



教育部高职高专规划教材
Jiaoyubu Gaozhi Gaozhuan Guihua Jiaocai

C++/VC 面向对象程序设计基础

张福祥 张莹 李健 于春花 孙永乐 编著

高等教育出版社
HIGHER EDUCATION PRESS



TP312C/1918

版社

教育部高职高专规划教材

C++ / VC 面向对象程序 设计基础

张福祥 张莹 李健 于春花 孙永乐 编著

高等教育出版社

图书在版编目(CIP)数据

C++/VC 面向对象程序设计基础/张福祥等编著. —北京:
高等教育出版社, 2001
教育部高职高专规划教材
ISBN 7-04-009948-9

I. C… II. 张… III. ①C语言—程序设计—高等
学校:技术学校—教材②面向对象语言—程序设计—高
等学校:技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2001)第 037016 号

C++/VC 面向对象程序设计基础

张福祥 张莹 李健 于春花 孙永乐 编著

出版发行 高等教育出版社

社 址 北京市东城区沙滩后街 55 号 邮政编码 100009
电 话 010-64054588 传 真 010-64014048
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>

经 销 新华书店北京发行所
印 刷 北京外文印刷厂

开 本 787×1092 1/16 版 次 2001 年 9 月第 1 版
印 张 15.75 印 次 2001 年 9 月第 1 次印刷
字 数 390 000 定 价 20.00 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

出版说明

教材建设工作是整个高职高专教育教学工作中的重要组成部分。改革开放以来,在各级教育行政部门、学校和有关出版社的共同努力下,各地已出版了一批高职高专教育教材。但从整体上看,具有高职高专教育特色的教材极其匮乏,不少院校尚在借用本科或中专教材,教材建设仍落后于高职高专教育的发展需要。为此,1999年教育部组织制定了《高职高专教育基础课程教学基本要求》(以下简称《基本要求》)和《高职高专教育专业人才培养目标及规格》(以下简称《培养规格》),通过推荐、招标及遴选,组织了一批学术水平高、教学经验丰富、实践能力强的教师,成立了“教育部高职高专规划教材”编写队伍,并在有关出版社的积极配合下,推出一批“教育部高职高专规划教材”。

“教育部高职高专规划教材”计划出版500种,用5年左右时间完成。出版后的教材将覆盖高职高专教育的基础课程和主干专业课程。计划先用2~3年的时间,在继承原有高职、高专和成人高等学校教材建设成果的基础上,充分汲取近几年来各类学校在探索培养技术应用性专门人才方面取得的成功经验,解决好新形势下高职高专教育教材的有无问题;然后再用2~3年的时间,在《新世纪高职高专教育人才培养模式和教学内容体系改革与建设项目计划》立项研究的基础上,通过研究、改革和建设,推出一大批教育部高职高专教育教材,从而形成优化配套的高职高专教育教材体系。

“教育部高职高专规划教材”是按照《基本要求》和《培养规格》的要求,充分汲取高职、高专和成人高等学校在探索培养技术应用性专门人才方面取得的成功经验和教学成果编写而成的,适用于高等职业学校、高等专科学校、成人高校及本科院校举办的二级职业技术学院和民办高校使用。

教育部高等教育司

2000年4月3日2

前 言

本书是教育部高职高专计算机应用与维护专业规划教材,主要讲授如何应用 Visual C++ 进行面向对象程序设计和可视化编程。全书在内容上可分为三大部分。第一部分介绍有关 C++ 的基本知识,主要包括:C++ 对 C 的扩充、类和对象、继承、函数重载及多态性,并以 DOS 程序为基础,使用控制台应用程序引入了最基本的 VC 开发环境。第二部分介绍 Visual C++ 程序设计的基本知识,主要包括:Visual C++ 基础类库 MFC 简介、使用向导建立应用程序框架、鼠标与键盘、资源及对话框、控件、图形用户界面和文件的输入输出。第三部分为 Visual C++ 程序设计的高级技术简介,主要包括 DLL 动态链接库和数据库编程技术。

本书内容翔实,深入浅出,通俗易懂,注重理论联系实际,结合应用开发实例,配备了丰富的例题和习题,系统讲解了书中所讲知识的使用方法。针对高职高专计算机专业教学的实际情况,本书重点讲解 Visual C++ 的基础知识,目的是为学生奠定坚实的程序设计基础。本书邀请了高职高专院校中具有丰富经验的教师编写。张福祥编写了第 1 章、第 3 章、第 7 章和第 8 章,张莹编写了第 2 章、第 4 章、第 5 章、第 6 章和第 9 章,于春花编写了第 10 章、第 11 章、第 12 章和第 13 章,孙永乐编写了第 14 章,李健与孙永乐合作编写了第 15 章、第 16 章、第 17 章和第 18 章,冯矢勇老师为本书的主审。本书在编写过程中得到了潍坊学院计算机科学系全体教师的大力帮助,在此一并表示感谢。

