

Js45/64

软件开发实例教程丛书



门槛创作室 编著

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 提 要

Microsoft Visual C++ 6.0 (简称 VC++6.0), 是微软公司 Visual 系列的最新杰作。Visual C++ 6.0 全面支持 Windows 编程, 与微软公司的 Windows 系列操作系统之间具有天然的亲和性。

本书的内容分为四部分。第一部分“基础篇”, 主要介绍 Visual C++ 6.0 集成操作环境的命令操作, 面向对象的方法和 Windows 编程原理。第二部分“应用篇”, 主要介绍使用 MFC 进行 Windows 编程, 介绍 Windows 消息循环和消息处理机制, 以及 Windows 图形显示和文件 I/O 操作。第三部分“资源篇”, 全面介绍使用 Windows 资源, 包括图标、光标、菜单、对话框以及对话框中的各种控件。第四部分“提高篇”, 深入介绍了 Visual C++ 6.0 编写的 Windows 应用程序的程序框架, 详细剖析了文本视图结构的原理, 向读者介绍了使用 Visual C++ 6.0 的编程向导 AppWizard 和 ClassWizard 编写 Windows 应用程序。

本书可作为 Visual Studio 培训班的教材及自学教程, 并可作为编程人员的参考书籍。

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

Visual C++6.0 实例教程 / 门槛创作室编著. —北京: 电子工业出版社, 1999.3

(软件开发实例教程)

ISBN 7-5053-5232-6

I. V… II. 门… III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (1999) 第 01182 号

丛 书 名: 软件开发实例教程丛书

书 名: Visual C++ 6.0 实例教程

编 著 者: 门槛创作室

策 划: 赵丽松

责任编辑: 吴 源

特约编辑: 刘 欣

印 刷 者: 北京天宇星印刷厂

装 订 者: 河北省涿州桃园装订厂

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: 787×1092 1/16 印张: 23.5 字数: 602 千字

版 次: 1999 年 4 月第 1 版 1999 年 4 月第 1 次印刷

书 号: ISBN 7-5053-5232-6
TP·2607

定 价: 30.00 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

总 序

计算机技术的广泛应用，正在改变着我们的生活。了解计算机、学习计算机、掌握计算机就意味着把握住了新时代的脉搏，把握住了新的机遇，也许意味着新生活的开始。在这一学习计算机技术的大潮中，掌握使用编程技术无疑是新时代弄潮儿所追求的梦想。

然而，编程语言这一统治计算机世界最有力的武器并不是那么容易驾驭的。要学好编程语言不仅要熟悉语言本身，还应能洞察计算机世界内在的运行规律。所以，未来的编程大师们正急待有力的引导和支持。

《软件开发实例教程丛书》正是一套能够帮助大家走进计算机编程奇妙天地并洞察计算机世界内在规律的优秀教程。本套丛书遵循编程语言可视化的最新趋势，向广大读者全面介绍广泛使用的可视化编程工具，如微软公司的最新力作 Visual Studio 6.0 系列语言（包括其中较常用的 Visual Basic 6.0、Visual C++ 6.0、Visual J++ 6.0、Visual FoxPro 6.0 及 Visual InterDev 6.0）等等。本套丛书还重点介绍了流行于 90 年代的面向对象的编程技术和计算机网络编程技术，为读者学习编写软件奠定了良好的理论基础。

《软件开发实例教程丛书》包含大量程序实例，通过这些生动有趣的实例向读者介绍可视化编程的技术和软件开发的思维方式。书中还提供了大量详细注释过的程序代码，对读者具有很高的参考价值。

