



21st CENTURY
规划教材

面向21世纪高等院校计算机系列规划教材
COMPUTER COURSES FOR UNDERGRADUATE EDUCATION

C语言程序设计实验与习题指导

汪志勤 主 编
张若冰 魏 文 虞 戟 副主编



科学出版社
www.sciencep.com



面向21世纪高等院校计算机系列规划教材
COMPUTER COURSES FOR UNDERGRADUATE EDUCATION

C 语言程序设计 实验与习题指导

汪志勤 主编

张若冰 魏文 虞戟 副主编

徐军 主审

科学出版社

北京

内 容 简 介

本书是《C 语言程序设计》配套的实验教材,内容包括 C 语言程序设计基础、数据类型、运算符和表达式、控制结构与语句、数组、函数、指针和结构体、共用体及枚举类型、文件、配套教材课后习题解答等。所有实验均按内容分类。

本书可作为高等院校相关专业以及其他 C 语言课程的实验教材,也可作为自学 C 程序设计的实验参考书或培训用的实验教材。

图书在版编目(CIP)数据

C 语言程序设计实验与习题指导/汪志勤主编. —北京:科学出版社,2006
(面向 21 世纪高等院校计算机系列规划教材)

ISBN 7-03-016793-7

I. C… II. 汪… III. C 语言-程序设计-高等学校-教学参考资料
IV. TP312

中国版本图书馆 CIP 数据核字(2006)第 003742 号

责任编辑:韩 洁/ 责任校对:刘彦妮

责任印制:吕春珉/ 封面设计:飞天创意

科学出版社出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

铭浩彩色印装有限公司 印刷

科学出版社发行 各地新华书店经销

*

2006 年 2 月第 一 版 开本:787×1092 1/16

2006 年 2 月第一次印刷 印张:10 1/4

印数:1—5 200 字数:234 000

定价:16.00 元

(如有印装质量问题,我社负责调换<环伟>)

销售部电话 010-62136131 编辑部电话 010-62138978-8203 (HP04)

前 言

C 语言程序设计是一门实践性非常强的课程，上机实践是学好该课程十分重要的环节。为了便于学习、掌握该课程、培养学生的编程能力，我们组织了中国计量学院计算机基础教育中心的一线任课教师共同编写了这本书。

本书以科学出版社出版的《C 语言程序设计》为主教材，与之配套。按照主教材的章节，并结合实验教学，为每一章设计了“实验目的”、“实验内容”、“实验指导”和“思考题”四部分内容。

“实验内容”是留给读者自己动手上机完成的练习题。每个实验的题量较多，如果作为学生上机练习，教师可根据学生情况，每次实验（2 学时）选做 2~3 题，其余可留作学生课外作业或上机练习。

“实验指导”由精心设计的多个具有较强针对性和实践性的实例组成，每一个实例都列出了切中要点的提示、源代码、程序分析，力求给读者一个操作示范，同时针对一些要求重点掌握或容易出错的问题提出“思考与讨论”，使读者通过这些实例的上机实践，起到举一反三的作用，加深对实验内容的理解和掌握，培养读者的实际动手能力。

本书中的所有实例、实验题参考程序都已在 Visual C++ 6.0 企业版中调试通过。由于解决一个问题通常采用不同方法、编写出多个程序，本教程给出的程序只是其中的一个，仅起到抛砖引玉的作用，读者完全可以编写出不同的、甚至更好的程序。

本书包含有主教材各章节习题与参考答案。

本书由长期工作在教学第一线并具有丰富计算机基础教学经验的多位教师共同编写完成。参加本书编写工作的有中国计量学院信息工程学院的汪志勤、张若冰、魏文、徐军、刘砚秋、郑文斌、计量技术工程学院的章皓。主教材的课后习题解答由汪志勤老师和虞戟老师负责解答和整理。全书由汪志勤担任主编并统稿。

本书可以作为各类高等院校非计算机专业学生的 C 语言程序设计课程的实验教学用书，也可以作为自学者学习 C 语言程序设计语言的参考书。

由于作者水平有限，加之时间仓促，书中定有一些不当或错误，谨请广大读者批评指正。

目 录

第一部分 上机实验指导	1
实验一 C 语言上机简介.....	1
实验二 数据类型.....	12
实验三 运算符和表达式.....	18
实验四 控制结构与语句.....	24
实验五 数组.....	30
实验六 函数.....	38
实验七 指针.....	45
实验八 结构体、共用体、枚举类型.....	52
实验九 文件.....	62
第二部分 习题及答案	70
习题一.....	70
习题二.....	73
习题三.....	76
习题四.....	82
习题五.....	94
习题六.....	109
习题七.....	121
习题八.....	123
习题九.....	133
习题十.....	141
习题十一.....	143
习题十二.....	149
主要参考文献	157



