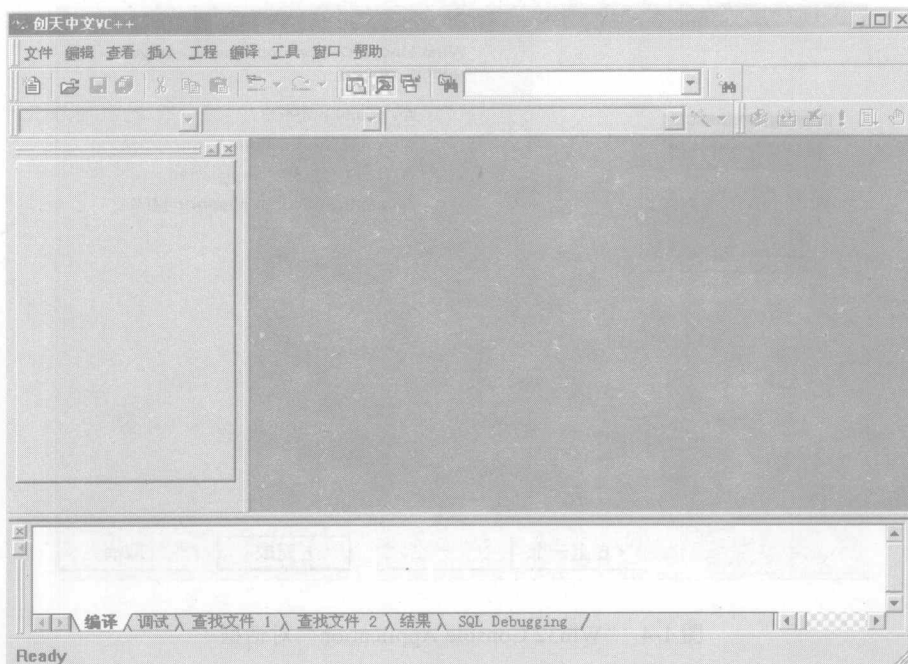


图 1-2 Microsoft Visual C++界面

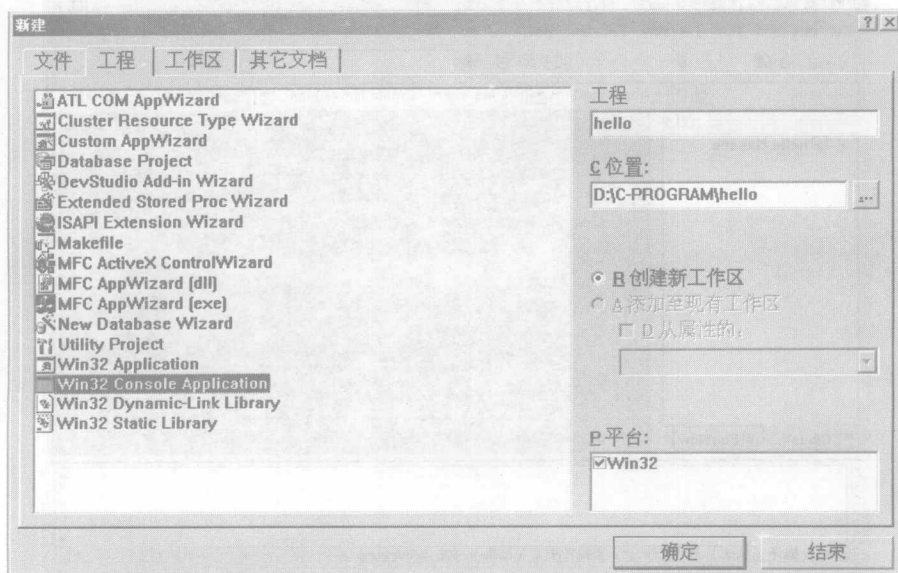


图 1-3 “工程”选项卡

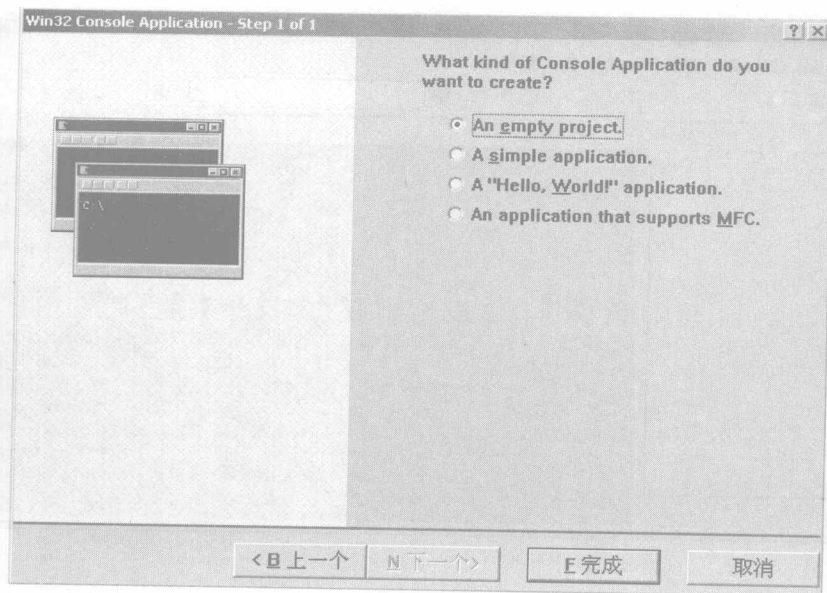


图 1-4 “Win32 Console Application”对话框

(4) 选中“An empty project”选项后，单击“完成”按钮，在弹出的“新建工程信息”对话框中，单击“确定”按钮，出现如图 1-5 所示的窗口。