《软件开发实例教程丛书》经过精心的策划及认真的组织编写，终于及时与广大读者见面。但由于编著者水平有限，时间仓促，书中不当之处，恳请各位同仁及广大读者批评指正。



Computerman Corporation

门槛创作室

<http://menkan.yeah.net>

E-mail: ldandxwh@public.bta.net.cn

目 录

第一篇 基础篇

第 1 章 走进 Visual C++ 的精彩世界	2
1.1 Visual C++ 6.0 概述	2
1.2 Visual C++ 的工作环境	2
1.2.1 Visual C++ 6.0 开发环境总览	2
1.2.2 File 菜单	3
1.2.3 Edit 菜单	5
1.2.4 View 菜单	6
1.2.5 Insert 菜单	10
1.2.6 Project 菜单	11
1.2.7 Build 菜单	12
1.2.8 Tools 菜单	13
1.2.9 Windows 菜单	14
1.2.10 Help 菜单	15
1.3 DOS 程序员的惊喜	16
1.3.1 生成工程	17
1.3.2 生成和编辑源文件	18
1.3.3 建立程序	19
1.3.4 运行程序	20
1.3.5 调试应用程序	20
1.4 小结	22
思考与练习	22
第 2 章 Windows 编程起步, 进入一个新天地	23
2.1 Windows 的历史	23
2.2 Windows 的窗口和控件	24
2.3 华美界面下的强大引擎	27
2.3.1 事件驱动模型和消息循环	27
2.3.2 多任务机制的内幕——进程和线程	29
2.3.3 应用程序共享的资源——动态链接库技术	30

2.3.4 Windows 内存管理技术	32
2.4 编写一个 Windows 程序	33
2.4.1 Windows 应用程序设计接口和开发工具包	33
2.4.2 用 Windows API 编写一个最小的 Windows 程序	34
2.5 小结	39
思考与练习	39
第 3 章 面向对象的程序设计技术基础	40
3.1 面向对象技术概述	40
3.2 封装性	42
3.3 继承性	43
3.4 多态性	45
3.5 Visual C++ 6.0 的基本类库 MFC 简介	50
3.6 C++ 的其它语法特点	51
3.7 小结	53
思考与练习	53

第二篇 应用篇

第 4 章 MFC 编程基础	56
4.1 MFC 的演变	56
4.2 MFC 类库层次	56
4.3 CObject 类	57
4.4 从 CObject 中派生出来的主要类	57
4.4.1 应用程序结构类	57
4.4.2 调试和异常类	58
4.4.3 文件服务类	58
4.4.4 图形处理类	59
4.4.5 菜单类	59
4.4.6 数据库类	59
4.4.7 同步类	59
4.4.8 数组类、列表类和映像类	60
4.4.9 Internet 支持类	60
4.4.10 非 CObject 派生类	60
4.5 MFC 程序剖析	60
4.6 小结	61
思考与练习	61
第 5 章 MFC 的消息和消息处理	62
5.1 MFC 事件驱动程序设计的引擎—CWinApp 类	62
5.2 消息映像和消息处理	66
5.2.1 消息	66

5.2.2	MFC 的消息映像及消息处理方法	68
5.2.3	MFC 预定义的消息和消息处理程序	70
5.3	消息映像例程	77
5.4	小结	79
	思考与练习	79
第 6 章	鼠标键盘与 Windows	80
6.1	鼠标消息和鼠标消息处理	80
6.1.1	用户区鼠标消息	80
6.1.2	非用户区的消息	81
6.1.3	“抓老鼠”——鼠标捕捉	83
6.2	键盘和键盘消息处理	84
6.2.1	键盘消息	84
6.2.2	按键消息及其处理	85
6.2.3	字符消息及其处理	87
6.3	鼠标键盘输入应用实例	88
6.4	小结	93
	思考与练习	93
第 7 章	开启多彩的窗口——设备描述表和图形对象	94
7.1	设备描述表与图形设备接口	94
7.2	MFC 设备描述类——CDC 类	95
7.3	CDC 的派生类	96
7.3.1	CPaintDC 类	96
7.3.2	CClientDC 类	97
7.3.3	CWindowDC 类	98
7.4	图形对象	99
7.4.1	笔——Cpen 类	99
7.4.2	刷子——CBrush 类	101
7.5	用 CDC 的绘图函数与图形对象绘图	102
7.5.1	画图的坐标系统	102
7.5.2	Timer——趣味的源泉	103
7.5.3	“膨胀的欲望”——CDC 绘图例程 1	105
7.5.4	“MFC 世界的晚霞”——CDC 绘图例程 2	108
7.5.5	“多彩的网络世界”——CDC 绘图例程 3	110
7.5.6	“贝叶斯万花筒”——CDC 绘图例程 4	112
7.6	用 CDC 输出文字	114
7.7	小结	120
	思考与练习	121
第 8 章	文件输入与输出	122
8.1	文件与 CFile()类	122

8.1.1	构造器类函数	123
8.1.2	输入输出方法	125
8.1.3	文件定位函数	126
8.1.4	其它 CFile 类成员函数	127
8.2	CFile 的派生类	128
8.2.1	CStdioFile 类	128
8.2.2	CMemFile 类	129
8.2.3	CSharedFile 类	129
8.3	文件操作的例程 FileDemo.CPP	130
8.4	小结	132
	思考与练习	133

