



21世纪普通高等院校规划教材 • 信息技术类

普通高等教育“十二五”规划教材

C YUYAN CHENGXU SHEJI SHIYAN ZHIDAO

# C语言程序设计 实验指导

主 编 肖志军  
副主编 张茂胜 杨夏妮  
闭吕庆 熊春荣

TP312C-33  
29



西南交通大学出版社  
[Http://press.swjtu.edu.cn](http://press.swjtu.edu.cn)

013034765

TP312C-33  
29

21 世纪普通高等院校规划教材——信息技术类

普通高等教育“十二五”规划教材

## C 语言程序设计实验指导

主 编 肖志军

副主编 张茂胜 杨夏妮 闭吕庆 熊春荣



TP 312C-33  
29

西南交通大学出版社

· 成 都 ·



北航

C1642081

## 内容简介

本书是为主教材《C 语言程序设计》编写的配套实验指导书。

全书由 11 个实验组成, 每个实验都包括了调试示例题、改错题和编程题。通过介绍分散在各个实验中的调试示例题的常用的程序调试方法来逐步熟悉编程环境, 掌握基本的程序调试方法。改错题能加深对易混淆知识点的理解和掌握。读者学习时可以先模仿示例操作, 然后再做改错题和编程题, 通过对调试示例题的“模仿”、改错题的“修改”和编程题的“编写”等上机实践训练, 可在循序渐进的引导中逐步熟悉编程环境, 理解和掌握程序设计的思想、方法和技巧, 以及程序调试方法。

---

### 图书在版编目 ( C I P ) 数据

C 语言程序设计实验指导 / 肖志军主编. —成都:  
西南交通大学出版社, 2013.3

21 世纪普通高等院校规划教材. 信息技术类 普通高  
等教育“十二五”规划教材

ISBN 978-7-5643-2235-9

I. ①C… II. ①肖… III. ①

C 语言—程序设计—高等学校—教学参考资料 IV.

①TP312

---

中国版本图书馆 CIP 数据核字 (2013) 第 042053 号

---

21 世纪普通高等院校规划教材——信息技术类

普通高等教育“十二五”规划教材

C 语言程序设计实验指导

主编 肖志军

---

|       |                                                                   |
|-------|-------------------------------------------------------------------|
| 责任编辑  | 黄淑文                                                               |
| 特邀编辑  | 黄庆斌                                                               |
| 封面设计  | 本格设计                                                              |
| 出版发行  | 西南交通大学出版社<br>(成都二环路北一段 111 号)                                     |
| 发行部电话 | 028-87600564 87600533                                             |
| 邮政编码  | 610031                                                            |
| 网 址   | <a href="http://press.swjtu.edu.cn">http://press.swjtu.edu.cn</a> |
| 印 刷   | 成都中铁二局永经堂印务有限责任公司                                                 |
| 成品尺寸  | 185 mm × 260 mm                                                   |
| 印 张   | 5.875                                                             |
| 字 数   | 147 千字                                                            |
| 版 次   | 2013 年 3 月第 1 版                                                   |
| 印 次   | 2013 年 3 月第 1 次                                                   |
| 书 号   | ISBN 978-7-5643-2235-9                                            |
| 定 价   | 16.00 元                                                           |

图书如有印装质量问题 本社负责退换

版权所有 盗版必究 举报电话: 028-87600562

# 目 录

|                        |    |
|------------------------|----|
| 实验 1 熟悉 C 语言编程环境 ..... | 1  |
| 实验 2 简单程序设计 .....      | 10 |
| 实验 3 分支结构程序设计 .....    | 17 |
| 实验 4 循环结构程序设计 .....    | 25 |
| 实验 5 函数程序设计 .....      | 33 |
| 实验 6 综合程序设计实例 .....    | 41 |
| 实验 7 数组程序设计 .....      | 47 |
| 实验 8 指针程序设计 .....      | 60 |
| 实验 9 结构体程序设计 .....     | 73 |
| 实验 10 函数与程序结构 .....    | 82 |
| 实验 11 文件操作 .....       | 87 |
| 参考文献 .....             | 90 |

## 实验 1 熟悉 C 语言编程环境

### 【实验目的】

- (1) 熟悉 C 语言编程环境 Microsoft Visual C++ 6.0，掌握运行（编辑、编译、连接和运行）一个 C 程序的基本步骤。
- (2) 了解 C 程序的基本框架，能参考教材例题编写简单的 C 程序。
- (3) 理解程序调试的思想，能找出并改正 C 程序中的语法错误。

### 【实验内容】

#### 1. 建立一个文件夹，存放 C 程序

在磁盘上新建一个文件夹，用于存放 C 程序，如 D:\C 程序。

#### 2. 编程示例

编写一个输出“Welcome to You!”的程序。

源程序：

```
#include <stdio.h>
void main()
{
    printf("Welcome to You!\n");
}
```

