

C语言 程序设计 案例实践

李丹程 刘莹 那俊 著



清华大学出版社

C 语言程序设计 案例实践

李丹程 刘 莹 那 俊 著

清华大学出版社

北 京

内 容 简 介

本书涵盖了C语言的重要基础知识,重点放在C语言的综合运用上,通过详细实例,结合具体的实际应用需求,循序渐进地启发学生独立完成实践题目,培养学生实际分析问题、解决问题的能力。本书设计了十个信息管理实例项目,以满足不同学校和学生的程序实践要求。

本书不仅可以作为计算机及相关专业的C语言实践课程教材,而且也可以作为其他C语言教材的参考书,亦可作为高等学校各专业C语言教辅教材。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C语言程序设计案例实践/李丹程,刘莹,那俊著. —北京:清华大学出版社,2009.2
ISBN 978-7-302-19101-8

I. C… II. ①李… ②刘… ③那… III. C语言—程序设计 IV. TP312

中国版本图书馆CIP数据核字(2008)第196823号

责任编辑:龙啟铭

责任校对:徐俊伟

责任印制:孟凡玉

出版发行:清华大学出版社

地 址:北京清华大学学研大厦A座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185×260 印 张:11.75

字 数:282千字

版 次:2009年2月第1版

印 次:2009年2月第1次印刷

印 数:1~3000

定 价:25.00元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:010-62770177 转 3103 产品编号:029627-01

前 言

程序实践是对编程设计能力的综合训练,对培养具有创新意识、创新能力的高素质软件人才有很大的推动作用。通过采用项目实例来进行 C 语言编程的综合训练,可以提高学生实际分析问题、解决问题、编程实践、自主创新的能力,同时,还可以培养学生的团队合作精神、创新协作意识及勇于探索的科学精神。本书程序实践题目是基于日常生活中常见的应用领域,可以提高学生的编程乐趣,切身体会 C 语言编程在现实生活中发挥的作用。

本程序实践的特点如下。

(1) 涵盖了 C 语言的重要基础知识,包括三种基本程序结构、结构体、链表、文件的使用等,重点放在结构体的处理方面。

(2) 每章的设计要求均结合实际应用的需求,激发学生的兴趣,调动学生主动学习的积极性,引导学生按照实际需要进行编程实践。

(3) 采用引导式学习的方式,通过详细实例介绍 Visual Studio 6.0 开发环境下工程建立、调试、测试的方法,将设计要求、算法和源程序分开,便于学生进行独立思考,学生可以在理解要求的基础上,脱离书中提供的代码来完成任务。

(4) 提供思考题引导学生拓展自己的思维,对实践过程进行总结和归纳。

(5) 全书的代码按照规范编写,给学生提供良好的范例,培养学生严谨的编程态度和良好的编程风格。

全书共分 11 章,第 0 章预备知识部分介绍本程序实践使用的开发环境;第 1~10 章提供了学生熟悉的 10 个生活中常用的信息管理项目,锻炼学生解决实际问题的能力。

参加本书编写的人员还有石凯、王学宾、滕志扬、廖丁石、于鹤鑫、闫振南、王重、刘景明、艾萌、郑晨、贾若、袁铮、黄守凯,他们参与了部分章节的编写、程序调试和校对等工作,为本书的出版花费了大量的心血。此外,朱志良教授对本书出版给予了大力支持,在此向他们表示衷心的感谢。

由于时间仓促及编者水平有限,书中不足之处在所难免,敬请读者批评指正。

目 录

第 0 章 预备知识——Visual Studio 6.0 开发环境	1
0.1 Visual C++ 6.0 的安装和启动	1
0.2 编辑一个 C 源程序	2
0.3 编译、链接和运行 C 程序	6
0.4 C 程序的调试	8
第 1 章 学生信息管理程序	11
1.1 实践目的	11
1.2 基本要求	11
1.3 算法分析	12
1.4 参考代码	15
1.5 代码测试	46
1.6 思考题	47
第 2 章 旅馆信息管理程序	48
2.1 实践目的	48
2.2 基本要求	48
2.3 算法分析	48
2.4 参考代码	52
2.5 代码测试	68
2.6 思考题	69
第 3 章 餐厅信息管理程序	70
3.1 实践目的	70
3.2 基本要求	70
3.3 算法分析	71
3.4 参考代码	73
3.5 代码测试	83
3.6 思考题	84
第 4 章 职工信息管理程序	85
4.1 实践目的	85
4.2 基本要求	85
4.3 算法分析	86
4.4 参考代码	88
4.5 代码测试	103
4.6 思考题	107