第三篇 资源篇

第 9 章	用图标光标装饰应用程序	136
9.1	资源	136
9.2	图标	136
9.2.1	创建图标	137
9.2.2	在工程中加入图标资源	138
9.2.3	在程序中使用图标资源	141
9.3	光标资源	143
9.3.1	创建光标	143
9.3.2	将光标加入应用程序	145
9.4	使用图标和动画光标的例程	145
9.5	小结	148
	思考与练习	148
第 10 章	使用菜单快捷键和字符串表	149
10.1	菜单的使用	149
10.1.1	菜单消息	150
10.1.2	用菜单编辑器生成菜单资源	150
10.1.3	在应用程序中使用主菜单	154
10.1.4	非主菜单资源的加载过程	156
10.2	快捷键	159
10.3	字符串表	161
10.4	状态栏	163
10.5	字体例程	164
10.6	小结	172
	思考与练习	173
第 11 章	发掘 Windows 资源——使用公用对话框	174
11.1	公用对话框概述	174

11.2	CColorDialog——选择色彩的好工具	175
11.3	CFileDialog——方便的文件操作界面	177
11.4	CfindReplaceDialog——编辑器中不可缺少的界面	179
11.5	CFontDialog 类——选择字体的好工具	182
11.6	CPrint Dialog 类——标准的打印设置界面	183
11.7	CPageSetupDialoge 类——页面设置的统一界面	185
11.8	字体显示例程的新版本——使用公用对话框	187
11.9	小结	197
	思考与练习	197
第 12 章	使用对话框和按键控件	198
12.1	对话框的基本概念	198
12.2	CDialog 类和创建对话框的一般过程	198
12.3	实现无控件对话框	200
12.3.1	用资源编辑器生成对话框	200
12.3.2	使用对话框	204
12.4	在对话框中使用按键	208
12.5	Cbutton 类	210
12.6	CBitmapButton 类——CButton 的优秀继承者	212
12.6.1	在对话框中使用 CBitmapButton 对象	214
12.7	在对话框中使用 CButton 类	214
12.8	小结	221
	思考与练习	221
第 13 章	编辑控件和对话框数据交换技术	222
13.1	CEdit 类——编辑控件的类库支持	222
13.1.1	编辑控件的消息	223
13.1.2	CEdit 类及其成员函数	224
13.2	CEdit 使用例程——编辑器	227
13.3	对话框数据交换(DDX)和验证(DDV)技术	234
13.3.1	对话框数据交换(DDX)	234
13.3.2	对话框数据验证(DDV)	235
13.3.3	DDX 和 DDV 在对话框中的实现	235
13.3.4	默默无闻的 CDataExchange 类	236
13.4	用编辑控制实现多种类型的数据输入	237
13.4	小结	244
	思考与练习	245
第 14 章	使用列表框和组合框	246
14.1	列表框控件和 CListBox 类	246
14.1.1	列表框控件的 MFC 支持 CListBox 类	247
14.1.2	资源编辑器和 CListBox 联合	248

14.1.3	CListBox 的方法	250
14.1.4	列表框消息	252
14.2	列表框应用例程	253
14.3	组合框控件和 CComboBox 类	259
14.3.1	组合框控件的 MFC 支持 CComboBox 类	260
14.3.2	资源编辑器和 CComboBox 的联合	261
14.3.3	CComboBox 方法	263
14.3.4	组合框消息	265
14.4	组合框应用例程	266
14.5	小结	270
	思考与练习	270
第 15 章	使用工具栏资源和滚动控件	271
15.1	工具栏和工具提示	271
15.1.1	CToolBar 类——工具栏的 MFC 支持	272
15.1.2	使用资源编辑器和 CToolBar 类相结合	274
15.1.3	工具提示	277
15.1.4	使用工具栏的一个应用程序框架	278
15.1.5	工具栏使用的提高——CToolBarCtrl 类	282
15.2	使用微调控件及其支持类 CSpin	286
15.2.1	CSpinButtonCtrl 类——微调控件的 MFC 支持	286
15.2.2	使用资源编辑器与 CSpinButtonCtrl 类结合创建微调控件	289
15.2.3	使用微调控件的例程	291
15.3	使用滚动控件及其支持类 CScrollBar	294
15.3.1	CScrollBar——滚动条控件的 MFC 支持	294
15.3.2	使用资源编辑器与 CScrollBar 类联合创建滚动条控件	298
15.3.3	使用滚动控件的例程	300
15.4	使用滑动控件及其支持的 CSLiderctrl 类	303
15.4.1	CSliderCtrl 类——滑动控件的 MFC 支持	303
15.4.2	使用资源编辑器与 CSLiderctrl 类结合创建滑动控件	306
15.4.3	使用滑动控件的例程	307
15.5	小结	311
	思考与练习	311

