



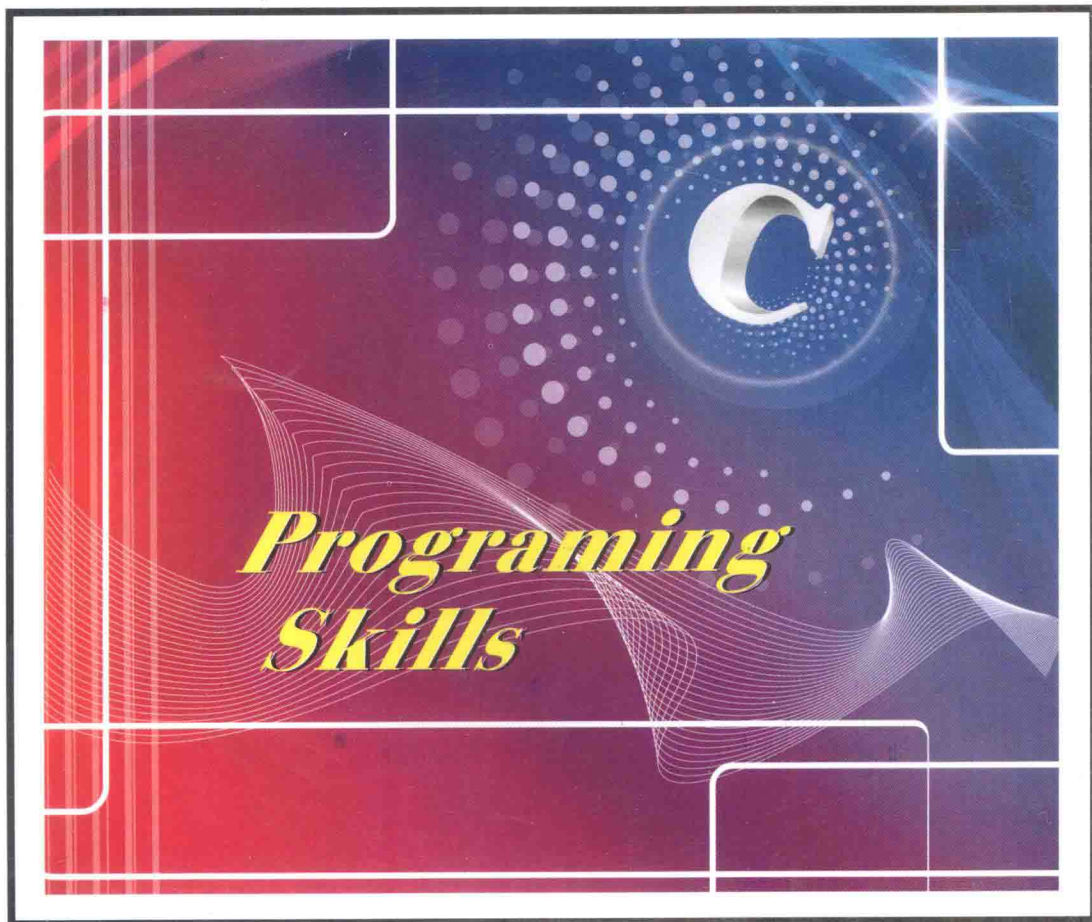
高等学校计算机类“十三五”规划教材
COMPUTER

省级精品课程配套教材

C语言程序设计

实践教程

主编 张卫国 朱宁洪
主审 李占利



西安电子科技大学出版社
<http://www.xduph.com>

高等学校计算机类“十三五”规划教材

省级精品课程配套教材

C 语言程序设计 实践教程

主 编 张卫国 朱宁洪

主 审 李占利

内 容 简 介

本书包括三大章和六个附录。第一章介绍了 VC++6.0 调试 C 语言程序的基础知识和方法,其目的是使读者掌握调试 C 语言程序的基本方法和技能,主要内容包括:VC++6.0 的安装、VC++6.0 的界面介绍、VC++6.0 错误类型及其查询方法、建立和运行 C 语言程序的方法以及 VC++6.0 调试方法及其工具的使用。第二章介绍了 C 语言程序设计实验,这一部分内容包括 VC++6.0 的运行环境、数据类型与表达式、三种基本结构、数组、函数、结构体与共用体、编译预处理、文件、指针的内容分别放在各个实验中讲述,便于将指针的应用与 C 语言的基本知识结合起来。第三章提供了与相应实验章节对应的习题与答案分析,供读者参考。附录部分分别介绍了 C 语言的学习方法、常见的编程错误、常用的库函数,以及全国计算机等级考试二级考试大纲和近年的笔试与机试真题和答案。

