



中等职业学校教学用书(计算机技术专业)

Visual C++ 6.0 程序设计案例教程

◎ 沈大林 杨旭 主编

本书

料包



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

中等职业学校教学用书（计算机技术专业）

Visual C++ 6.0 程序设计案例教程

沈大林 杨 旭 主编

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

C++语言是当今最流行的面向对象程序设计语言,而 Visual C++是 Microsoft 公司开发的基于 Windows 操作系统的 C++语言程序的可视化编程工具。Visual C++ 6.0 是众多 Visual C++中的最高版本。Visual C++的执行速度快,对操作系统的访问权限高,这些是其他许多编程工具所无法比拟的。因此,Visual C++成为众多程序员在 Windows 操作系统下编程的首选工具。

本书共分为 8 章,全书提供了 37 个案例,结合案例介绍相关的知识点。本书采用任务驱动式的教学方法,读者可以边模仿练习各章的程序设计案例,边学习各案例中涉及的知识和概念,将理论知识学习与实际操作有机地结合在一起,达到事半功倍的效果。本书的作者由计算机教师和程序员组成,他们都有着丰富的教学经验和程序设计经验。

本书适应社会的需求、企业的需求、人才的需求和学校的需求,可以作为中职中专和高职高专的教材及培训学校的培训教材,还可以作为程序设计爱好者的自学用书。

为了方便教师教学,本书还配有电子教学参考资料包(包括教学指南、电子教案及习题答案),详见前言。

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,侵权必究。

图书在版编目(CIP)数据

Visual C++ 6.0 程序设计案例教程/沈大林, 杨旭主编. —北京: 电子工业出版社, 2005.10
中等职业学校教学用书(计算机技术专业)
ISBN 7-121-01303-7

I. V… II. ①沈…②杨… III. C 语言—程序设计—专业学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2005)第 103832 号

责任编辑: 李 影 刘真平

印 刷: 北京季蜂印刷有限公司

出版发行: 电子工业出版社

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: 21.75 字数: 556.8 千字

印 次: 2005 年 10 月第 1 次印刷

印 数: 5 000 册 定价: 27.00 元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。联系电话:(010) 68279077。质量投诉请发邮件至 zlts@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

中等职业学校教材工作领导小组

组 长：陈贤忠 安徽省教育厅厅长

副 组 长：李雅玲 信息产业部人事司技术干部处处长

尚志平 山东省教学研究室副主任

睦 平 江苏省教育厅职社处副处长

苏渭昌 教育部职业技术教育中心研究所主任

王传臣 电子工业出版社副社长

组 员：（排名不分先后）

唐国庆 湖南省教科院

张志强 黑龙江省教育厅职成教处

李 刚 天津市教委职成教处

王润拽 内蒙古自治区教育厅职成教处

常晓宝 山西省教育厅职成教处

刘 晶 河北省教育厅职成教处

王学进 河南省职业技术教育教学研究室

刘宏恩 陕西省教育厅职成教处

吴 蕊 四川省教育厅职成教处

左其琨 安徽省教育厅职成教处

陈观诚 福建省职业技术教育中心

邓 弘 江西省教育厅职成教处

姜昭慧 湖北省职业技术教育研究中心

李栋学 广西自治区教育厅职成教处

杜德昌 山东省教学研究室职教室

谢宝善 辽宁省基础教育教研培训中心职教部

安尼瓦尔·吾斯曼 新疆自治区教育厅职成教处

秘 书 长：李 影 电子工业出版社

副秘书长：蔡 葵 电子工业出版社

前言



C++语言是当今最流行的面向对象程序设计语言，而 Visual C++ 是 Microsoft 公司开发的基于 Windows 操作系统的 C++ 语言程序的可视化编程工具。Visual C++ 6.0 是众多 Visual C++ 中的最高版本。Visual C++ 的执行速度快，对操作系统的访问权限高，这些是其他许多的编程工具所无法比拟的。因此，Visual C++ 成为众多程序员在 Windows 操作系统下编程的首选工具。