第 5 章 设备信息管理程序.....	108
5.1 实践目的.....	108
5.2 基本要求.....	108
5.3 算法分析.....	109
5.4 参考代码.....	111
5.5 代码测试.....	124
5.6 思考题.....	126
第 6 章 图书信息管理程序.....	127
6.1 实践目的.....	127
6.2 基本要求.....	127
6.3 算法分析.....	127
6.4 参考代码.....	130
6.5 代码测试.....	136
6.6 思考题.....	137
第 7 章 医院信息管理程序.....	138
7.1 实践目的.....	138
7.2 基本要求.....	138
7.3 算法分析.....	139
7.4 参考代码.....	141
7.5 代码测试.....	150
7.6 思考题.....	151
第 8 章 超市信息管理程序.....	152
8.1 实践目的.....	152
8.2 基本要求.....	152
8.3 算法分析.....	152
8.4 参考代码.....	154
8.5 代码测试.....	160
8.6 思考题.....	162
第 9 章 票务信息管理程序.....	163
9.1 实践目的.....	163
9.2 基本要求.....	163
9.3 算法分析.....	163
9.4 参考代码.....	167
9.5 代码测试.....	171
9.6 思考题.....	172
第 10 章 超级终端仿真程序.....	173
10.1 实践目的.....	173
10.2 基本要求.....	173

10.3	算法分析	173
10.4	参考代码	176
10.5	代码测试	178
10.6	思考题	178

第 0 章 预备知识——Visual Studio 6.0 开发环境

本书以 Visual C++ 6.0 作为 C 源程序的实践开发环境，本章将首先介绍 Visual C++ 6.0 环境的基本操作，包括 Visual C++ 6.0 的安装和启动、C 源程序的编辑、运行与调试。

0.1 Visual C++ 6.0 的安装和启动

如果计算机上未安装 Visual C++ 6.0，则按照安装向导直接安装，具体步骤此处不再详述。安装过程中建议各位同时安装 MSDN，以便日后自学。

成功安装 Visual C++ 6.0 后，可以在桌面上看到如图 0.1 所示的图标或通过“开始”->“程序”->Microsoft Visual Studio 找到 Visual C++ 6.0 启动程序。



图 0.1 Visual C++ 6.0 图标

双击 Visual C++ 6.0 图标或者单击“开始”->“程序”->Microsoft Visual Studio->Visual C++ 6.0 即可启动 Visual C++ 6.0 集成开发环境，正常启动开发环境后，可见如图 0.2 所示的主窗口操作界面。

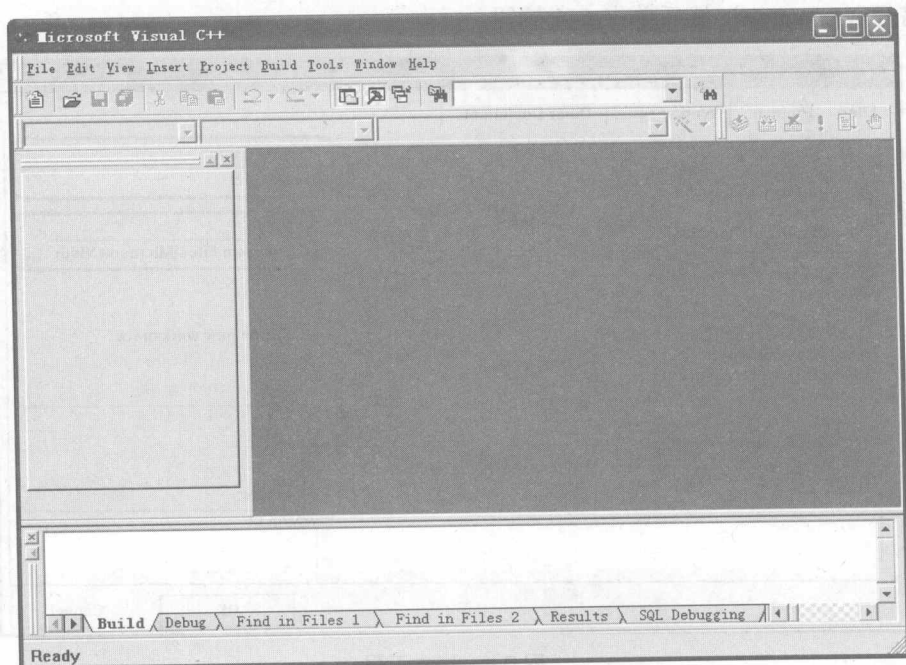


图 0.2 Visual C++ 6.0 主窗口

0.2 编辑一个 C 源程序

编辑一个 C 源程序的前提是新建一个程序或者打开现有的程序。本节首先介绍如何新建一个程序，如何打开现有程序，并在此基础上对文件进行编辑。

1. 新建一个工程

在 Visual C++ 6.0 主窗口的界面中的主菜单栏中单击 File（文件）菜单项，然后单击 New（新建）菜单项，也可按 Ctrl+N 组合键进入，如图 0.3 所示。