下面就是在 Microsoft Visual C++ 6.0 的编程环境下，以上述 C 语言源程序为例，介绍从新建到运行一个 C 程序的基本步骤。

#### 3. 新建并运行一个 C 程序

第一步，启动 Microsoft Visual C++ 6.0 (VC++)。

启动 VC++，可以通过“开始”菜单，也可以通过桌面快捷方式。启动之后，将看到的是空白的 VC++ 开发环境，如图 1.1 所示。

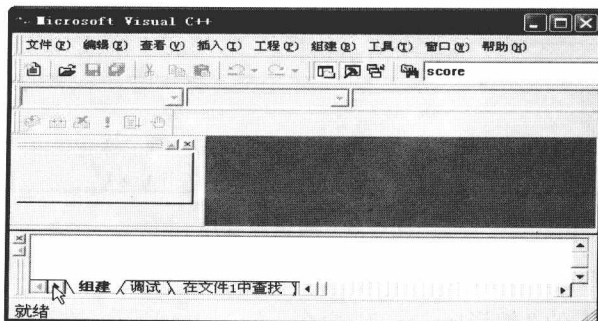


图 1.1 VC++ 窗口

第二步，新建一个文件。

选择“文件”菜单下的“新建”命令，在弹出的“新建”对话框中选择“文件”选项卡，并做如图 1.2 所示的设置。

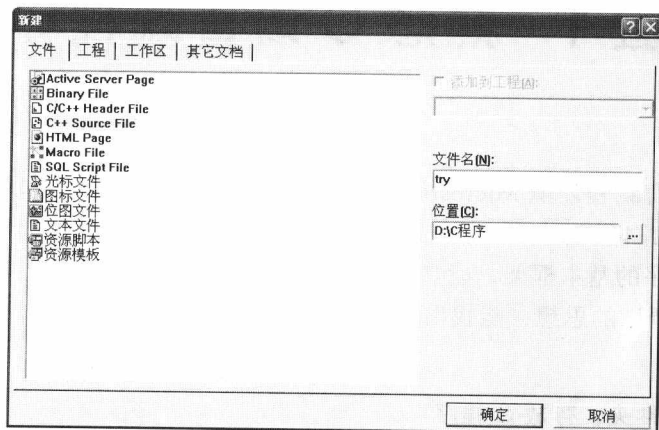


图 1.2 新建文件

在“文件”选项卡中，进行如下操作：

- (1) 选择“C++ Source File”文件类型；
- (2) 输入文件名 try，默认创建的是 .cpp 文件。

(3) 程序保存在 D 盘根目录下的“C 程序”文件夹。该选项不需要手工输入，可以单击在“位置”右下方的  (choose directory) 按钮，来选择已经建立好的文件夹。

第三步，编辑和保存源程序。

当 try.cpp 源文件创建后，就会出现空白的程序编辑窗口，即可编辑 C 程序源代码。在程序编辑窗口中输入源程序，如图 1.3 所示。然后执行“文件”菜单下的“保存”命令，保存源文件。

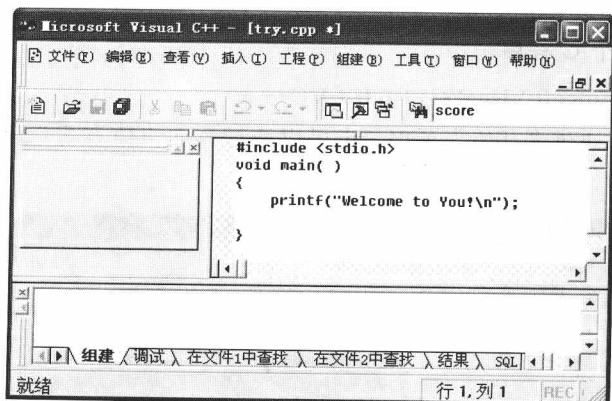


图 1.3 编辑源程序

第四步，编译源程序。

执行“组建”菜单下的“编译[try.cpp]”命令，如图 1.4 所示，在弹出的消息框中单击“是”按钮，如图 1.5 所示，开始编译，并在信息窗口中显示编译信息，如图 1.6 所示。