要真正地掌握 Visual C++，必须对 C/C++ 语言编程有深入的了解，应理解 MFC 库和 Windows 操作系统下的编程方法。在 Windows 操作系统下编程，通常通过调用 Windows API 得以实现。Visual C++ 6.0 将大量的 Windows API 以 C++ 格式进行封装，通过 MFC 方式提供给开发人员，大大简化了开发人员的编程工作。因而，要学好 Visual C++，MFC 的掌握是关键。

Visual C++ 6.0 还提供了两个功能强大的编程工具：AppWizard（应用程序向导）和 ClassWizard（类向导）。利用 AppWizard 可以在很短的时间内创建出 Windows 应用程序的框架；而 ClassWizard，则可以在应用程序框架之上迅速增加新的类、成员变量与函数，使 Windows 编程得以快速实现。两者结合，让 Windows 操作系统下的编程变得非常简单。

本书共分为 8 章，第 1 章介绍 Visual C++ 6.0 程序设计基础，第 2 章介绍面向对象程序设计，第 3 章介绍可视化程序设计，第 4 章介绍 Visual C++ 6.0 资源应用，第 5 章介绍图形与文本，第 6 章介绍基于对话框的应用程序，第 7 章介绍文档与视图，第 8 章介绍综合应用程序开发。全书提供 37 个案例，并结合案例介绍了相关的知识点。

本书的作者由计算机教师和程序员组成，他们都有着丰富的教学经验和程序设计经验。在工作中，作者们总结了一套任务驱动式的教学方法，比传统教学方法更容易讲授和学习。采用这种方法，可以使读者快速掌握软件的操作方法、程序设计方法及程序设计技巧。读者可以边模仿练习各章的程序设计案例，边学习各案例中涉及的知识和概念，将理论知识学习与实际操作有机地结合在一起，达到事半功倍的效果。

本书由沈大林和杨旭主编，王爱贞、关点、曾浩、王浩轩、曲彭生、关山、张伦、董鑫、赵亚辉、朱学亮、胡野红、生强、季明辉、石淳、沈昕、肖柠朴、姜树昕、耿文杰、刘宇昕、耿大伟、张铮、张磊、马开颜、陈一兵、毕凌云、李征、郝侠、刘庆荣、李瑞梅、崔元如、于金霞、赵艳霞等也参加了本书的编写工作。

本书适应了社会的需求、企业的需求、人才的需求和学校的需求，可以作为中职中专和高职高专的教材及培训学校的培训教材，还可以作为程序设计爱好者的自学用书。

由于技术的不断变化以及操作过程中的疏漏，书中难免有错误和不足之处，恳请广大读者批评指正。

为了方便教师教学，本书还配有教学指南、电子教案及习题答案（电子版），请有此需要的教师登录华信教育资源网（<http://www.hxedu.com.cn>）下载或与电子工业出版社联系，我们将免费提供。E-mail:ve@phei.com.cn。

编者

2005年6月



目 录



第 1 章 Visual C++ 6.0 程序设计基础	(1)
1.1 Visual C++ 6.0 概述	(1)
1.1.1 Visual C++ 6.0 简介	(1)
1.1.2 面向对象程序设计	(1)
1.2 案例 1: 欢迎进入 Win32 的世界	(2)
1.2.1 学习目标	(2)
1.2.2 程序实现	(2)
1.2.3 知识要点	(8)
习题 1-1	(14)
1.3 案例 2: 快速可视化程序的开发	(15)
1.3.1 学习目标	(15)
1.3.2 程序实现	(15)
1.3.3 知识要点	(17)
习题 1-2	(25)
1.4 案例 3: 让程序正确运行起来	(25)
1.4.1 学习目标	(25)
1.4.2 程序实现	(26)
1.4.3 知识要点	(29)
习题 1-3	(39)
第 2 章 面向对象程序设计	(41)
2.1 案例 4: 圆、圆周长与圆面积	(41)
2.1.1 学习目标	(41)
2.1.2 程序实现	(41)
2.1.3 知识要点	(43)
习题 2-1	(54)
2.2 案例 5: 相似三角形的比较	(54)
2.2.1 学习目标	(54)
2.2.2 程序实现	(55)
2.2.3 知识要点	(56)
习题 2-2	(66)
2.3 案例 6: 圆与圆柱	(66)

