



高等学校“十二五”应用型特色规划教材

C语言程序设计

上机实验与习题解答

C Programming Language

◎ 蔺德军 贾小珠 主 编 ◎ 高长铎 夏方遁 副主编



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

高等学校“十二五”应用型特色规划教材

C 语言程序设计上机实验与 习题解答

蔺德军 贾小珠 主 编

高长铎 夏方道 副主编

侯东昌 赵希梅 参 编

刘 堃 崔良中

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书是《C 语言程序设计》(ISBN 978-7-121-27422-0)的配套教材,内容包括三部分:上机实验、教材的习题解答、部分实验教材中的习题答案。其中实验部分严格按照教材的内容次序,针对 C 语言中的知识点,从简单到复杂,精心组织了实验过程,每个实验大约需要 2 学时,练习从最简单的示例开始,逐步提高难度,完成一个练习,也就掌握了一个章节的内容。教材的习题解答部分给出了教材中全部习题的答案及必要的解析。

实验教材的答案中之所以只给出了部分习题的答案,是避免学生直接查看答案,缺少了思考的过程,也就失去了上机练习的意义,因此只附带部分所谓“难题”的答案,完整的实验教材答案可以到华信教育资源网下载电子版。

本书特别适合作为大学 C 语言公选课实验教材及自学 C 语言的实验教材,也适合作为二级 C 语言考试的参考用书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

C 语言程序设计上机实验与习题解答 / 蔺德军, 贾小珠主编. —北京: 电子工业出版社, 2015.11

ISBN 978-7-121-27423-7

I. ①C… II. ①蔺… ②贾… III. ①C 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(2015)第 246573 号

策划编辑: 任欢欢

责任编辑: 任欢欢

印 刷: 三河市华成印务有限公司

装 订: 三河市华成印务有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 787×1092 1/16 印张: 12.75 字数: 326.4 千字

版 次: 2015 年 11 月第 1 版

印 次: 2015 年 11 月第 1 次印刷

定 价: 29.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010)88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010)88258888。

前 言

学习计算机语言，在计算机上编程练习是至关重要的，否则就无法掌握一门语言，本书旨在帮助学生上机练习，从浅入深，逐步引导学生学习 C 语言的各项内容。

本实验教材的安排如同上山的台阶，从第一题开始做起，逐题提升一点高度，最后便能掌握整章内容，达到顶峰，因此使用本教材要按次序逐题练习，因为后面习题的答案可能就隐藏在前面的题中，这样做反而会省时省力并且高效，反之如同要一下跳上百米高台，尽管很努力，最后结果可能是白费的。

翻看答案，功效减半！学习和教学的经验告诉我，编程的学习，锻炼的就是逻辑思维能力，只记住别人写的一段代码，对自己编程能力没有任何的增益，相反，自己去摸索，去尝试，即使最后没有完成程序的编写，能力反倒有所提高。

本课程介绍了使用 Visual C++ 6.0 编译环境(推荐)、Dev-Cpp 软件环境、Visual Studio 2013 环境下调试 C 语言程序的方法。而本书所有示例仅在 Visual C++ 6.0 环境下调试通过。三个软件中，Visual C++ 6.0 SP1 使用最为普遍，可以在 Windows 2000、Windows XP、Windows 7 下运行；Dev-Cpp（或 Dev-C++）占用空间较小，也可以在 Windows 2000、Windows XP、Windows 7 下运行；Visual Studio 2013 比较庞大，在 Windows 7 下运行没有问题，能否在 Windows 8 下安装运行，尚未检验。

