



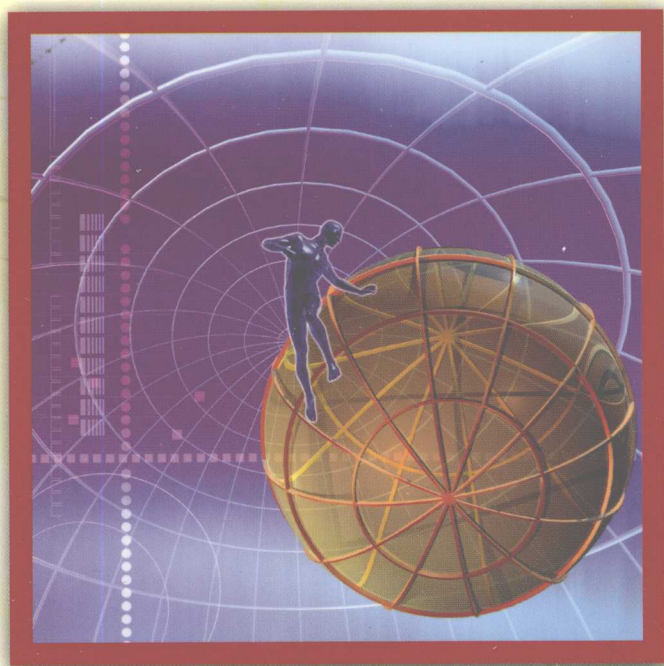
高等学校计算机教育系列规划教材



C++ 程序设计实践教程

刘卫国 杨长兴 主 编

李小兰 蔡旭晖 副主编



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

高等学校计算机教育系列规划教材

C++程序设计实践教程

刘卫国 杨长兴 主 编

李小兰 蔡旭晖 副主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书是《C++程序设计》配套的实践教程。全书包括上机实验指南、结构化程序设计方法和习题选解三部分内容。第一部分内容帮助读者熟悉 C++ 上机环境,方便读者上机操作练习。通过有针对性的上机实验,可以更好地掌握 C++ 程序设计的方法。第二部分内容将问题进行分类,总结每一类问题的编程思路,并给出大量的程序实例,以引导读者掌握基本的结构化程序设计方法和技巧。此外,还介绍了程序测试与调试的常用方法,以帮助读者提高调试程序的能力。第三部分内容可以作为课程学习或参加各种计算机考试的辅导材料。

本书内容丰富,实用性强,适合作为高等院校计算机程序设计课程的实践教材,也可作为社会各类计算机应用人员及参加各类计算机等级考试的读者的辅助教材。

图书在版编目(CIP)数据

C++程序设计实践教程 / 刘卫国, 杨长兴主编. —北京:
中国铁道出版社, 2008.2
高等学校计算机教育规划教材
ISBN 978-7-113-08664-0

I. C… II. ①刘…②杨… III. C 语言—程序设计—高等
学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2008) 第 022769 号

书 名: C++程序设计实践教程

作 者: 刘卫国 杨长兴 等

出版发行: 中国铁道出版社 (100054, 北京市宣武区右安门西街 8 号)

策划编辑: 严晓舟 秦绪好

责任编辑: 李 旸 郑 楠

封面设计: 付 巍

封面制作: 白 雪

印 刷: 三河市华晨印务有限公司

开 本: 787×1092 1/16 印张: 16.5 字数: 389 千

版 本: 2008 年 3 月第 1 版 2008 年 3 月第 1 次印刷

印 数: 1~5 000 册

书 号: ISBN 978-7-113-08664-0/TP·2731

定 价: 24.00 元

版权所有 侵权必究

本书封面贴有中国铁道出版社激光防伪标签, 无标签者不得销售

凡购买铁道版的图书, 如有缺页、倒页、脱页者, 请与本社计算机图书批销部调换。

计算机程序设计基础是从技术角度学习计算机的主要基础课程,已成为大多数专业的必修课。随着软件工程技术的不断发展,面向对象的程序设计方法已成为当今软件开发的主流技术,因此,越来越多的学校选择 C++ 作为程序设计课程的语言环境。C++ 程序设计是一门实践性很强的课程,学习程序设计不能仅限于纸上谈兵,许多程序设计方法和技巧不是光靠听课和看书就能学到的,而是通过大量的上机实践慢慢积累起来的,程序设计能力的培养必须以实践为重。本书是《C++ 程序设计》配套的实践教程,全书包括上机实验指南、结构化程序设计方法和习题选解三部分内容。

第一部分是上机实验指南,包括 C++ 程序的集成开发环境、实验要求与实验项目两章。

C++ 程序的集成开发环境一章介绍了两种最为流行的 C++ 集成开发环境——Visual C++ 6.0 和 Turbo C++ 3.0,熟悉和掌握这方面的操作是 C++ 程序设计上机实践的基础。

实验要求与实验项目一章设计了 15 个实验,实验内容和教学内容紧密衔接。最后一个实验是综合程序设计,可作为课程设计的内容。这些实验和课堂教学紧密结合,通过有针对性的上机实验,可以更好地掌握 C++ 程序设计的方法,并培养较强的程序应用开发能力。