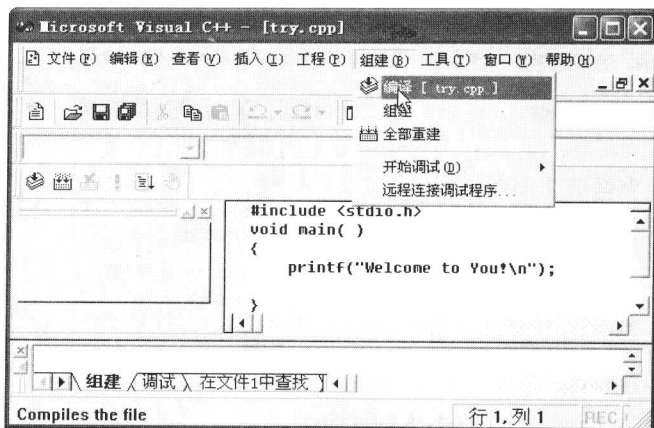


图 1.4 编译源程序

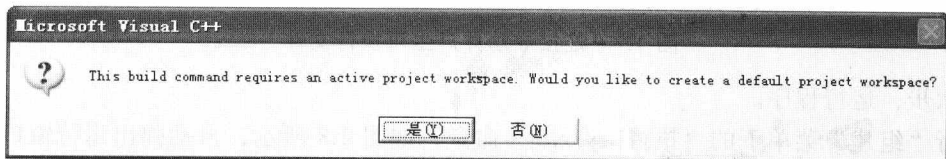


图 1.5 创建工作空间消息框

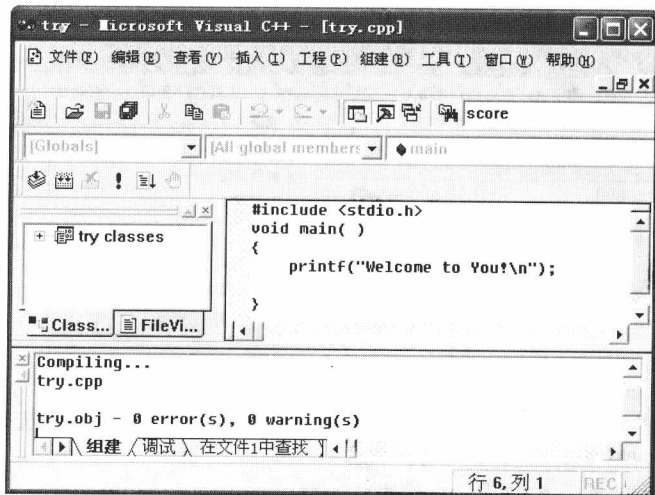


图 1.6 编译完全成功

在图 1.6 中，如果编译完全成功，会显示“try.obj - 0 error(s), 0 warning(s)”，表示没有发现错误和警告，并生成了目标文件 try.obj。如果显示错误信息，说明程序中存在错误，必须改正后重新编译；如果显示警告信息，说明这些警告并不影响目标文件的生成，但一般来说，也应该改正。

第五步，连接。

执行“组建”菜单下的“组建[try.exe]”命令，开始连接，并在信息窗口中显示连接信息，如图 1.7 所示。信息窗口中出现的“try.exe - 0 error(s), 0 warning(s)”表示连接成功，并生成了可执行文件 try.exe。

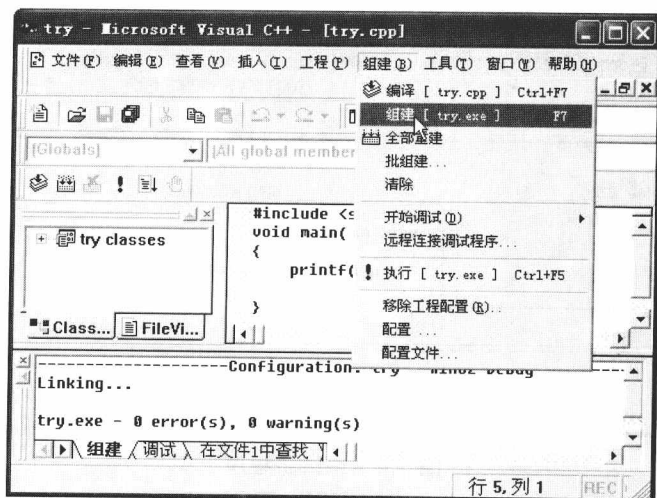


图 1.7 连接成功并产生可执行文件

第六步，运行程序。

执行“组建”菜单下的“执行[try.exe]”命令，如图 1.8 所示，自动弹出运行窗口，如图 1.9 所示，显示运行结果“Welcome to You!”。其中“Press any key to continue”提示用户按任意键退出运行窗口，返回到 VC++ 编辑窗口。