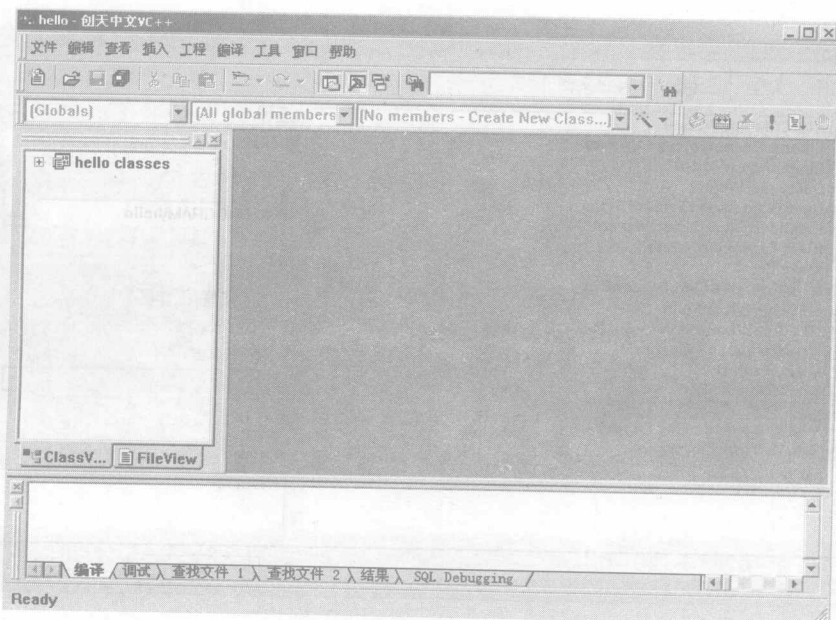


图 1-5 Visual C++ IDE

示例(5) 选择命令“工程”→“添加工程”→“新建”，出现如图 1-6 所示的对话框。

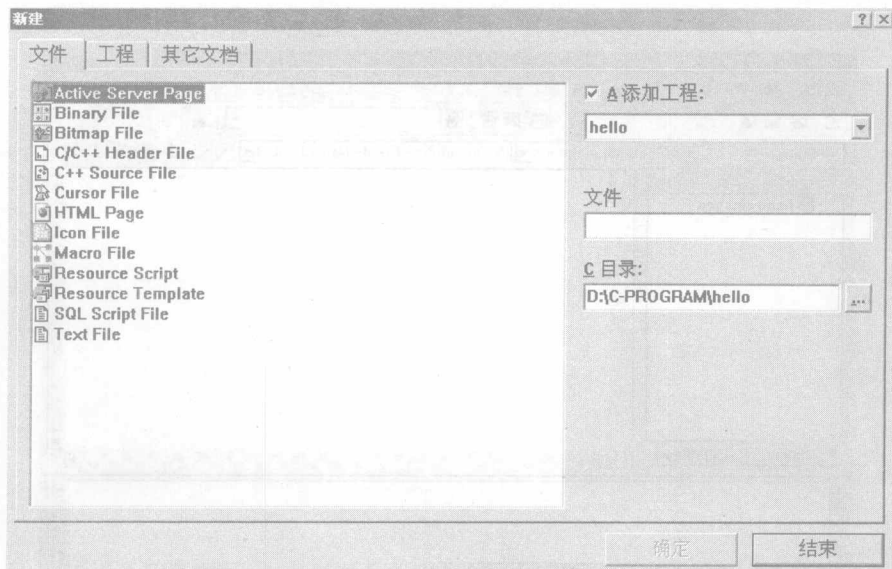


图 1-6 “新建”对话框

(6) 在“文件”选项卡下，选择“C++ Source File”选项，在“文件”文本框中输入“hello.c”，如图 1-7 所示。