第四篇 提高篇

第 16 章	MFC 的应用程序框架	314
16.1	应用程序和主窗口模型	314
16.1.1	CWinApp 类——MFC 中的 WinMain()	314
16.1.2	CFrameWnd 类——MFC 中的基本框架窗口	314
16.2	基于对话框的应用程序框架	318

16.2.1 建立对话框应用程序框架的基本步骤	318
16.2.2 第一个基于对话框的应用程序	319
16.3 文档/视图结构的应用程序框架	320
16.3.1 文档和 CDocument 类	321
16.3.2 视图和 CView 类	323
16.3.3 CView 支持的打印及打印预览	325
16.3.4 文档/视图结构的管家——文档模板和 CDocTemplate 类	328
16.3.5 文档/视图结构应用程序的消息循环和消息处理	331
16.3.6 文档/视图结构的应用程序示例	334
16.4 MDI 程序的文档/视图结构	343
16.5 小结	350
思考与练习	350
第 17 章 VC 编程自动化——AppWizard 和 ClassWizard	351
17.1 全方位提供服务的 Wizard	351
17.2 AppWizard 的应用	353
17.3 使用 CClassWizard 为应用程序增加功能	356
17.4 小结	360
思考与练习	360

第一篇

基础篇

本篇导读

本篇简要介绍 Visual C++ 的工作环境，并综述了 Windows 编程技术和面向对象的编程技术。这将有利于读者迅速熟悉 Visual C++，有利于理解 Windows 的工作原理，为掌握 Visual C++ 编制 Windows 程序奠定基础。

第一部分涉及的内容有：

VC 的发展历程和技术特点

用控制台类程序移植 DOS 程序

Windows 程序的内在运行机制

Windows 系统的一般技术

面向对象编程的概念

第1章 走进Visual C++的精彩世界

本章将介绍Visual C++的发展历史和技术特点。介绍Visual C++的工作环境，使读者快速熟悉Visual C++的界面操作，并介绍用控制台类程序移植DOS程序，介绍程序的生成、编译和调试。

1.1 Visual C++ 6.0 概述

Microsoft Visual C++，这套功能强大的Windows应用程序开发系统是由Microsoft C/C++演化而来的。从Microsoft C/C++ 8.0开始改称Visual C++，发展到今天Visual C++ 6.0。

由于先天“血缘”关系，Visual 平台是当前Windows应用开发系统中与Windows平台本身最具亲和性的开发平台。特别是Visual C++ 6.0，它包含了支持Windows 95、Windows NT 4.0以及Windows 98的Win32 SDK文档，即人们可以在Visual C++环境下使用Windows SDK提供的SDK文档提供的API来进行Windows应用程序的开发。Visual C++ 6.0同Visual C++ 5.0一样彻底放弃了对16位操作系统的支持。

Visual C++的最主要的技术特点是：可视化编程和支持面向对象的编程技术。Visual C++提供了一系列可视化编程的工具，如AppWizard和ClassWizard等。它们使Windows编程更为直观。Visual C++支持的面向对象的编程技术包装了Windows内在的复杂的运行机制，使Windows编程更为简单易学。此外，Visual C++ 6.0还具有其他技术特点：如通过ActiveX技术，支持Internet编程；支持ODBC数据库编程等。

Visual C++就象一个奇妙的魔盒，在其美丽的外表下包装着强大的“魔力”，一旦掌握它，你将进入Windows编程的自由王国。

1.2 Visual C++的工作环境

1.2.1 Visual C++ 6.0开发环境总览

Visual C++ 6.0作为32位应用程序的开发工具，对支持环境有较高的要求。硬件配置的最低要求是：486 CPU、100MB硬盘空间、16MB内存（Windows 95下可为8MB）；软件要求32位操作系统，即Windows 95，Windows NT 3.5, Windows 98, Windows NT 4.0及更高版本。