在本书编写过程中，得到姜彬、刘晓洁、张云红、郑世秀、朱威同等各位 C 语言专家的耐心指导和大力支持，在此表示感谢。

本书编写仓促，编者水平有限，书中难免有疏漏和错误，还希望大家多加指正，共同进步。

编 者

目 录

实验 1	操作环境与过程	1
1.1	实验目的	1
1.2	实验预习	1
1.3	使用 VC++ 6.0 上机过程	1
1.4	使用 Dev-C++上机过程	4
1.5	在 VS 2013 中如何调试 C 语言程序	7
1.6	编译时常见错误	8
1.7	链接时常见错误	9
1.8	常见错误及注意事项	10
实验 2	变量类型与输入/输出	11
2.1	实验目的	11
2.2	实验预习	11
2.3	实验内容	11
2.4	常见错误及注意事项	14
实验 3	运算符与表达式	15
3.1	实验目的	15
3.2	实验预习	15
3.3	实验内容	15
3.4	常见错误及注意事项	19
实验 4	顺序结构程序设计	20
4.1	实验目的	20
4.2	实验预习	20
4.3	实验内容	20
4.4	常见错误及注意事项	24
实验 5	选择结构程序设计	25
5.1	实验目的	25
5.2	实验预习	25
5.3	实验内容	25
5.4	常见错误及注意事项	30
实验 6	循环结构程序设计（一）	31
6.1	实验目的	31
6.2	实验预习	31

6.3	实验内容	31
6.4	常见错误及注意事项	34
实验 7	循环结构程序设计 (二)	35
7.1	实验目的	35
7.2	实验预习	35
7.3	实验内容	35
7.4	常见错误及注意事项	38
实验 8	综合程序设计练习 (一)	39
8.1	实验目的	39
8.2	实验预习	39
8.3	实验内容	39
8.4	常见错误及注意事项	43
实验 9	数组	44
9.1	实验目的	44
9.2	实验预习	44
9.3	实验内容	44
9.4	常见错误及注意事项	48
实验 10	字符串处理	49
10.1	实验目的	49
10.2	实验预习	49
10.3	实验内容	49
10.4	常见错误及注意事项	53
实验 11	指针 (一)	54
11.1	实验目的	54
11.2	实验预习	54
11.3	实验内容	54
11.4	常见错误及注意事项	59
实验 12	指针 (二) (选做)	60
12.1	实验目的	60
12.2	实验预习	60
12.3	实验内容	60
12.4	常见错误及注意事项	64
实验 13	函数 (一)	65
13.1	实验目的	65
13.2	实验预习	65
13.3	实验内容	65
13.4	常见错误及注意事项	70

实验 14	函数（二）	71
14.1	实验目的	71
14.2	实验预习	71
14.3	实验内容	71
14.4	常见错误及注意事项	75
实验 15	综合程序设计练习（二）	76
15.1	实验目的	76
15.2	实验预习	76
15.3	实验内容	76
实验 16	结构体、共用体、枚举	81
16.1	实验目的	81
16.2	实验预习	81
16.3	实验内容	81
16.4	常见错误及注意事项	87
实验 17	链表（选做）	88
17.1	实验目的	88
17.2	实验预习	88
17.3	实验内容	88
17.4	常见错误及注意事项	94
实验 18	编译预处理	95
18.1	实验目的	95
18.2	实验预习	95
18.3	实验内容	95
18.4	常见错误及注意事项	98
实验 19	文件	99
19.1	实验目的	99
19.2	实验预习	99
19.3	实验内容	99
19.4	常见错误及注意事项	103
实验 20	C 语言二级机试真题	104
20.1	实验目的	104
20.2	实验预习	104
20.3	实验内容	104
实验 21	综合程序设计练习（三）	121
21.1	实验目的	121
21.2	实验预习	121
21.3	实验内容	121

附录 A 教材课后习题答案解析	122
习题 1	122
习题 2	123
习题 3	124
习题 4	127
习题 5	129
习题 6	134
习题 7	140
习题 8	152
习题 9	154
习题 10	160
习题 11	171
习题 12	171
附录 B 部分实验参考答案	177
实验 12 指针 (二) (选做)	177
实验 15 综合程序设计练习 (二)	178
实验 17 链表 (选做)	182
实验 19 文件	187
实验 20 C 语言二级机试真题	190
实验 21 综合程序设计练习 (三)	193

实验 1 操作环境与过程

1.1 实验目的

1. 学习如何在 VC++ 6.0 环境中编写 C 语言程序。
2. 学习如何在 VC++ 6.0 环境中编译、连接和运行 C 语言程序。

1.2 实验预习

1. 熟悉文字的编辑修改。
2. 熟悉 C 语言程序的基本结构。
3. 熟悉 C 语言注释的表示方法。
4. 理解 C 语言原程序转换为可执行代码的过程。

1.3 使用 VC++ 6.0 上机过程

【步骤 1】选择[开始]→[程序]→[Microsoft Visual Studio]→[Microsoft Visual C++]菜单选项进入 VC++ 6.0，出现如图 1.1 所示界面。

