



高等学校教材

C++ 语言程序设计 ——上机指导与习题解答

● 吕凤翥 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

URL: <http://www.phei.com.cn>

高等学校教材

C++语言程序设计

——上机指导与习题解答

吕凤翥 编著

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》的学习辅导书。主要介绍了 Visual C++ 6.0 编译系统的使用,在此系统基础上进行 C++ 语言的程序调试上机指导。本书着眼于教材中的重点难点知识分析以及习题解析,并为初学者指出了在 C++ 语言学习过程中需要注意的问题。

本书适合高等理工院校学生在学习 C++ 语言时作为参考,以帮助自学。

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,翻版必究。

图书在版编目(CIP)数据

C++ 语言程序设计:上机指导与习题解答/吕凤翥编著. - 北京:电子工业出版社,2001.9

高等学校教材

ISBN 7-5053-7029-4

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

中国版本图书馆 CIP 数据核字(2001)第 067146 号

丛 书 名: 高等学校教材

书 名: C++ 语言程序设计——上机指导与习题解答

编 著 者: 吕凤翥

责任编辑: 束传政

特约编辑: 赫 嘉

排版制作: 电子工业出版社计算机排版室

印 刷 者: 北京东光印刷厂

装 订 者: 三河市万和装订厂

出版发行: 电子工业出版社 URL: <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×1 092 1/16 印张: 9.25 字数: 236.8 千字

版 次: 2001 年 9 月第 1 版 2001 年 9 月第 1 次印刷

书 号: ISBN 7-5053-7029-4
TP·4030

印 数: 8 000 册 定价: 12.00 元

凡购买电子工业出版社的图书,如有缺页、倒页、脱页、所附磁盘或光盘有问题者,请向购买书店调换;
若书店售缺,请与本社发行部联系调换。电话 68279077

目 录

第1章 C++ 程序的实现	1
1.1 Visual C++ 6.0 编译系统部分功能介绍	1
1.2 C++单文件应用程序的实现	12
1.3 C++多文件应用程序的实现	13
第2章 上机练习指导	16
2.1 第1章上机练习指导	16
2.2 第2章上机练习指导	19
2.3 第3章上机练习指导	21
2.4 第4章上机练习指导	24
2.5 第5章上机练习指导	30
2.6 第6章上机练习指导	40
2.7 第7章上机练习指导	47
2.8 第8章上机练习指导	50
2.9 第9章上机练习指导	56
2.10 第10章上机练习指导	63
2.11 第11章上机练习指导	69
2.12 第12章上机练习指导	79
2.13 第13章上机练习指导	85
第3章 习题答案和难点分析	88
3.1 第1章习题答案和解答	88
3.1.1 习题答案	88
3.1.2 习题解答	89
3.2 第2章习题答案和解答	90
3.2.1 习题答案	90
3.2.2 习题解答	90
3.3 第3章习题答案和解答	92
3.3.1 习题答案	92
3.3.2 习题解答	92
3.4 第4章习题答案和解答	94
3.4.1 习题答案	94
3.4.2 习题解答	95
3.5 第5章习题答案和解答	97
3.5.1 习题答案	97
3.5.2 习题解答	98
3.6 第6章习题答案和解答	100
3.6.1 习题答案	100

3.6.2	习题解答	101
3.7	第7章习题答案和解答	106
3.7.1	习题答案	106
3.7.2	习题解答	106
3.8	第8章习题答案和解答	109
3.8.1	习题答案	109
3.8.2	习题解答	110
3.9	第9章习题答案和解答	113
3.9.1	习题答案	113
3.9.2	习题解答	115
3.10	第10章习题答案和解答	119
3.10.1	习题答案	119
3.10.2	习题解答	120
3.11	第11章习题答案和解答	122
3.11.1	习题答案	122
3.11.2	习题解答	123
3.12	第12章习题答案和解答	125
3.12.1	习题答案	125
3.12.2	习题解答	126
3.13	第13章习题答案和解答	127
3.13.1	习题答案	127
3.13.2	习题解答	127
第4章	面向对象方法的小结	129
4.1	面向对象方法的概括	129
4.2	实例说明	136