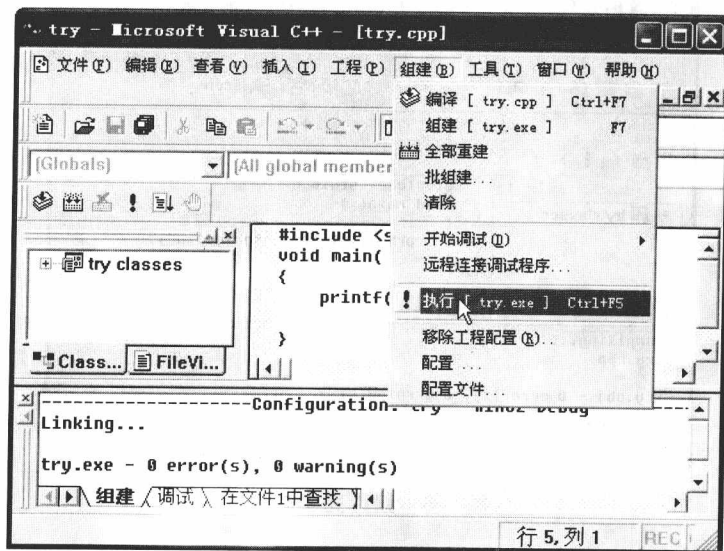


图 1.8 运行程序

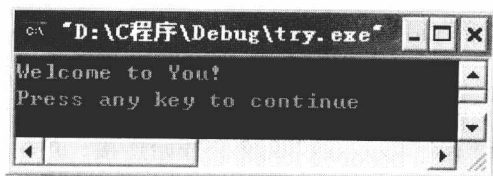


图 1.9 显示运行结果



如果该程序在上一次编译后又被修改, 将会弹出如图 1.10 所示的消息框, 问是否要把最新的代码重新编译。选择“是”, 随后该程序就会被重新编译、连接, 再运行。

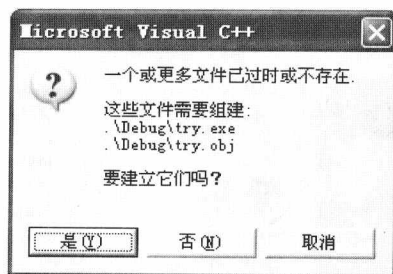


图 1.10 重新编译消息框

第七步, 关闭程序工作区。执行“文件”菜单下的“关闭工作空间”命令, 如图 1.11 所示, 在弹出的对话框(见图 1.12)中单击“是”按钮, 关闭工作空间。

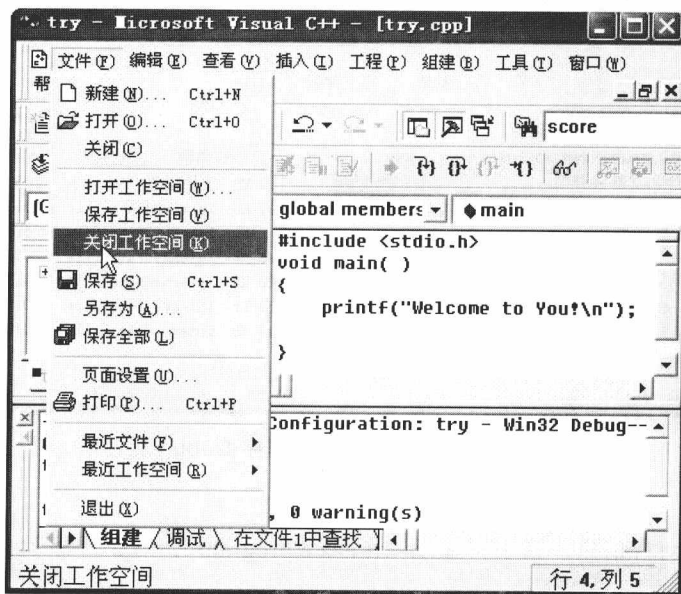


图 1.11 关闭工作空间

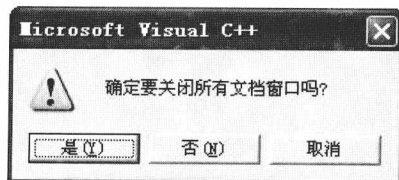


图 1.12 关闭所有文档窗口

#### 4. 查看 C 源文件、目标文件和可执行文件的存放位置

经过编辑、编译、连接和运行后, 在文件夹“D:\C 程序”中存放着源文件 try.cpp, 如图 1.13 所示。在文件夹“D:\C 程序\Debug”中存放着目标文件 try.obj 和可执行文件 try.exe, 如图 1.14 所示。

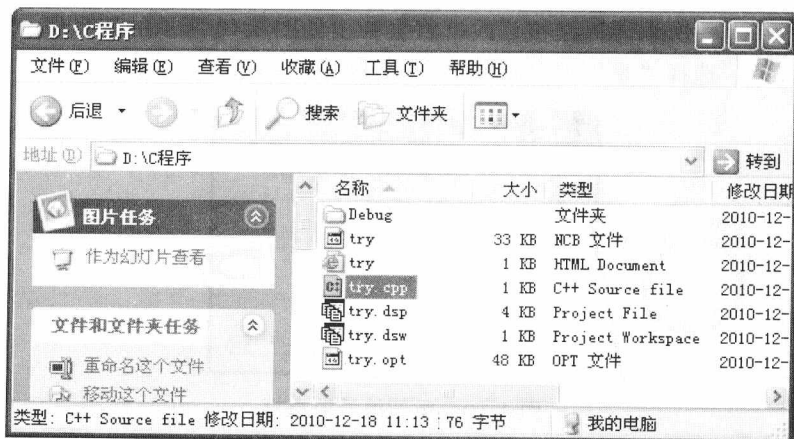


图 1.13 文件夹“D:\C程序”