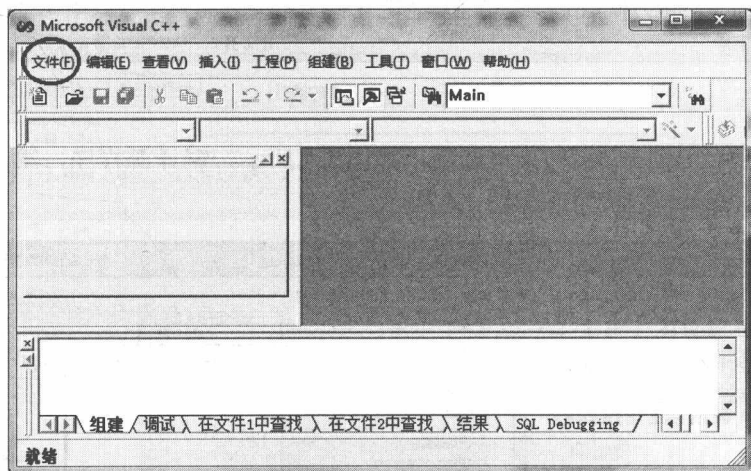


图 1.1 VC++ 6.0 主界面

【步骤 2】选择[文件]→[新建]菜单选项，打开“新建”对话框，在[文件]选项卡中，选择[C++ Source File]选项，如图 1.2 所示。

【步骤 3】选择保存文件的位置，并输入文件名（如图 1.2 所示），扩展名可以省略，系统将自动添加扩展名.cpp，单击[确定]按钮，在打开的界面中开始输入源程序，如图 1.3 所示。