本书根据“夯实基础、面向应用、培养创新”的指导思想,加强了教材的基础性、应用性和创新性,注重培养读者程序设计的思维方式和技巧,为后续课程打下坚实基础。

本书适合作为高等院校 C 语言程序设计课程的上机教材,也适合程序设计初学者自学使用,还可作为全国计算机等级考试应试者的参考书。

图书在版编目(CIP)数据

C 语言程序设计实践教程/张卫国,朱宁洪主编. —西安:西安电子科技大学出版社,2015.9

高等学校计算机类“十三五”规划教材

ISBN 978-7-5606-3845-4

I. ① C… II. ① 张… ② 朱… III. ① C 语言—程序设计—高等学校—教材 IV. ① TP312

中国版本图书馆 CIP 数据核字(2015)第 203006 号

策 划 李惠萍

责任编辑 许青青 王文秀

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfb001@163.com

经 销 新华书店

印刷单位 陕西大江印务有限公司

版 次 2015 年 9 月第 1 版 2015 年 9 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 10.75

字 数 251 千字

印 数 1~4000 册

定 价 19.00 元

ISBN 978-7-5606-3845-4/TP

XDUP 4137001-1

如有印装问题可调换

前 言

程序设计是高等院校重要的计算机基础课程，它以编程语言为平台，介绍程序设计的思想和方法。通过该课程的学习，学生不仅可掌握高级程序设计语言的相关知识，更重要的是可在实践中逐步掌握程序设计的思想和方法，培养其应用程序语言解决实际问题的能力。

C 语言是一门实践性很强的课程，该课程的学习有其自身的特点。学习者必须通过大量的编程训练，在实践中掌握程序设计语言，培养程序设计的基本能力，并逐步理解和掌握程序设计的思想和方法。因此，C 语言程序设计的教学重点应该是培养学生的实践编程能力，教材也要适应这种要求。

目前，计算机等级考试中使用 VC++6.0，为了适应等级考试的需要，本书在编写过程中，程序编译调试的工具采用 VC++6.0，同时在附录中介绍了等级考试的大纲和真题，供学生参加等级考试时参考；同时，编者在教学实践中发现学生调试程序的能力较弱，因此在书稿编写过程中注重加强学生程序调试能力的培养。

全书共包括三章和六个附录。

第一章介绍了 VC++6.0 调试 C 语言程序的基础知识和方法，其目的是使读者掌握调试 C 语言程序的基本方法和技能。本章内容主要包括：VC++6.0 的安装、VC++6.0 的界面介绍、VC++6.0 错误类型及其查询方法、建立和运行 C 语言程序的方法以及 VC++6.0 调试方法及其工具的使用。这一部分内容教师可以根据学生的情况和课时的情况，重点讲授或让学生自学。

第二章介绍了 C 语言程序设计实验，这一部分内容包括 VC++6.0 的运行环境、数据类型与表达式、三种基本结构、数组、函数、结构体与共用体、编译预处理、文件，指针的内容分别放在各个实验中讲述，便于将指针的应用与 C 语言的基本知识结合起来。每一个实验都提供了实验目的和实验需要掌握的基本语法知识，同时根据所给实验的内容提供一些经典算法和应用实例供读者参考，并对相应示例进行了注释，实验包括改错、注释、比较、填空、分析、编程等多种类型。

第三章提供了与相应实验章节对应的习题与答案分析，供读者参考，多数编程答案都逐行添加了注释，方便读者理解。

附录部分分别介绍了 C 语言的学习方法、常见的编程错误、常用的库函数，

以及全国计算机等级考试二级考试大纲和近年的笔试与机试真题和答案。

书中所用到的所有示例代码均在 VC++6.0 中调试通过，使用者可以直接将代码输入 VC++6.0 中编译运行。书中很多例题都很经典，是作者精心选编的，这些范例提出并解决了很多常见问题。读者完成这些练习题并理解程序的思路，将有助于开阔眼界、丰富编程知识，学会如何解决程序设计的实际问题。

本书是陕西省精品资源共享课程建设的一部分。本书由西安科技大学计算机学院张卫国老师和朱宁洪老师编著。其中，李占利老师编写了第一章，张卫国老师编写了第二章，朱宁洪老师编写了第三章和附录 6，王昱哲老师编写了附录 1~5，全书由李占利老师审定。