第 1 章 C++ 程序的实现

1.1 Visual C++ 6.0 编译系统部分功能介绍

主窗口

启动 Visual C++ 6.0 进入 Developer Studio 编译环境，如图 1.1 所示。

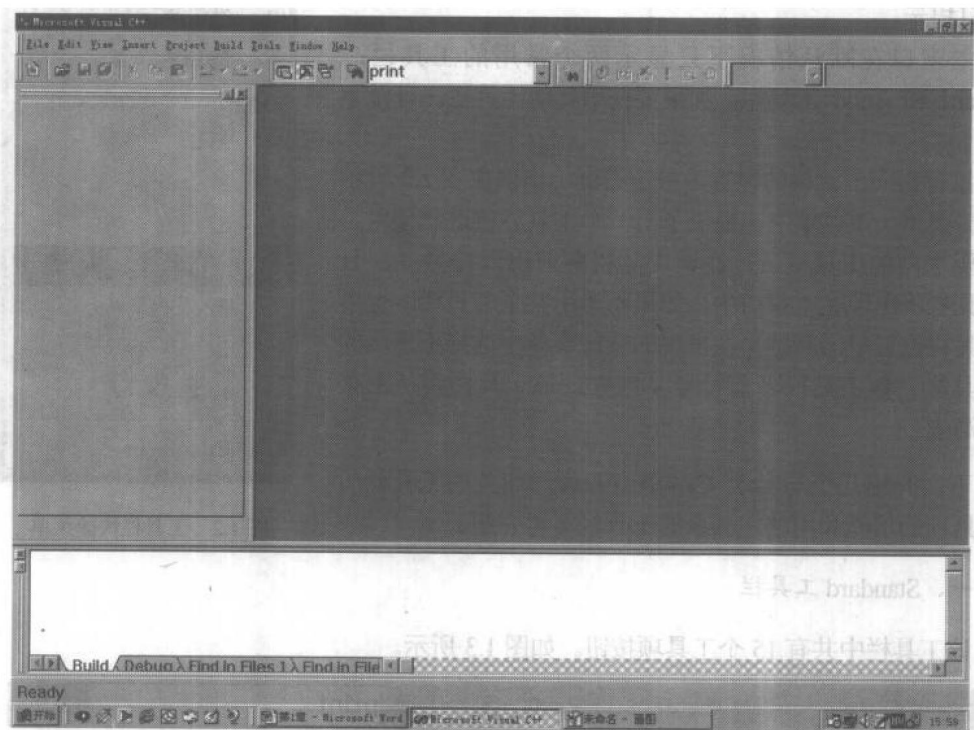


图 1.1 Visual C++ 6.0 编译环境窗口

主窗口由标题栏、菜单栏、工具栏、工作区窗口、源代码编辑窗口、输出窗口和状态栏组成。

屏幕窗口中最上方是标题栏，显示所打开的应用程序名。标题栏左端是控制菜单图标，单击后弹出窗口控制菜单。标题栏右端从左至右有三个控制按钮，分别为最小化、最大化(还原)和关闭按钮，可用它们快速设置窗口的大小。

标题栏下方是菜单栏，由 9 个菜单项组成。单击菜单项弹出下拉式菜单，可使用这些菜单项实现集成环境的各种功能。

菜单栏下方是工具栏，它由若干个功能按钮组成，单击按钮可实现某种操作功能。

工具栏的下方有左右两个窗口，左窗口是项目工作区窗口，右窗口是源代码编辑窗口。

在项目工作窗口和源代码编辑窗口的下方有一个输出窗口，在创建项目(Build)时，用来显示项目创建过程中的错误信息。

屏幕最底部是状态栏，它可给出当前操作或所选命令的提示信息。

在上述窗口组成部分中，工作区窗口可通过单击工具栏中“Workspace”按钮隐藏或显示；输出窗口可通过单击工具栏中“Output”按钮隐藏或显示。隐藏这些窗口可以扩大源代码编辑区的大小。

工具栏

为了用户操作方便，该系统在主窗口中提供了多种工具栏，每种工具栏中有若干个按钮，每个工具栏中的按钮表示某种操作。在鼠标指向某个按钮时，将显示出该按钮的功能信息提示。