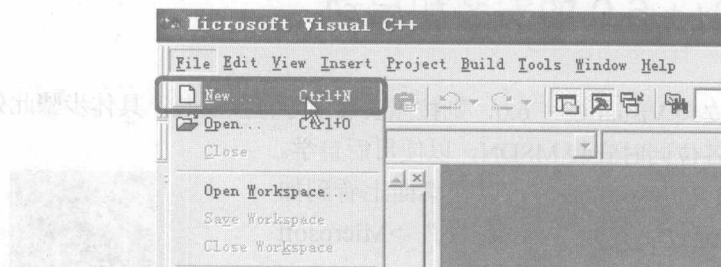


图 0.3 新建工程

屏幕上出现一个 New（新建）对话框，如图 0.4 所示。单击此对话框上方的 Projects（工程）标签，在其下方菜单选择 Win32 Console Application 项，输入工程名，设定工程存储路径（Location）然后单击 OK 按钮。

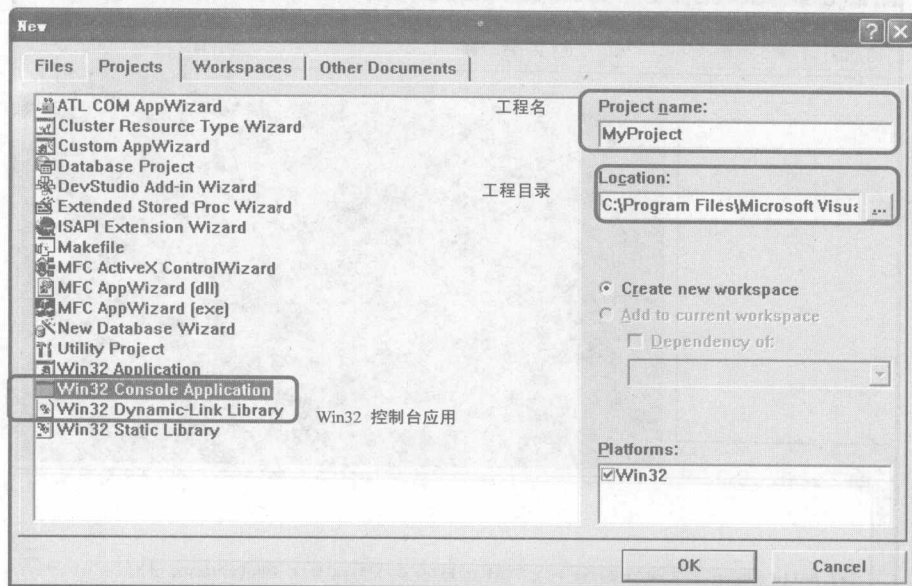


图 0.4 建立工程

单击 OK 按钮后，会弹出如图 0.5 所示的界面，选择 An empty project，单击 Finish

按钮。

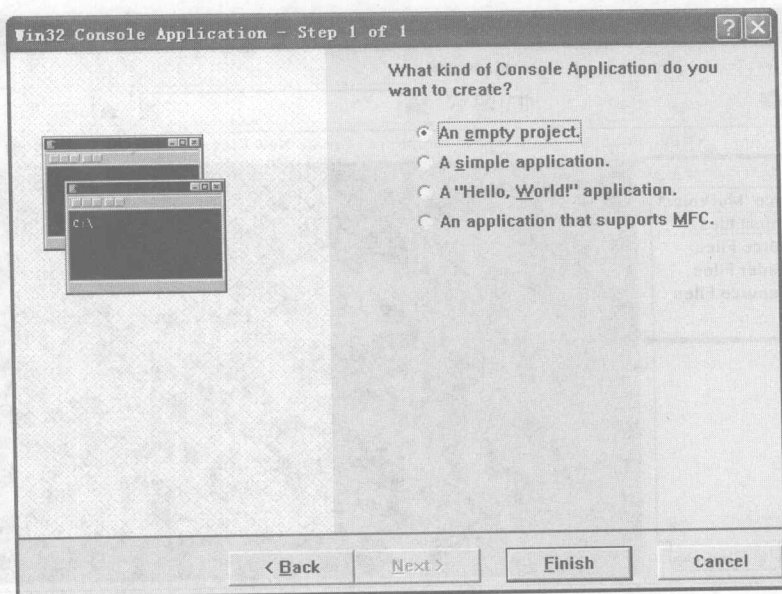


图 0.5 工程类型选择

之后，弹出如图 0.6 所示的界面。注意：窗口上显示了工程文件的路径。

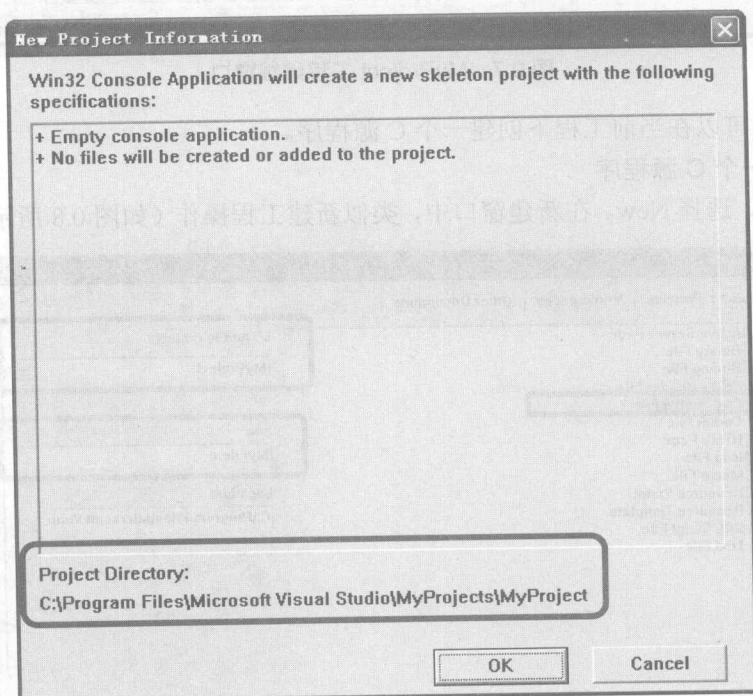


图 0.6 新工程信息

单击 OK 按钮，则进入工程编辑窗口，如图 0.7 所示。在工程窗口可以看到 MyProject

工程下只有三个空白的文件夹。