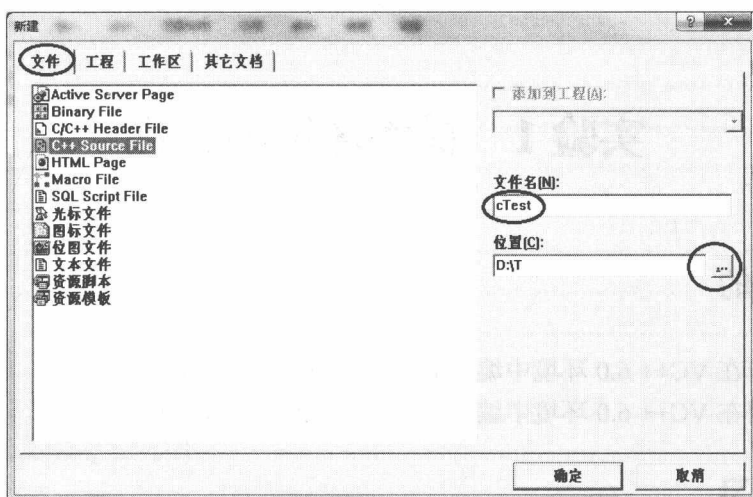


图 1.2 “新建”对话框

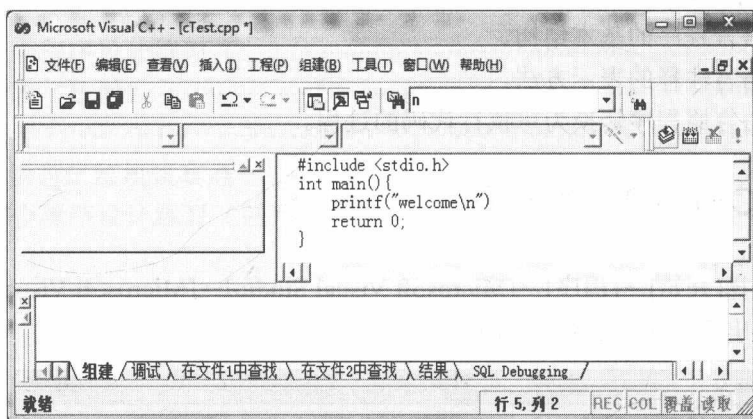


图 1.3 输入源程序

【步骤 4】选择[组建]→[编译]菜单选项，如图 1.4 所示，编译源程序，第一次编译会出现建立工作区请求，单击[是]按钮，如图 1.5 所示。

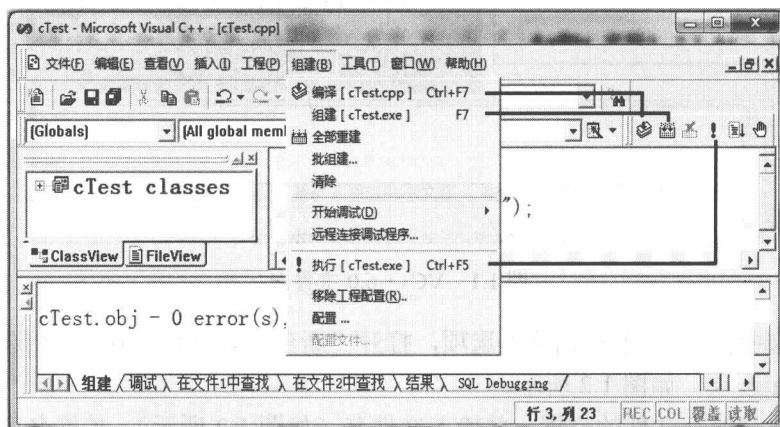


图 1.4 [编译]菜单选项

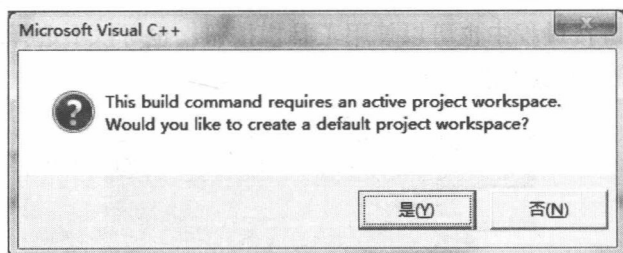


图 1.5 建立工作区请求

在出现保存请求时单击[是]按钮，如图 1.6 所示。

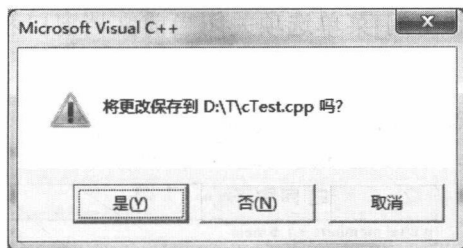


图 1.6 保存请求

【步骤 5】查看调试窗口内有无提示错误信息，若有，则根据提示的行号和信息进行修改，例如：若源代码中语句缺少结束符号“;”，则会出现如图 1.7 所示的错误提示信息。

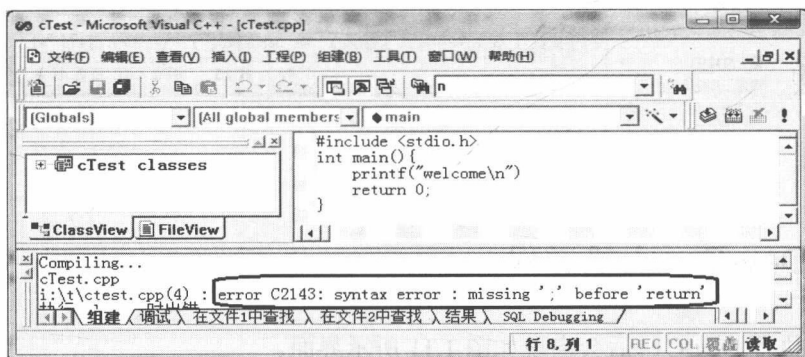


图 1.7 语句缺少“;”错误提示

【步骤 6】改正后错误重新编译，直至出现 0 个错误提示后，选择[组建]→[组建]菜单选项进行链接，若有错则改，若没有错误就可以选择[组建]→[执行]菜单命令开始运行程序了，然后在控制台窗口内查看结果，如果结果不对，则修改程序，重复上面的过程，如图 1.8 所示。