由于编写时间仓促,编写水平有限,书中难免有些缺点和错误,请读者批评指正。

作 者
2001 年 2 月

目 录

第 1 章 VC 的开发环境入门	(1)
1.1 Visual C++ 概述	(1)
1.2 建立控制台应用程序	(1)
1.3 编辑源程序文件	(4)
1.4 编译、连接和运行	(8)
思考与练习	(10)
第 2 章 C++ 对 C 的扩充	(11)
2.1 新的关键字	(11)
2.2 流的输入输出	(11)
2.3 注释	(12)
2.4 强制转换	(13)
2.5 引用	(14)
2.6 函数中的引用调用	(15)
2.7 函数原型	(17)
2.8 内联函数	(18)
2.9 带默认参数的函数	(19)
第 3 章 类和对象	(21)
3.1 类与对象的定义	(21)
3.2 成员函数	(23)
3.3 构造函数	(26)
3.4 析构函数	(27)
3.5 对象与指针	(28)
3.6 类与结构	(28)
思考与练习	(29)
第 4 章 继承	(30)
4.1 基类与派生类	(30)
4.2 多重继承	(35)
4.3 派生类的对象和构造函数	(37)
思考与练习	(40)
第 5 章 重载	(41)
5.1 函数重载	(41)
5.2 操作符重载	(46)
思考与练习	(52)
第 6 章 多态性	(53)
6.1 静态联编与动态联编	(53)
6.2 虚拟函数	(53)
思考与练习	(59)

第 7 章 Windows 编程基础	(60)
7.1 Windows 的性能特点	(60)
7.2 Windows 用户界面对象	(62)
7.3 Windows 的编程特点	(64)
7.4 应用程序设计接口和开发工具包	(67)
思考与练习	(68)
第 8 章 VC 的开发环境	(69)
8.1 集成工作环境	(69)
8.2 工程项目生成	(83)
思考与练习	(84)
第 9 章 MFC 简介	(85)
9.1 MFC 概述	(85)
9.2 MFC 类库层次	(86)
9.3 COject 类	(86)
9.4 主要派生类	(87)
9.5 MFC 程序设计	(93)
思考与练习	(95)
第 10 章 用 AppWizard 建立应用框架	(96)
10.1 建立工程项目	(96)
10.2 应用框架分析	(103)
10.3 ClassWizard(类向导)	(105)
10.4 文档和视图	(109)
思考与练习	(111)
第 11 章 键盘与鼠标	(112)
11.1 键盘与鼠标消息	(112)
11.2 使用键盘	(113)
11.3 鼠标与光标资源	(116)
附录 1 usekey 源程序代码	(120)
附录 2 cursor 源程序代码	(128)
附录 3 mouse 源程序代码	(136)
思考与练习	(144)
第 12 章 菜单和工具栏	(145)
12.1 菜单	(145)
12.2 补充说明	(149)
12.3 工具栏	(152)
附录 usemenu 源程序代码	(156)
思考与练习	(164)
第 13 章 对话框	(165)
13.1 对话框的分类	(165)
13.2 创建对话框	(165)
13.3 对话框的详细设计	(167)
附录 usedialog 源程序代码	(172)

思考与练习	(182)
第 14 章 控件	(183)
14.1 静态控件、编辑控件和按钮控件	(183)
14.2 列表框和组合框	(185)
14.3 标签控件	(191)
14.4 树控件	(194)
思考与练习	(197)
第 15 章 图形用户界面	(198)
15.1 图形设备接口 GDI	(198)
15.2 映射模式和绘图坐标	(198)
15.3 CDC 类	(202)
15.4 CGdiObject 类	(204)
15.5 实例	(205)
思考与练习	(209)
第 16 章 文件的输入输出	(210)
16.1 串行化	(210)
16.2 CFile 类	(213)
16.3 实例	(215)
思考与练习	(220)
第 17 章 DLL 动态链接库	(221)
17.1 DLL 基础知识	(221)
17.2 DLL 入出口函数	(221)
17.3 从 DLL 中导出函数	(221)
17.4 DLL 中的数据	(222)
17.5 程序链接	(224)
17.6 实例	(225)
思考与练习	(228)
第 18 章 数据库编程	(229)
18.1 MFC 中的 ODBC 类	(229)
18.2 建立数据源	(230)
18.3 编写应用程序	(232)
18.4 增加控件	(233)
18.5 编译并运行	(234)
18.6 增加和删除记录	(235)
思考与练习	(238)
参考文献	(239)

第 1 章 VC 的开发环境入门

本章以 DOS 程序为基础,以 C 程序为例,引入最基本的 VC 开发环境,使读者初步认识 Visual C++ 程序设计。

1.1 Visual C++ 概述

Visual C++ 6.0 是微软公司推出的一种功能强大而复杂的编译器,用于建立在 Windows 95、Windows 98 和 Windows NT 下的应用程序,它不仅适用于 Windows 编程,还为网络编程、数据库编程提供了强大的支持。Visual C++ 提供了功能强大的 Windows 应用程序框架。微软 MFC 封装了 Win32 API 的函数,提供了大量编写好的程序代码,以实现多数 Windows 标准操作。在 Visual C++ 中,编写 Windows 应用程序时,可以继续使用 SDK 调用 Windows 本身提供的 Win32 API 函数编程,可以利用 MFC 封装的类库编写程序,而最方便的是利用 AppWizard 向导建立程序的基本框架。与 Visual C++ 5.0 相比,Visual C++ 6.0 具有更快的编译速度、更加友好的编辑界面和更加强大的数据库支持功能。

Visual C++ 的主要技术特点为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工具,如应用程序向导 AppWizard 和类向导 ClassWizard。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 编程技术变得简单易学。

Visual C++ 可用来开发各种类型、不同规模和复杂程度的应用程序,开发效率很高,生成的应用软件代码品质优良。现在的 MFC 已经成为 C++ 进行 Windows 编程的工业标准,其易用性和强大的功能促使对 Visual C++ 程序员的需求越来越多,可以预见,将来的 Visual C++ 将变得更加强大。

1.2 建立控制台应用程序

Visual C++ 控制台应用程序允许把程序中的一些语句从 DOS 直接移植到 Windows 中,最新版本的 Visual C++ 6.0 提供的控制台应用程序类型可以让 DOS 程序员方便地进行 MFC 应用程序的开发,下面用实例来介绍控制台应用程序创建方法。

例 1.2.1 一篮鸡蛋,不超过 400 个。若每次拿 2 个(或者 3、4、5、6 个),篮里总剩下一个蛋;若每次拿 7 个,则刚好拿完。问篮里原有多少个鸡蛋?