由于编者水平所限，书中难免有不足之处，敬请读者指正。

编 者

2015 年 8 月

目 录

第一章 C/C++实验基础知识.....	1
1.1 C/C++语言实验环境配置	1
1.2 Visual C++6.0 的使用	1
1.2.1 Visual C++介绍	1
1.2.2 Visual C++的安装和启动	2
1.2.3 常用功能键及其意义	2
1.2.4 输入和编辑源程序	3
1.2.5 编译、连接和运行	6
1.2.6 建立和运行包含多个文件的程序	10
1.3 Visual C++6.0 工程相关文件介绍.....	16
1.4 Visual C++6.0 错误信息.....	16
1.4.1 程序错误的类型	16
1.4.2 错误信息的查询	18
1.5 Visual C++6.0 调试工具的使用.....	19
1.5.1 设置工程的调试属性	20
1.5.2 学习设置断点进行单步调试	20
第二章 C 语言程序设计实验	25
实验 1 VC++运行环境	25
实验 2 数据类型	29
实验 3 顺序结构	34
实验 4 选择结构	38
实验 5 循环结构	44
实验 6 数组	51
实验 7 函数	57
实验 8 结构体和共用体	65
实验 9 编译预处理	69
实验 10 文件	71
第三章 C 语言程序设计习题及参考答案.....	73
习题一及参考答案	73
习题二及参考答案	74
习题三及参考答案	76
习题四及参考答案	90

习题五及参考答案	101
习题六及参考答案	112
习题七及参考答案	121
习题八及参考答案	125
 附录 1 C 语言程序设计实验报告格式	 134
附录 2 C 语言学习指南	135
附录 3 C 语言常见编程错误	137
附录 4 C 语言常用的库函数	145
附录 5 全国计算机等级考试二级 C 语言程序设计考试大纲	149
附录 6 全国计算机等级考试二级 C 语言笔试与机试真题	152

第一章 C/C++实验基础知识

1.1 C/C++语言实验环境配置

C/C++语言实验环境配置包括硬件配置和软件配置。

1. 硬件配置

586 以上 PC 兼容机或品牌机, 配有彩色显示器、鼠标、键盘, 内存不小于 20 MB, 硬盘自由空间不少于 60 MB。推荐配置为内存 32 MB 或 64 MB(或以上), 硬盘自由空间为 500 MB 以上。

2. 软件配置

(1) 操作系统: Windows 98、Windows 2000、Windows XP、Linux、UNIX。

(2) 集成开发环境:

① 在 Windows 98、Windows 2000、Windows XP 系统下, 主要的开发编译环境有 Turbo C/C++ 3.0、Borland C++ 3.1、Microsoft Visual C++6.0、DJGPP, 其中 DJGPP 是 GCC 在 DOS/Windows 操作系统下的移植。为配合全国计算机等级考试的要求, 本实验指导书使用的开发环境为 Microsoft Visual C++6.0 英文版(汉化版本只是把部分菜单汉化了且用词很不准确)。

② 在 Linux、UNIX 系统下, 采用 GCC 编译环境。本书不涉及相关内容, 有兴趣的读者可以自行学习。

1.2 Visual C++ 6.0 的使用

1.2.1 Visual C++介绍

Visual C++是 Microsoft 公司的 Visual Studio 开发工具箱中的一个 C/C++程序开发包。Visual Studio 提供了一整套开发 Internet 和 Windows 应用程序的工具, 包括 Visual C++、Visual Basic、Visual FoxPro、Visual InterDev、Visual J++以及其他辅助工具, 如代码管理工具 Visual SourceSafe 和联机帮助系统 MSDN。Visual C++包中除包括 C++编译器外, 还包括所有的库、例子和为创建 Windows 应用程序所需要的文档。

从最早期的 1.0 版本到 6.0 版本, 再到最新的.NET 版本, Visual C++已经有了很大的变化, 在界面、功能、库支持方面都有所增强。6.0 版本在编译器、MFC 类库、编辑器以及

联机帮助系统等方面都在以前版本的基础上做了较大改进。

Visual C++一般分为三个版本：学习版、专业版和企业版，不同的版本适合于不同类型的应用开发。实验中可以使用这三个版本中的任意一种。

1.2.2 Visual C++的安装和启动

如果计算机未安装 Visual C++6.0，则应先安装 Visual C++6.0。Visual C++是 Visual Studio 的一部分，因此需要 Visual Studio 的光盘，执行其中的 setup.exe，按屏幕上的提示安装即可。安装完毕后，在 Windows 的“开始”菜单的“程序”子菜单中会出现“Microsoft Visual Studio”子菜单，如图 1-1 所示。