为了达到理想的实验效果,实验前应认真准备,要根据实验目的和实验内容,复习好实验中可能要用到的知识,想好编程的思路,做到胸有成竹,提高上机效率。实验过程中积极思考,要深入分析程序的运行结果以及各种屏幕信息的含义、出现的原因并提出解决办法。实验后认真总结,要总结本次实验有哪些收获,还存在哪些问题,并写出实验报告。

第二部分是结构化程序设计方法,包括常用算法设计方法、程序测试与调试两章。

尽管 C++ 是一种面向对象的程序设计语言,但就程序实现而言,仍需要使用有关结构化程序设计的方法,首先进行算法设计。例如,一个类的操作方法(成员函数)就要用结构化程序设计来实现。所以在学习 C++ 程序设计时,算法设计仍然是不可忽视的问题。考虑到算法设计是学习的难点,特意增加这部分内容。这部分内容将常见的程序设计问题进行分类,总结每一类问题的编程思路与技巧,并给出大量的程序实例,以引导读者掌握基本的程序设计方法和技巧。教学实践表明,这种方法对提高读者的编程能力是大有裨益的。同样这些内容对学习其他高级语言程序设计也是有帮助的。

程序测试与调试是程序设计的重要环节。通过测试发现程序中的错误,然后找出错误的原因和位置并加以改正,这就是程序调试的目的。编写程序时出错是难免的,关键是如何在短时间内发现并纠正程序中的错误,这就要培养较强的调试程序的能力。这部分内容介绍程序测试与调试的常用方法,并结合 Visual C++ 6.0 集成开发环境说明如何运用这些方法来调试 C++ 程序。

第三部分是习题选解,根据教学要求和教学内容分成 10 章,这部分内容包括大量的习题,涉及目前各类计算机考试中常见的题型,并附有参考答案。这部分内容可供读者进行课外练习之用,也可作为各种计算机考试的辅导材料。

本书由刘卫国、杨长兴担任主编，李小兰、蔡旭晖担任副主编。第1、3、4章由刘卫国编写，第2章由杨长兴、刘卫国、周春艳、朱从旭、周欣然、李小兰、曹岳辉、吕格莉、李利明、蔡旭晖共同编写，第5~14章分别由杨长兴、周春艳、朱从旭、周欣然、李小兰、刘卫国、曹岳辉、吕格莉、李利明、蔡旭晖编写。参与程序调试及资料整理的还有杨斌、刘勇、张志良、李斌和康维等。全书由刘卫国和杨长兴统稿，李小兰和蔡旭晖协助主编做了大量工作。

在本书编写过程中，得到了施荣华教授的大力支持与帮助，在此表示衷心感谢。中南大学计算机基础教学实验中心的全体教师是本课程教学改革的实践者，为本书的编写积累了许多有价值的素材，在本书的编写过程中提出了许多很好的建议，在此一并表示感谢。此外，在编写本书的过程中，编者还参考了大量的文献资料，在此也向这些文献资料的作者表示感谢。

由于编者水平有限，成稿时间仓促，书中如有疏漏及不妥之处，敬请读者不吝赐教。

编者

2008年1月

第一部分 上机实验指南

第 1 章 C++程序的集成开发环境	3
1.1 Visual C++ 6.0 集成开发环境	3
1.1.1 Visual C++ 6.0 的安装与启动	3
1.1.2 Visual C++ 6.0 主界面的组成	4
1.1.3 Visual C++ 6.0 环境下 C++程序的运行	13
1.2 Turbo C++ 3.0 集成开发环境	15
1.2.1 Turbo C++ 3.0 的安装与启动	15
1.2.2 Turbo C++ 3.0 主界面的组成	16
1.2.3 Turbo C++ 3.0 环境下 C++程序的运行	17
1.2.4 系统设置	19
第 2 章 实验要求与实验项目	21
实验 1 C++基础知识	21
实验 2 程序控制结构（一）	26
实验 3 程序控制结构（二）	29
实验 4 结构化程序设计的应用	33
实验 5 函数和编译预处理	36
实验 6 数组	40
实验 7 指针	43
实验 8 自定义数据类型	46
实验 9 类与对象（一）	50
实验 10 类与对象（二）	55
实验 11 重载与模板	60
实验 12 继承与派生	68
实验 13 多态性和虚函数	72
实验 14 输入输出流	76
实验 15 综合程序设计	81

