

郑阿奇 主编

丁有和 编著

高等院校程序设计规划教材

Visual C++ 实训 (第3版)



清华大学出版社

郑阿奇 主编

丁有和 编著

高等院校程序设计规划教材

Visual C++ 实训 (第3版)

清华大学出版社
北 京

内 容 简 介

本书第3版以当前最流行的 Visual C++ 6.0 中文版为平台, 内容包括实验和综合应用实习两大部分。实验包括: Visual C++开发环境, C++程序基础, Windows 编程基础, 常用控件, 菜单、工具和状态栏, 框架窗口、文档和视图, 图形和文本, 数据库编程。综合应用实习为设计一个学生信息管理的较完整的程序系统, 包括用 InstallShield 进行应用程序发布。本书实验在第2版的基础上进行了重新整合和优化, 增加了 C++程序基础和简单计算器实训。本书在《Visual C++教程》(第3版) 内容的基础上进行了延伸和拓展, 并自成体系, 既可作为配套用书, 也可单独使用。

本书适合作为大学本科、高职高专等各类学校的实训教材, 也可作为 Visual C++培训和用户的自学参考书。

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

Visual C++实训/郑阿奇主编; 丁有和编著. —3版. —北京: 清华大学出版社, 2016
高等院校程序设计规划教材
ISBN 978-7-302-44195-3

I. ①V… II. ①郑… ②丁… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2016) 第 148268 号

责任编辑: 张瑞庆

封面设计: 常雪影

责任校对: 焦丽丽

责任印制: 杨 艳

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 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: 三河市中晟雅豪印务有限公司

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 14.5 字 数: 352 千字

版 次: 2005 年 7 月第 1 版 2016 年 9 月第 3 版 印 次: 2016 年 9 月第 1 次印刷

印 数: 1~2000

定 价: 29.00 元

产品编号: 070041-01

前言

本书以最流行的 Visual C++ 6.0 中文版为平台, 仍然以第 2 版的体系结构为基础, 包括实验和综合应用实习两大部分。实验在第 2 版的基础上进行重新整合和优化, 增加了 C++ 程序基础和简单计算器实训。综合应用实习在上一版的基础上进行了优化。同时, 本书在下列三个方面进一步完善和创新:

- (1) 本书既是 Visual C++ 教程内容的实验, 也是教材内容的延伸和拓展。
- (2) 本书既是初学者的 Visual C++ 基础实训, 也是解决实际问题的模板。
- (3) 本书既可以有选择地进行 Visual C++ 课程实践, 也可以独立成册, 成为 Visual C++ 开发者的工具之一。

本书与《Visual C++ 教程》(第 3 版) 配套使用时, 可与下列内容配合组成 Visual C++ 课程包。

(1) Visual C++ 教程 (第 3 版): 教程以“跟着学→模仿→自己应用”为思路, 力争使问题简单化。翻开书, 整篇体现较强的应用特色, 把介绍内容和实际应用有机地结合起来。选用的实例既不太大, 这样程序不太长, 同时实例又涉及一定的范围和意义, 读者可通过实例来消化主要内容。为了解决用户对 Visual C++ 较高层的内容需要, 在介绍有关基本知识后引入一个小规模可运行的例子来消化内容、上机实验和供用户模仿。

(2) Visual C++ 教程 (第 3 版) 课件: 包含了本教程的主要内容, 在网上同步免费提供该课件下载, 教师可据此备课和教学。同时, 附上本教程所有实例源代码。

(3) Visual C++ 应用系统: 在网上同步免费提供包含教程和实验中形成的学生信息管理系统的所有源文件, 实习形成的人员信息管理系统的所有源文件。教师据此在课上演示, 学生可据此上机模仿。