2.3.1 学习目标	(66)
2.3.2 程序实现	(67)
2.3.3 知识要点	(68)
习题 2-3	(76)
2.4 案例 7: 对象的通用接口	(76)
2.4.1 学习目标	(76)
2.4.2 程序实现	(77)
2.4.3 知识要点	(79)
习题 2-4	(84)
2.5 案例 8: 个人文档编辑器	(85)
2.5.1 学习目标	(85)
2.5.2 程序实现	(85)
2.5.3 知识要点	(87)
习题 2-5	(92)
第 3 章 可视化程序设计	(93)
3.1 案例 9: 窗口程序	(93)
3.1.1 学习目标	(93)
3.1.2 程序实现	(93)
3.1.3 知识要点	(96)
习题 3-1	(102)
3.2 案例 10: MFC 框架程序	(102)
3.2.1 学习目标	(102)
3.2.2 程序实现	(103)
3.2.3 知识要点	(104)
习题 3-2	(111)
3.3 案例 11: 动画窗口	(111)
3.3.1 学习目标	(111)
3.3.2 程序实现	(112)
3.3.3 知识要点	(116)
习题 3-3	(121)
3.4 案例 12: 文字的输入与显示	(122)
3.4.1 学习目标	(122)
3.4.2 程序实现	(122)
3.4.3 知识要点	(125)
习题 3-4	(127)
3.5 案例 13: 鼠标坐标实时探测	(127)
3.5.1 学习目标	(127)
3.5.2 程序实现	(128)
3.5.3 知识要点	(131)

习题 3-5	(133)
3.6 案例 14: 滚动的图像	(133)
3.6.1 学习目标	(133)
3.6.2 程序实现	(134)
3.6.3 知识要点	(137)
习题 3-6	(139)
3.7 案例 15: 应用程序的生存周期	(140)
3.7.1 学习目标	(140)
3.7.2 程序实现	(140)
3.7.3 知识要点	(142)
习题 3-7	(144)
第 4 章 Visual C++ 6.0 资源应用	(146)
4.1 案例 16: 动态光标	(146)
4.1.1 学习目标	(146)
4.1.2 程序实现	(146)
4.1.3 知识要点	(150)
习题 4-1	(154)
4.2 案例 17: 图标变换	(154)
4.2.1 学习目标	(154)
4.2.2 程序实现	(154)
4.2.3 知识要点	(159)
习题 4-2	(164)
4.3 案例 18: 上下文菜单	(164)
4.3.1 学习目标	(164)
4.3.2 程序实现	(165)
4.3.3 知识要点	(168)
习题 4-3	(174)
4.4 案例 19: 动态菜单	(174)
4.4.1 学习目标	(174)
4.4.2 程序实现	(174)
4.4.3 知识要点	(177)
习题 4-4	(178)
4.5 案例 20: 图像窗口	(179)
4.5.1 学习目标	(179)
4.5.2 程序实现	(179)
4.5.3 知识要点	(183)
习题 4-5	(186)
4.6 案例 21: 图像浏览器	(187)
4.6.1 学习目标	(187)

4.6.2 程序实现	(187)
4.6.3 知识要点	(194)
习题 4-6	(200)
4.7 案例 22: 绘图工具栏	(201)
4.7.1 学习目标	(201)
4.7.2 程序实现	(202)
4.7.3 知识要点	(207)
习题 4-7	(213)
4.8 案例 23: 状态栏消息	(214)
4.8.1 学习目标	(214)
4.8.2 程序实现	(214)
4.8.3 知识要点	(219)
习题 4-8	(222)
第 5 章 图形与文本	(224)
5.1 案例 24: 绘图画板	(224)
5.1.1 学习目标	(224)
5.1.2 程序实现	(224)
5.1.3 知识要点	(229)
习题 5-1	(237)
5.2 案例 25: 五子棋盘	(237)
5.2.1 学习目标	(237)
5.2.2 程序实现	(238)
5.2.3 知识要点	(240)
习题 5-2	(245)
5.3 案例 26: 字体展示	(246)
5.3.1 学习目标	(246)
5.3.2 程序实现	(246)
5.3.3 知识要点	(248)
习题 5-3	(252)
5.4 案例 27: 图形的显示比例	(253)
5.4.1 学习目标	(253)
5.4.2 程序实现	(253)
5.4.3 知识要点	(256)
习题 5-4	(259)
第 6 章 基于对话框的应用程序	(260)
6.1 案例 28: 模态对话框与非模态对话框	(260)
6.1.1 学习目标	(260)
6.1.2 程序实现	(260)
6.1.3 知识要点	(265)