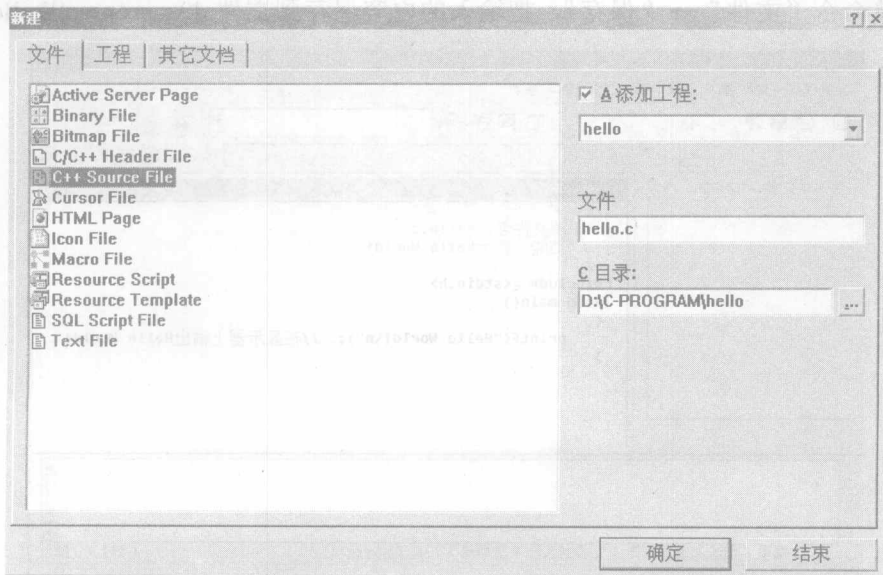


图 1-7 “文件”选项卡

(7) 单击“确定”按钮，出现如图 1-8 所示的窗口，右边有字符输入光标闪烁，提示输入程序。

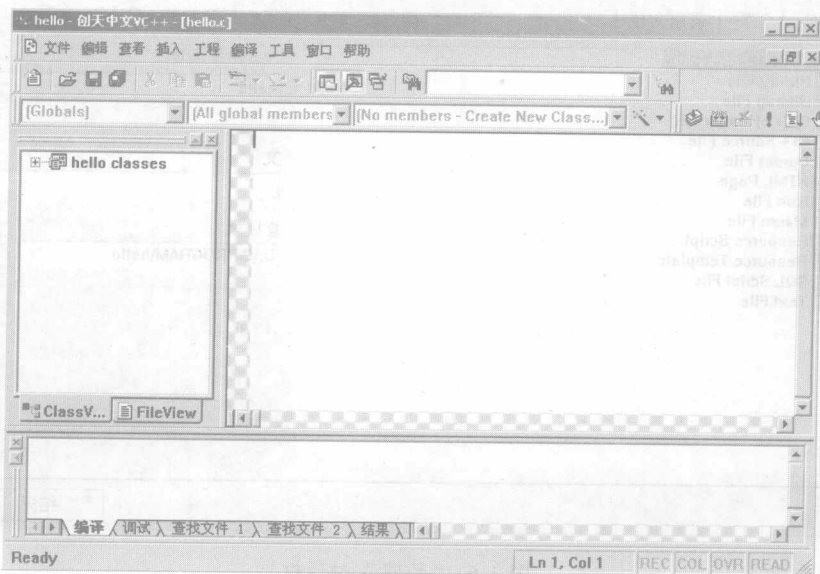


图 1-8 编辑模式下的 Visual C++ IDE

(8) 输入程序的全部内容，如图 1-9 所示，在输入的时候不要输入中文标点符号。然后选择菜单命令“文件”→“保存”，把输入的内容保存到文件 d:\c-program\hello.c。