本书配套的教学资源可以在清华大学出版社网站免费下载, 网站地址为: <http://www.tup.com.cn>。

本教程不仅适合于教学, 也非常适合于 Visual C++ 的各类培训和用 Visual C++ 开发应用程序的用户学习和参考。

本书由丁有和 (南京师范大学) 编写, 郑阿奇 (南京师范大学) 统编、定稿。参加本书编写的还有梁敬东、曹弋、郑进、刘中、刘金定、姜宁秋、刘怀、吴明祥、刘建等。

由于作者水平有限, 不当之处在所难免, 恳请读者批评指正。

意见、建议请发送到: easybooks@163.com。

编 者

2016 年 5 月

第 1 部分 实验

1

实验 0	Visual C++开发环境	1
0.1	认识 Visual C++ 6.0 开发环境	2
0.1.1	创建工作文件夹	2
0.1.2	启动 Visual C++	2
0.1.3	创建并运行控制台应用程序	3
0.1.4	认识开发环境布局	6
0.2	工具栏操作	7
0.2.1	认识工具栏	7
0.2.2	显示和隐藏	8
0.2.3	浮动与停泊	9
0.3	使用项目工作区窗口	10
0.3.1	认识项目工作区窗口	10
0.3.2	项目工作区窗口常用操作	11
0.4	C++程序的基本开发	11
0.4.1	C++程序基本框架	12
0.4.2	新建 C++程序	13
0.4.3	修正语法错误	14
0.4.4	退出 Visual C++ 6.0	16
0.5	常见问题处理	16
0.6	思考与练习	17
实验 1	C++程序基础	18
1.1	类的设计	19
1.1.1	设计基类 CPerson	19
1.1.2	派生 CStudent 类和 CTeacher 类	22
1.2	虚函数	24
1.2.1	设计输入虚函数 Input	24
1.2.2	设计输出虚函数 Output	27
1.3	数据模型和操作	29
1.3.1	动态数组	29

1.3.2	添加、删除和查找	30
1.3.3	较完整的人员信息管理	33
1.4	常见问题处理	35
1.5	思考与练习	36
实验 2	Windows 编程基础	37
2.1	SDK 编程	38
2.1.1	基于 SDK 的 Win32 程序框架	38
2.1.2	创建控件和提示文本	40
2.1.3	获取并输出计算结果	42
2.2	MFC 编程	43
2.2.1	MFC 程序框架	44
2.2.2	WM_CREATE 消息及其映射	45
2.2.3	按钮消息映射	47
2.3	MFC 向导	48
2.3.1	创建对话框应用程序	48
2.3.2	添加和布局控件	49
2.3.3	映射消息并完善代码	50
2.4	常见问题处理	51
2.5	思考与练习	52
实验 3	常用控件	53
3.1	简单计算器	53
3.1.1	设计计算器对话框	54
3.1.2	映射并控制输入	56
3.1.3	解析并输出结果	60
3.2	控件图案绘制	64
3.2.1	设计图案绘制对话框	64
3.2.2	WM_PAINT 和控件绘制	65
3.2.3	图案及其颜色调整	68
3.3	管理学生的个人信息	70
3.3.1	设计主对话框	70
3.3.2	添加并设计个人信息对话框	71
3.3.3	完善个人信息操作	73
3.4	常见问题处理	77
3.5	思考与练习	78
实验 4	菜单、工具和状态栏	79
4.1	菜单设计和实现	79
4.1.1	CRichEditView 框架	80
4.1.2	设计“格式”菜单	81
4.1.3	映射和更新菜单命令	82

4.1.4	快捷菜单和快捷键	83
4.2	工具栏和状态栏	84
4.2.1	添加并设计“格式”工具栏	85
4.2.2	设计 CStyleBar 类	87
4.2.3	使用 CStyleBar	92
4.2.4	显示行号和列号	95
4.3	简单调试	97
4.3.1	设置断点	97
4.3.2	控制程序运行	98
4.3.3	查看和修改变量的值	100
4.4	常见问题处理	102
4.5	思考与练习	103
实验 5	框架窗口、文档和视图	104
5.1	表单 Ex_Form	104
5.1.1	设计表单	105
5.1.2	可序列化类	107
5.1.3	实现数据操作	109
5.2	视图切换	113
5.2.1	添加列表视图	113
5.2.2	视图的切换	115
5.3	切分窗口	117
5.3.1	目录树	118
5.3.2	文件列表	121
5.3.3	切分实现	124
5.4	常见问题处理	126
5.5	思考与练习	126
实验 6	图形和文本	127
6.1	针式时钟	127
6.1.1	设计对话框	128
6.1.2	绘制时钟	128
6.1.3	WM_TIMER	133
6.2	一个简单的 CAD 程序	135
6.2.1	框架和数据流	135
6.2.2	动态绘制	138
6.2.3	对象拾取	145
6.2.4	属性修改	149
6.3	枚举系统字体	152
6.3.1	设计枚举字体对话框	153
6.3.2	枚举字体	154