主窗口在默认情况下只显示两个常用的工具栏：Standard 和 Build 工具栏。如果要使用其他工具栏，可做下述操作：

鼠标指向工具栏的位置，单击右键，出现如图 1.2 所示的快捷菜单。该菜单中，提供了 10 个工具栏供选择使用。单击要选用的工具栏后，在该工具栏名前出现符号√，且该工具栏将出现在主窗口中。如果不想用某个工具栏，也可在工具栏位置单击右键，在出现的快捷菜单中选择不想用的某工具栏，该工具栏名前符号√消失，该工具栏将从主菜单中消失。

下面介绍几个与实现 C++ 语言源程序相关的工具栏所含工具项的功能和用法，其余的工具栏不再介绍。

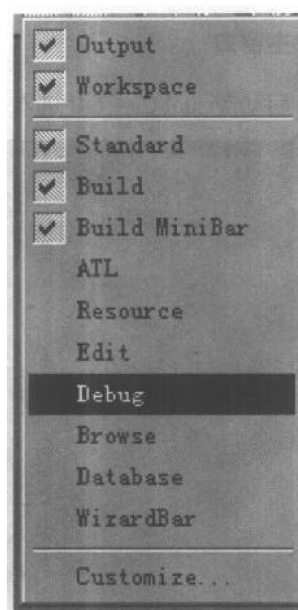


图 1.2 工具栏快捷菜单

一、Standard 工具栏

该工具栏中共有 15 个工具项按钮。如图 1.3 所示。



图 1.3 Standard 工具栏

自左至右各按钮的功能介绍如下：

- New Text File 创建新的文本文件
- Open 打开已有文档
- Save 保存当前文档内容
- Save All 保存所有打开的文档
- Cut 将选定的文档内容从文档中删除，并将之复制到剪贴板中
- Copy 将选定的文档内容复制到剪贴板中
- Paste 在当前插入点处粘贴剪贴板中的内容
- Undo 取消最近一次编辑操作

- Redo 恢复前一次取消的编辑操作
- Workspace 显示或隐藏工作区窗口
- Output 显示或隐藏输出窗口
- Windows list 管理当前打开的窗口
- Find in Files 在多个文件中搜索字符串
- Find 激活查找工具
- Search 搜索联机文档

二、Build 工具栏

该工具栏中有 8 个工具项按钮，如图 1.4 所示。

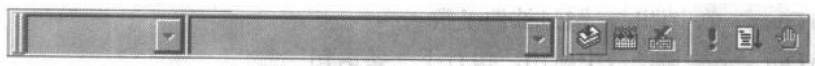


图 1.4 Build 工具栏

这些工具项按钮用来对已建好的应用文件或项目进行编译、连接和运行。自左至右，各工具项按钮的功能介绍如下：

- Select Active Project 选择当前活动项目
- Select Active Configuration 选择活动的配置

Visual C++ 提供两种活动配置：Win32 Release 和 Win32 Debug。前者是基于 Win32 平台的发行版，后者是基于 Win32 平台的调试版。

- Compile 编译文件
- Build 创建项目
- Stop Build 停止创建项目
- Execute Program 执行程序
- Go 启动或继续程序的执行
- Insert/Remove Breakpoint 插入或删除断点

三、Build MiniBar 工具栏



图 1.5 Build MiniBar 工具栏

该工具栏中包含 6 个工具项按钮，如图 1.5 所示。

该工具栏 6 个工具项按钮依次为：

- Compile
- Build
- Stop Build
- Execute program
- Go
- Insert/Remove Breakpoint

它们的功能与工具栏 Build 中的相应按钮功能相同。

四、Debug 工具栏

该工具栏中含有 16 个工具项按钮，如图 1.6 所示。

这些工具项按钮可用来调试已编译的 C++ 源文件及项目，查找所存在的问题。它们只有处于调试运行状态时才有效。

第一行自左至右，各个工具项按钮的功能为：

- Restart 重新启动程序，并处于调试状态
- Stop Debugging 停止调试运行的程序
- Break Execution 中断程序的执行
- Apply Code Change 使用改变代码进行调试
- Show Next Statement 显示下一条要执行的语句
- Step Into 单步调试，进入被调函数内
- Step Over 单步调试，跳过被调函数
- Step Out 从被调函数中跳出，执行下一条语句
- Run to Cursor 运行到当前光标处