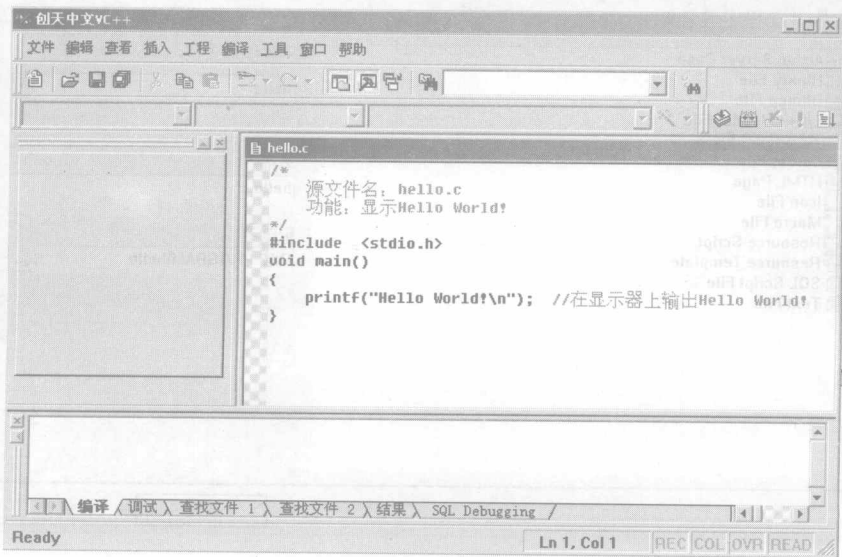


图 1-9 输入 hello 程序

(9) 选择命令“编译”→“编译 hello.c”，结果如图 1-10 所示。窗口下部的显示框内最后一行说明在程序中发现了多少错误。如果不是“0 error(s), 0 warning(s)”，则要检查输入的程序，纠正错误，再重复此步骤，直到没有错误为止。修改完成后，请注意按照步骤(8)保存修改后的程序。