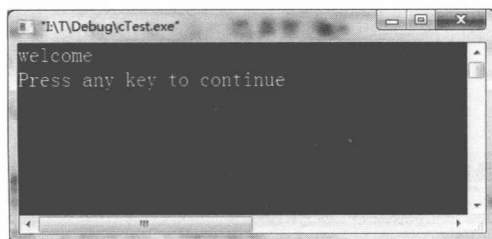


图 1.8 查看结果

【步骤 7】在编译链接过程中也可以使用工具栏按钮，也可以不选择编译，而直接选择运行，系统会自动先编译、链接后执行，如图 1.9 所示。

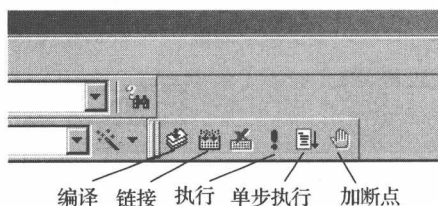


图 1.9 工具栏按钮

【步骤 8】选择[关闭工作空间]菜单选项关闭当前文档窗口和工作空间，开始下一个程序，如图 1.10 所示。

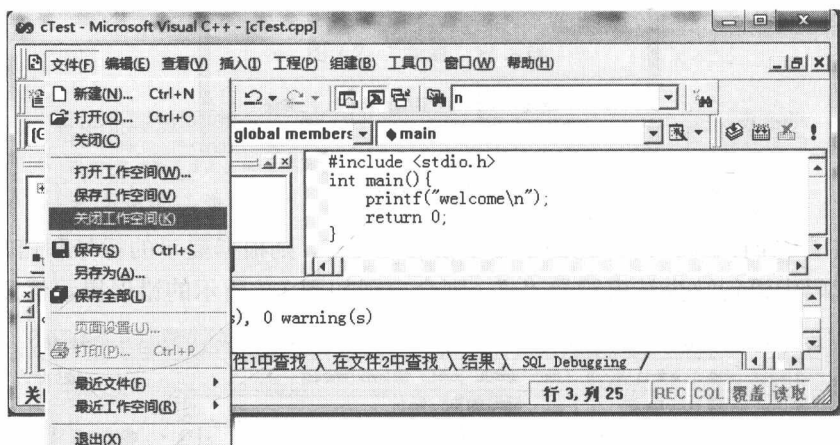


图 1.10

1.4 使用 Dev-C++上机过程

【步骤 1】执行 Dev-C++后会出现如图 1.11 所示界面。

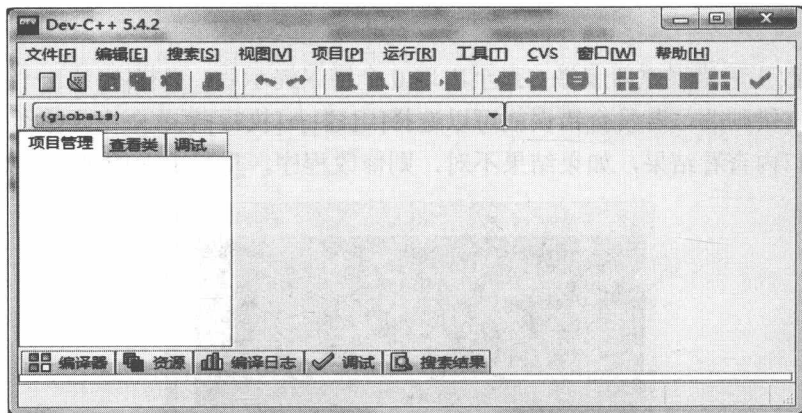


图 1.11 Dev-C++界面

【步骤 2】选择[文件]→[新建]→[源代码]菜单选项，输入源代码（图 1.12）。



图 1.12 输入源代码

【步骤 3】选择[运行]→[编译]菜单选项或单击对应按钮，如图 1.13 所示。

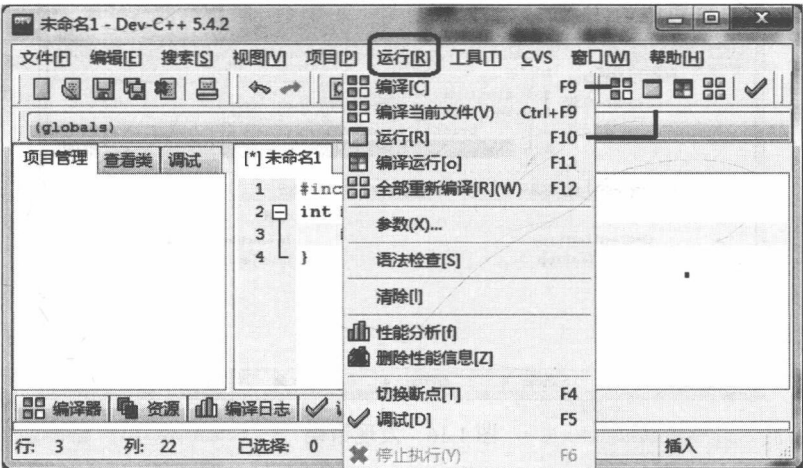


图 1.13 [编译]菜单命令

【步骤 4】在弹出的询问对话框中单击[Yes]按钮，如图 1.14 所示，然后选择保存的位置和文件名，如图 1.15 所示。