这是一个和我国古代孙子定理相似的问题。分析题意,首先推断蛋的数目必为奇数,故可设初值 $n = 3$ 。然后用 3 除,若余数非 1 就加 2,直至余数为 1 止;再用 4 除,若余数非 1 就加 6,直至余数为 1 止;再用 5 除,若余数非 1 就加 12,直至余数为 1 止;再用 6 除,若余数非 1 就加 60,直至余数为 1 止;最后用 7 除,若不能整除就加 60,直至整除为止。这时的 n 值就是所求的

数目。程序如下：

```
#include <iostream.h>
void main()
{
    int n = 3;
    while (n % 3! = 1)
        n += 2;
    while (n % 4! = 1)
        n += 6;
    while (n % 5! = 1)
        n += 12;
    while (n % 6! = 1)
        n += 60;
    while (n % 7! = 0)
        n += 60;
    cout << "There were" << n << "eggs" << endl;
}
```

假如要建立的工程项目取名为 Example1,源程序文件名同时为 Example1,首先运行 Visual C++ 6.0,启动成功后单击 File 菜单,选择 New 选项,出现 New 对话框,切换到 Project 表页,如图 1-2-1 所示。

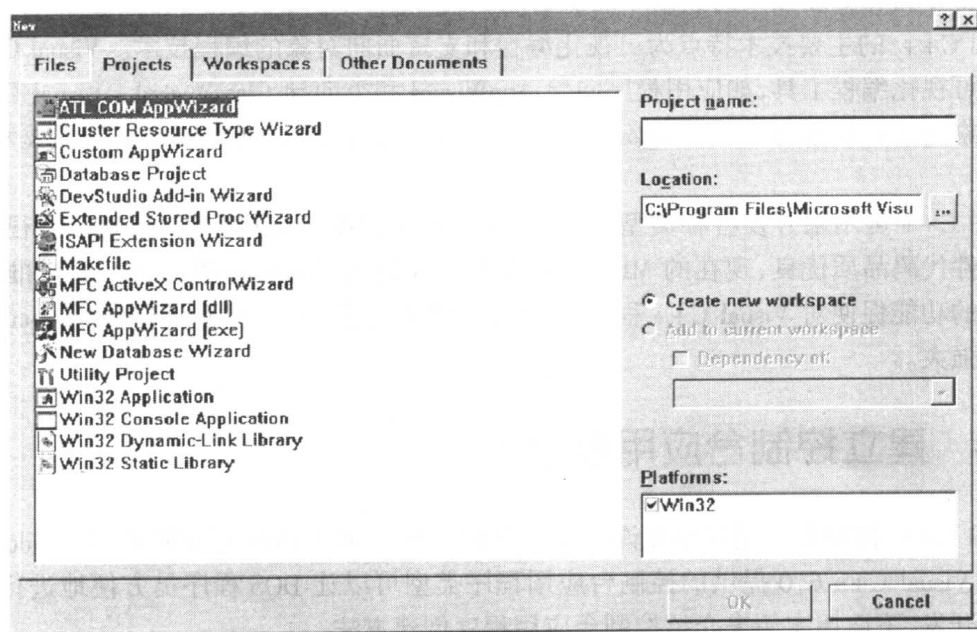


图 1-2-1 New 对话框 Project 表页

在 Project 表页列表框中选择 Win32 Console Application 单击,在 Project name 输入框中输入要建立的项目文件名 Example1,在 Location 输入框中输入用户希望存放项目的子目录名,在默认情况下,系统会在安装 Microsoft Visual Studio 6.0 时自动建立的 Microsoft Visual Studio 目录下的 MyProjects 子目录中自动建立 Example1 子目录,用于存放项目文件。输入完毕后单击 OK 按

钮,一个询问项目类型的 Win32 应用程序向导将出现,如图 1-2-2 所示。

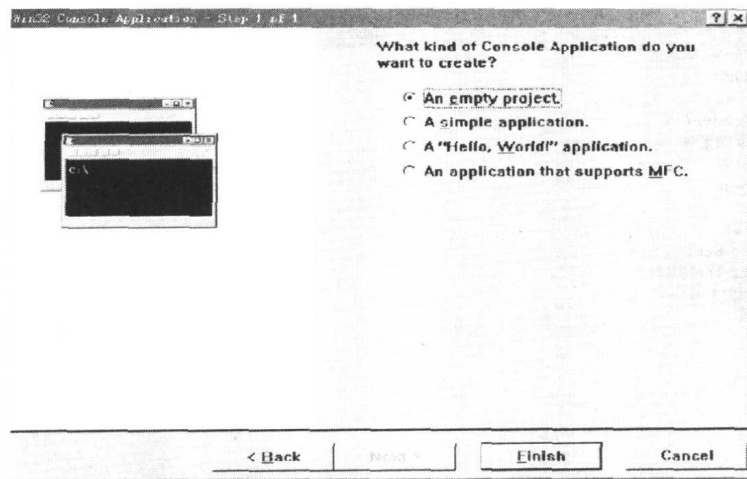


图 1-2-2 选择项目类型

选择 An empty project 单选按钮,然后单击 Finish 按钮,弹出如图 1-2-3 所示的信息对话框。单击 OK 按钮后,系统开始创建此应用程序。