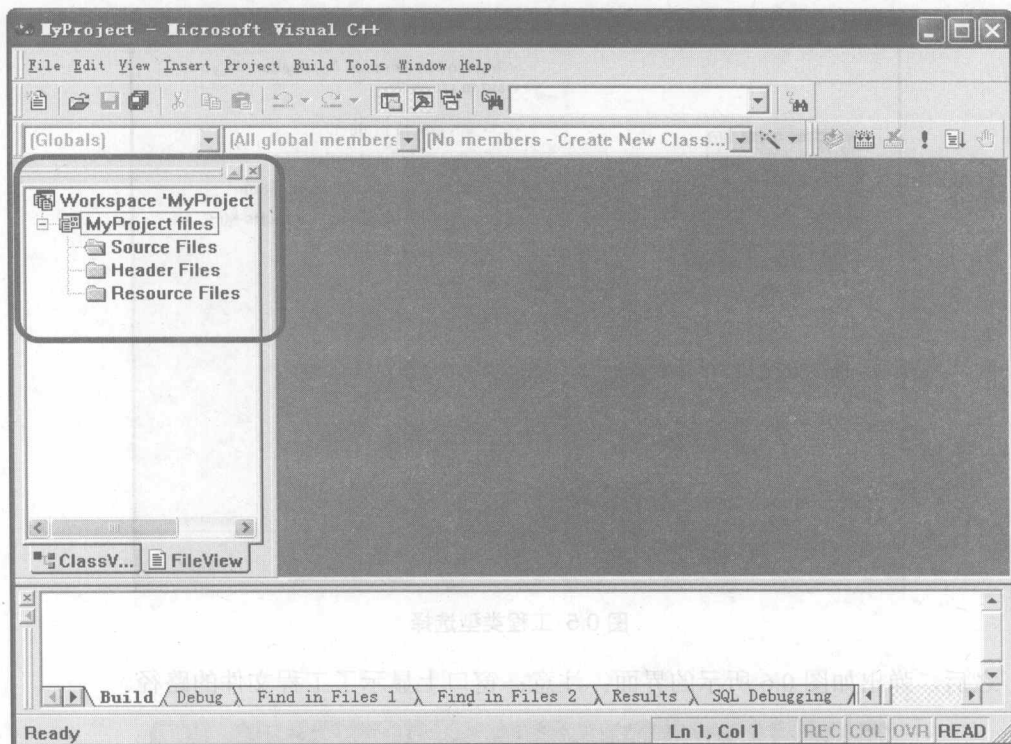


图 0.7 MyProject 工程编辑窗口

此后，就可以在当前工程下创建一个 C 源程序。

2. 新建一个 C 源程序

单击 File，选择 New。在新建窗口中，类似新建工程操作（如图 0.8 所示），在 Files

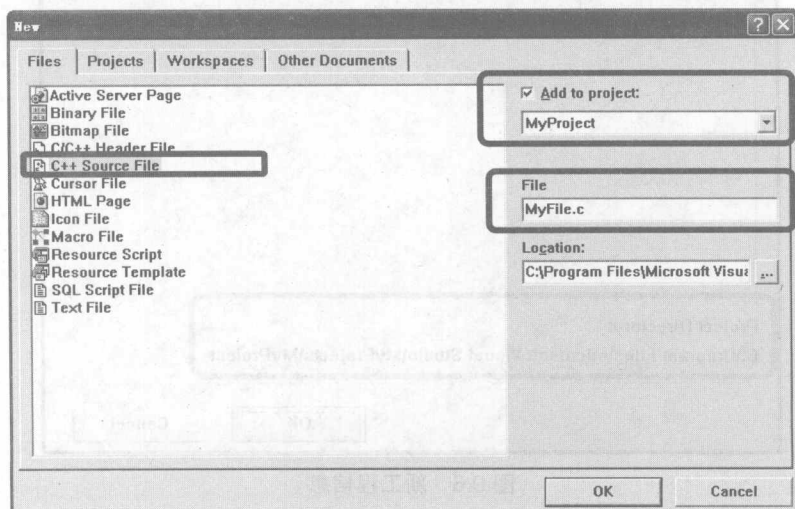


图 0.8 新建 C 源程序

标签选择 C++ Source File; 在 Add to Project 前打钩, 选择添加到刚新建的工程 MyProject; 在 File 输入框中, 输入后缀名是.c 的文件名, 单击 OK 按钮。

注意, 如果在文件名中不显式地输入扩展名.c, 则 VC++ 将为文件附上默认扩展名.cpp, 并按照 C++ 语言的语法进行检查。由于 C++ 的语法检查要比 C 语言的语法更为严格, 因此, 建议各位还是显式地输入文件扩展名.c。

单击 OK 按钮后, 可以进入 MyFile.c 的编辑界面, 如图 0.9 所示。在窗口的标题栏和工程窗口中都显示出了当前要编辑的文件名字 MyFile.c。