6.3.3	显示样本	155
6.4	文字特效	157
6.4.1	特效框架	158
6.4.2	设计文字变形	161
6.4.3	使用文字变形	163
6.5	常见问题处理	166
6.6	思考与练习	166
实验 7	数据库编程	167
7.1	MFC ODBC	167
7.1.1	数据库和数据源	168
7.1.2	记录列表显示	170
7.1.3	添加、修改和删除	172
7.2	MFC DAO	177
7.2.1	界面框架	177
7.2.2	DAO 支持	178
7.2.3	操作 MDB	179
7.3	ADO 编程	183
7.3.1	数据库和框架	183
7.3.2	多表项显示	185
7.3.3	记录添加	187
7.4	常见问题处理	191
7.5	思考与练习	192

第2部分 综合应用实习 193

实验 8	学生信息管理系统设计	193
8.1	系统需求分析	193
8.1.1	系统功能	193
8.1.2	数据库	194
8.2	系统设计	195
8.2.1	界面设计	195
8.2.2	模块及接口	196
8.3	编程与实现	199
8.4	系统测试	213
8.5	应用程序发布	214

实 验

实验 0 Visual C++ 开发环境

Visual C++ 是 Microsoft 公司推出的基于 Windows 平台的可视化编程环境。Visual C++ 6.0 是在以往版本不断更新的基础上形成的, 由于其功能强大、灵活性好、完全可扩展以及具有强有力的 Internet 支持, 因此在各种 C++ 语言开发工具中脱颖而出, 成为目前最为流行的 C++ 语言集成开发环境。

Visual C++ 6.0 分为标准版、专业版和企业版 3 种, 但其基本功能是相同的。Visual C++ 6.05 中文版是在 Visual C++ 6.0 基础上进行汉化的一个版本, 本书以此版本作为编程环境。为统一起见, 本教程仍称之为 Visual C++ 6.0, 并以 Windows XP 作为操作系统平台。

本实验(实训)先来介绍如何启动 Visual C++ 6.0, 并以一个简单的控制台应用程序创建和运行过程为例, 来帮助认识并操作 Visual C++ 6.0 开发环境, 其中包括开发环境的菜单、工具栏等的具体布置和用法。同时, 还以一个基本的 C++ 程序为实例, 操作项目工作区窗口, 并对简单的程序错误进行修改。在这些内容之后还将以“常见问题”的形式来总结一些技巧和方法, 如编译死机时的处理等。

实验目的

- ▶ 熟悉 Visual C++ 开发环境。
- ▶ 学会编译 C++ 程序。
- ▶ 能通过项目工作区窗口对程序进行打开、定位、添加等编辑操作。
- ▶ 学会修正常见的程序错误。
- ▶ 熟悉常见问题的处理方法和技巧。

实验内容

- ▶ 认识 Visual C++ 6.0 开发环境。
- ▶ 掌握工具栏操作。
- ▶ 使用项目工作区窗口。
- ▶ 熟悉 C++ 程序的基本开发。

实验准备和说明

- ▶ PC 中已安装好 Visual C++6.0 最新中文版。
- ▶ 熟悉 Windows 操作系统的环境和基本操作。

0.1 认识 Visual C++ 6.0 开发环境

这里首先通过一个“控制台应用程序”的创建和运行来介绍 Visual C++ 6.0 开发环境。所谓“控制台应用程序”，是指那些需要与传统 DOS (Disk Operating System, 磁盘操作系统) 保持程序上的某种兼容，同时又不需要为用户提供完善界面的程序。简单地讲，就是指在 Windows 环境下运行的 DOS 程序。

具体的实验（实训）过程如下：

- 创建工作文件夹。
- 启动 Visual C++ 6.0。
- 创建并运行控制台应用程序。
- 认识开发环境布局。

0.1.1 创建工作文件夹

由于 Visual C++对应用程序是采用文件夹的方式来管理的，即一个程序项目的所有源代码、编译的中间代码、连接的可执行文件等内容均放置在与程序项目名同名的文件夹中及其下的 debug（调试）或 release（发行）子文件夹中。因此，在用 Visual C++进行应用程序开发时，一般先要创建一个工作文件夹，以便于集中管理和查找。

为本实验（实训）创建 Visual C++ 6.0 的工作文件夹“D:\Visual C++程序\LiMing”（LiMing 是自己的名字），以后所有创建的应用程序项目都在此文件夹下。在文件夹“LiMing”下再创建一个子文件夹“0”，这样本次课实验（实训）程序均在该文件夹下。下一次就在“LiMing”文件夹下创建子文件夹“1”，以此类推。