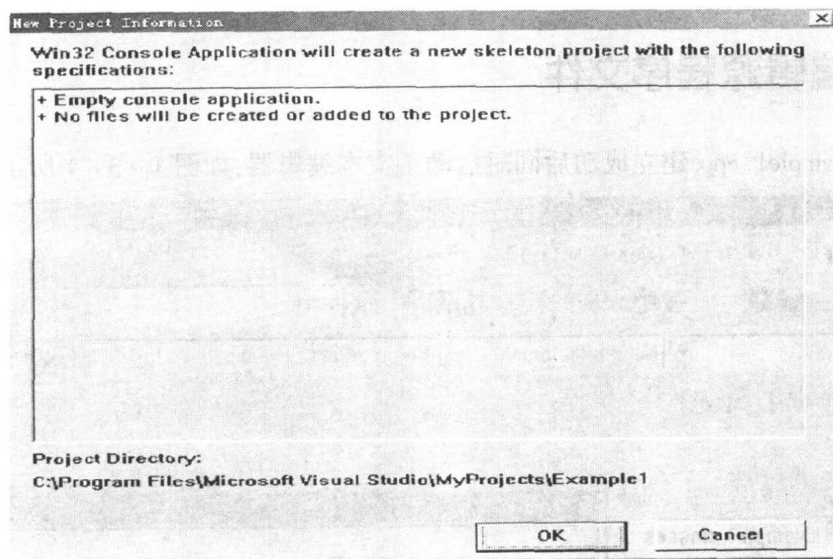


图 1-2-3 信息对话框

生成源程序文件。选择 File 菜单中 New 命令,再次打开 New 对话框,切换到 File 表页,如图 1-2-4 所示。从 File 表页列表框中可以看到 Visual C++ 6.0 支持多种文件的编辑。选择 C++ Source File 选项,然后在右边的 Add to project、File、Location 输入框中输入参数。Add to project 表示新建的文件要加入到工程中,输入框中要输入工程名 Example1 或者选择默认值。File 表示新建的文件名,输入框中要输入为新建文件起的名 Example1。Location 表示新建的文件存放的目录,默认值为存放工程的子目录,输入框中要输入存放文件 Example1.cpp 的目录或者选择默认值。单击 OK 按钮后系统建立源文件 Example1.cpp。