图 1-1 Microsoft Visual Studio 启动子菜单

当需要使用 Visual C++时，只需从桌面上顺序选择开始→程序→Microsoft Visual Studio→Visual C++6.0 命令即可，此时屏幕上在短暂显示 Visual C++6.0 的版权页面后，出现 Visual C++6.0 的由窗口、工具条、菜单、工具及其他部分组成的主界面。通过这个界面，用户可以创建、测试、调试应用程序，如图 1-2 所示。



图 1-2 Visual C++6.0 界面

1.2.3 常用功能键及其意义

为了使程序员能够方便快捷地完成程序开发，开发环境提供了大量快捷方式来简化一些常用操作的步骤。键盘操作直接、简单，非常方便，因而程序员非常喜欢使用键盘命令来控制操作。下面给出的表 1-1 是一些最常用的功能键，希望读者在实验中逐步掌握。

表1-1 VC++6.0最常用的功能键一览表

操作类型	功能键	对应菜单	含 义
文件操作	Ctrl+N	File New	创建新的文件、项目等
	Ctrl+O	File Open	打开项目、文件等
	Ctrl+S	File Save	保存当前文件
编辑操作	Ctrl+X	Edit Cut	剪切
	Ctrl+C	Edit Copy	复制
	Ctrl+V	Edit Paste	粘贴
	Ctrl+Z	Edit Undo	撤消上一个操作
	Ctrl+Y	Edit Redo	重复上一个操作
	Ctrl+A	Edit Select All	全选
	Del	Edit Del	删除光标后面的一个字符
建立程序 操作	Ctrl+F7	Build Compiler current file	编译当前源文件
	Ctrl+F5	Build Run exe	运行当前项目
	F7	Build Build exe	建立可执行程序
	F5	Build Start Debugging	启动调试程序
调试	F5	Debug Go	继续运行
	F11	Debug Step into	进入函数体内部
	Shift+F11	Debug Step out	从函数体内部运行出来
	F10	Debug Step over	执行一行语句
	F9		设置/清除断点
	Ctrl+F10	Debug Run to Cursor	运行到光标所在位置
	Shift+F9	Debug Quick Watch	快速查看变量或表达式的值
	Shift+F5	Debug Stop Debugging	停止调试

上述这些快捷键命令都有对应的菜单和工具条按钮，如图 1-3 所示。

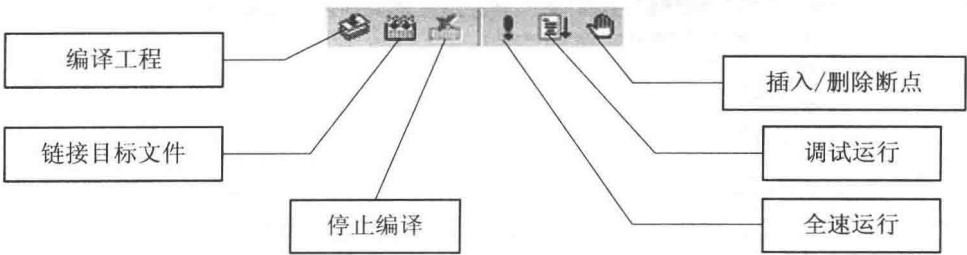


图 1-3 快捷键命令菜单与按钮

1.2.4 输入和编辑源程序

由于 C 语言是 C++的子集，C++完全兼容 C 语言，在 VC++6.0 中，调试 C 语言程序就是使用调试 C++程序的框架。在本书中以后除特别声明外，讲 C 语言或 C++都是同一个概念。

先介绍最简单的情况,即程序只由一个源程序文件组成,亦即单文件程序(有关对多文件程序的操作在本章稍后部分介绍)。

1. 新建 C++源程序

如果要新建一个 C++源程序,可采取以下步骤:

在 Visual C++主窗口的主菜单栏中选择 File(文件)命令,然后选择 New(新建)命令,如图 1-4 所示。这时,展幕上出现一个 New(新建)对话框,单击此对话框上方的 Files(文件)属性页,在列表中选择 C++ Source File 项,表示要建立新的 C++源程序文件,然后在对话框右半部分的 Location(目录)文本框中输入准备编辑的源程序文件的存储路径(假设为 D:\VC++6.0\测试程序\其他测试\demo),表示准备编辑的源程序文件将存放在“D:\VC++6.0\测试程序\其他测试\demo”子目录下。在其上方的 File(文件)文本框中输入准备编辑的源程序文件的名字(输入 MyFirstC),见图 1-5。这样,即将进行输入和编辑的源程序就以 MyFirstC.cpp 为文件名存放在“D:\VC++6.0\测试程序\其他测试\demo”目录下,见图 1-6。当然,读者完全可以指定其他路径名和文件名。