0.1.2 启动 Visual C++

启动 Visual C++ 6.0 有以下两种方法：

- 在 Windows XP 窗口左下角选择“开始”→“程序”→Microsoft Visual C++ 6.0→Microsoft Visual C++ 6.0。需要说明的是，若是安装了 Visual Studio 版本，则应选择“开始”→“程序”→Microsoft Visual Studio 6.0→Microsoft Visual C++ 6.0。
- 在安装文件夹“...\Program Files\Microsoft Visual Studio\Common\MSDev98\Bin\”下找到可执行文件 MSDEV.EXE，然后双击它。

Visual C++ 6.0 一旦启动后，将显示如图 0.1 所示的“每日提示”对话框。单击  按钮，可看到有关各种操作的提示。如果去除“启动时显示提示”复选框的选中标记“✓”，那么下一次运行 Visual C++ 6.0 将不再出现此对话框。

单击  按钮关闭此对话框，进入 Visual C++ 6.0 开发环境，如图 0.2 所示。

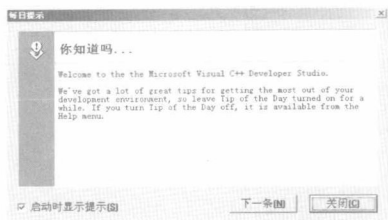


图 0.1 “每日提示”对话框

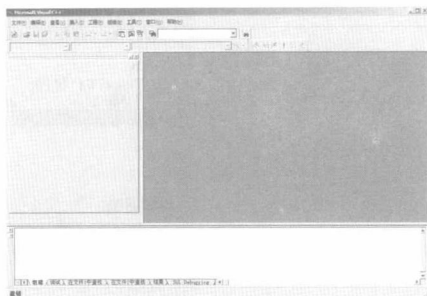


图 0.2 无项目的 Visual C++ 界面

0.1.3 创建并运行控制台应用程序

创建并运行控制台应用程序具体步骤如下：

(1) 在 Visual C++ 6.0 开发环境的最上层菜单中选择“文件”→“新建”菜单命令，打开应用程序向导，弹出“新建”对话框，如图 0.3 所示。单击“工程”标签，从列表框中选中 Win32 Console Application (Win32 控制台应用程序) 项 (图 0.3 中的标记 1)。

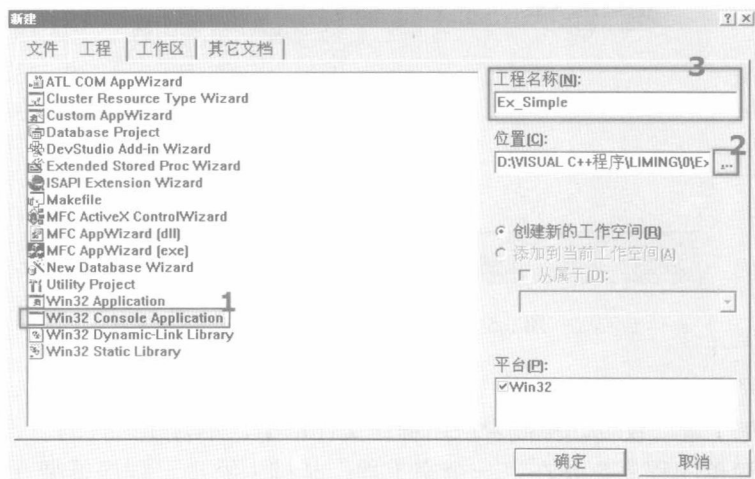


图 0.3 “新建”对话框工程页面

(2) 单击“位置”编辑框右侧的浏览按钮  (图 0.3 中的标记 2)，从弹出的“选择目录”对话框指定项目所在的文件夹，如图 0.4 所示 (图中的数字标记表示最经常的操作次序，以后同，特此说明)。单击  按钮，“选择目录”对话框退出，回到如图 0.3 所示的“新建”对话框。

(3) 在“新建”对话框的“工程名称”编辑框 (图 0.3 中的标记 3) 中输入名称 Ex_Simple，保留“平台”下“Win32”复选框的默认选中 ☒ 状态，单击  按钮进入下一步。

(4) 出现 Win32 Console Application (Win32 控制台应用程序) 向导的步骤 1 (共 1 步) 对话框，从中可选择要创建的应用程序类型，如图 0.5 所示。