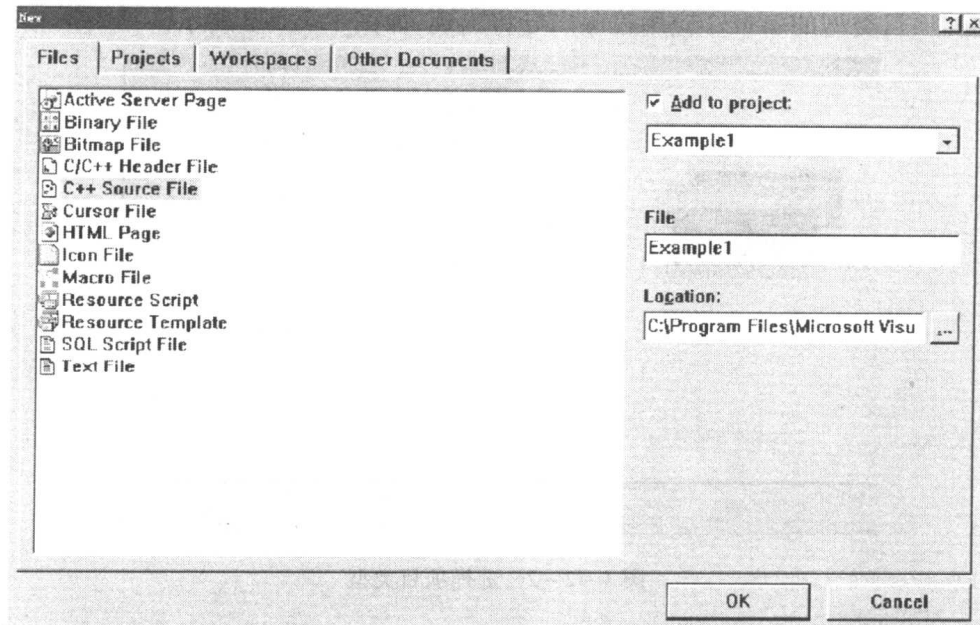


图 1-2-4 New 对话框 File 表页

1.3 编辑源程序文件

源文件 Example1.cpp 建立成功后同时启动了文本编辑器,如图 1-3-1 所示。

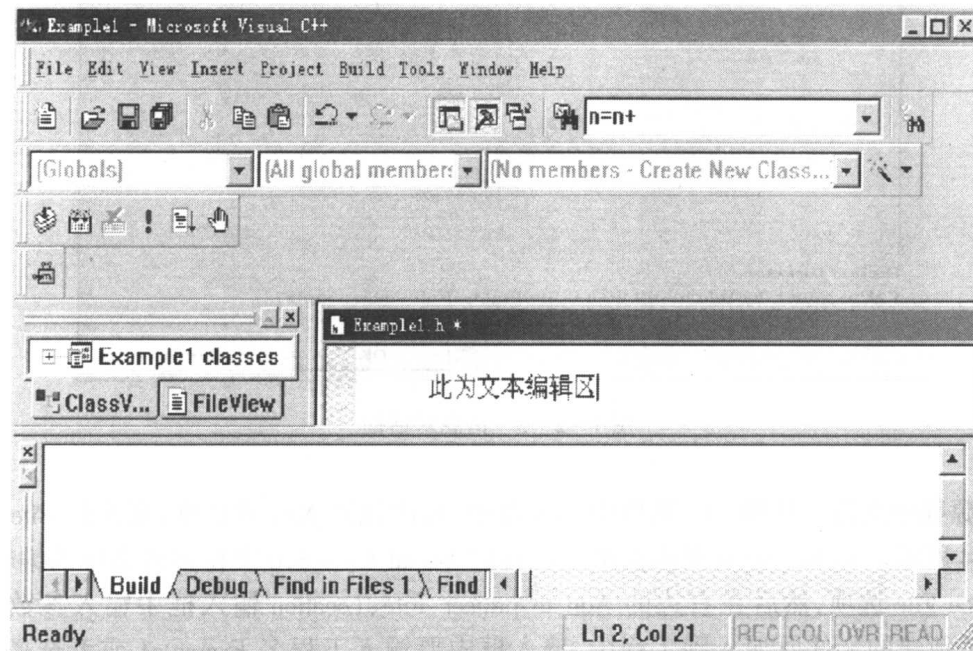


图 1-3-1 文本编辑器

在文本编辑器中,用一闪烁的短线条表示编辑位置,通过键盘输入的文字在此位置插入文

本。用鼠标左键点击文本中的某个字符可以改变编辑位置。

文本编辑器的基本操作包括：

--> : 光标向后移动一个字符。

<-- : 光标向前移动一个字符。

↑ : 光标向上移动一行。

↓ : 光标向下移动一行。

Home: 光标移动到行首。

End: 光标移动到行尾。

Ctrl + Home: 光标移动到文件头。

Ctrl + End: 光标移动到文件尾。

PgUp: 光标向上滚动一屏。

PgDn: 光标向下滚动一屏。

Ctrl + Y: 删除行。

Del: 删除光标右边字符。

Backspace: 删除光标左边字符。

Ins: 插入/改写方式切换。

除此之外,一些高级编辑功能可以通过 Developer Studio 的 Edit 子菜单或者工具栏按钮实现,大致可分为以下几类。

1. 撤消和重复操作

Edit 子菜单中的 Undo 命令是编辑中最重要的命令之一,因为当进行误操作时可以改正。如果要取消最近的几次编辑,只要选择 Undo 命令,或者单击常用工具栏上的撤消按钮。Undo 命令与当前完成的操作密切相关,它不断地变换,以反映上一次的操作。Undo 命令能够逐步恢复操作;而常用工具栏的撤消按钮有一下拉列表,可以有选择地进行恢复,如果你发现在前几步出现了错误,可以一次将这几步恢复过来。

除了可以撤消上一次的操作,还可以用 Edit 子菜单中的 Redo 命令来取消前一次所撤消的操作,像 Undo 命令一样,Redo 命令也是与当前完成的操作紧密联系,所以它不断变化以反映上一次的操作。另外,Redo 命令能够逐步恢复操作,常用工具栏的重复按钮有一下拉列表,该表列出了所有可以重复的操作。

2. 剪贴、复制、粘贴和删除

Cut: 将选定的内容复制到剪贴板,然后再从当前活动窗口中删除。“Cut”选项与“Paste”选项联合使用可以移动选定的内容。要选定文本内容,可将鼠标移到选定内容的开始位置,按下鼠标左键,然后拖动鼠标到选定内容的末尾放开鼠标左键。此时整个被选定内容将反白显示(黑底白字)。

Copy: 将选定内容复制到剪贴板,但不从当前活动窗口中删除。“Copy”选项和“Paste”选项联合使用可以复制选定的内容。

Paste: 将剪贴板中的内容插入到当前编辑位置。注意,要先剪切或复制选定内容到剪贴板后,才能进行粘贴。

Delete: 删除选定内容。删除以后,还可用“Undo”命令来恢复误删的内容。

Select All: 选定当前正在编辑文本的所有内容。

3. 查找和替换

Find: 在正在编辑的文本中查找指定的字符串。如果希望在 Example1.cpp 中查找 n += 12, 单击 Edit 菜单, 选择 Find 选项, 弹出对话框如图 1-3-2 所示, 在 Find What 输入框中输入 n += 12, 在 Direction 选项中选择 Up(向上查找)或者 Down(向下查找), 然后单击 Find Next 开始查找 n += 12 这个字符串。