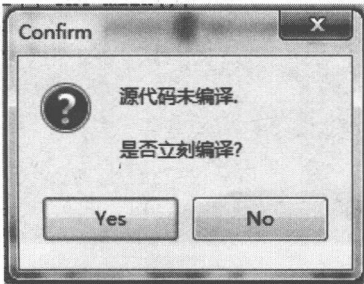


图 1.14 询问对话框

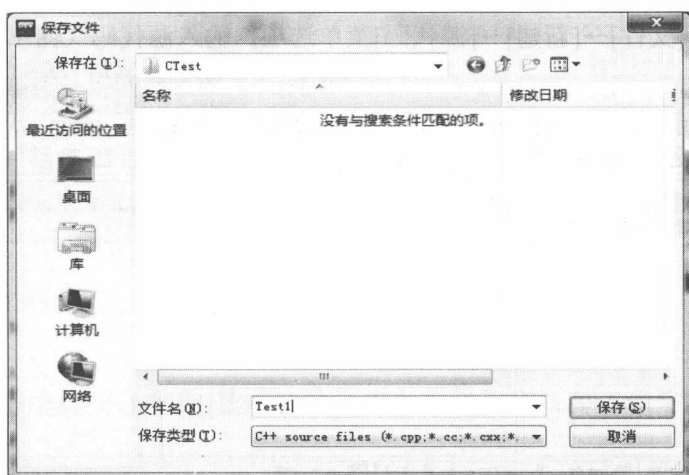


图 1.15 “保存文件”对话框

【步骤 5】若发现错误，如图 1.16 所示，则修改错误后，重新编译。

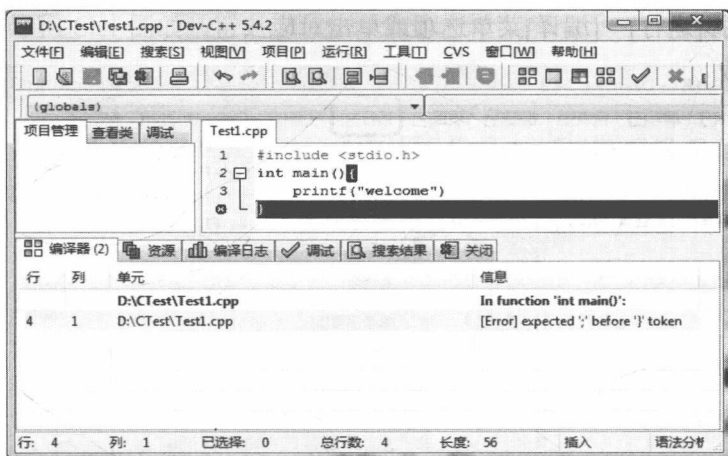


图 1.16 发现错误

【步骤 6】直到没有错误后，选择[运行]菜单命令或单击相应按钮，如图 1.13 所示，查看运行结果，如图 1.17 所示。

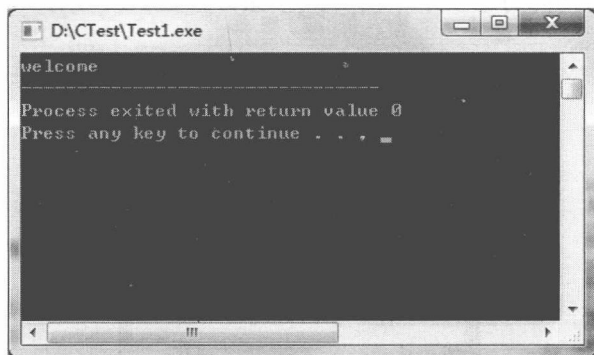


图 1.17 查看运行结果

【步骤 7】一个程序调试完成后, 选择[文件]→[全部关闭]菜单命令, 然后重新开始。

1.5 在 VS 2013 中如何调试 C 语言程序

【步骤 1】选择[文件]→[新建]→[项目]菜单命令, 如图 1.18 所示。

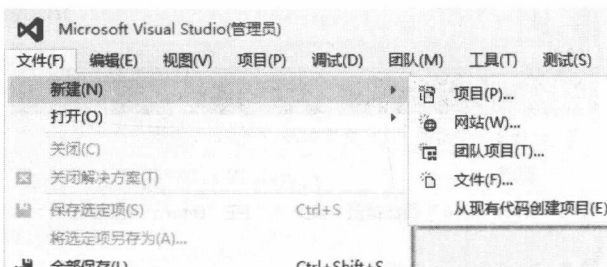


图 1.18 [项目]菜单命令

【步骤 2】选择[Visual C++]→[Win32]→[Win32 控制台应用程序], 设置项目名和存储位置后, 单击[确定]按钮完成项目的新建, 如图 1.19 所示。