第一部分 上机实验指导

实验一 C 语言上机简介

C 语言是目前使用最广泛的高级程序设计语言。C 语言功能丰富、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，既有高级语言的优点，又具有低级语言的许多特点。Turbo C 是美国 Borland 公司研制生产的，是一个集程序编辑、编译、连接、调试为一体的 C 语言程序开发软件，它具有速度快、效率高等优点，是编制 C 语言程序的工具。

Visual C++ 6.0 是 Microsoft 公司推出的目前使用极为广泛的基于 Windows 平台的可视化编程环境，这对于习惯图形界面的用户来说是比较易学的。由于 Visual C++ 6.0 功能强大、灵活性好，以及具有强有力的 Internet 支持，成为目前最流行的 C++ 语言集成开发环境。

利用 Visual C++ 6.0 集成开发环境，也可以有效地实现 C 语言程序的编写及运行。本书中所有实验都是以 Visual C++ 6.0 作为开发平台。

一、实验目的

- 1) 熟悉开发工具 Visual C++ 6.0 的使用。
- 2) 学习编写简单的 C 语言源程序。
- 3) 掌握 C 语言程序的编辑、编译和运行过程。
- 4) 掌握 C 语言程序的单步跟踪调试方法以及常用的编译方法。

二、实验内容

- 1) 启动 Visual C++ 6.0，熟悉窗口组成。
- 2) 编写一个简单的程序。在屏幕上打印输出如下：

```
Hello! World
```

"This is my first program!"

学会如何新建、编辑、编译及运行 C 程序。

3) 编写一个简单的程序。输入两个数 a、b 的值， $s=a+b$ 。打印输出 $s=2*s$ 的值。单步跟踪 s 当前值的变化，掌握程序的单步跟踪调试方法。

三、实验指导

实例 1 启动 Visual C++ 6.0，熟悉窗口组成。

(1) 启动

选择任务栏中的“开始”、“程序”、“Microsoft Visual Studio 6.0”、“Microsoft Visual C++ 6.0”，如图 1.1 所示，就可以启动如图 1.2 所示的 Visual C++ 6.0 集成开发环境。