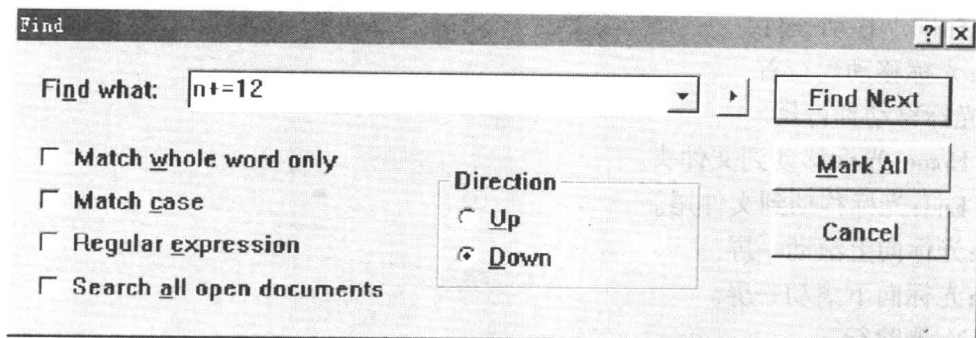


图 1-3-2 Find 对话框

Find in Files: 在多个文件中查找指定的字符串。

Replace: 在正在编辑的文本中替换指定的字符串。如果希望在 Example1.cpp 中将 n += 全部替换成 n = n + , 单击 Edit 菜单, 选择 Replace 选项, 弹出对话框如图 1-3-3 所示, 在 Find What 输入框中输入 n += , 在 Replace With 输入框中输入 n = n + , 然后单击 Replace All 开始替换。

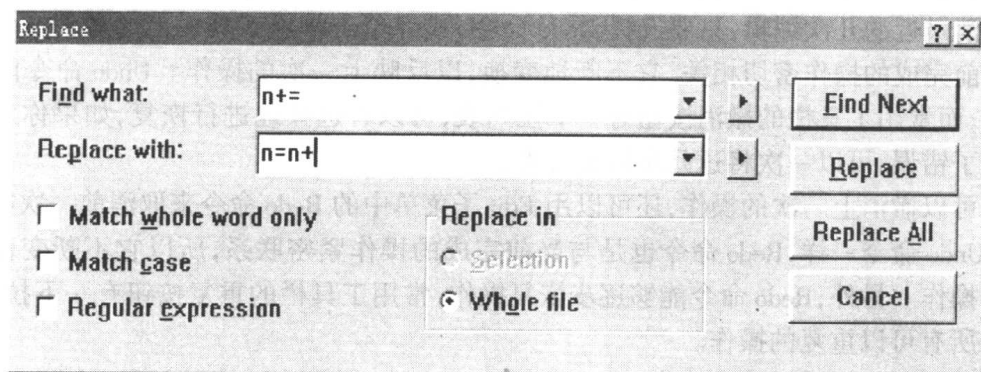


图 1-3-3 Replace 对话框

4. 书签

Go To: 弹出“Go To”对话框如图 1-3-4 所示, 指定如何将编辑位置移到当前活动窗口的指定位置, 如指定的号、地址和书签等。如果希望在 Example1.cpp 中将编辑位置移到第 8 行, 在 Go to what 列表框中选择 Line 选项, 在 Enter line number 输入框中输入 8, 然后单击 Go To 按钮即可。