第二行自左至右分别为：

- Quick Watch 快速查看当前的调试状态
- Watch 打开一个独立窗口，用来显示用户要查看的变量值和类型。当用户输入变量名时，调试程序自动显示变量的值和类型
- Variables 打开一个独立窗口

该窗口内有 3 个标签，分别用来显示当前语句和上一条语句所用的变量、正在执行函数的局部变量以及 this 指针所指的对象的信息。

- Registers 打开一个独立窗口，显示 CPU 各个寄存器的状态
- Memory 打开一个独立窗口，显示内存的当前状态
- Call Stack 打开一个独立窗口，显示当前语句调用的所有函数，当前函数在顶部
- Disassembly 打开一个独立窗口显示反汇编代码

菜单栏

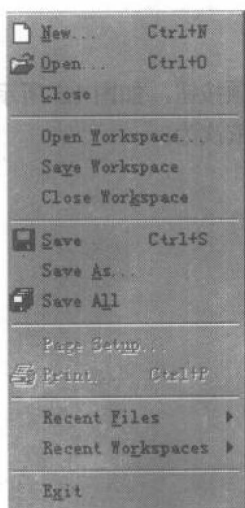


图 1.7 File 菜单

Visual C++ 6.0 主窗口的菜单栏中包含有下面 9 个主菜单项：File, Edit, View, Insert, Project, Build, Tools, Window 和 Help。

下面简单介绍这些菜单项的功能。

一、File 菜单

打开 File 菜单，出现如图 1.7 所示的下拉菜单项，共有 14 个选项。

该菜单的各个命令选项主要功能是用来对文件进行创建、打开、关闭、保存和打印等操作。常用命令介绍如下：

- New 命令

选择该命令，出现如图 1.8 所示的对话框。

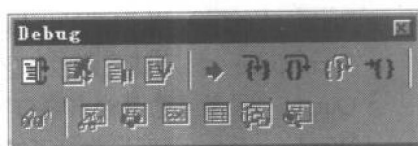


图 1.6 Debug 工具栏

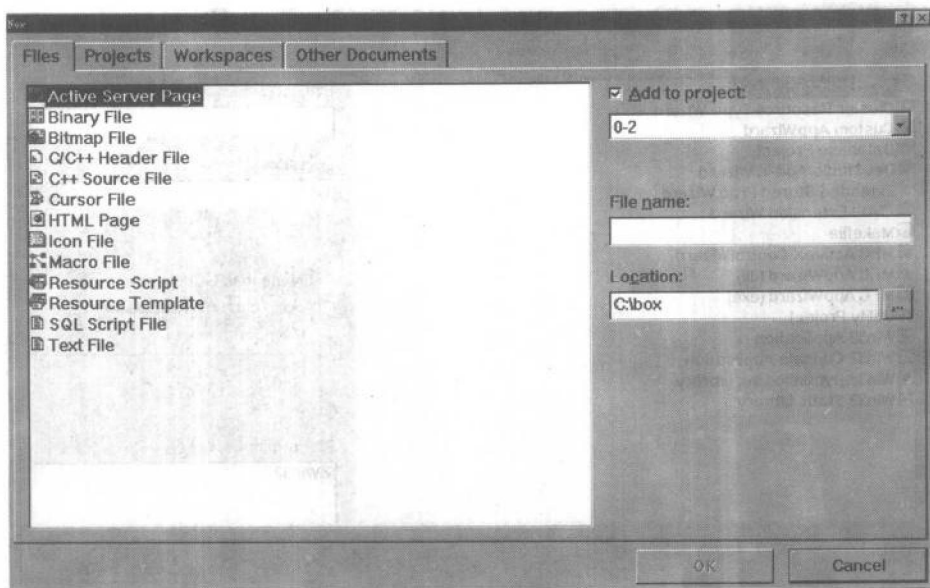


图 1.8 New 命令对话框

该对话框是用来创建文件、项目、工作区以及其他文档的。它有 4 个标签: Files, Projects, Workspace 和 Other Documents。