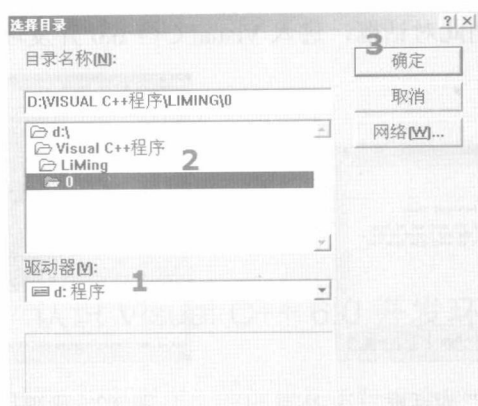


图 0.4 “选择目录”对话框

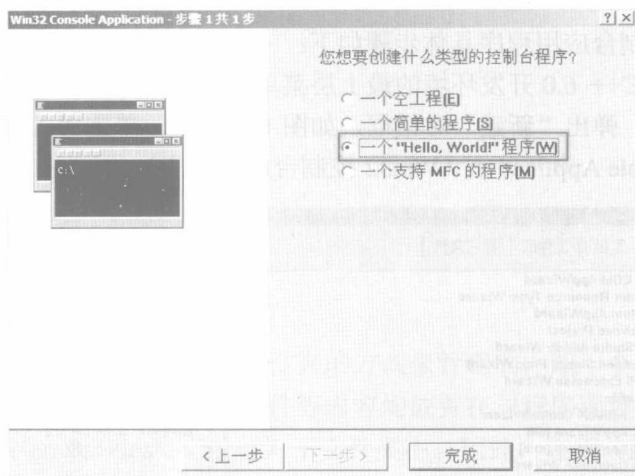


图 0.5 应用程序的向导对话框

这些应用程序类型的区别在于:

- “一个空工程”仅创建控制台应用程序文件框架,不含任何代码。
- “一个简单的程序”是在“一个空工程”基础上添加了程序框架(有入口函数、`#include`指令等)。
- “一个“Hello, World!”程序”在“一个简单的程序”基础上增加了 C 语言的 `printf` 函数调用,用来输出“Hello World!”。
- “一个支持 MFC 的程序”则是支持 MFC 的控制台应用程序框架,由于其相对复杂且一般人也不会使用它,所以这里不再赘述。

(5) 选中“一个“Hello, World!”程序”,单击  按钮,弹出“新建工程信息”对话框,如图 0.6 所示。单击  按钮,系统将按前面的选择自动创建此应用程序。

(6) 又回到 Visual C++开发环境,在菜单栏下面可以看到许多工具条,单击编译工具条  上的生成工具按钮  或直接按快捷键 F7,系统开始对 Ex_Simple 项目工程中的文件进行编译、连接,同时在输出窗口中观察出现的内容,当出现

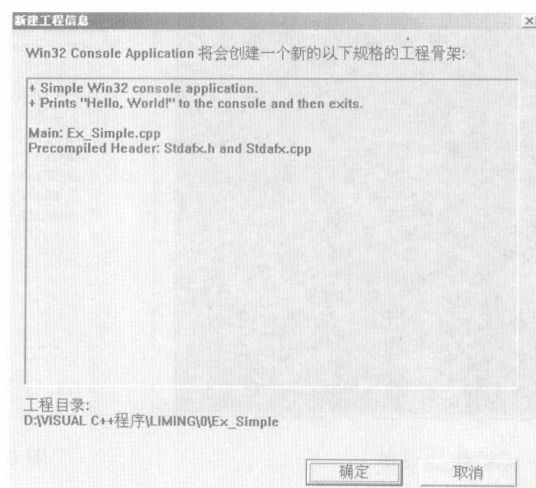


图 0.6 “新建工程信息”对话框

Ex_Simple.exe - 0 error(s), 0 warning(s)

表明 Ex_Simple.exe 可执行文件已经正确无误地生成了。单击编译工具条上的运行工具按钮或直接按快捷键 Ctrl+F5, 就可以运行刚刚生成的 Ex_Simple.exe, 结果如图 0.7 所示。

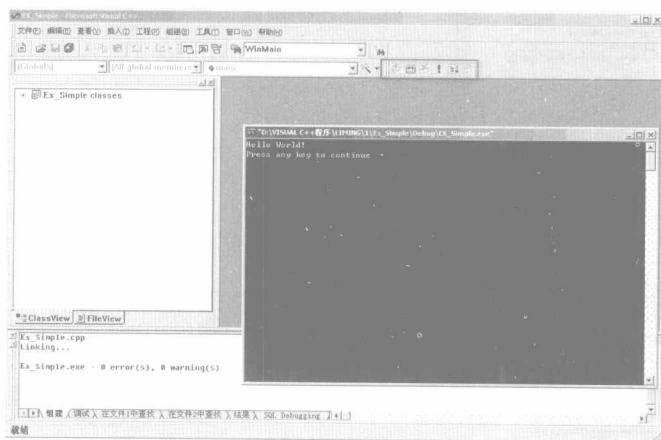


图 0.7 开发环境和运行结果

需要说明的是:

(1) 控制台窗口中, “Press any key to continue” 是系统自动加上去的, 表示 Ex_Simple 运行后, 按任意键返回到 Visual C++ 6.0 开发环境。

(2) 默认情况下, 控制台应用程序运行后的控制台窗口是黑底白字, 必要时可对其属性进行修改, 具体方法如下: 在控制台窗口标题上的最左边单击图标, 从弹出的快捷菜单中选择“属性”菜单命令, 如图 0.8 所示, 弹出“属性”对话框, 如图 0.9 所示。从中可以看出, “属性”对话框含有“选项”、“字体”、“布局”和“颜色”4 个标签页面。通过这些页面可以设置相应的属性。

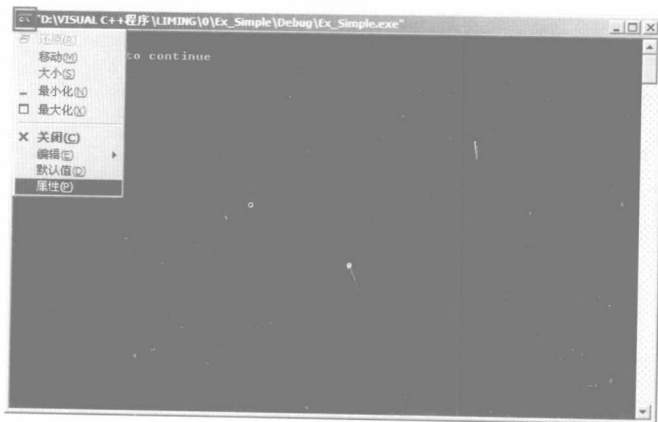


图 0.8 标题快捷菜单

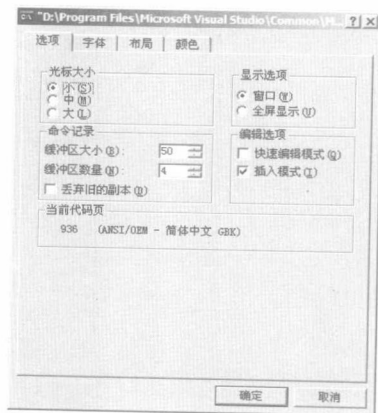


图 0.9 “属性”对话框

0.1.4 认识开发环境布局

项目创建的 Visual C++ 6.0 开发环境如图 0.10 所示, 它由标题栏、菜单栏、工具栏、项目工作区窗口、文档窗口、输出窗口以及状态栏等组成。

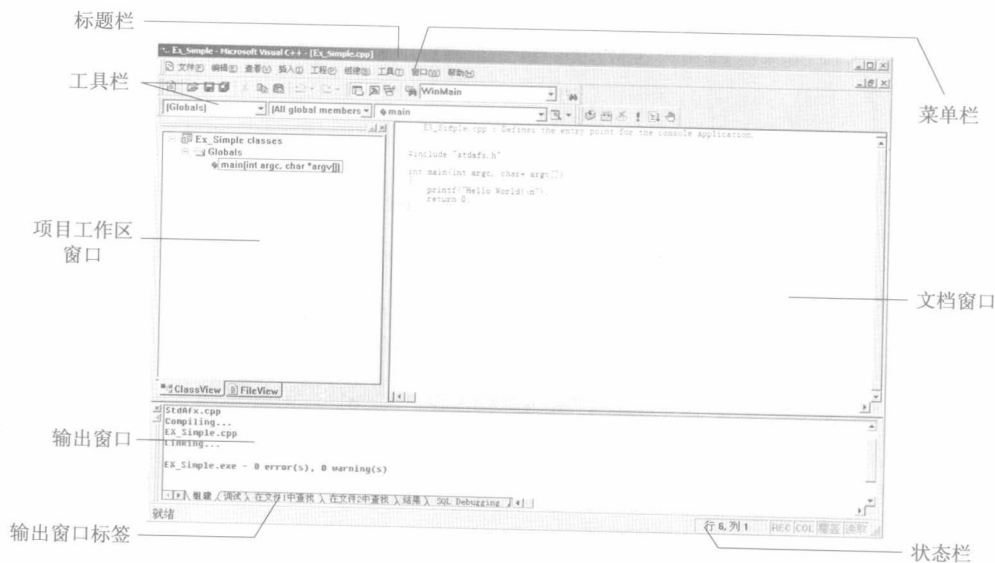


图 0.10 Visual C++ 6.0 开发环境(有项目)