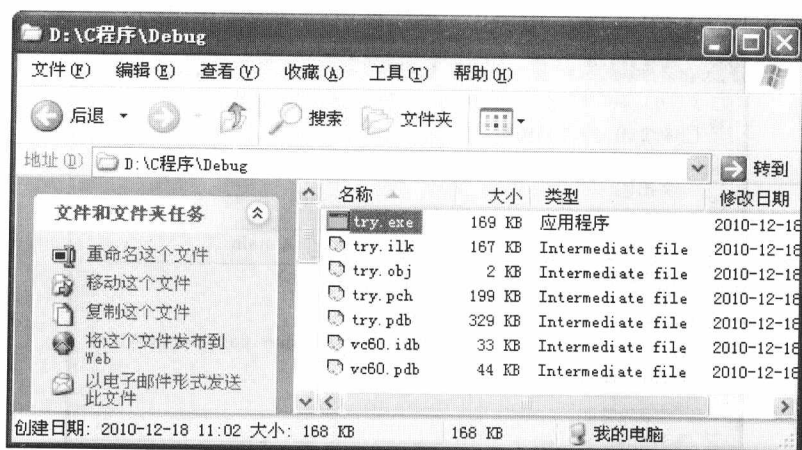


图 1.14 文件夹“D:\C程序\Debug”

## 5. 编程题

在屏幕上显示两行信息。第 1 行显示 “It is a c program”，第 2 行显示自己的姓名和学号。

## 6. 编程题

在屏幕上显示如下星号图形：

```

*
*  *
*

```

## 7. 调试示例

改正下面例子中源程序的错误，在屏幕上显示 “Hello World!”。

```

#include <stdio.h>
void maie()
{
    printf(Hello World!\n")
}

```

(1) 在 VC++ 中编辑以上源程序, 单击“组建”菜单下的“编译”命令, 或者菜单栏的  (compile) 按钮, 出现的编译错误信息如图 1.15 所示。

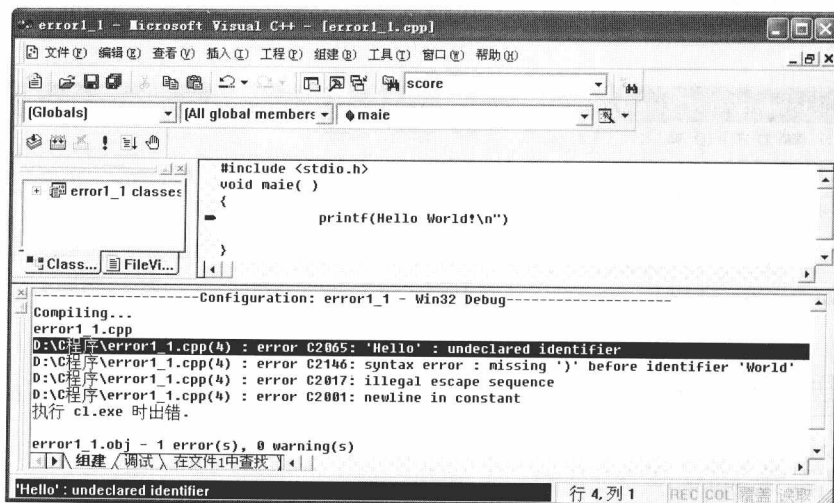


图 1.15 编译产生的错误信息

(2) 找出错误。在信息窗口中双击第一条错误, 编辑窗口就会出现一个指向程序出错的位置, 如图 1.15 所示。一般在箭头的当前行或上一行, 可以找到出错的语句。在图 1.15 中, 箭头指向第 4 行, 错误信息提示“Hello”是一个未定义的变量, 但“Hello”并不是变量, 出错的原因是“Hello”前少了一个双引号。

(3) 改正错误。在“Hello”前加上双引号。

(4) 重新编译。信息窗口显示本次编译的错误信息, 如图 1.16 所示。双击该错误信息, 箭头指向在源程序中的出错位置, 错误信息指出在“}”前缺少分号。改正错误, 在“}”前一条语句最后加上一个分号。