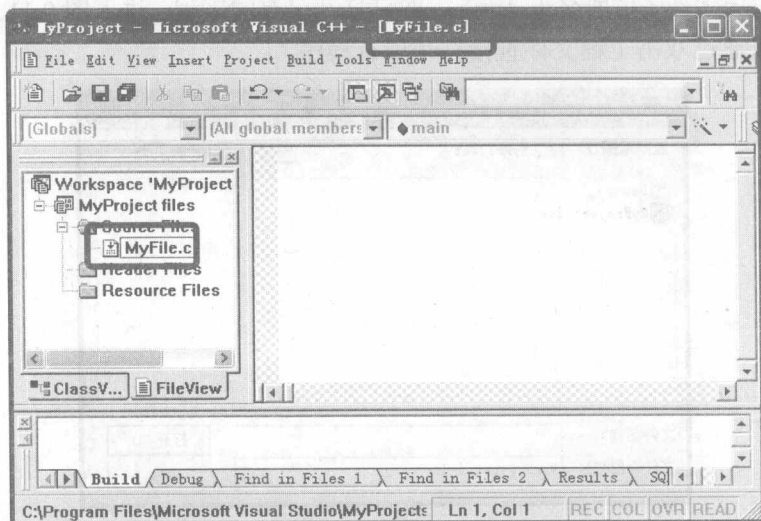


图 0.9 MyFile.c 编辑窗口

此时, 程序编辑窗口被激活, 可以输入和编辑源程序了。这里, 我们编写了一个简单的 C 程序, 输出 Hello World!, 如图 0.10 所示。

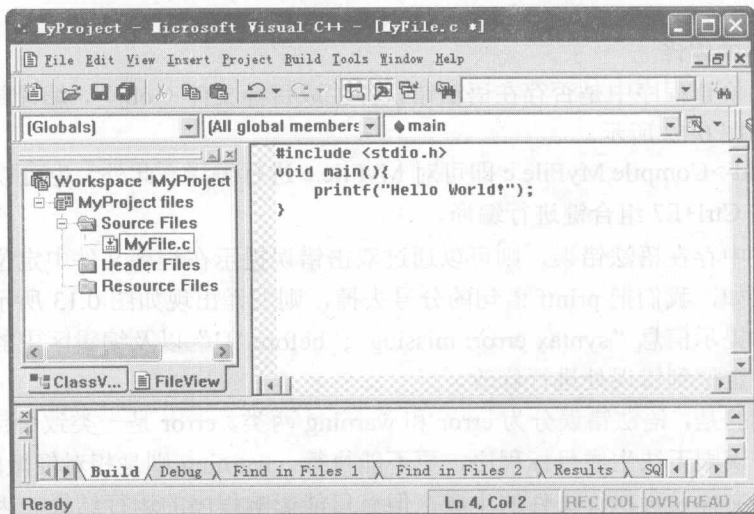


图 0.10 MyFile.c 的编辑

进入编辑状态后, 如果对源程序进行了修改且未保存则在标题栏中文件名字后面会出现 “*” 提示。选择 File->Save, 单击  按钮或者使用快捷键 Ctrl+S 都可以保存文件。保存之后, 标题栏中的 “*” 消失。

如果不想将源程序保存到指定的文件中, 可以选择 File->Save As 项, 重新指定文件保存的位置以及文件名字。

3. 打开一个现存程序

与新建的操作类似, 可以通过 File->Open Workspace 项进入打开工程窗口, 如图 0.11 所示。双击工程文件 (扩展名为 .dsw), 即可打开已有的工程, 进入图 0.11 所示的界面。此外, 也可以通过双击工程文件直接打开现有程序。