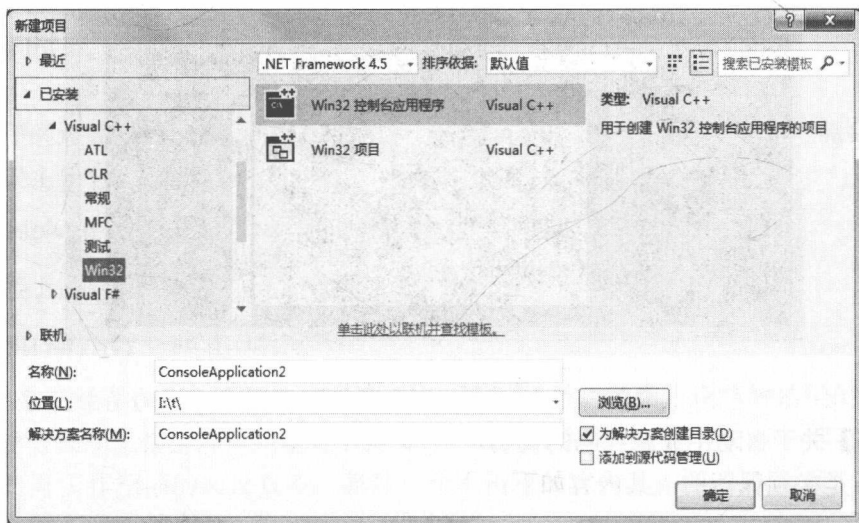


图 1.19 新建项目

【步骤 3】原有代码不必改动, 添加程序代码, 如图 1.20 所示。

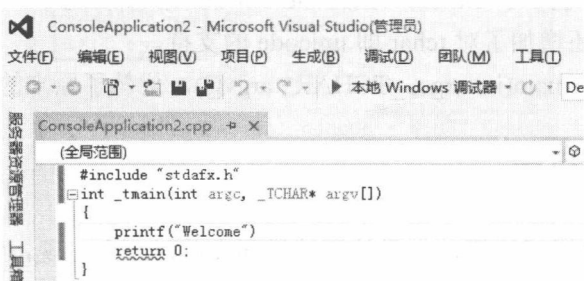


图 1.20 添加程序代码

【步骤 4】单击[生成]→[生成解决方案]菜单命令，根据错误提示修改错误，如图 1.21 所示，然后重新生成，直到成功。

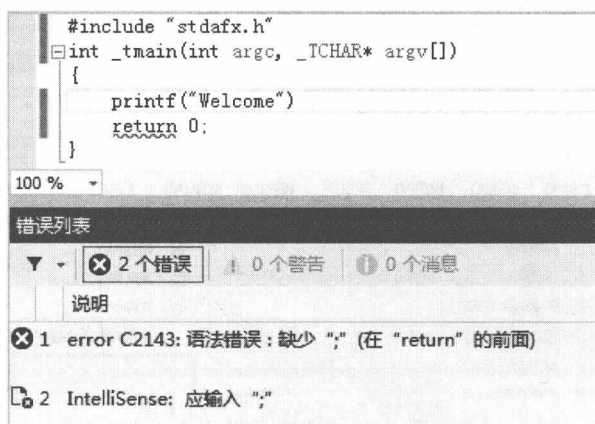


图 1.21 根据错误提示修改错误

【步骤 5】单击[调试]→[开始执行]菜单命令，结果如图 1.22 所示。

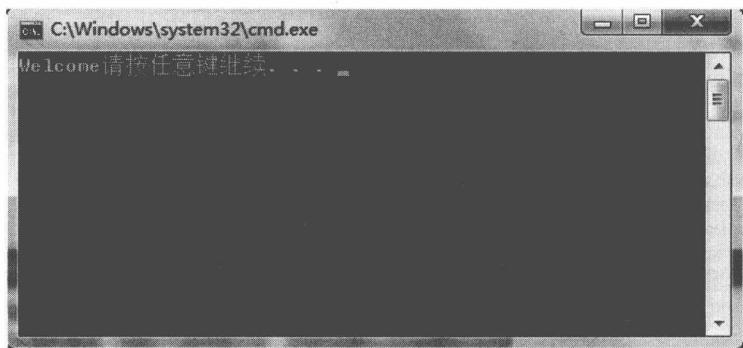


图 1.22 执行程序

【步骤 6】关于自动生成源代码的说明。

stdafx.h 是必须保留的，其内容如下：

```
#include "targetver.h"
#include <stdio.h>
#include <tchar.h>
```

可以看出，其中已包含了 stdio.h，因此 stdio.h 就不需要再写了，需要其他函数时，仍然需要添加，stdafx.h 中还增加了对 tchar 即 unicode 的支持。

入口函数改为 int _tmain(int argc, _TCHAR* argv[])，代替了原来的 int main() 函数。

1.6 编译时常见错误

写完程序后，先要进行编译，大量的语法错误在此时便会出现，常见的语法错误有以下几种。