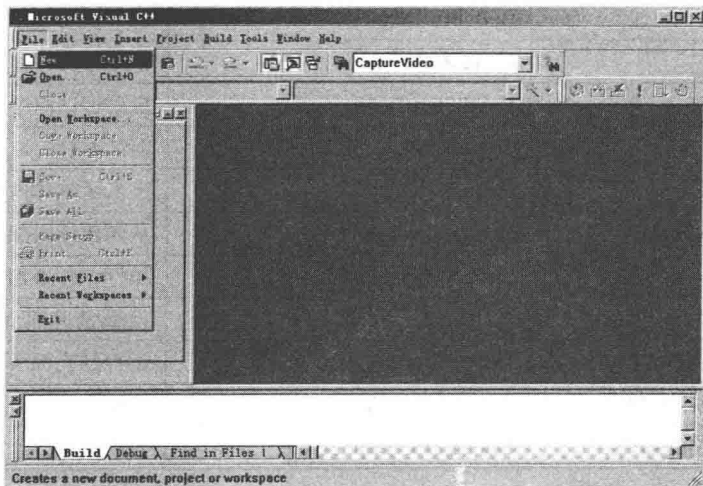


图 1-4 选择 File 菜单中的 New 操作

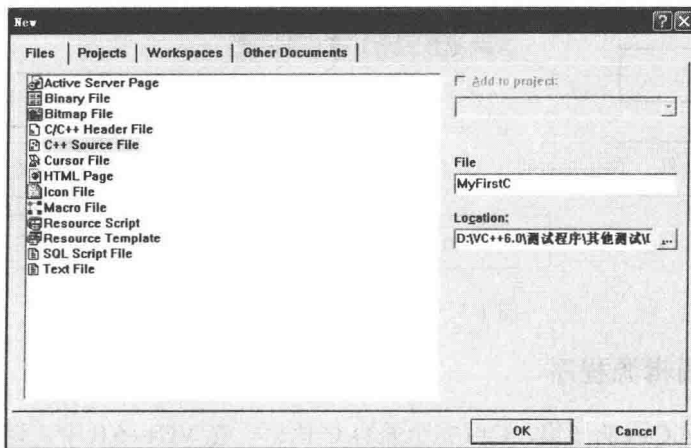


图 1-5 创建工程文件

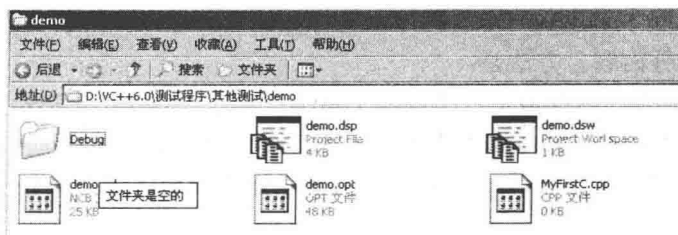


图 1-6 在指定的目录下建立的源文件

在单击 OK 按钮后, 回到 Visual C++主窗口, 由于在前面已指定了路径(D:\VC++6.0\测试程序\其他测试\demo)和文件名(MyFirstC.cpp), 因此在窗口的标题栏中显示出文件名(MyFirstC.cpp)。可以看到光标在程序编辑窗口闪烁, 表示程序编辑窗口已激活, 可以输入和编辑源程序了。这里输入如图 1-7 窗口中所示的程序。检查无误后, 将源程序保存在前面指定的文件中, 方法是: 在主菜单栏中选择 File(文件)命令, 并在其下拉菜单中选择 Save(保存)命令。也可以用快捷键 Ctrl+S 来保存文件。如果不想将源程序存放到原先指定的文件中, 可以不选择 Save 命令, 而选择 Save As(另存为)命令, 并在弹出的 Save As 对话框中指定文件路径和文件名。