如果已经在上述环境中安装了Visual C++ 6.0，用鼠标单击“开始”，在“程序”组中选“Microsoft Visual C++ 6.0”工作组，在“Microsoft Visual C++ 6.0”项上双击，即进入Visual C++ 6.0的集成环境，如图1.1所示。使用过QuickC for Windows，或Borland IDE的读者对Visual C++ 6.0的工作过程一定不会陌生。在这一集成环境中，最重要的是工程（Project）的概念。“工程”是相关源文件的集合，包括源程序、头文件及资源定义文件。Visual平台是自动化很高的编译系统，它能自动处理源文件间的关系，利用其内在推理规则来激活编译器、连接器和资源编译器，最后生成可执行文件。读者要做的只是从Project菜单中选择Build命令，

系统就会自动生成可执行文件。

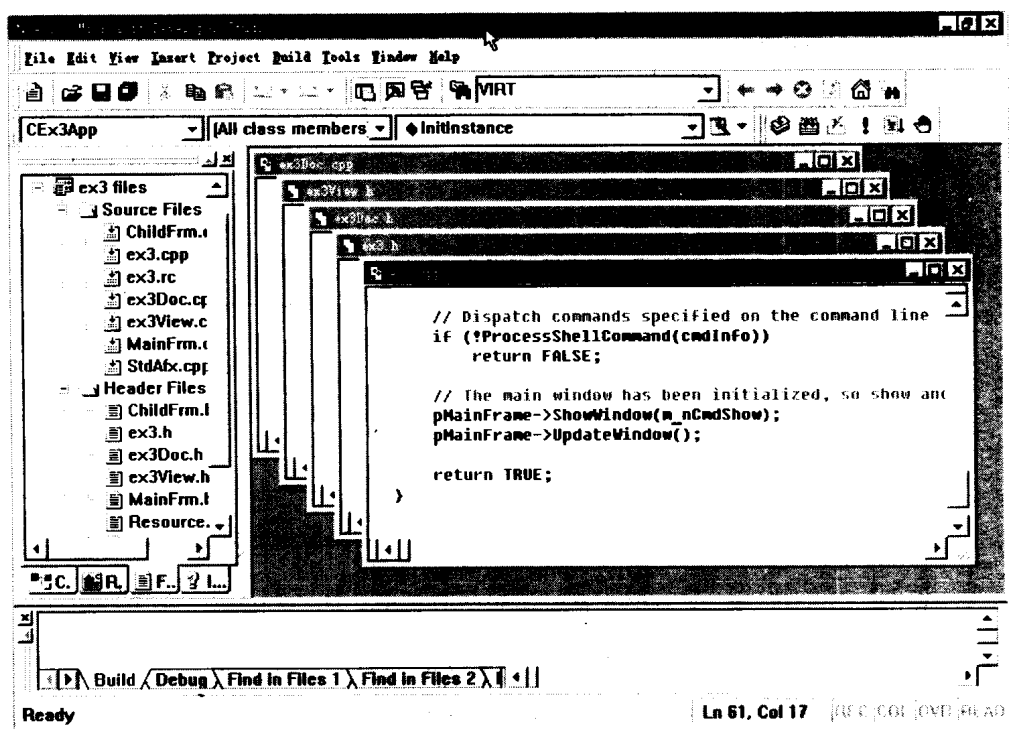


图1.1 Visual C++ 6.0集成环境界面

下面将简要介绍各菜单功能。

1.2.2 File菜单

File菜单项主要用于各种编程过程中用到的源文件进行操作和管理。File菜单包括以下菜单命令。

New(Ctrl+N)

New命令用于创建新文件，包括源文件、头文件和资源文件、工程(Projects)或工程工作区(Workspaces)甚至Word文档，该命令运行时将打开一个对话框，Visual C++将根据你选择的文件类型和指定文件名及路径自动创建文件，并且打开相应的编辑器，如图1.2所示。

Open(Ctrl+O)

Open命令打开一个已存在的文件，同New命令一样，Open命令可打开多种文件，包括源文件、头文件、各种资源文件、工程文件、图形文件等并打开相应的编辑器，使文件内容在工作区显示出来，以供查阅和编辑。