1. Cannot open include file

嵌入文件打不开, #include 后的文件名拼写错误。

2. syntax error : missing ';'

缺少 “;” 漏写 “;” 或有多余的 “}”。

3. newline in constant

常量跨行漏写 “\n”。

4. unexpected end of file found

未发现文件结尾, 缺少 “}” 或 “{” 不配套。

5. unknown character

错用了中文标点。

6. XXX:undeclared identifier

不认识的关键字, 一般是变量名未声明, 或未加函数所在的头文件。

1.7 链接时常见错误

一个程序经过编译没有错误, 仅仅表示没有语法错误了, 但是链接时还可能会出现错误, 链接时的错误主要有以下三种。

1. unresolved external symbol _main

无法链接上 main 函数 (如写成了 mian)。

2. _main already defined in XXX

main 函数已经存在, 一个程序中出现多个 main 函数。通常出现这种情况的原因是, 写完程序后, 没有关闭工作空间, 又建立了新文件, 导致一个程序, 两个文件, 都有 main 函数。解决办法是打开工作空间的文件页签, 删除一个不需要的文件, 如图 1.23 所示。

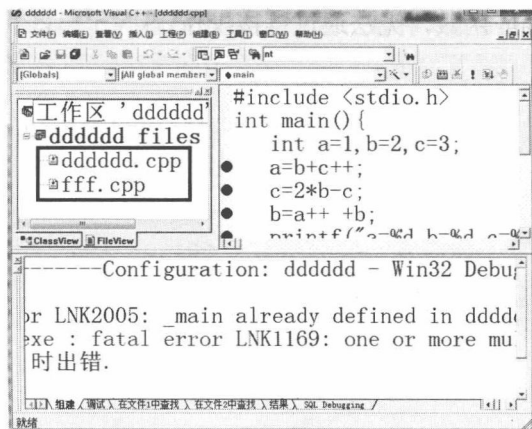


图 1.23 删除一个不需要的文件