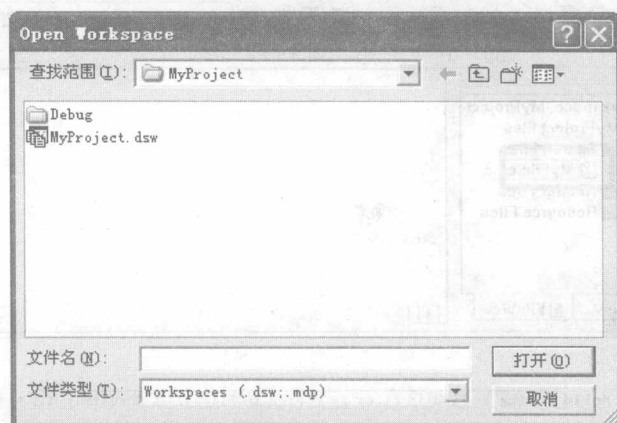


图 0.11 打开工程窗口

0.3 编译、链接和运行 C 程序

1. C 程序的编译

编译可以检查程序中是否存在语法错误并生成目标文件 (.obj), 编译结果显示在输出窗口中, 如图 0.12 所示。

选择 Build->Compile MyFile.c 即可对 MyFile.c 进行编译。此外, 也可以通过单击  按钮或直接按 Ctrl+F7 组合键进行编译。

如果程序中存在语法错误, 则可以通过双击错误提示在程序文件中定位到错误所在的代码行。例如, 我们把 printf 语句的分号去掉, 则编译出现如图 0.13 所示的结果。根据提示框中的提示信息 “syntax error: missing ‘;’ before ‘}’” 以及编辑区中的蓝色提示箭头则可以直接定位到错误处进行修改。

值得注意的是, 语法错误分为 error 和 warning 两类。error 是一类致命错误, 程序中如果有此类错误则无法生成目标程序, 更不能执行。warning 则是相对轻微的一类错误, 不会影响目标文件及可执行文件的生成, 但有可能影响程序的运行结果。因此, 建议最好把所有错误 (不论是 error 还是 warning) 都一一修正。