习题 6-1	(267)
6.2 案例 29: 密码登录框	(267)
6.2.1 学习目标	(267)
6.2.2 程序实现	(268)
6.2.3 知识要点	(270)
习题 6-2	(277)
6.3 案例 30: 电子试卷	(278)
6.3.1 学习目标	(278)
6.3.2 程序实现	(278)
6.3.3 知识要点	(281)
习题 6-3	(284)
6.4 案例 31: 花纹图案	(284)
6.4.1 学习目标	(284)
6.4.2 程序实现	(285)
6.4.3 知识要点	(291)
习题 6-4	(294)
6.5 案例 32: 字体演示	(295)
6.5.1 学习目标	(295)
6.5.2 程序实现	(295)
6.5.3 知识要点	(298)
习题 6-5	(300)
6.6 案例 33: 通用对话框	(301)
6.6.1 学习目标	(301)
6.6.2 程序实现	(301)
6.6.3 知识要点	(304)
习题 6-6	(311)
第 7 章 文档与视图	(312)
7.1 案例 34: 文件访问	(312)
7.1.1 学习目标	(312)
7.1.2 程序实现	(312)
7.1.3 知识要点	(314)
习题 7-1	(320)
7.2 案例 35: 我的记事本	(321)
7.2.1 学习目标	(321)
7.2.2 程序实现	(321)
7.2.3 知识要点	(322)
习题 7-2	(323)
第 8 章 综合应用程序开发	(325)
8.1 案例 36: 媒体播放器	(325)

8.1.1 学习目标	(325)
8.1.2 程序实现	(325)
8.1.3 知识要点	(328)
习题 8-1	(328)
8.2 案例 37: 网页浏览器	(329)
8.2.1 学习目标	(329)
8.2.2 程序实现	(329)
8.2.3 知识要点	(332)
习题 8-2	(333)

第1章 Visual C++ 6.0 程序设计基础



1.1 Visual C++ 6.0 概述

1.1.1 Visual C++ 6.0 简介

C++语言是当今最流行的面向对象程序设计语言。可视化与面向对象程序设计是当今最流行的编程技术，Visual C++就是 Microsoft 公司开发的基于 Windows 操作系统的 C++ 语言程序的可视化编程工具。

由于同为 Microsoft 公司的产品，在 Windows 操作系统强有力的支持下，Visual C++ 6.0 的执行速度及对操作系统的访问权限之高，是其他许多的编程工具所无法比拟的。因此，Visual C++ 成为众多程序员在 Windows 操作系统下编程的首选工具。

对于初学者而言，也许是受名称的误导，会觉得 Visual C++ 是一个完全可视化的编程工具，就如同 Visual Basic 那样。其实不然，虽然 Visual C++ 的集成开发环境提供了一个快速编程的框架，极大地提高了编程效率，但是，要真正地掌握 Visual C++，还必须对 C/C++ 语言编程有深入的了解，理解 MFC 库和 Windows 操作系统下的编程方法。

在 Windows 操作系统下编程，通常通过调用 Windows API (Application Program Interface, 应用程序编程接口) 得以实现。Visual C++ 6.0 将大量的 Windows API 以 C++ 格式进行封装，通过 MFC 方式提供给开发人员，大大简化了开发人员的编程工作，提高了工作效率。因而，要学好 Visual C++，关键是掌握 MFC。