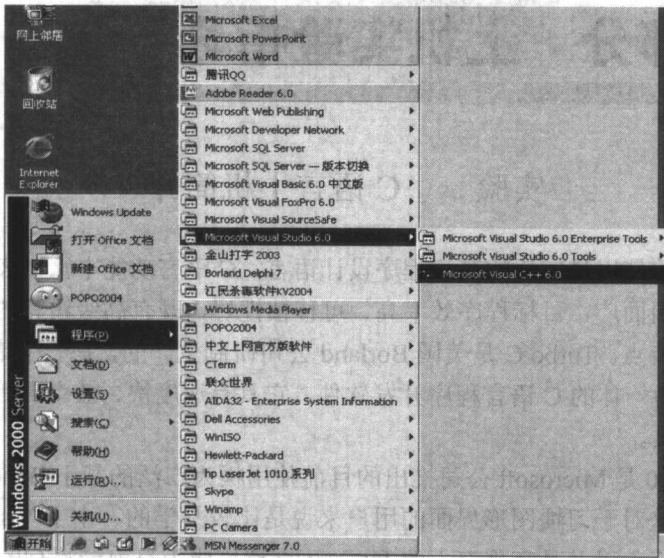


图 1.1 启动 Visual C++ 6.0

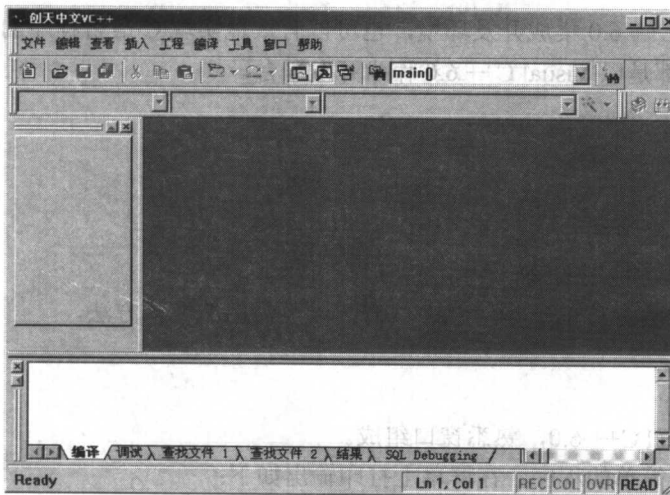


图 1.2 Visual C++ 6.0 开发环境

(2) 窗口组成

窗口主要组成如图 1.3 所示。

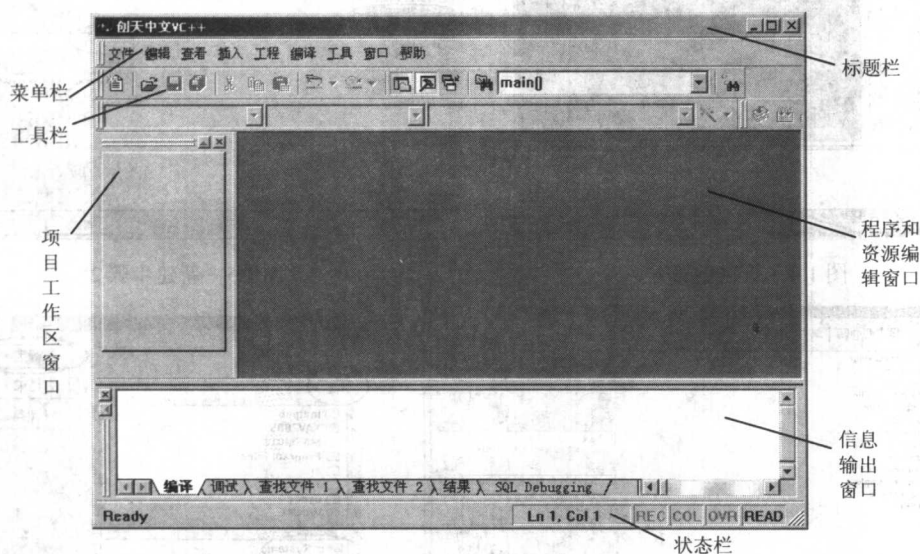


图 1.3 Visual C++ 6.0 开发环境窗口组成

- 1) 程序和资源窗口：设计及处理源程序代码及项目资源的空间。
- 2) 信息输出窗口：显示编译及调试的结果，方便用户修改程序的错误。提示错误信息包括错误的条数、错误处所在位置和错误的原因等。
- 3) 项目工作区窗口：主要包含项目文件，资源等基本信息。
- 4) 状态栏：能显示当前操作状态、光标所在的行列号等信息。

实例 2 编写一个简单的在屏幕上打印输出“Hello world”。

(1) 新建文件

根据上面的图 1.1 的操作步骤，打开如图 1.2 所示的 Visual C++ 6.0 的操作界面后，选择主菜单中的“文件”、“新建”，如图 1.4 所示。在新建对话框中选择“文件”选项卡。在下面列出的选项中选择“C++ Source File”，在右边相应的“文件”对话框中，输入文件名，如“hello.c”，如图 1.5 所示。然后，在 C 目录对话框中选择文件的保存位置。单击如图 1.6 所示的圈中的按钮，会弹出一个如图 1.7 所示的对话框，选择要存放的路径，单击“确定”按钮，出现如图 1.8 所示的界面。

对比图 1.6 可以看出，此时文件名为 hello.c，文件保存的位置在 C 盘下面。完成上述的操作后单击如图 1.8 所示的“确定”按钮，就新建了一个文件，即可进行编辑，如图 1.9 所示。

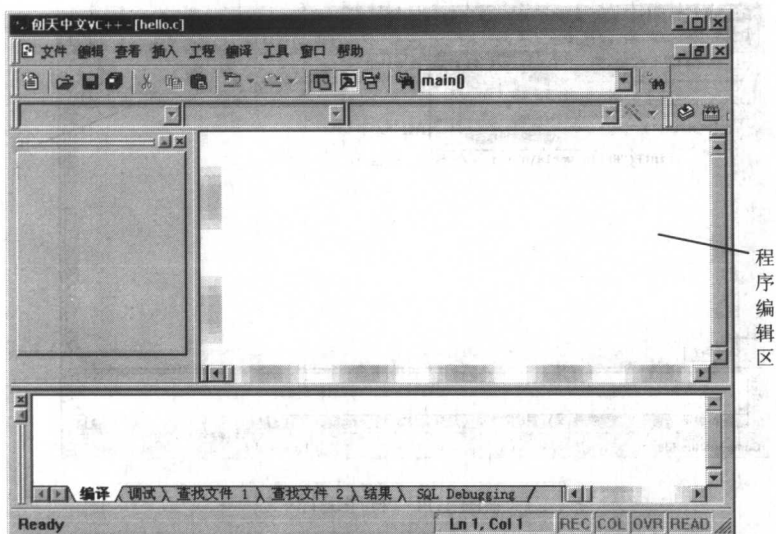



图 1.9 程序编辑窗口

(3) 程序的编译

接下来我们将上面的 C 语言代码编译成计算机能够执行的目标代码。

选择“编译”菜单下的“编译”选项，如图 1.11 所示，或者是单击工具栏上的按钮（图 1.11 圈中部分）或按快捷键 Ctrl+F7。如果源程序第一次编译时，会弹出如图 1.12 所示的对话框，此时单击“是”按钮，开始编译。