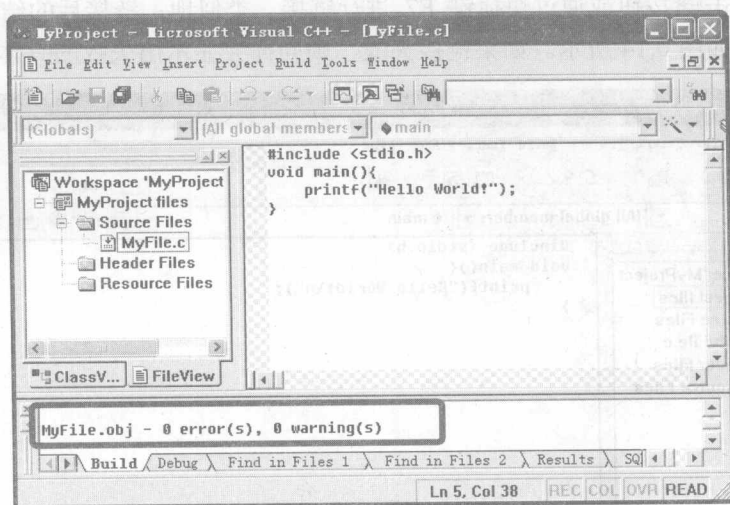


图 0.12 程序编译结果

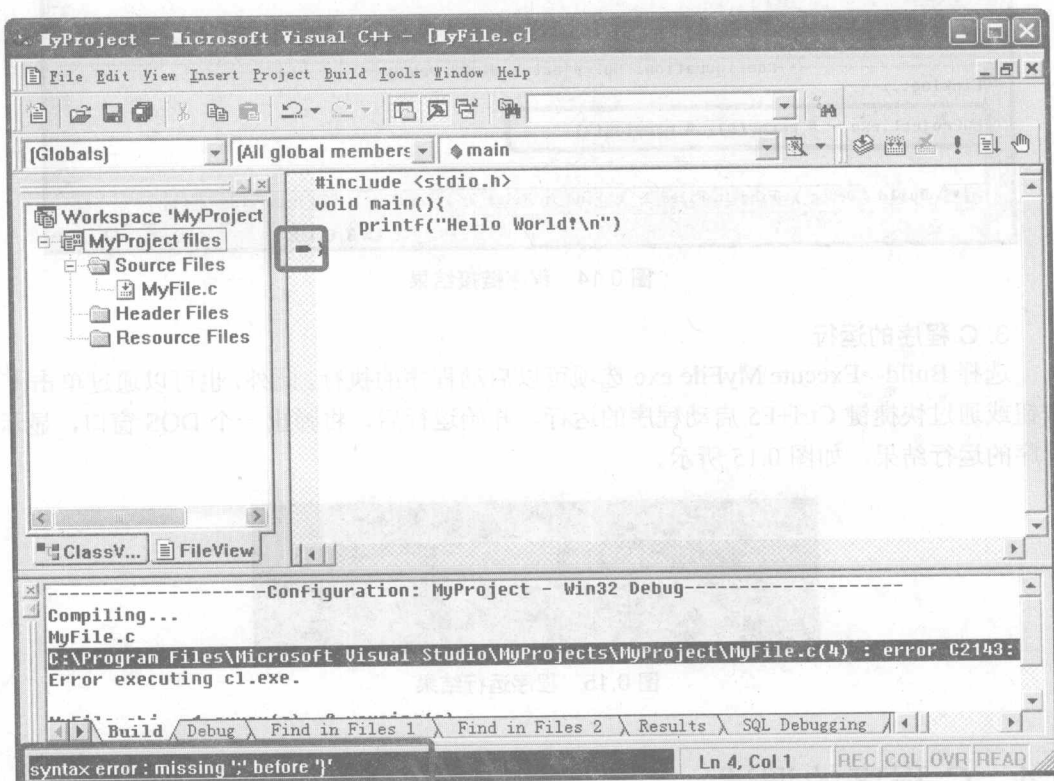


图 0.13 语法错误

2. C 程序的链接

链接将生成可执行文件(.exe)。选择 Build->Build 选项可以完成程序的连接。此外，

也可以通过单击  按钮或通过快捷键 F7 进行链接。类似地，链接后的结果也会出现在输出窗口中，如图 0.14 所示。如果链接失败，则同样会显示出失败的具体原因。

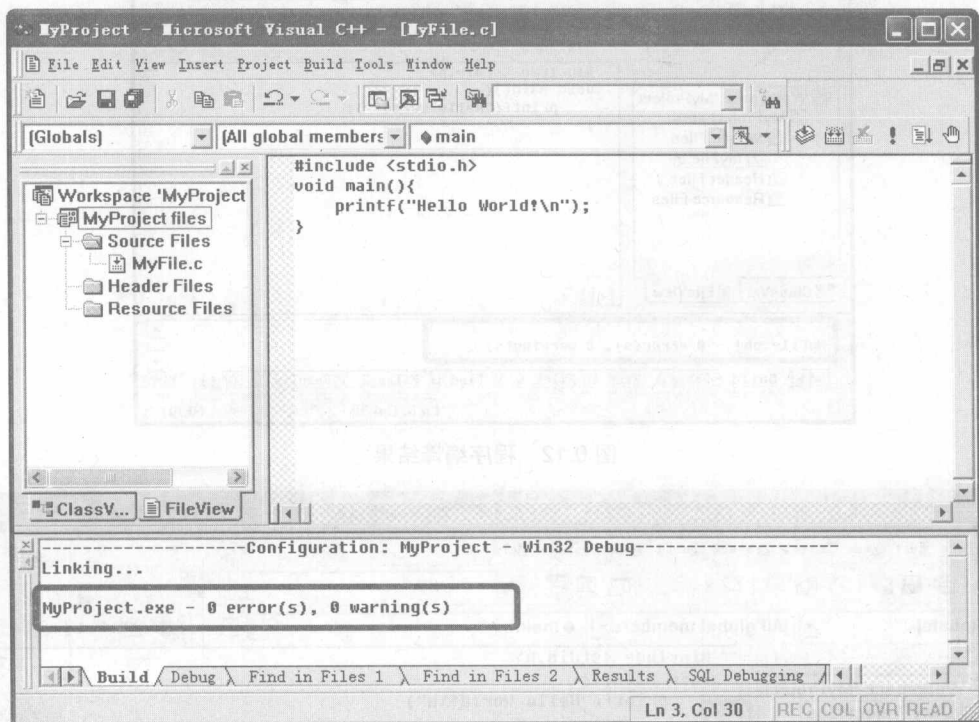


图 0.14 程序链接结果

3. C 程序的运行

选择 Build->Execute MyFile.exe 选项可以启动程序的执行。此外，也可以通过单击  按钮或通过快捷键 Ctrl+F5 启动程序的运行。开始运行后，将弹出一个 DOS 窗口，显示程序的运行结果，如图 0.15 所示。

```
Hello World!  
Press any key to continue.
```

图 0.15 程序运行结果

0.4 C 程序的调试

调试是为了发现程序中的错误，包括语法错误和逻辑错误。其中，语法错误能够在编译的过程中发现并修改；而逻辑错误往往无法直观地被发现，即程序通常能够被成功地编译和连接甚至执行，然而其执行结果却与预期结果不一致。逻辑错误的调试是比较