在 Visual C++ 6.0 中，可以使用 MFC 完成大多数的工作，也可以直接调用 Windows API 完成一些底层的开发，用户可以完全掌握 Windows 下的资源。用 Visual C++ 6.0 开发的程序代码简练，运行速度快。

在 Visual C++ 6.0 中，提供了两个功能强大的编程工具：AppWizard (应用程序向导) 和 ClassWizard (类向导)。

利用 Visual C++ 6.0 的 AppWizard，可以在很短的时间内创建出 Windows 应用程序的框架；而 ClassWizard 则可以在应用程序框架之上迅速增加新的类、成员变量与函数，使 Windows 编程得以快速实现。两者结合，让 Windows 操作系统下的编程变得非常简单。

1.1.2 面向对象程序设计

Visual C++ 是面向对象的编程工具，对对象及对象在程序中的实现——类的了解，对于学习 Visual C++ 是非常重要的。



面向对象设计的思想是将世界看成由具有独立行为能力的各种对象所组成的，所有对象都有其自身的特性及相关行为。比如，学生可以看成一种对象，学生具有学号、姓名、性别、年龄、班级、成绩等不同的特征，学生还具有吃、喝、运动、学习、娱乐等行为。同样，汽车也是一种对象，汽车具有品牌、车型、载重、排气量等特征，此外还具有加速、减速、刹车等行为。

对象中还可以包含其他对象，也就是说对象可以由多个子对象组成。例如，汽车由车身、车架、发动机、离合器、车轮等子对象组成，计算机由主机、显示器、键盘、鼠标、打印机、扫描仪等子对象组成。

面向对象的程序设计方法（OOP），把具有共性的程序和数据封装起来视为一个对象，每个对象都作为一个完整的独立组件出现在程序中，在程序中只要针对该对象进行简单的语句调用，就能执行所需要完成的功能。

比如，将计算机看做一个对象，当用户需要启动计算机时，所需做的只是按一下开机按钮，然后静静地等待计算机启动，直到出现操作系统界面。但是在启动的过程中，计算机内部经历了接通电源、检测硬件、引导操作系统等一系列的步骤，而这些工作对于使用者来说是无须了解的，使用者只需要知道按下开机按钮，等待出现操作系统界面就可以了。

在面向对象的程序设计方法中，对象（Object）是程序设计的基本单元，是代码和数据的集合。对象都具有属性（数据）和方法（作用于对象的操作）。对象的属性和方法被封装成一个整体，供程序设计者使用。对象之间的作用通过消息传送来实现。