Close命令

Close命令关闭右侧工作区内的当前活动窗口。如果窗口内的内容有改动，系统会提示保存当前窗口的内容。

Open Workspace命令

相当于Open的子命令，只负责打开已存在的工程，即把所有开发过程中相关的文件，包括源程序文件、资源文件等都调到工程工作区。

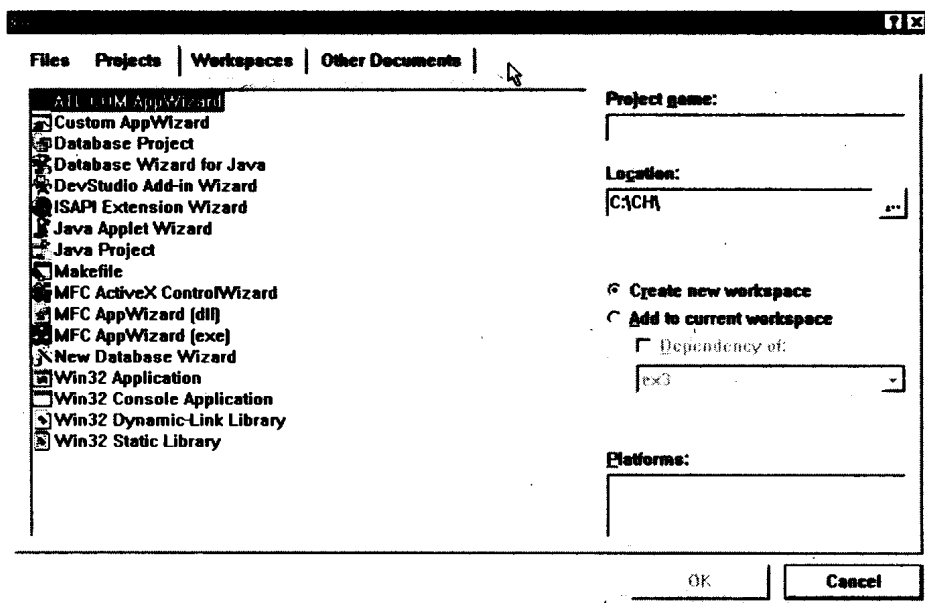


图1.2 File菜单New对话框

Save Workspace命令

将保存工程中所有相关文件及编译、连接所需的信息。

Close Workspace命令

Close Workspace 关闭工程中所有相关文件，并提示保存修改过的文件和设置。

Save命令(Ctrl+S)

Save 命令用于将编辑区中当前活动窗的内容保存到相关的文件中。

Save as 命令

Save as命令用于将编辑区中当前活动窗的内容保存到一个由用户指定的文件中。

Save All命令

Save All命令将保存所有打开的文件、文档和工程。

Rename命令

Rename命令将对指定文件重新命名。

Page Setup命令

Page Setup命令，即页面设置。其中可以对要求输出的文档进行输出效果的控制。其中包括设置上边界、下边界、左边界、右边界、头标和尾标。头标和尾标可以有文件名、页码、当前时间、当前日期及左对齐、右对齐、居中等格式设置，如图1.3所示。

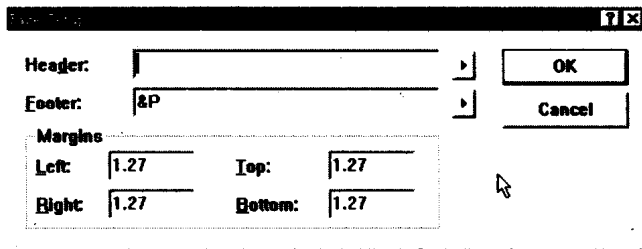


图1.3 Page Setup命令对话框

Print命令

Print命令用于打印当前工作区的活动窗口的内容。在Print对话框中，可以选打印机打印范围，并可通过Setup激活打印设置的对话框，对纸张、输出质量等参数进行设置，如图1.4所示。

Recent Files命令

Recent Files 命令将列出4个最近打开过的文件，可以通过双击鼠标打开所选的文件。

Recent Workspaces命令

Recent Workspaces命令将列出4个最近打开过的工程。可以通过双击鼠标打开所选的工程。

Exit命令

Exit命令将先关闭所有的窗口，再关闭Visual C++工作平台。在关闭的过程中，系统会提示保存用户改动过的文件。

1.2.3 Edit菜单

Edit菜单主要支持文本文件的编辑和查找。

Undo命令(Ctrl+Z)

Undo命令用于取消最近一次的误操作。

Redo命令(Ctrl+Y)

Redo命令用于取消Undo操作，抵消Undo命令的效果。

Cut命令(Ctrl+X)

用于删除选定的文本块，同时写入剪贴板，覆盖剪贴板原有的内容。

Copy命令(Ctrl+C)

Copy命令将选定文本块复制到剪贴板上。

Paste命令(Ctrl+V)