困难的，因此，一般的程序开发环境都会提供完整的程序调试工具。本节将主要介绍如何使用 Visual C++ 6.0 中的调试工具发现程序中的逻辑错误。

1. 设置断点

当需要调试程序的时候，可以首先大致判断出程序中可能从哪条语句开始出现问题，并将光标移动到该语句行，单击  按钮，则在语句行左侧出现一个红点，称为断点，如图 0.16 所示。此后，程序调试运行过程中遇到断点时，将自动暂停执行，进入调试状态。

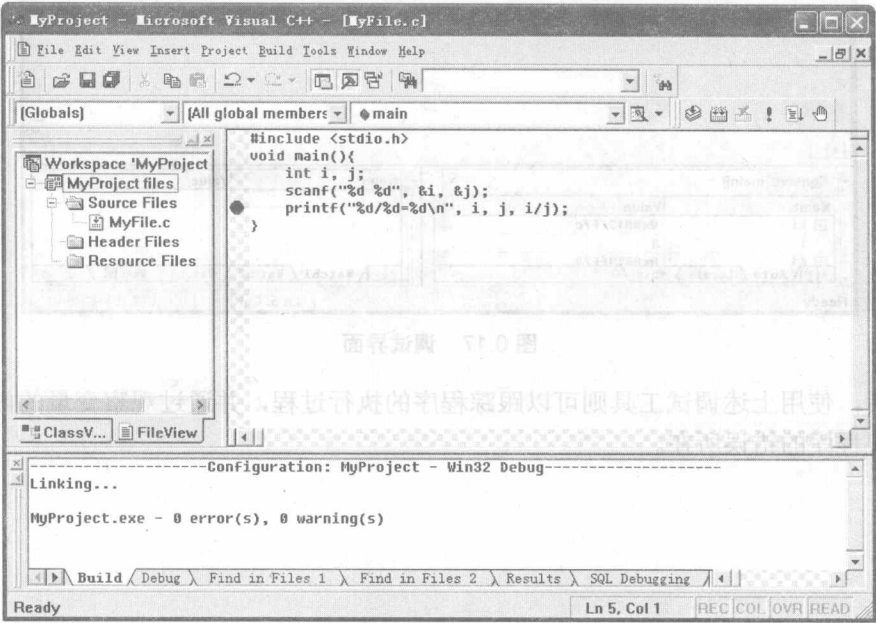


图 0.16 设置断点

2. 进入调试界面

设置好断点之后，单击  按钮，以调试的方式启动程序的运行。当运行到断点所在的语句时，将直接进入调试状态，如图 0.17 所示。代码行左侧的黄色小箭头表示了程序的当前执行位置。调试工具条中的按钮是常用的调试工具按钮，表 0.1 中列举了常用按钮的功能。屏幕下方左右两个窗口则分别是自动变量框和观察框，可以看到变量的当前值，作为判断程序是否出错的参考。其中，观察框支持对各种变量值的观察，在 Name 列输入变量的名字，则在 Value 列就会显示出该变量的当前值。

表 0.1 调试工具条常用按钮说明

按 钮	功 能	按 钮	功 能
	重新开始调试		停止/退出调试
	跟踪到函数中		逐个函数调试
	从跟踪的函数中跳出		运行到光标处

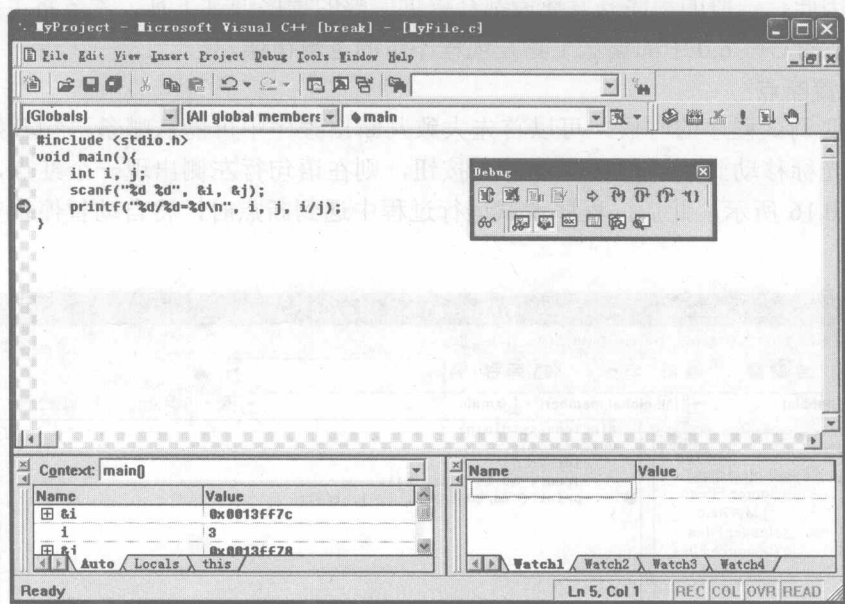


图 0.17 调试界面

此后，使用上述调试工具则可以跟踪程序的执行过程，并通过观察变量的内容变化及时发现程序的错误所在。