① Files 标签, 显示出可创建的文件类型, 包括有:

- Active Server Page(服务器页文件)
- Binary File(二进制文件)
- Bitmap File(位图文件)
- C/C++ Header File(C/C++ 头文件)
- C++ Source File(C++ 源程序文件)
- Cursor File(光标文件)
- HTML Page(HTML 页文件)
- Icon File(图标文件)
- Macro File(宏文件)
- Resource Script(资源脚本文件)
- Resource Template(资源模板文件)
- SQL Script File(SQL 脚本文件)
- Text File(文本文件)

选择某种文件类型后, 在对话框内右边的 File name 文本框内键入要创建的文件名。

② Projects 标签, 显示出各种可供选择的项目类型, 如图 1.9 所示。

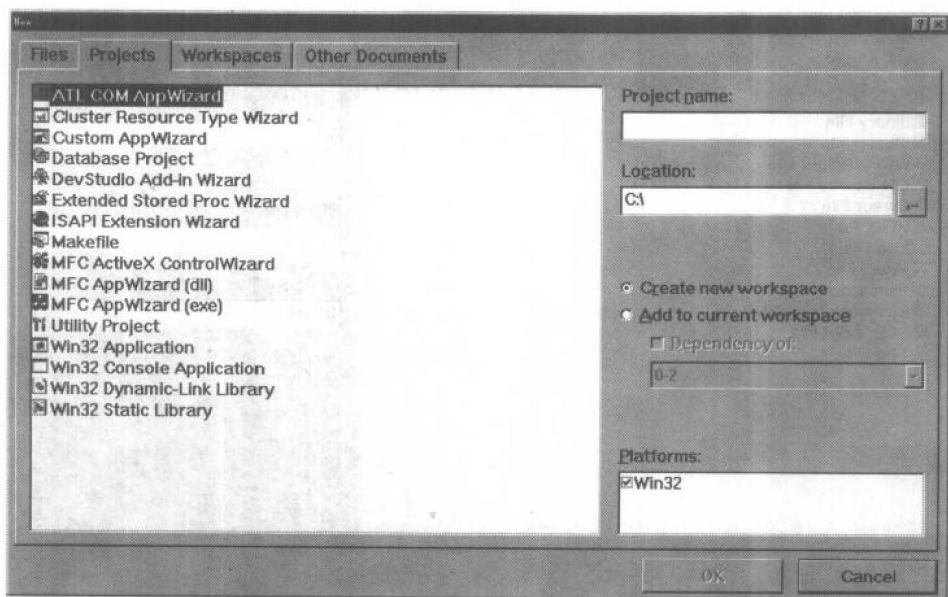


图 1.9 Projects 标签窗口

可供选择的项目类型包含有:

- ATL COM AppWizard(ATL 应用程序创建向导)
- Cluster Resource Type Wizard(簇资源类型创建向导)
- Custom AppWizard(自定义的应用程序创建向导)
- Database Project(数据库项目)
- DevStudio Add-in Wizard
- Extended Stored Proc Wizard
- ISAPI Extension Wizard
- Makefile(C/C++生成文件)
- MFC ActiveX Control Wizard(MFC Activex 控制程序创建向导)
- MFC AppWizard(exe)(MFC 可执行程序创建向导)
- MFC AppWizard(dll)(MFC 动态链接库创建向导)
- New Database Wizard(新数据库创建向导)
- Utility Project(单元项目)
- Win32 Application(Win32 应用程序)
- Win32 Console Application(Win32 控制台应用程序)
- Win32 Dynamic-Link Library(Win32 动态链接库)
- Win32 Static Library(Win32 静态库)

选择某种项目类型后,在该对话框内右边的 Project name 文本框中输入项目名。在 Location 框内输入或修改项目所在路径。

该对话框的 Workspaces 标签和 Other Document 标签可以创建各种类型的工作区文件和文档,在这里就不做详细介绍了。

- Open 命令

选择该命令后，弹出“打开”对话框，如图 1.10 所示。



图 1.10 “打开”对话框

该对话框可用来打开 C++ 源文件、项目文件和其他文件。具体操作方法如下：

先在“搜寻”栏中选定要打开的文件的路径，再通过“文件类型”栏指定要打开文件的类型。此时，在文件名列表框中会出现所要选的文件名。双击文件名，或单击文件名再单击“打开”按钮，两种方式都可以打开所选的文件。