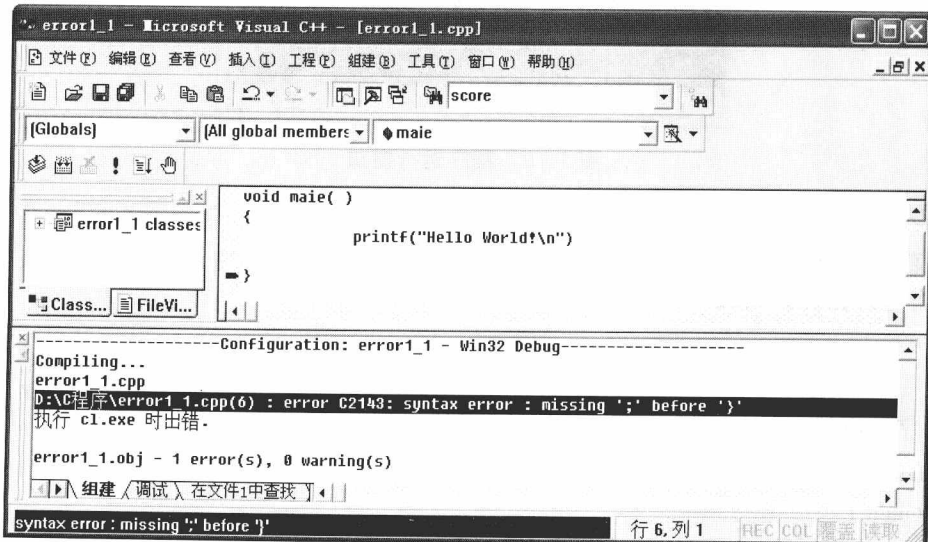


图 1.16 重新编译后产生的错误信息

(5) 再次编译。信息窗口中显示编译正确。

(6) 连接。执行“组建”菜单下的“组建[error1\_1.exe]”命令，开始连接，并在信息窗口中显示连接错误信息，如图 1.17 所示。仔细观察后发现，主函数名“main”拼写错误，被误写为“maie”。

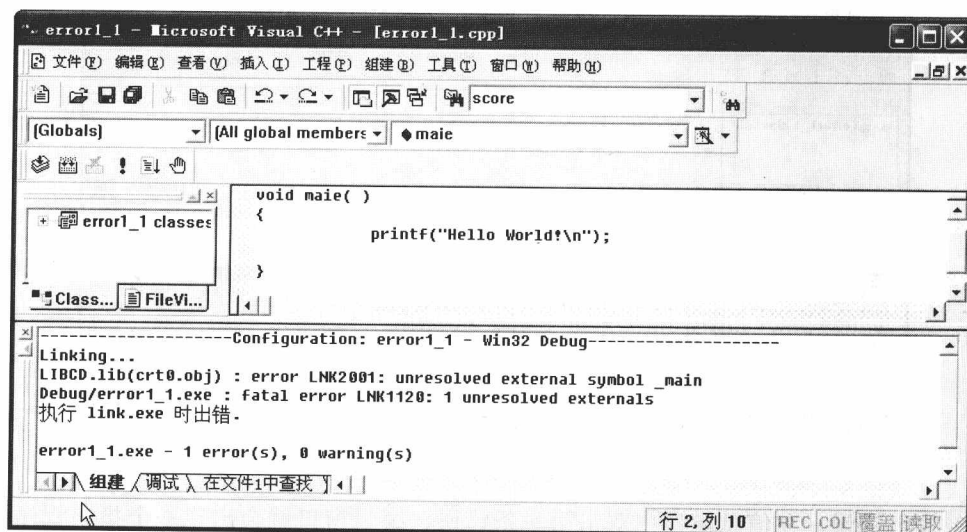


图 1.17 连接产生的错误信息

(7) 改正错误。把“maie”改写为“main”后，重新编译和连接，信息窗口中没有出现错误信息。

(8) 运行。执行“组建”菜单下的“执行[error1\_1.exe]”命令，自动弹出运行窗口，如图 1.18 所示，运行结果与题目要求一致，按任意键返回。

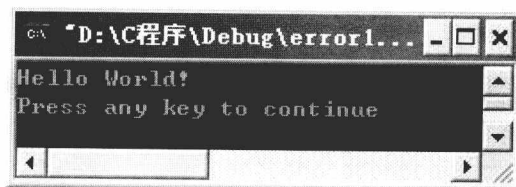


图 1.18 程序运行窗口

## 8. 改错题

在屏幕上显示以下三行信息。

```
*****
```

```
Hello world!
```

```
*****
```

源程序（有错误的程序）：

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

```
void mian()
```

```
{
```

```
    printf("*****\n");
```

```
printf("Hello world!\n")
printf("*****\n");
}
```

### 【实验结果与分析】

将源程序、运行结果、分析情况以及实验中遇到的问题和解决问题的方法写在实验报告上。

## 实验 2 简单程序设计

### 【实验目的】

- (1) 掌握基本算术运算符的使用。
- (2) 掌握输入与输出函数的使用。
- (3) 掌握顺序结构程序设计。
- (4) 掌握常见错误的排除与纠正。
- (5) 掌握简单的单步调试方法。
- (6) 掌握库函数的调用。