Paste命令将剪贴板上的内容插到光标处。

Delete命令(Del)

Delete命令将删除选定文本块。选定的文本块也不会留在剪贴板上。

Select All命令(Ctrl+A)

Select All命令用于选定当前窗口中的全部内容。

Find命令(Ctrl+F)

Find 命令将激活对话框。通过对话框输入在当前窗口中要查找的字符中，同时可以选择查找的方向和匹配的模式，如图1.5所示。

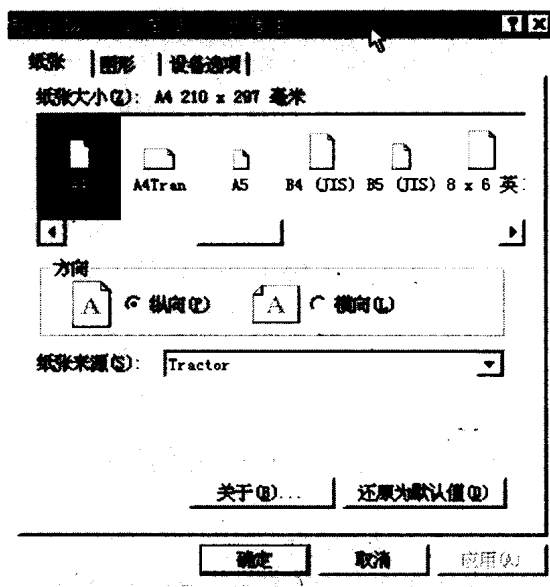


图1.4 Print Setup对话框

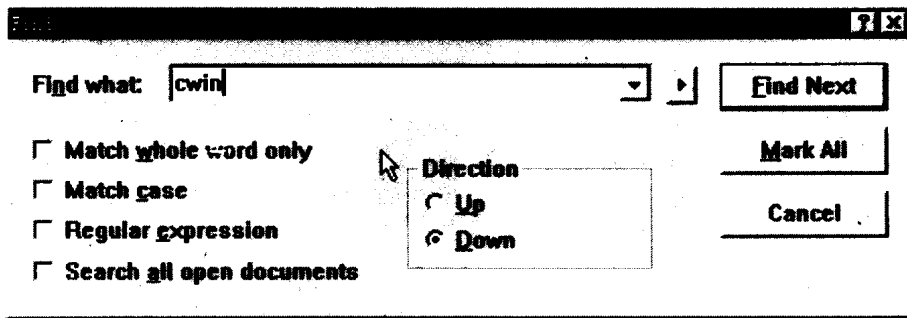


图1.5 Find对话框

Find in Files命令

该命令在多个文件中查找指定的字符串。查找结果将显示在OutPut窗口的Find in File页面上。

Replace命令(Ctrl+L)

该命令可以进行文本替换，输入要查找字符串的要替换成的字符串，即可替换。同时可以通过对话框选择替换的方式和范围。

GoTo命令(Ctrl+G)

GoTo命令用于在文件中迅速找到指定行，可以按下列类型指定：地址(Address),书签(Bookmark)、定义(Definition)、错误(Error/Tag)等，如图1.6所示。

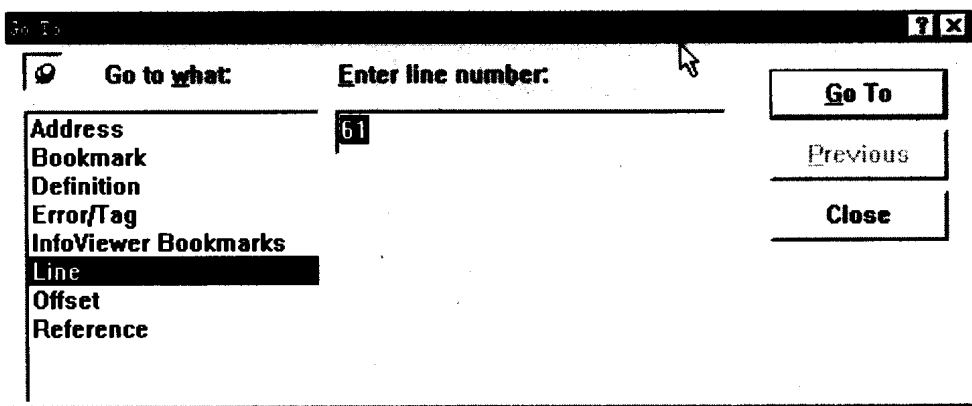


图1.6 Go To对话框

Bookmark 命令(Alt+F2)