标题栏处在开发环境的最上层, 一般有“最小化”()、“最大化”()或“还原”()以及“关闭”()按钮, 单击 按钮将退出开发环境。事实上, 标题栏上还会显示出当前文档窗口中正在操作的文档的文件名。

标题栏的下面就是菜单栏, 它包含开发环境中几乎所有的命令, 提供文档操作、程序编译、调试、窗口操作等一系列的功能。菜单栏往往和工具栏同处在一个区域, 并且菜单中的一些常用命令还被排列在相应的工具栏上, 以便更快捷地操作。

在开发环境的左侧是一个带有多个标签页面的项目工作区窗口，它包含当前项目的几乎所有的信息，包括项目中的类（ClassView 页面）信息、文件（FileView 页面）信息以及 Windows 项目的资源（ResourceView 页面）信息等。在这些项目信息中的文字或图标处右击鼠标都会弹出相应的快捷菜单，从中可选择当前状态下的一些快捷操作。

文档窗口一般位于开发环境中的右边，各种程序代码的源文件、资源文件、文档文件等都可以通过文档窗口显示出来。

输出窗口一般出现在开发环境窗口的底部，包括编译（Build，组建）、调试（Debug）、在文件中查找（Find in Files）等相关信息的输出。这些输出信息以多页面标签的形式出现在输出窗口中，例如，“组建”页面标签显示的是程序在编译和连接时的进度及错误信息。

状态栏一般位于开发环境的最底部，它用来显示当前操作状态、注释、文本光标所在的行列号等信息。

0.2 工具栏操作

菜单栏下面是工具栏。工具栏上的按钮通常和一些菜单命令相对应，提供执行经常使用的命令的一种快捷方式。具体的实验（实训）过程如下：

- 认识工具栏。
- 工具栏的显示和隐藏。
- 工具栏的浮动和停泊。

0.2.1 认识工具栏

Visual C++ 6.0 开发环境默认显示的工具栏有：编译微型条（Build MiniBar）工具栏、标准（Standard）工具栏、向导（WizardBar）工具栏。

1) 编译微型条工具栏

编译微型条工具栏提供了常用的编译、连接操作命令，如图 0.11 所示，前面已使用过。表 0.1 列出了各个按钮命令的含义。

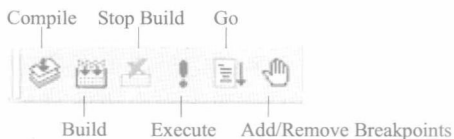


图 0.11 编译微型条工具栏

表 0.1 编译微型条工具栏按钮命令及功能描述

按钮命令	功能描述
Compile	编译 C 或 C++源代码文件
Build	生成应用程序的 EXE 文件
Stop Build	停止编译、连接
Execute	执行应用程序
Go	单步执行
Add/Remove Breakpoints	插入或删除断点