注意：选择图 1.12 中的“是”按钮，Visual C++ 6.0 集成开发环境会自动在 hello.c 文件所在的文件夹中创建相应的项目文件。

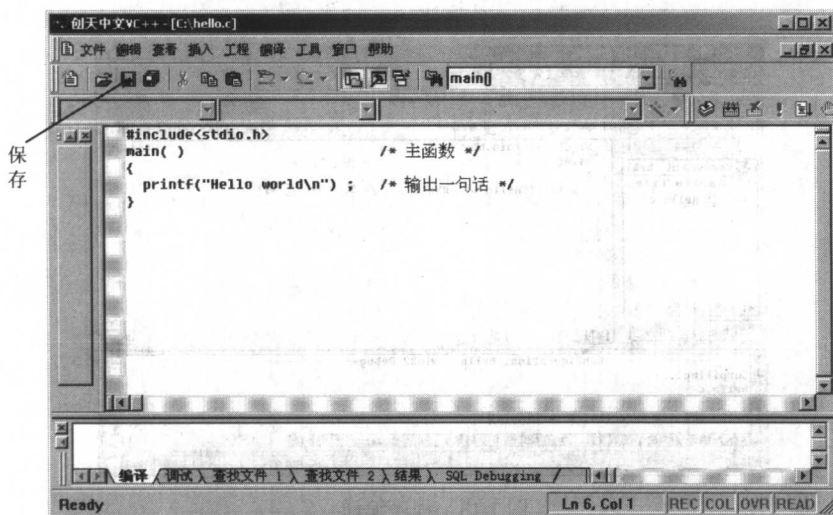


图 1.10 程序编辑窗口

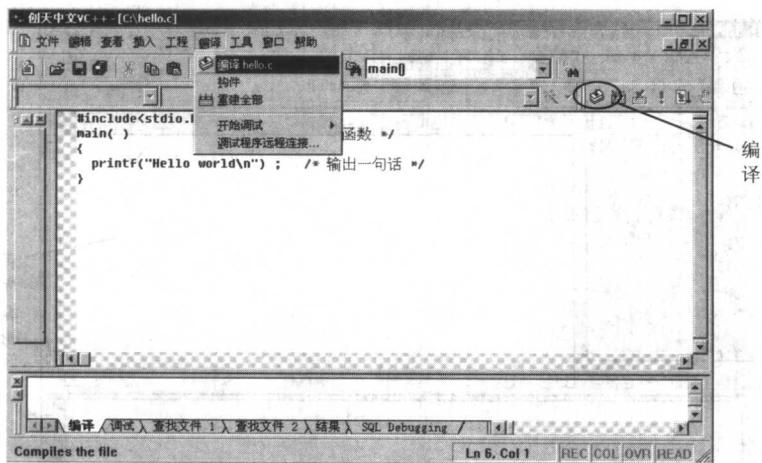


图 1.11 程序编译

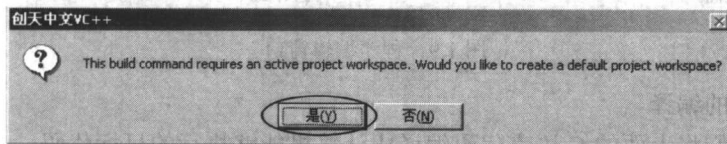


图 1.12 程序编译

编译时，下方的输出框中将显示相应的编译说明，如果源程序没有词法和语法错误，那么窗口将显示：

```
hello.obj - 0 error(s), 0 warning(s)
```