- Close 命令

该命令用来关闭在活动窗口中打开的文件。若该文件修改后尚未保存，系统会提示用户是否应该保存该文件。

- Open Workspace 命令

选择该命令将弹出“Open Workspace”对话框，用来打开该工作区的文件，也可打开其他文件。

- Close Workspace 命令

该命令用来关闭当前工作区的文件，选择该命令后，弹出一个对话框，提示用户是关闭所有文件(选择“是”)还是保留这些文件(选择“否”)。

- Save 命令

该命令用于保留当前窗口中的文件内容，并存放放到原文件名中。如果该文件是未命名的新文件，则系统提示“Save As”对话框。

- Save As 命令

该命令用来将已打开的文件保存到一个新的文件名中。选择该命令，出现“Save As”对话框，用户可将新的文件名输入到该对话框中的文件名文本框内。

- Save All 命令

该命令用来保存当前窗口中所有被打开的文件的内容。如果某个文件尚未命名，系统将会提示用户先输入文件名。

- Page Setup 命令

该命令用来设置和格式化打印结果。选择该命令后，出现“Page Setup”对话框，使用该对话框为打印文档设置标题、脚注以及边距等。

- Recent Files 命令

该命令用来显示打开过的最近 4 个文件，单击该文件名可以将该文件打开。

- Exit 命令

该命令用来退出 Visual C++ 6.0 编译系统。在退出系统前，应将打开的文件保存。

二、Edit 菜单

该菜单的功能是对文档进行编辑和搜索，例如，包含撤消/恢复命令 Redo/Undo，剪贴命令 Cut/Copy/Paste 等。此外，常用命令还有：

- Find 命令

该命令的功能是用来在当前打开的文件中查找指定的字符串。选择该命令后，出现“Find”对话框，如图 1.11 所示。

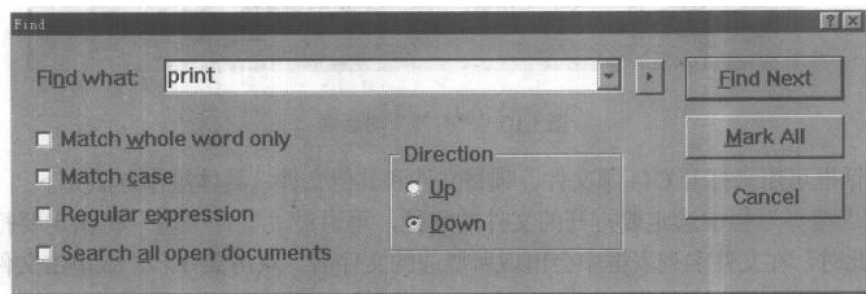


图 1.11 Edit 菜单中的“Find”命令对话框

在该对话框的 Find What 文本框中输入要查找的字符串，在 Direction 框内，选择查找方向：向前(Up)和向后(Down)。再在 4 个复选框内根据需要进行选择。

Match whole word only(仅匹配整个单词)

Match case(区分大小写字母)

Regular expression(正则表达式)

Search all open Documents(搜索所有打开的文档)