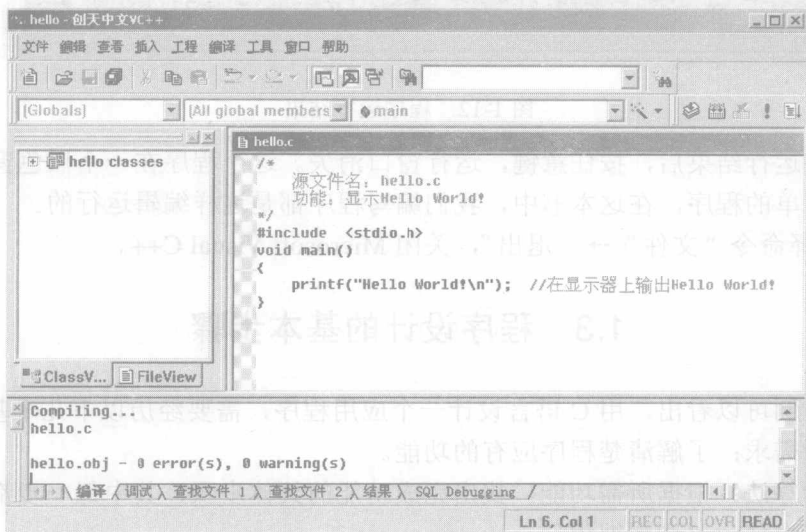


图 1-10 编译 C 源代码

(10) 选择命令“编译”→“构件 hello.exe”，结果如图 1-11 所示。

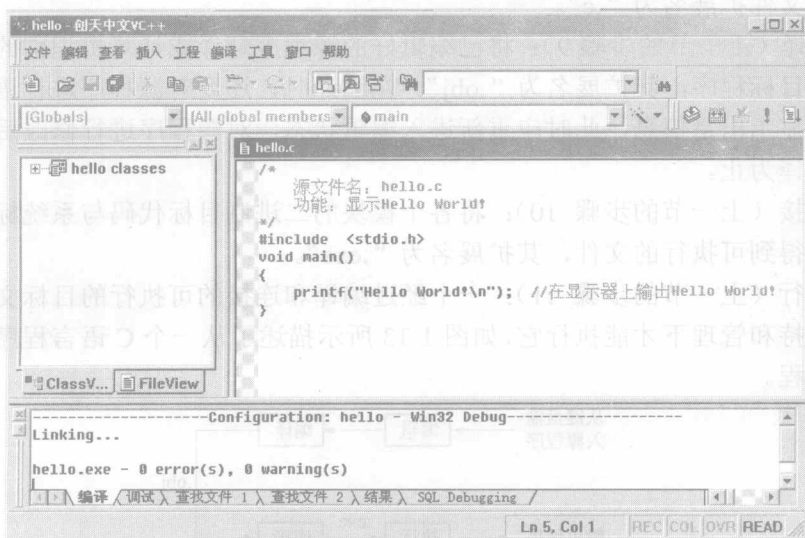


图 1-11 通过连接生成的可执行程序

(11) 选择命令“执行 hello.exe”，结果如图 1-12 所示。

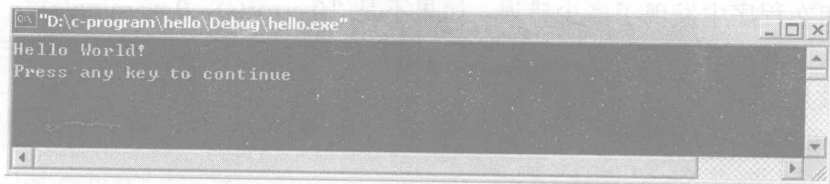


图 1-12 程序运行结果

观察程序运行结果后，按任意键，运行窗口消失。这个程序在一个黑色窗口中运行，这是一种最简单的程序，在这本书中，我们编写程序都是这样编辑运行的。

(12) 选择命令“文件”→“退出”，关闭 Microsoft Visual C++。

1.3 程序设计的基本步骤

从以上举例可以看出，用 C 语言设计一个应用程序，需要经历以下几个基本步骤：

- (1) 分析需求：了解清楚程序应有的功能。
- (2) 设计算法：根据所需功能，找出完成功能的具体步骤，其中每一步都应当是简单的、确定的。也称为“逻辑编程”。
- (3) 编写程序：根据上一步骤设计的算法，编写符合 C 语言语法规则的源程序。
- (4) 输入与编辑程序（上一节的步骤 8）：将源程序逐个字符输入到计算机内存，并保存为文件，文件扩展名为“.c”。
- (5) 编译（上一节的步骤 9）：将已编辑好的源程序翻译成计算机识别的二进制代码文件，也称为目标程序，其扩展名为“.obj”。在编译时，还要对源程序进行语法检查，如发现错误，则显示出错信息，此时应重新进入编辑状态，对源程序进行修改后再重新编译，直到通过编译为止。
- (6) 连接（上一节的步骤 10）：将各个模块的二进制目标代码与系统标准模块经过连接处理后，得到可执行的文件，其扩展名为“.exe”。
- (7) 执行（上一节的步骤 11）：一个经过编译和连接的可执行的目标文件，只有在操作系统的支持和管理下才能执行它，如图 1-13 所示描述了一个 C 语言程序到生成可执行文件的全过程。

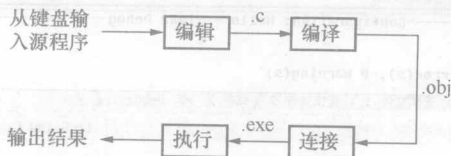


图 1-13 C 程序实现过程示意图