说明编译没有错误（error）和警告（warning），生成目标文件 `hello.obj`，如图 1.13 所示。

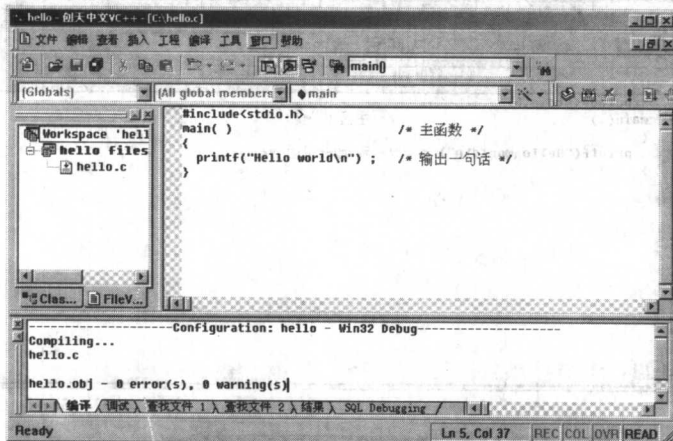
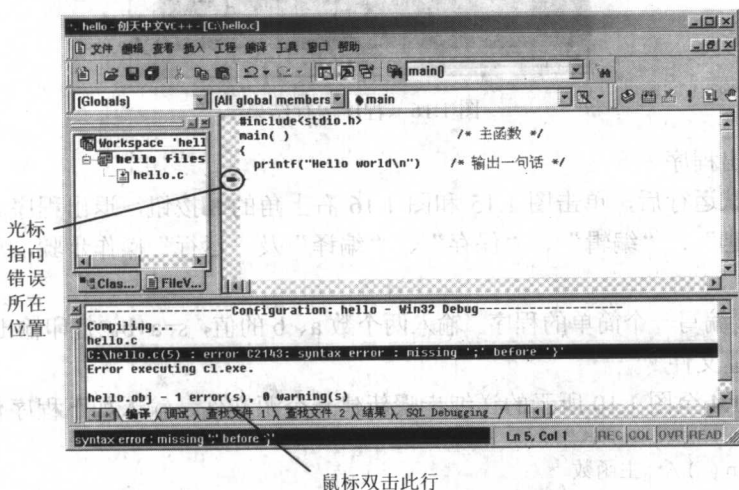


图 1.13 程序编译

如果源程序有词法和语法错误，编译时会得到错误和警告。编译的错误会在下方的输出窗口中显示出来。如上面的源程序，如果把 `printf` 语句后面的“;”去掉，那么编译后会出现如图 1.14 所示的窗口。双击输出窗口中的错误提示行，光标可以自动移动到发

生错误的源程序相应地点，再根据错误提示信息，检查源程序，改正错误。然后重复图 1.11 所示的编译过程，直到编译通过为止。




鼠标双击此行

图 1.14 程序编译出错

(4) 连接与运行

上面的程序编译好后，系统会生成一个 hello.obj 的目标文件。目标文件不能被计算机直接执行，必须将目标文件 (.obj) 和相关的库函数或目标程序连接成一个可执行文件。

选择“编译”菜单下的“执行 hello.exe”，如图 1.15 所示，或者是单击工具栏上的  按钮或按快捷键 Ctrl+F5。生成可执行文件 hello.exe。该文件保存在与 hello.c 同一文件夹下的 Debug 文件夹中。

程序运行以后将显示一个控制台程序窗口，显示程序的输出结果，如图 1.16 所示。注意：图 1.16 中的“Press any key to continue”是系统提示你按任意键关闭，如图 1.16 所示，退回到 Visual C++ 6.0 开发环境中。

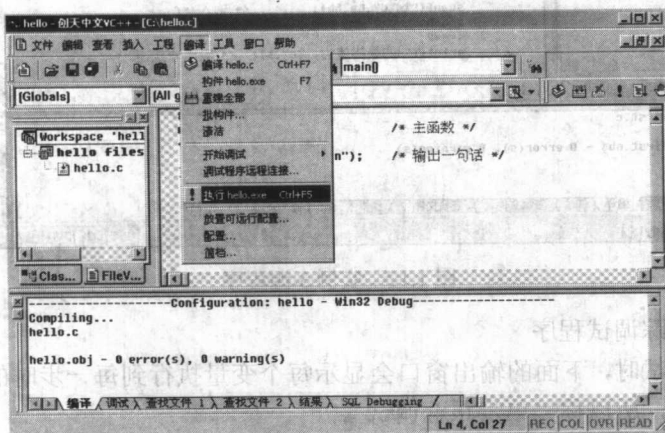


图 1.15 程序运行

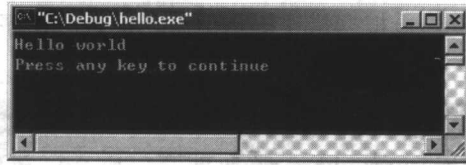



图 1.16 程序运行结果

(5) 退出程序

程序调试运行后,单击图 1.15 和图 1.16 右上角的  按钮,退出程序。然后,重复上面的“新建”、“编辑”、“保存”、“编译”及“运行”操作步骤,重新开始一个新的程序。

实例 3 编写一个简单的程序。输入两个数 a 、 b 的值, $s=a+b$ 。打印输出 $s=2*s$ 的值。

(1) 新建文件

根据图 1.4 至图 1.10 所示的详细步骤新建一个源文件,输入如下程序代码:

```
#include<stdio.h>
main() /* 主函数 */
{
    int a,b,s;
    printf("输入 a,b 两个数\n");
    scanf("%d%d",&a,&b);          /* 输入 */
    s=a+b;
    s=s*2;
    printf("s 的最终值为: %d",s); /* 输出 s 的结果 */
}
```