在使用正则表达式(Regular expression)查找文件中的匹配文本时，可使用特殊的字符序列去匹配文件中的某种文本模式，常用的查找模式有下述几种：

- *：匹配多个字符。例如，a*可匹配的有 a1, a12, abc 等。

- ：匹配一个字符。例如，a•可匹配的有 ab, a1, a5 等。

- ^：匹配以指定字符串开头的一行。例如，print^，表示所有的以 print 开头的行都可匹配。

- +: 匹配以指定字符串结尾的字符串。例如，+X 可匹配 StartX, EndX 等。

- Go To 命令

该命令用来指定如何将光标移到当前活动窗口的指定位置。选择该命令后弹出“Go To”对话框。

- Bookmarks 命令

该命令用来设置、命名、删除和读取书签。

- Breakpoint 命令

该命令是用来设置、删除和查看断点的。断点可分为位置断点、数据断点、消息断点

和条件断点。断点将告诉调试器何时何处中断程序的执行，以便检查程序代码、变量和寄存器的值。

- **List Members 命令**

该命令用来列出当前光标处对象类属的成员。可以通过双击相应成员名，将该成员添加到光标处。

- **Parameter Info 命令**

该命令用来显示当前光标处函数的参数信息，可在书写函数调用时提供参考，保证调用函数的实参与形参的一致性，从而减少错误。

三、View 菜单

该菜单包含有调试信息和控制屏幕显示方式的命令项，并提供从这里访问“MFC ClassWizard”的方法。常用命令项有：

- **ClassWizard 命令**

该命令用来显示“MFC ClassWizard”对话框，使用它可以进行 MFC 的常规操作。

- **Full Screen 命令**

该命令用来使源代码编辑区扩大到全屏幕。

- **Workspace 命令**

该命令用于显示项目工作区窗口。

- **Output 命令**

该命令用于显示数据输出窗口，在编译时该窗口将会显示出编译信息，包括出错信息。

- **Debug Window 命令**

选择该命令出现级联菜单，在级联菜单中列出了调试窗口的若干操作。

四、Insert 菜单

该菜单可以创建新类、资源、窗体，并将它们插入到文档中，也可以将文件作为文本插入到文档中，还可以添加新的 ATL 对象到项目中。常用的命令有：

- **Resource Copy 命令**

该命令用来复制选定的资源。

- **File As Text 命令**

该命令用来选择插入到文档中的文件。

- **New ATL Object 命令**

该命令用来启动 ATL Object Wizard，将添加新的对象到项目中。

五、Project 菜单

该菜单用于对项目和工作区的管理。可以选择指定项目为工作区中的当前(活动)项目，也可以将文件、文件夹等添加到指定的项目中去，还可以编辑和修改项目间的依赖关系。常用的命令有：

- **Set Active Project 命令**

该命令用来选择当前活动项目。

- **Add To Project 命令**

该命令用来将新文件或已有文件或者部件及控制加到指定的项目中去。

- **Insert Project into Workspace 命令**

该命令用来将项目插入到工作区中。

六、Build 菜单

该菜单包括用于编译、连接和运行应用程序的命令。常用命令有：

- **Compile 命令**

该命令用于编译显示在源代码编辑窗口中的源文件。在编译过程检查源文件中是否有语法错。如果发现错误(warning 或 error)，则将错误信息显示在输出窗口中。使用鼠标双击某行错误信息时，将在源代码编辑窗口中用粗箭头指向出错的代码行，方便用户修改。

- **Build 命令**

该命令用来创建当前文件项目。该命令实际上包含了对源文件或项目的编译和连接，最终生成可执行文件。如果被创建的文件或项目已被编译，则该命令将用来连接，生成可执行文件。在编译或连接中检查出语法错误时，将出错信息显示在输出窗口中。用户修改后，再进行创建，直到生成可执行文件为止。

- **Rebuild All 命令**

该命令用来对所有文件进行重新编译、连接，包含已编译过的文件。因此，此项操作耗时稍长。

- **Execute 命令**

该命令用来运行已生成好的可执行文件，并将运行结果显示到相应的环境中(例如 MS-DOS, Windows98 或 Windows NT 等)。

- **Start Debug 命令**

选择该命令出现级联菜单，选取该菜单项便可启动调试器。这时，将用 Debug 菜单项代替 Build 菜单项。

- **Go 命令**

该命令用在调试过程中，从当前语句启动或者继续运行。

- **Restart 命令**

该命令将系统重新装载程序到内存中，并且将放弃所有变量的当前值。

- **Stop Debugging 命令**

该命令将中断当前调试过程，并返回到原来的编辑状态。

- **Step into 命令**

该命令用来设置单步执行程序。当程序执行到某一函数调用语句时，进入该函数体，并从第一行语句开始单步执行。

- **Step over 命令**

该命令也是单步操作命令，不同的是当程序执行到某一函数调用语句时，不进入该函数体内，直接执行该调用语句，然后停在该调用语句后面的语句。

- **Step out 命令**

该命令用来在单步执行时从某个函数体内跳出，调试该函数调用语句后面的语句。该命令是与 Step into 命令配合使用，先用 Step into 命令将单步执行植入某函数体内，发现不