1.2 案例 1：欢迎进入 Win32 的世界

1.2.1 学习目标

我们的第 1 个程序将是一个在 Visual C++ 6.0 的 AppWizard 指导下完成的可视化 Win32 应用程序，如图 1-1 所示。



图 1-1 Win32 应用程序

在实现这个案例的过程中，我们将熟悉 Visual C++ 6.0 集成开发环境，以及 Win32 应用程序的开发过程。

1.2.2 程序实现

1. 启动 Visual C++ 6.0 集成开发环境

在 Windows 操作系统中，单击【开始】→【程序】→【Microsoft Visual Studio 6.0】→



【Microsoft Visual C++ 6.0】项，启动 Visual C++ 6.0 集成开发环境，如图 1-2 所示。

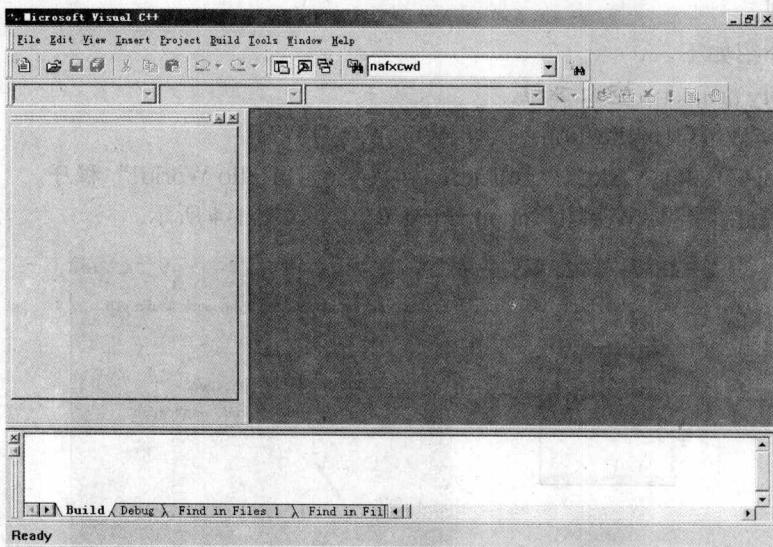


图 1-2 Visual C++ 6.0 集成开发环境

2. 创建 Win32 可视化应用程序框架

在 Visual C++ 6.0 集成开发环境中，单击【File】→【New】菜单命令（或按快捷键 Ctrl+N），此时将弹出【New】对话框，在【New】对话框中选择【Projects】选项卡，然后在左侧的选项框中选择“Win32 Application”选项，在右侧的 Location 框中选择保存项目的路径，在 Project name 框中输入项目的名称“MyPro1”，如图 1-3 所示。

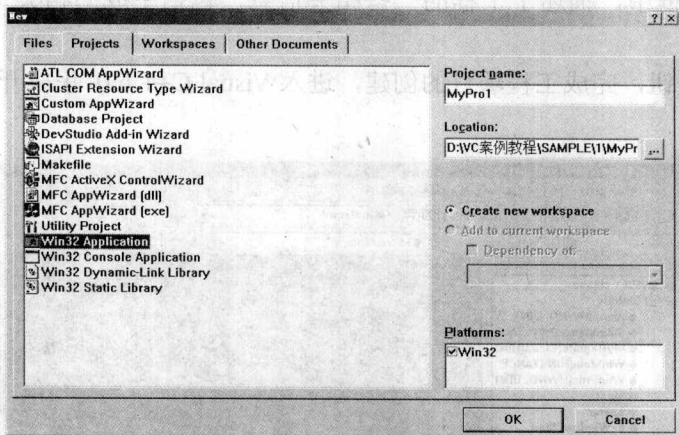


图 1-3 新建 Win32 应用程序



注意

在右下方的 Platforms（平台）框中，需要选中“Win32”。只有在 Platforms 框中选中“Win32”选项，才能建立 32 位的 Windows 程序；否则，建立的将是 16 位的 Windows 程序。



单击【OK】按钮，在接下来的【Win32 Application-Step 1 of 1】对话框中，选择要创建工程项目的类型。

类型选项介绍如下。

- An empty project: 空白工程。
- A simple Win32 application: 简单 Win32 应用程序。
- A typical “Hello World!” application: 典型的 “Hello World!” 程序。