源文件保存为“first.c”,如图 1.17 所示。

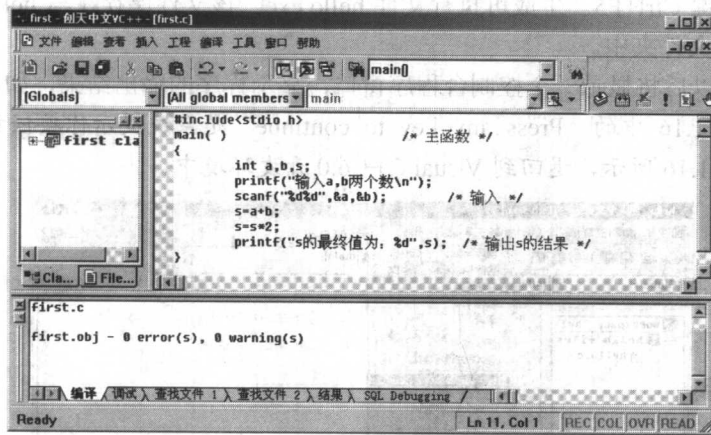


图 1.17 实例 3 源程序

(2) 单步跟踪调试程序

单步跟踪调试时,下面的输出窗口会显示每个变量执行到每一步时的当前值,即实时跟踪变量的值,所以称“单步跟踪调试”。

在“编译”菜单下的“开始调试”子菜单中选择“Step Into”选项,或按快捷键 F11。如图 1.18 所示。

此时，在下面的输出窗口中的“name”处可添加想要跟踪的变量，则单步跟踪时，在“value”处会时时显示变量的当前值。如图 1.19、图 1.20 所示分别为添加跟踪变量前和添加跟踪变量后的情况。

如图 1.20 所示，a、b、s 变量的值显示为“CXX0069: Error: variable needs stack frame”，这是因为程序单步运行时，尚未运行到 scanf 语句，所以 a、b 还没有确定的值，即 s 也无确定的值。

添加好跟踪的变量后，可反复的按快捷键 F10，每按一次 F10 键，程序都执行一步，可以从跟踪标记的移动来观察程序执行的所在处。当程序执行到 scanf 语句时，在控制台程序窗口输入变量 a、b 的值，如给变量 a 输入 5，变量 b 输入 8，如图 1.21 和图 1.22 所示。