2) 标准工具栏

如图 0.12 所示，标准工具栏中的工具按钮命令大多数是常用的文档编辑命令，如新建、保存、撤销、恢复、查找等，表 0.2 列出了各个按钮命令的含义。

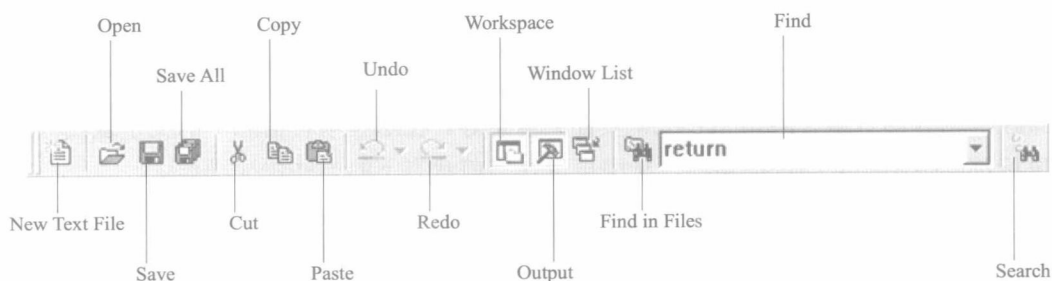


图 0.12 标准工具栏

表 0.2 标准工具栏按钮命令及功能描述

按钮命令	功能描述
New Text File	新建一个文本文件
Open	打开已存在的文件
Save	保存当前文件
Save All	保存所有打开的文件
Cut	将当前选定的内容剪切掉,并移至剪贴板中
Copy	将当前选定的内容复制到剪贴板中
Paste	将剪贴板中的内容粘贴到光标当前位置处
Undo	撤销上一次操作
Redo	恢复被撤销的操作
Workspace	显示或隐藏项目工作区窗口
Output	显示或隐藏输出窗口
Window List	文档窗口操作
Find in Files	在指定的多个文件中查找字符串
Find	指定要查找的字符串,按 Enter 键进行查找
Search	在当前文件中查找指定的字符串

3) 向导工具栏

向导工具栏是将 MFC ClassWizard (MFC 类向导,以后还会专门对其操作)对话框的类似功能体现为 3 个相互关联的组合框和一个 Actions 控制按钮,如图 0.13 所示。



图 0.13 WizardBar

3 个组合框分别表示要操作的类 (Class)、选择相应类的过滤器 (Filter) 和相应类的成员函数 (Members) 等。单击 Actions 控制按钮可将文本指针移动到指定类成员函数在相应的源文件的定义或声明的位置处,单击 Actions 向下按钮 (▼) 会弹出一个快捷菜单,从中可以选择要执行的命令。

需要说明的是,上述工具栏上的按钮有时是处于未激活状态,例如,标准工具栏的复制按钮  在没有选定对象前是灰显的,此时用户无法使用它。

0.2.2 显示和隐藏

在 Visual C++开发环境中,可以对有  (把手) 的工具栏 (或菜单栏) 进行显示与隐

藏、浮动与停泊等操作。就工具栏的显示或隐藏来说，有“定制”对话框和快捷菜单两种方式：

- 选择“工具”菜单→“定制”菜单命令项，弹出“定制”对话框，如图 0.14 所示；单击“工具栏”页面标签，将显示出所有的工具栏，名称前面将带有选中标记✓的 是在开发环境上显示的工具栏。
- 在开发环境的工具栏区域的空白处右击鼠标，这时就会弹出一个包含工具栏名称的快捷菜单，如图 0.15 所示。若要显示某工具栏，只需要单击该工具栏名称，使得前面的复选框带有选中标记即可。同样的操作再进行一次，工具栏名称前面的复选框的选中标记将消失，该工具栏就会从开发环境中消失。

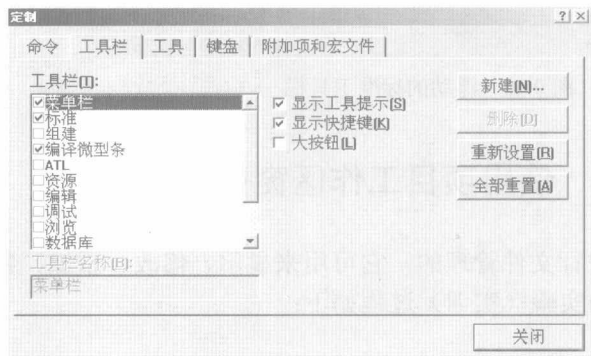


图 0.14 “定制”对话框



图 0.15 快捷菜单

边练边试

用“定制”对话框方式显示所有的工具栏，然后用快捷菜单方式隐藏，使工具栏恢复到默认的界面。

0.2.3 浮动与停泊

对于工具栏的“浮动”与“停泊”来说，当 Visual C++ 6.0 启动后，系统默认将常用工具栏“停泊”在主窗口的顶部。若将鼠标指针移至工具栏的☞（把手）处或其他非按钮区域，然后按住鼠标左键，可以将工具栏拖动到主窗口的四周或中央，此时会出现预停泊的细框线，松开鼠标，被拖动的工具栏停泊。

若将工具栏拖动到窗口的中央处松开鼠标左键，则工具栏成为“浮动”的工具窗口，此时窗口的标题就是该工具栏的名称，拖动工具栏窗口的边或角框可以改变其形状。例如，图 0.16 是“标准”工具栏浮动的状态，其大小已被改变过。

当然，浮动和停泊两种状态可以进行切换。在“浮动”的工具窗口标题栏处双击鼠标或将其拖放到主窗口的四周，都能使其停泊在相应的位置处。在“停泊”工具栏的非按钮区域双击鼠标，可切换成“浮动”的工具窗口。

边练边试

将“标准”工具栏依次向窗口的四周“停泊”，然后恢复到默认的位置。