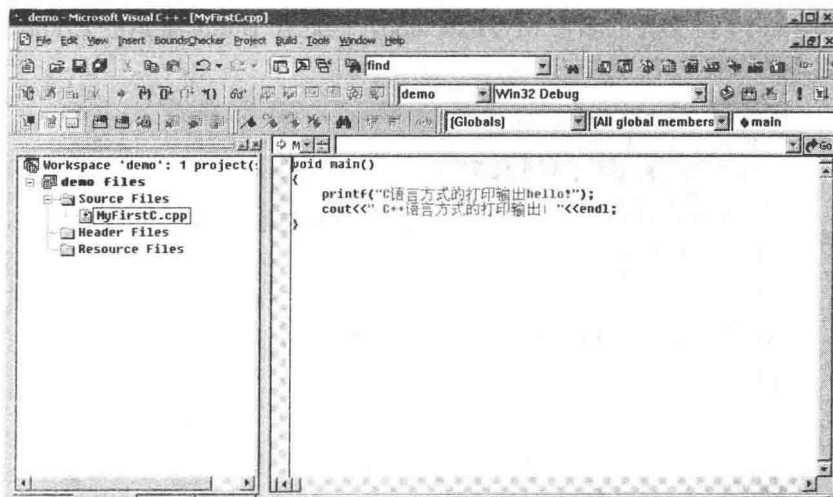


图 1-7 C++程序的输入

2. 打开一个已有的程序

如果已经编辑并保存过 C++源程序, 而希望打开所需要的源程序文件, 并对它进行修改, 方法是:

- (1) 在“资源管理器”或“我的电脑”中按路径找到已有的 C++程序名(如 C1-1.cpp)。
- (2) 双击此文件名, 则进入 Visual C++集成环境, 并打开了该文件, 程序将显示在编辑窗口中。也可以用 Ctrl+O 或单击工具栏中的小图标来打开文件。

(3) 如果修改后仍想保存在原来的文件中, 可以选择 File(文件)→Save(保存)命令, 或用快捷键 Ctrl+S, 或单击工具栏中的小图标来保存文件。

3. 通过已有程序建立新程序

如果已编辑并保存过 C++源程序(而不是第一次在该计算机上使用 Visual C++), 则可

以通过一个已有的程序来建立一个新程序，这样做比重新输入一个新文件方便，因为可以利用原有程序的部分内容。方法是：

(1) 打开任何一个已有的源文件(如 cl-1.cpp)。

(2) 利用该文件的基础修改成新的文件，然后通过选择 File(文件)→Save as(另存为)命令将它以另一文件名另存(如以 test.cpp 名字另存)，这样就生成了一个新文件(test.cpp)。

用这种方法很方便，但应注意：

① 保存新文件时，不要错用 File→Save(保存)操作，否则原有文件(cl-1.cpp)的内容就被修改了。

② 在编译新文件前，应首先选择 File(文件)→Close Workspace(关闭工作区)命令将原有的工作区关闭，以免新文件在原有的工作区进行编译。

1.2.5 编译、连接和运行

1. 程序的编译

在编辑和保存了源文件以后，需要对该源文件进行编译。单击主菜单栏中的 Build(编译)命令，在其下拉菜单中选择 Compile MyFirstC.cpp(编译 MyFirstC.cpp)命令。由于建立(或保存)文件时已指定了源文件的名称 MyFirstC.cpp，因此在 Build 菜单的 Compile 命令中显示了现在要编译的源文件名 MyFirstC.cpp，见图 1-8。

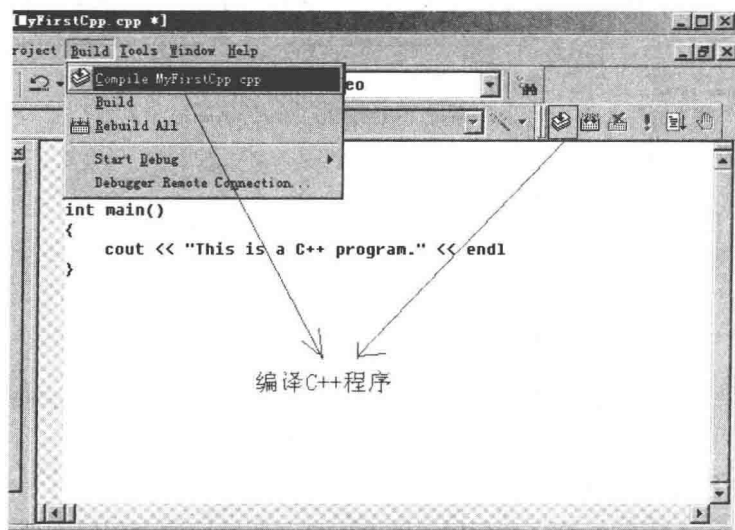


图 1-8 编译 C++ 程序

在选择“编译”命令后，如果屏幕上出现一个对话框，内容是“This build command requires an active project workspace. Would you like to create a default project workspace?”(此编译命令要求一个有效的项目工作区。你是否同意建立一个默认的项目工作区?)，见图 1-9。单击是(Y)按钮，表示同意由系统建立默认的项目工作区，然后开始编译。也可以不用选择菜单的方法，而用 Ctrl+F7 或小图标来完成编译。