### 【实验内容】

#### 1. 调试示例 1

输入一个大于 1 的正整数，利用等差数列的求和公式计算并输出从 1 到该数间所有整数之和。等差数列的求和公式如下：

$$S_n = \frac{n \times (a_1 + a_n)}{2}$$

源程序（有错误的程序）：

```
#include<stdio.h>
void main ( )
{
    int i, sum;
    Scanf("%d", &i);
    Sum=i*(1+i)/2;
    Printf("%d\n",sum);
}
```

(1) 在示例 1 中，使用菜单来完成编译、连接和运行操作，在本节实验里将介绍使用工具栏来完成上述操作的方法。在工具栏或菜单栏上单击鼠标右键，弹出如图 2.1 所示的完整的工具箱菜单，单击“编译微型条”选项，该工具条即出现在工具栏的下方，如图 2.2 所示。其中，第一个按钮 （Compile）表示编译，第二个按钮 （Build）表示组建（用于连接），第四个按钮  表示运行。

(2) 将以上源代码输入 VC++ 中的编辑窗口，点击工具栏的编译按钮 ，编译完成后，在信息窗口的下方将出现如图 2.3 所示的错误。

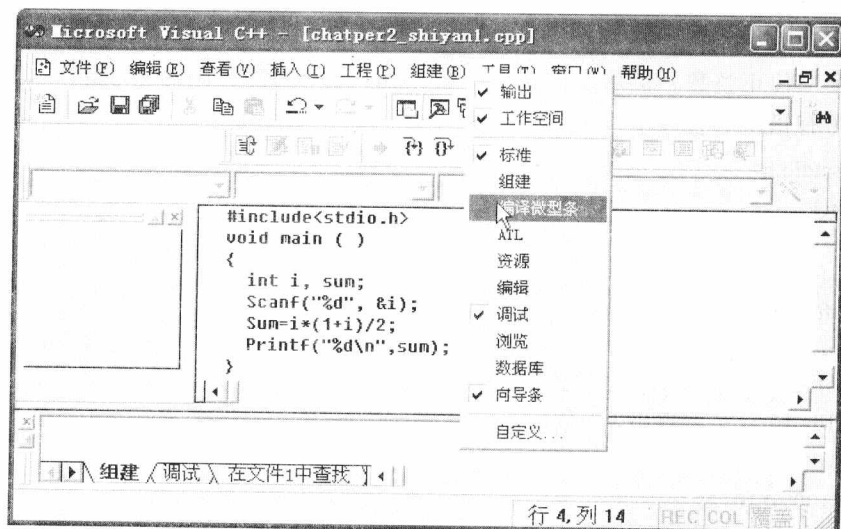


图 2.1 显示完整的工具箱菜单



图 2.2 编译微型条

```

chatper2_shiyan1.cpp
D:\C程序\chatper2_shiyan1.cpp(5) : error C2065: 'Scanf' : undeclared identifier
D:\C程序\chatper2_shiyan1.cpp(6) : error C2065: 'Sum' : undeclared identifier
D:\C程序\chatper2_shiyan1.cpp(7) : error C2065: 'Printf' : undeclared identifier
执行 cl.exe 时出错。

chatper2_shiyan1.obj - 1 error(s), 0 warning(s)

```

图 2.3 错误提示

双击“D:\C 程序\chatper2\_shiyan1.cpp(5) : error C2065: 'Scanf' : undeclared identifier”，光标就会自动跳转到源代码中出现该错误所在的行。此错误含义是：单引号中的'Scanf'是 undeclared identifier，即未声明(定义)的标识符。出现 undeclared identifier 的错误，一般是由于未定义变量、未定义函数等原因引起的。本示例中提示是标识符 Scanf 未声明，仔细检查发现，输出函数 scanf 的第一个字母是小写，而程序中写成了大写，因此，需将大写 S 改为小写的 s。同理，'Sum'和'Printf'中的大写字母也应改成小写。

C 语言严格区分大小写，不可混用。undeclared identifier 表示未声明(定义)标识符，需仔细检查出现该错误所在行的变量、函数等。

改正所有的错误后，再次点击编译按钮，提示：0 error(s), 0 warning(s)，即无错误。编译无误后即可点击执行按钮 ，运行程序，运行结果如图 2.4 所示。

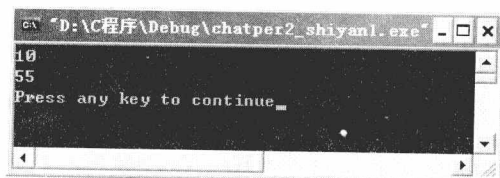


图 2.4 示例程序运行结果

## 2. 调试示例 2

输入一个三位正整数，然后反向输出此数，如：输入 123，则输出 321。

源程序（有错误的程序）：

```
#include<stdio.h>

void main()
{
    int n,i,j,k;
    printf("请输入一个正整数：");
    scanf("%d",&n);
    i=n%10;          /*取出个位数*/
    j=n/10%10;       /*取出十位数*/
    k=n%100;         /*取出百位数*/

    n=i*100+j*10+k;
    printf("\n%d\n",n);
}
```