Bookmarks: 弹出“Bookmarks”对话框, 设置或取消书签。书签用于在文件中做标记。

5. 高级选项

Advanced: 选择该项将弹出子选择菜单如图 1-3-5 所示, 其中含有用于编辑或修改的高

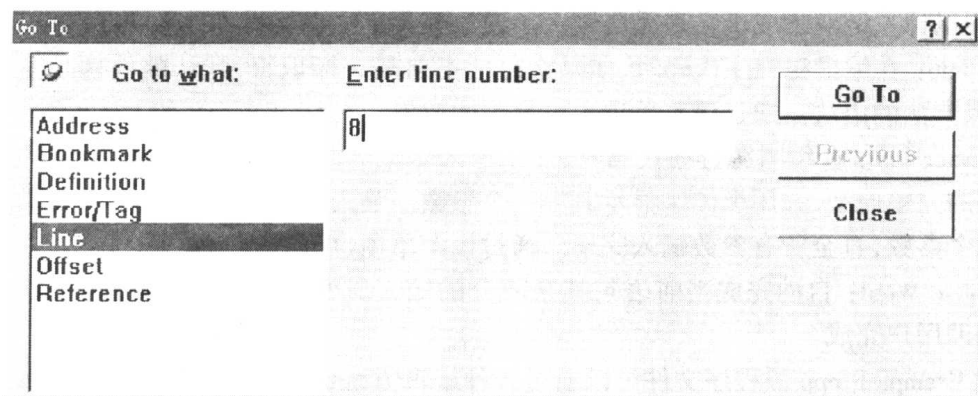


图 1-3-4 Go To 对话框

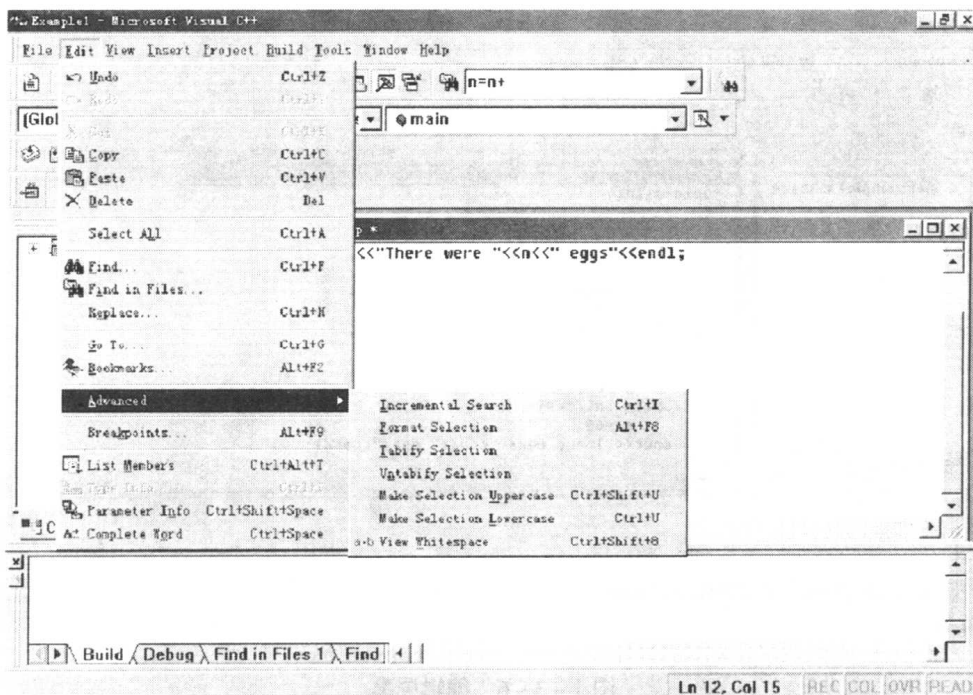


图 1-3-5 Advanced 对话框

级命令,如增量式搜索、将选定内容全部转为大写或小写、显示或隐藏制表符等。

6. 断点设置

Breakpoints:弹出“Breakpoints”对话框,以便设置、删除和查看断点。断点用来通知调试器应该在何处或何时中断程序的执行过程,以便检查程序代码、变量和寄存器值。“Breakpoints”对话框的“Location”,“Data”和“Messages”选项卡分别用于设置位置断点、数据断点和消息断点。如果要设置条件断点,必须先设置位置断点,然后再设置中断程序执行的条件。

7. 编程指导信息

List Members:输入程序代码时,如果在变量名之前键入“.”或“-->”(即要访问某对象的成员),默认的处理是自动列表显示所有有效的成员。这时,键入待输入成员的前几个字母即可从列表中选中该成员,按 Tab 键即可完成成员名的输入;也可通过滚动条找到待输入的成

员,然后用鼠标左键双击成员名来完成输入。通过该菜单项也可得到此效果。

Type Info: 在编辑源程序代码时,如果将鼠标指向某一标识符,默认的处理是显示所指变量、函数或方法的语法。通过该菜单项也可得到此效果。

Parameter Info: 在编辑源程序代码时,如果输入函数名之后键入左括号,默认的处理是显示该函数的完整原型,并用黑体字显示其第一个参数。输入第一个参数值之后,接着会以黑体显示第二个参数,直至所有参数输入完毕。通过该菜单项也可得到此效果。

Complete Word: 自动完成当前语句其余部分的输入。如果不能自动完成,则给出适当的提示来辅助用户完成。

实例 Example1.cpp 源程序文件可以通过各种编辑方法编辑,编辑完成后的屏幕画面如图 1-3-6 所示。

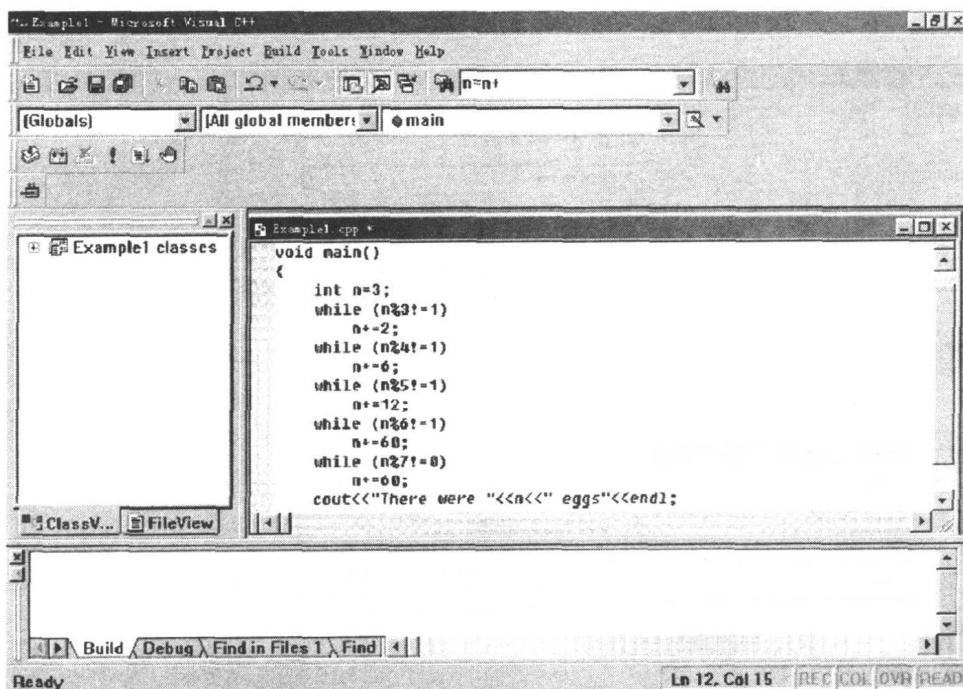


图 1-3-6 编辑屏幕

1.4 编译、连接和运行

实例 Example1.cpp 源程序文件编辑完毕后,要对工程项目 Example1 进行编译和连接,对编译和连接中发现的错误,经更改后继续进行编译和连接,直至正确为止。Developer Studio 提供了一整套功能,用于应用程序的编译、连接和运行,其功能集中在 Build 子菜单中,但其中多数功能也可通过快捷键和工具条调用。