在进行编译时，编译系统检查源程序中是否有语法错误，然后在主窗口下部的调试信息窗口输出编译的信息，如果有错，就会指出错误的位置和性质，见图 1-10。

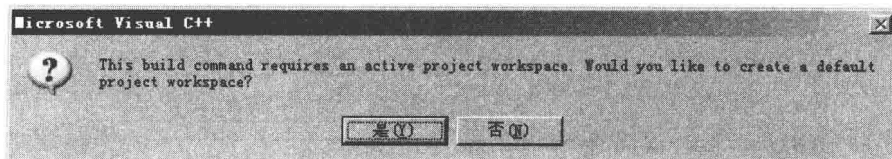


图 1-9 建立默认工作区提示

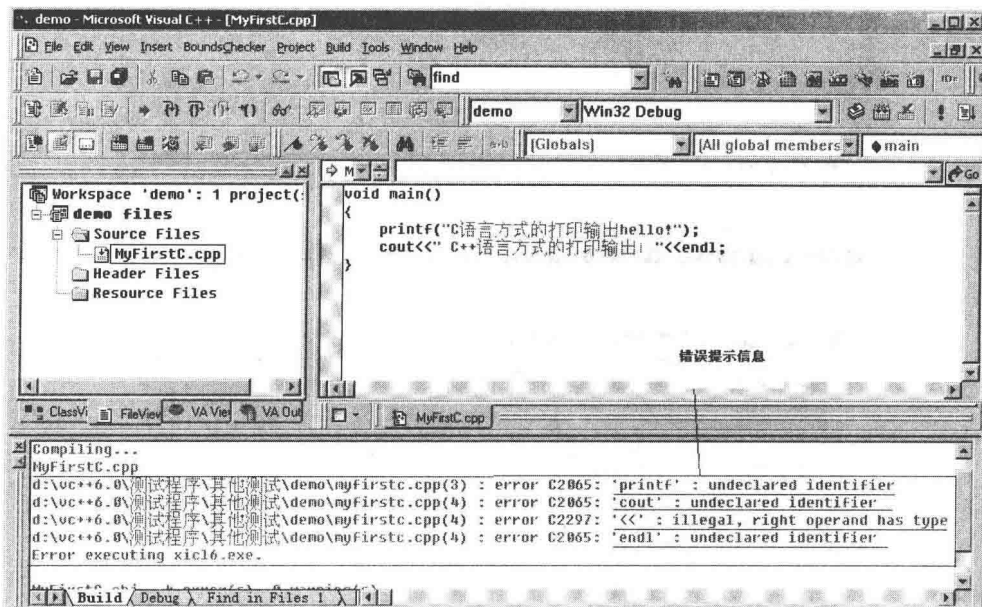


图 1-10 编译错误提示信息

2. 程序错误的修改

修改程序错误的任务是发现和改正程序中的错误，使程序能正常运行。编译系统能检查出程序中的语法错误。语法错误分两类：一类是致命错误，以 **error** 表示，如果程序中有这类错误，就无法通过编译，不能形成目标程序，更谈不上运行了。另一类是轻微错误，以 **warning**(警告)表示，这类错误不影响生成目标程序和可执行程序，但有可能影响运行的结果。因此也应当尽量改正，使程序既无 **error**，又无 **warning**。

从图 1-10 中的编译输出窗口中可以看到编译的信息，指出源程序有 **error** 和 **warning**。有时候明明是第 6 行有错，怎么在报错时说成是第 7 行的错呢？这是因为 C++ 允许将一个语句分写成几行，因此检查完第 6 行末尾无分号时还不能判定该语句有错，必须再检查下一行，直到发现第 7 行的“}”前都没有“;”，才能判定出错，因此在第 7 行报错。在根据编译报错信息检查程序时，应检查其上下行。

进行改错时，双击编译输出窗口中的某个报错信息的第 1 行，光标就自动移到程序窗口中被报错的程序行，并用粗箭头指向该行，见图 1-11。

根据提示信息“**error C2065: 'printf': undeclared identifier**”，判断没有加入预编译头文件 **stdio.h**，根据提示信息“**error C2065: 'cout': undeclared identifier**”，判断没有加入头文件 **iostream.h**。

修改后如图 1-12 所示，增加编译头文件 **stdio.h** 和 **iostream.h**。

3. 程序的连接

在得到目标程序后,就可以对程序进行连接了。此时应选择 Build(构建)→Build MyFirstC.exe(构建 MyFirstC.exe)命令,见图 1-14。表示要求连接并建立一个可执行文件 MyFirstC.exe。