选择 A typical “Hello World!” application 选项，如图 1-4 所示。

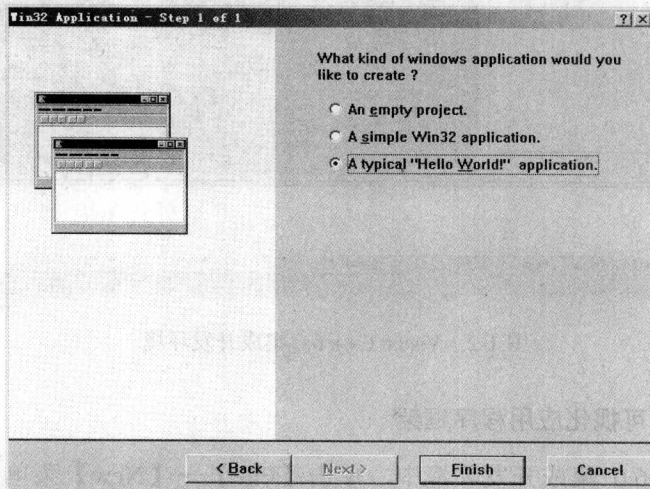


图 1-4 选择创建典型的 “Hello World!” 程序

单击【Finish】按钮，完成工程的创建。

接下来的对话框中，描述了工程的一些相关信息，包括生成的相关文件及文件保存位置等。

单击【OK】按钮，完成工程项目的创建，进入 Visual C++ 6.0 集成开发环境，如图 1-5 所示。

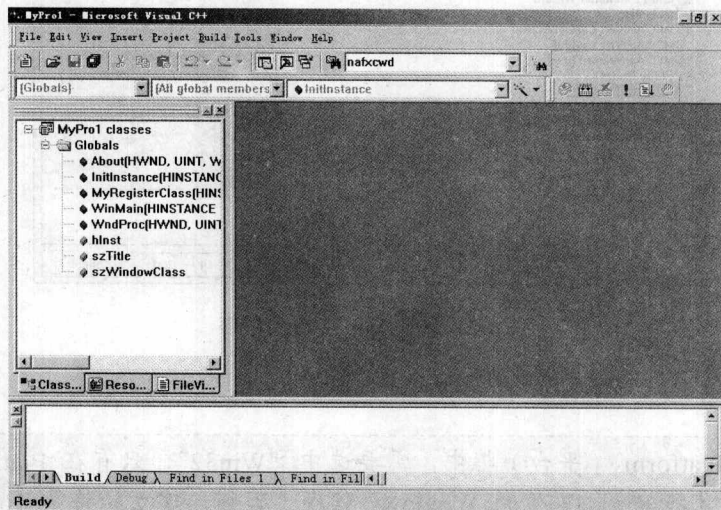


图 1-5 完成创建后的项目



3. 修改程序代码

在 Visual C++ 6.0 集成开发环境左侧【项目工作区】窗口的【ClassView】面板中双击 WndProc 项，在集成开发环境中将出现代码编辑窗口，该窗口中包含将要进行编辑的 WndProc() 函数代码，如图 1-6 所示。

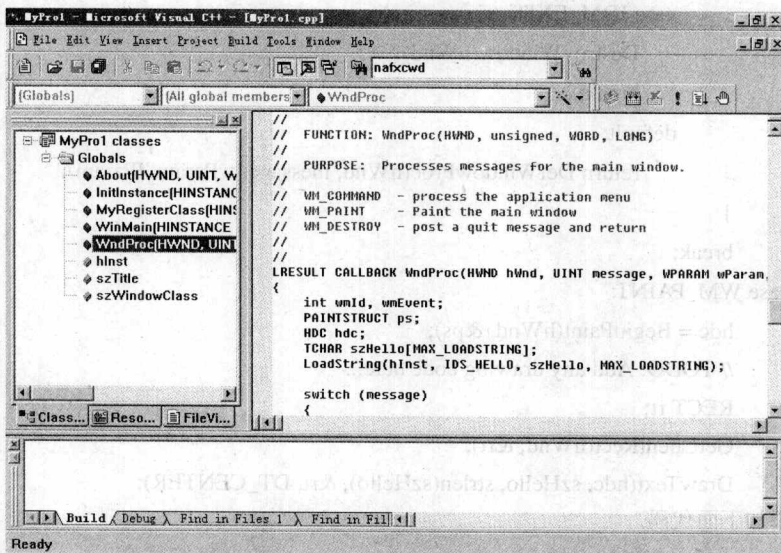


图 1-6 代码编辑窗口

现在，我们可以看到 Visual C++ 6.0 集成开发环境所生成的程序框架，在代码编辑窗口中可以看到大量自动生成的代码。不要被这些代码所吓倒，我们现在并不需要了解所有这些代码都能做些什么，只需要在相应的地方进行简单的修改就可以完成这个案例。

按下面的代码对 WndProc() 函数进行修改（仅需要添加代码中标出的部分即可）。

程序代码：

```
LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam)
{
    int wmId, wmEvent;
    PAINTSTRUCT ps;
    HDC hdc;
    TCHAR szHello[MAX_LOADSTRING];
    LoadString(hInst, IDS_HELLO, szHello, MAX_LOADSTRING);

    switch (message)
    {
        case WM_COMMAND:
            wmId    = LOWORD(wParam);
            wmEvent = HIWORD(wParam);
            // Parse the menu selections:
```