将以上源代码输入 VC++ 中的编辑窗口并编译、运行，运行结果如图 2.5 所示。

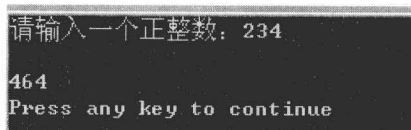


图 2.5 示例程序 2 运行结果

输入 234，输出 464，发现语法上没有错误，但运行结果是错误的。在这种情况下，需要通过调试来发现并改正错误。

单步调试的具体步骤如下：

(1) 在工具栏或菜单栏上单击鼠标右键，弹出如图 2.6 所示的完整的工具箱菜单，单击“调试”选项，该调试工具栏即可出现，如图 2.7 所示。

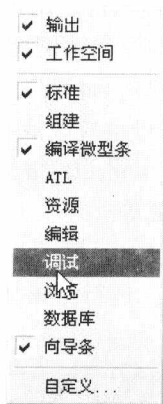


图 2.6 显示完整的工具箱菜单

(2) 单击调试工具栏中的 Step Over 按钮 ，每次执行一行语句，则编辑窗口出现如图 2.7 所示的变化。



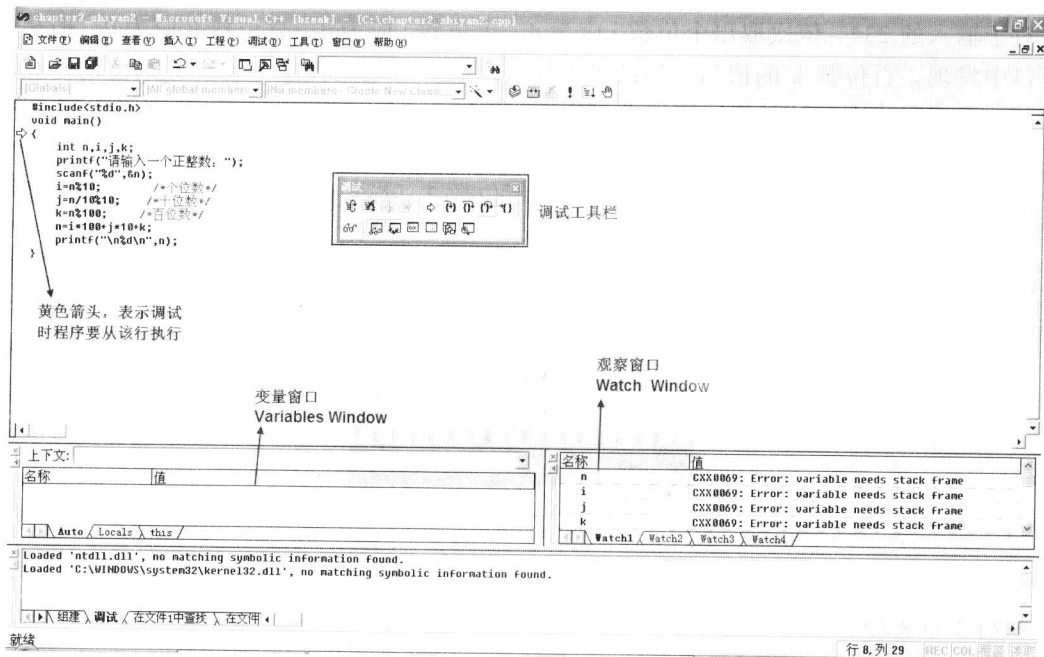


图 2.7 调试窗口

(3) 每点击一次 Step Over 按钮 , 程序执行一行(黄色箭头依次往下移动), 当执行到 scanf 函数所在的行时, 需用户输入数据, 如图 2.8 所示。



图 2.8 输入数据窗口

(4) 继续点击 Step Over 按钮 , 让程序按顺序往下执行, 在变量窗口中可发现, 当程序执行完语句 “i=n%10;” 时, i 的值为 4, 如图 2.9 所示。

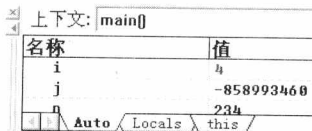


图 2.9 变量窗口

依次往下执行, 可逐个显示其他变量的值。

当程序变量太多, 或者循环内部变量的值不断变化, 或者跟踪函数的局部变量时, 也可以在观察窗口中输入变量名, 来实时观察变量的值。如本示例中, 在观察窗口的名称栏的空格中依次输入 i、j、k, 并点击 Step Over 按钮 , 观察此三个变量的值, 如图 2.10 所示。

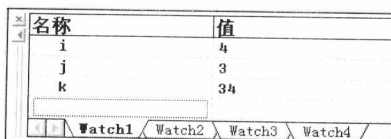


图 2.10 观察窗口