Bookmark命令用于设置书签。

1.2.4 View菜单

View菜单中的选项可以用不同方式对工作平台中的不同窗口进行观察，可以在工作平台上显示各种工具。

Class Wizard 命令(Ctrl+W)

Class Wizard命令将激活MFC Class Wizard 对话框。该工具帮助用户进行应用程序类的增减、编辑和管理，是Visual C++可视化编程的重要工具，如图1.7所示。

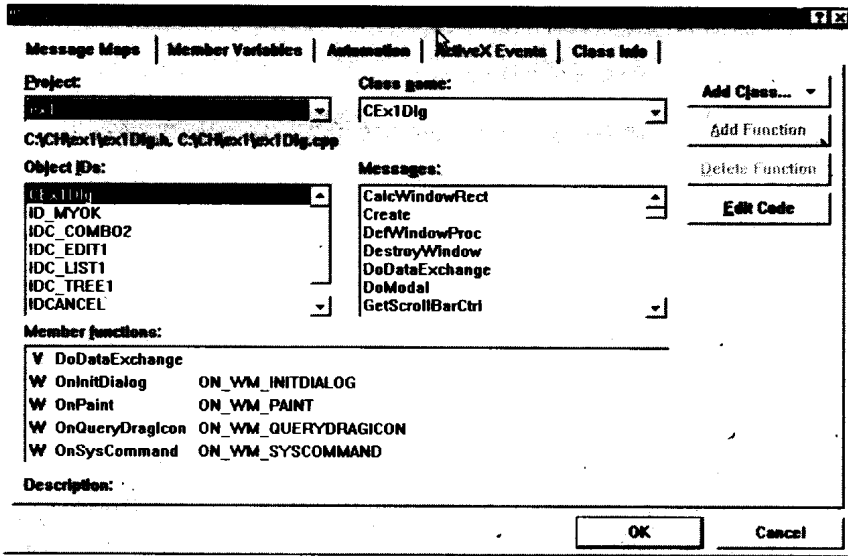


图1.7 Class Wizard 对话框

Resource Symbols命令

Resource Symbols命令管理程序中所用的符号，可以对符号常量进行修改、添加和删除，但Resource Symbols命令只能管理程序由Appwizard等可视工具生成的工程，如图1.8所示。

Resource Includes命令

Resource Includes命令将激活Resource Includes对话框。该对话框的功能是：改变符号头文件名；将只读符号和计算符号包含到程序中；将程序连接过程中要加入用户工程中的资源包含到程序中。

Full Screen命令

Full Screen命令的作用是将工作区的当前窗口放大到全屏幕。按Esc键即可返回。

Workspace命令(Alt+O)

Workspace命令(Alt+O)命令控制显示工程工作区，如图1.9所示。

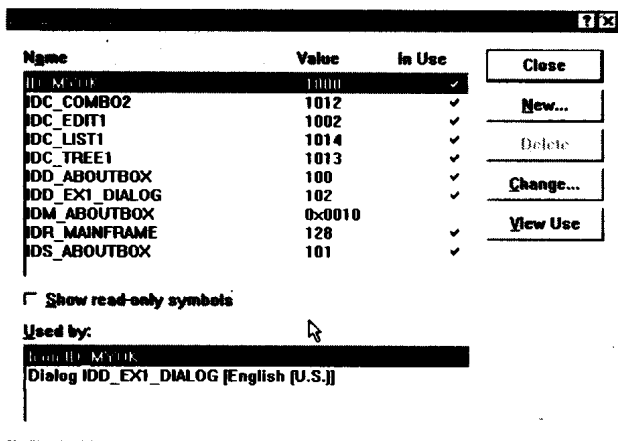


图1.8 Resource Symbols对话框

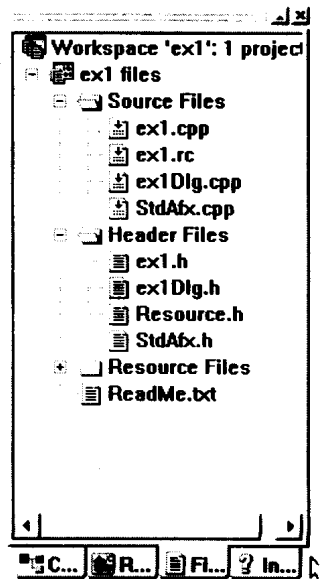


图1.9 Workspace窗口