需要对该函数体内进行单步调试时，使用该命令跳出该函数体。

- Quick Watch 命令

选择该命令，将弹出“Quick Watch”对话框，通过该对话框可以查看和修改变量和表达式，或将变量和表达式添加到 Watch 窗口中。

七、Tools 菜单

该菜单中的命令用来浏览用户程序中定义的符号、定制菜单与工具栏，激活常用的工具或更改选项和变量的设置。

八、Window 菜单

该菜单的命令用来进行有关窗口的操作，常用的命令有：

- New Window 命令

该命令为当前项目文件打开一个新窗口。

- Split 命令

该命令用于将活动窗口进行分区。

- Close 命令

该命令用来关闭当前打开的窗口。

- Close All 命令

该命令用来关闭所有的窗口。

- Next 命令

该命令用来显示下一个窗口。

- Previous 命令

该命令用来显示前一个窗口。

九、Help 菜单

Help 用来获得大量的帮助信息。读者应该养成遇到问题查“Help”的习惯。

项目工作区

一、项目(Project)

项目是一些相互关联的源文件的集合，这些源文件组成一个程序，它们被编译、连接后生成一个可执行文件。

创建一个项目后，可以添加任何其他目录的文件到该项目中。添加文件到项目中并不改变文件的位置，而项目只是记录文件的名字和位置。

二、项目工作区(Project Workspace)

在 Visual C++ 中，文件、项目和项目配置是由项目工作区组织起来的。项目工作区的内容和设置通过项目工作区文件(.dsw)来描述，在建立一个项目工作区文件的同时，还生成项目文件(.dsp)和工作区选项文件(.opt)，用来保存工作区的设置。

三、项目工作区窗口

项目工作区窗口用来查看和修改项目中的所有元素。该窗口的底部提供了三种面板：类面板(Class View)、资源面板(Resource View)和文件面板(Files View)。它们的功能如下所述：

- 类面板

该面板在项目工作区窗口中显示该项目中所有类及其成员函数。单击“+”号，打开树形结构的每一项，显示出某类的成员函数和数据成员。双击某一项，则在右边的源代码编辑窗口中显示该成员的源代码。如果该源代码已被显示，则使用光标进行指示。

- 文件面板

该面板在项目工作区窗口中显示项目中的所有文件及其相互关系。单击“+”号时，依次打开树形结构的每一页，并显示出所有的资源文件、头文件和源代码文件。双击某一项时，则会在右边的源代码编辑窗口内打开该文件，显示其源代码。

- 资源面板

该面板在项目工作区窗口中显示项目中的所有资源。单击“+”号时，依次打开树形结构的某一项，并显示出所有资源，包括字符串表、对话框图符及其版本信息。双击某一项时，则在右边的源代码编辑窗口内显示该资源的图形编辑窗口，可直接在该窗口内增添资源或修改资源特性。

1.2 C++ 单文件应用程序的实现

本节介绍使用 Visual C++ 6.0 实现 C++ 单文件应用程序的方法。

一、编辑

单击主窗口菜单栏中的 File 菜单项，弹出如图 1.7 所示的下拉式菜单。

单击下拉式菜单中的选项 New，弹出“New”对话框(如图 1.8 所示)，该对话框中有 4 个标签。选择 Files 标签后，弹出的对话框中列出了 13 个选项。在编辑单个 C++ 源文件时，双击 C++ Source File 选项，返回到 Microsoft Visual C++ 的主窗口。这时可在源代码编辑窗口中输入源代码程序。例如，输入下述源程序：

```
#include<iostream.h>
void main( )
{
    cout<<"This is a C++ source File.\n";
}
```

按上述格式将该源代码程序输入到机器后，适当进行编辑，然后将其文件内容存入磁盘。存盘方法如下：

单击菜单栏中 File 菜单项，在其下拉菜单中选择 Save 选项或者 Save As 选项，弹出“保存为”对话框。选择存放文件的路径后，在“文件名”框内输入文件名(默认的扩展名为 cpp)后按回车键，或按“保存”按钮即可。