1. 编译当前项目。单击 Build 菜单选择 Compile 选项或者击快捷键 Ctrl + F7,将源程序编译为目标代码(.obj)。编译结果及编译错误信息在输出窗口(Output)显示。实例 Example1 编译画面如图 1-4-1 所示,注意输出窗口中显示的信息。

2. 编译、连接当前项目。单击 Build 菜单,选择 Build 选项,或者击快捷键 Ctrl + F7,将当前

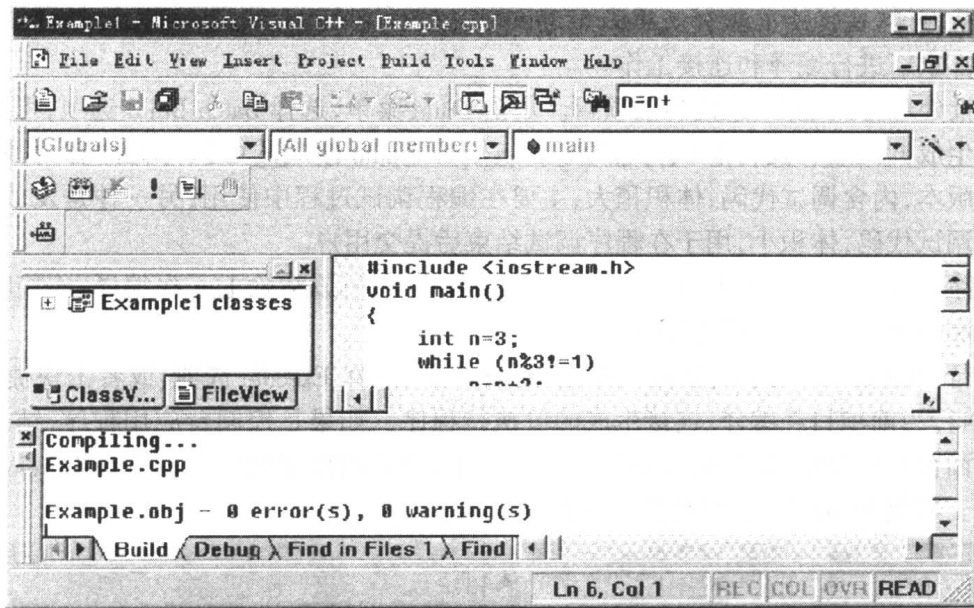


图 1-4-1 编译信息图

项目的源程序编译之后,还进行连接(Link)操作,即对目标代码与系统或用户类库连接并生成可执行的程序(或动态连接库.DLL等)。编译、连接结果及编译、连接的错误信息在输出窗口(Output)显示。该选项的特点是可对新修改的文件进行编译和连接,因此速度较快,是编程、调试的常用功能。通常可用鼠标点击工具条上的对应的图标调用该功能,非常方便。实例 Example1 编译并连接的画面如图 1-4-2 所示,注意输出窗口中显示的信息与编译输出窗口中显示的信息有何不同。

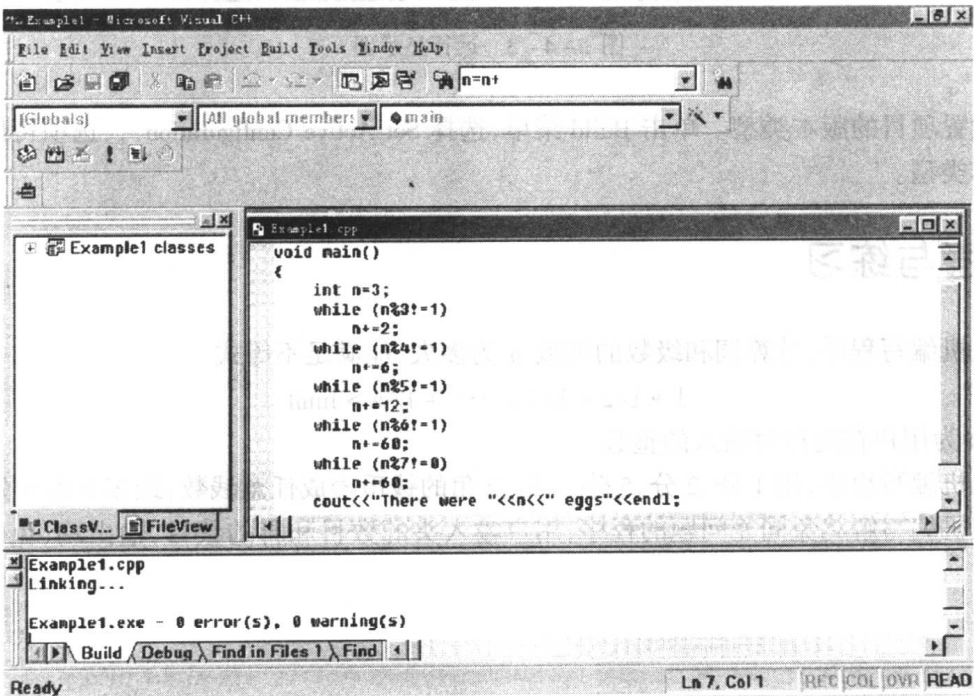


图 1-4-2 编译连接信息图