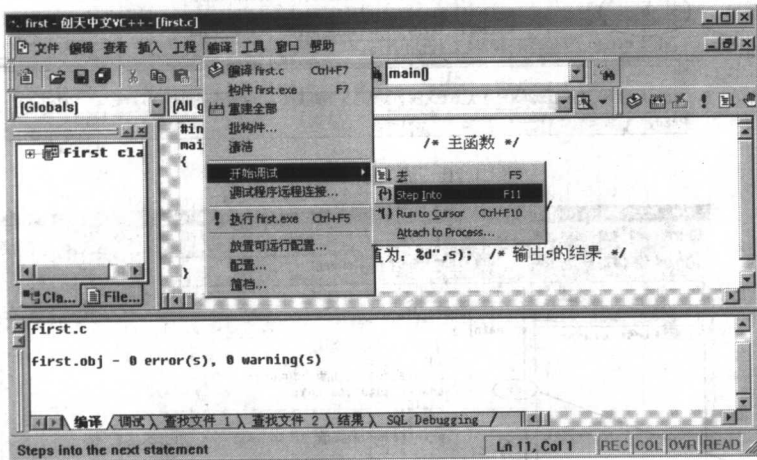


图 1.18 单步跟踪选项

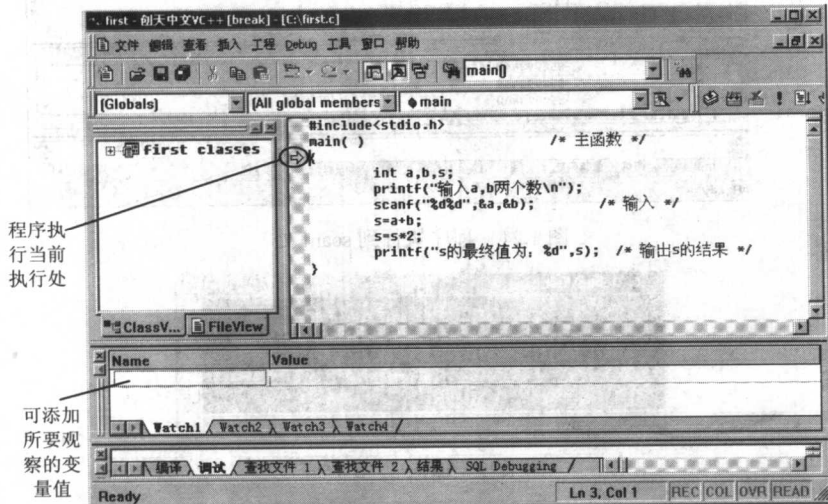


图 1.19 单步跟踪

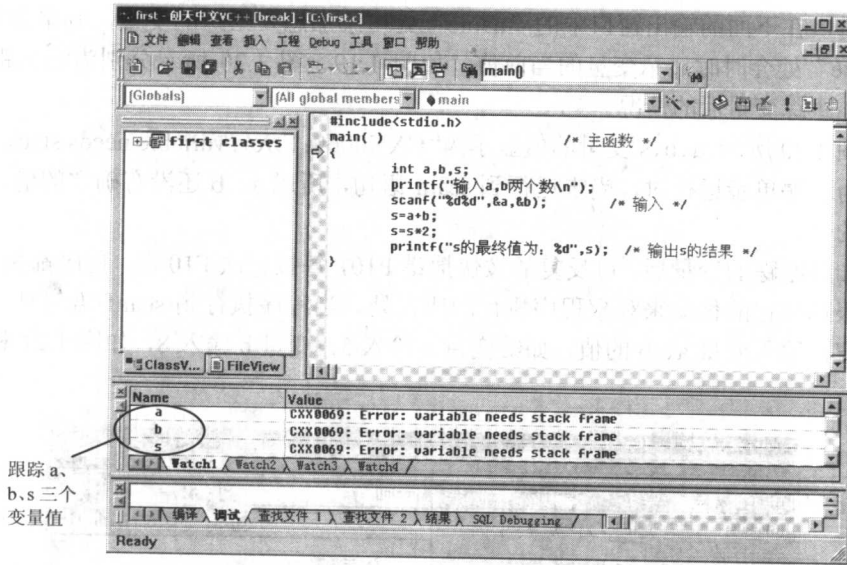


图 1.20 添加跟踪变量值

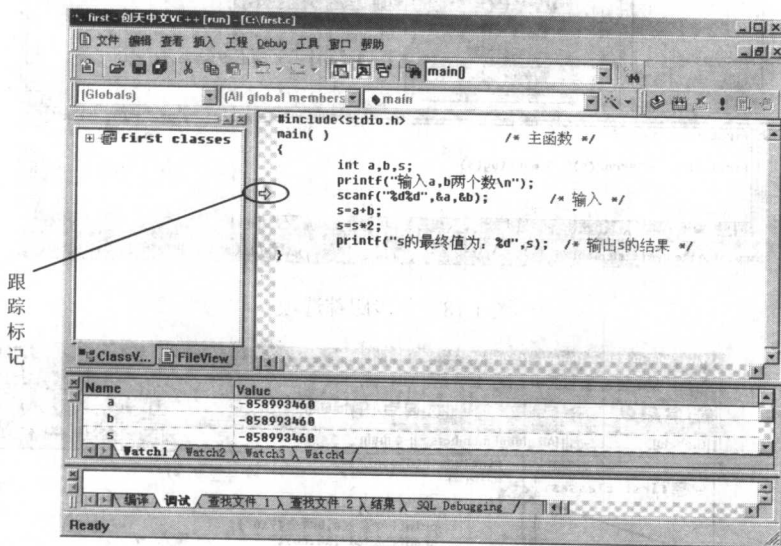


图 1.21 程序执行到 scanf 处

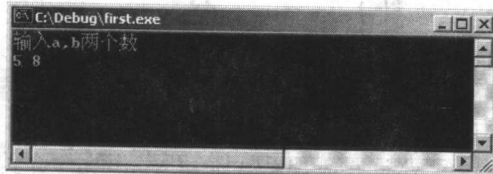


图 1.22 输入 a, b 的值

输入后，回到程序的编译窗口。继续按 F10 键单步跟踪，通过下面的观察窗口，可以观察变量 a、b 和 s 的值，如图 1.23 至图 1.25 所示。直至单步跟踪到程序结束，在控制台程序窗口显示程序的输出结果，如图 1.26 所示。