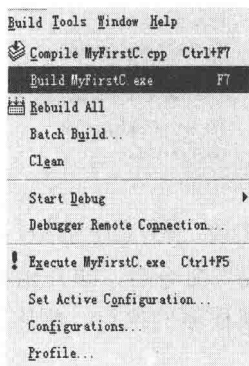


图 1-14 连接建立可执行文件

在执行连接后,在调试输出窗口显示连接时的信息,说明没有发现错误,生成了一个可执行文件 MyFirstC.exe,见图 1-15。

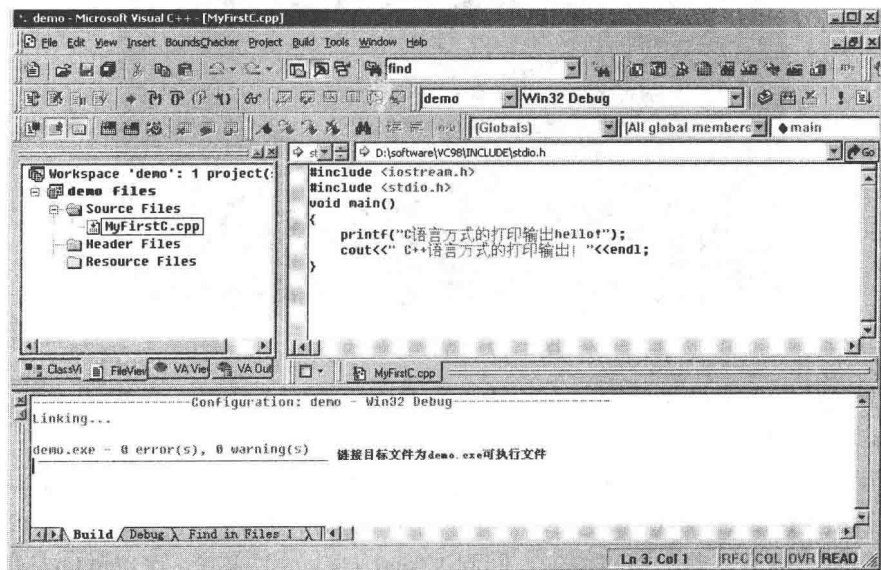


图 1-15 可执行文件建立成功

以上介绍的分别是进行程序的编译与连接,也可以选择 Build→命令(或按 F7 键)一次完成编译与连接。对于初学者来说,还是建议分步进行程序的编译与连接,这样有利于对程序编译、连接过程的理解。待积累一定经验后,可以一步完成编译与连接。

4. 程序的执行

在得到可执行文件 MyFirstC.exe 后,就可以直接执行 MyFirstC.exe 了。选择 Build→!Execute MyFirstC.exe(执行 MyFirstC.exe)命令,见图 1-16。

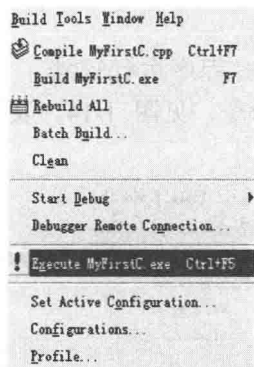


图 1-16 运行程序可执行文件

在选择“! Execute MyFirstC.exe”命令后，即开始执行 MyFirstC.exe。也可以不通过选择菜单命令，而使用 Ctrl+F5 来实现程序的执行。程序执行后，屏幕切换到输出结果的窗口，显示出运行结果，见图 1-17。



图 1-17 程序运行结果

可以看到，在输出结果的窗口中的第 1 行是程序的输出结果：C 语言方式的打印输出 hello! C++语言方式的打印输出！

第 2 行“Press any key to continue”并非程序所指定的输出，而是 Visual C++ 在输出完运行结果后由 Visual C++6.0 系统自动加上的一行信息，通知用户“按任何一键以便继续”。当你按下任何一键后，输出窗口消失，回到 Visual C++ 的主窗口，可以继续对源程序进行修改、补充或进行其他工作。

如果已完成对一个程序的操作，并不再对它进行其他处理，应当选择 File(文件)→Close Workspace(关闭窗口)命令，以结束对该程序的操作。

1.2.6 建立和运行包含多个文件的程序

上面介绍的是最简单的情况，一个程序只包含一个源程序文件。如果一个程序包含多个源程序文件，则需要建立一个项目文件(project file)，在这个项目文件中包含多个文件(包括源文件和头文件)。项目文件是放在项目工作区(Workspace)中并在项目工作区的管理之下