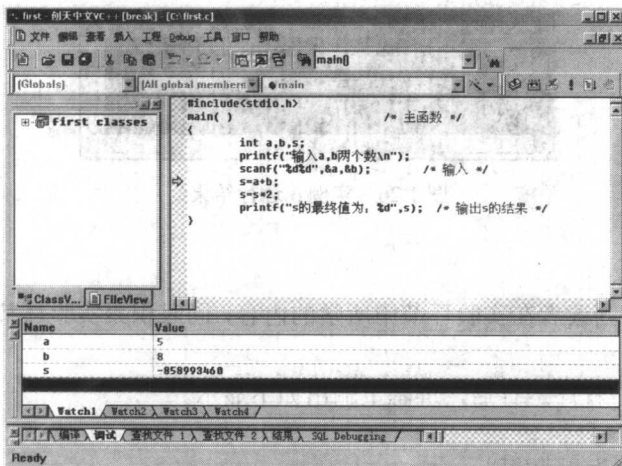


图 1.23 单步跟踪 (1)

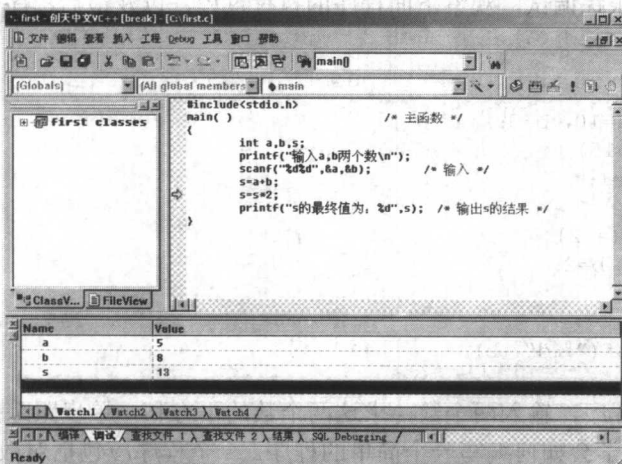


图 1.24 单步跟踪 (2)

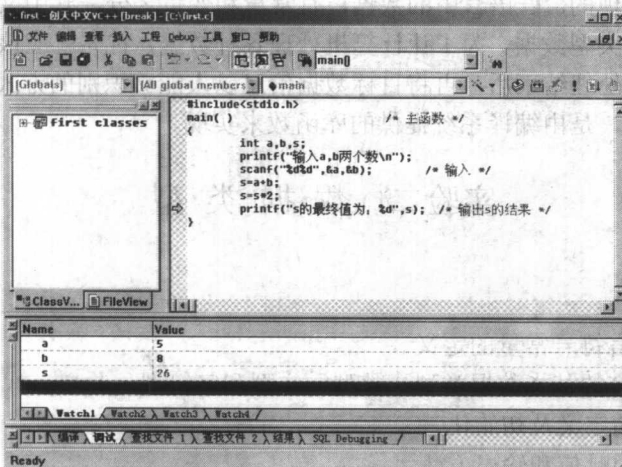


图 1.25 单步跟踪 (3)

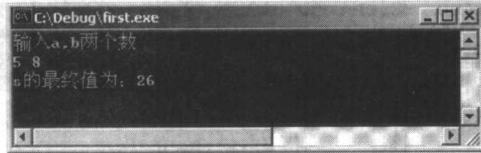


图 1.26 实例 3 运行结果

四、思考题

此环节可以巩固读者在实验环节中所学的内容，同时积累实验环节外进行独立编程的经验。

1. 编写程序，程序运行后，屏幕上输出如下显示。

```
*****
This is a test!
*****
```

2. 通过单步跟踪调试，观察下面程序的执行过程，以及最后程序的输出结果。

```
#include<stdio.h>
main()
{
    int a=10, b=0;
    if(a>15)
        b=1;
    else
        if(a<2)
            b=2;
        else
            b=3;
    printf("%d", b);
}
```

3. 编写简单程序，输入两个数，求这两个数的平方和，单步跟踪并运行程序。

前面我们已经学会如何编写一个简单的程序。一个程序应包括对数据的描述和对数据处理描述。对数据的描述，即数据结构。在 C 语言中，系统提供的数据结构，是以数据类型形式出现的。C 语言中的数据，有常量和变量之分，其中又包括整型、实型和字符型三种基本类型数据。为了让计算机处理各种数据，首先就应该把源数据输入到计算机中；计算机处理结束后，再将目标数据信息以人能够识别的方式输出。C 语言中实现输入输出操作，是由编译系统提供的库函数来实现。

实验二 数据类型

一、实验目的

- 1) 掌握 C 语言符号常量的定义。
- 2) 掌握 C 语言的基本数据类型：整型、实型和字符型。
- 3) 熟悉变量的定义及初始化。
- 4) 熟悉变量的赋值和输出方法。
- 5) 掌握 C 语言输入输出中的格式说明符，熟练使用赋值语句及输入、输出函数。