第二部分 结构化程序设计方法

第 3 章 常用算法设计方法	111
3.1 累加与累乘问题	111
3.2 数字问题	115

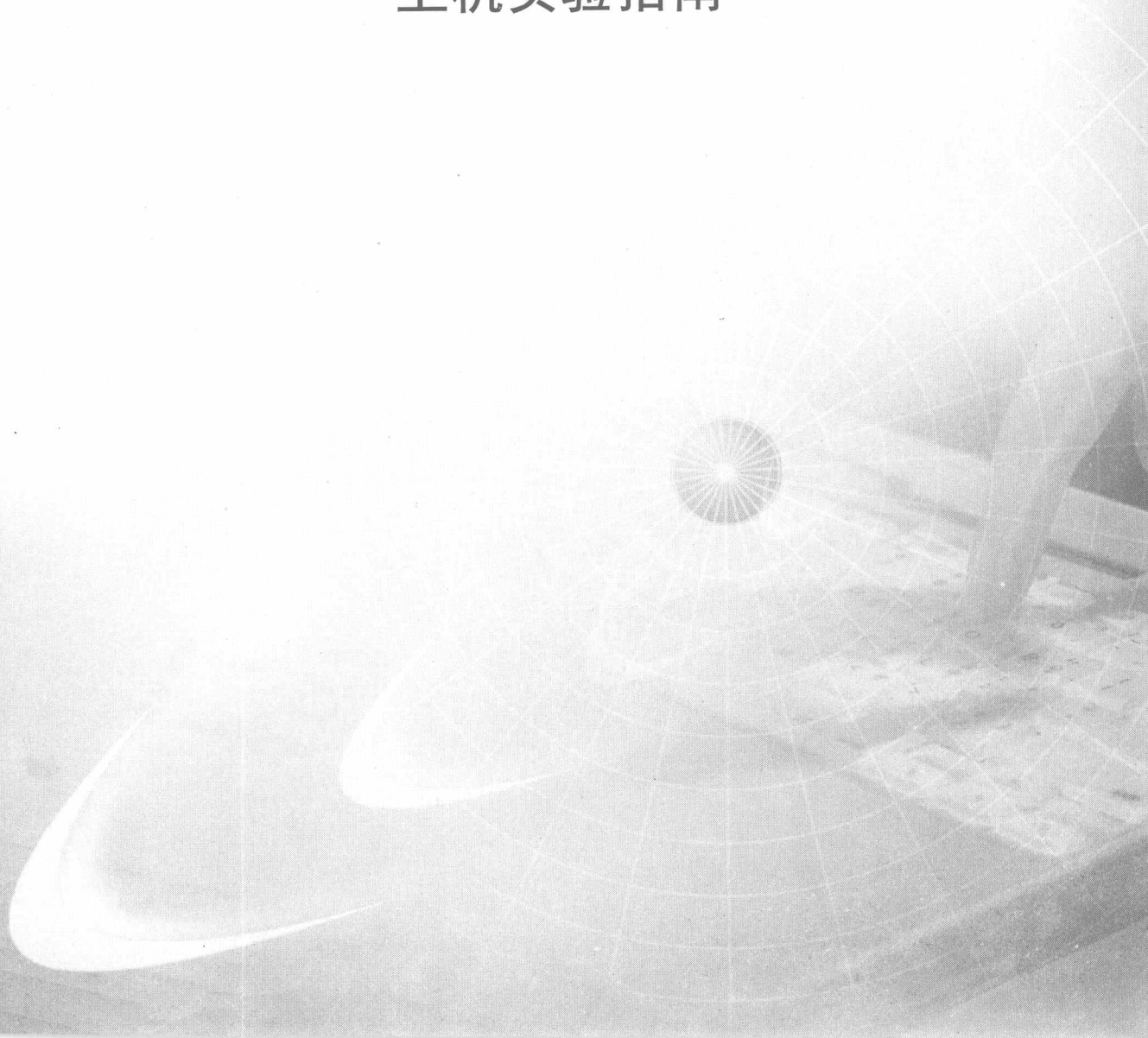
3.3 数值计算问题.....	118
3.4 数组的应用	120
3.5 函数的应用	123
3.6 解不定方程	124
思考题	125
参考答案	127
第4章 程序测试与调试.....	129
4.1 程序测试	129
4.2 程序调试	130
4.2.1 诊断错误的实验方法	130
4.2.2 错误诊断的推理技术	132
4.2.3 错误修改的原则	132
4.3 Visual C++ 6.0 程序调试	133

第三部分 习题选解

第5章 C++基础知识.....	139
第6章 程序控制结构	146
第7章 函数和编译预处理	157
第8章 数组与指针	167
第9章 自定义数据类型	178
第10章 类与对象	188
第11章 运算符重载与模板	208
第12章 继承与派生	227
第13章 多态性和虚函数	231
第14章 输入输出流	246
参考文献	257

第一部分

上机实验指南



第 1 章

C++程序的集成开发环境

运行 C++ 语言程序需要相应编译系统的支持。C++ 的编译系统有很多，常用的都是集成开发环境，即源程序的输入、修改、调试及运行都可以在同一环境下完成，功能齐全，操作方便。本章介绍 Visual C++ 6.0 和 Turbo C++ 3.0 两种流行的 C++ 语言程序集成开发环境，使读者掌握编辑、编译、连接和运行一个 C++ 程序的全过程。

1.1 Visual C++ 6.0 集成开发环境

Visual C++ 6.0 (简称 VC 6.0) 是 Microsoft 公司研制开发的、基于 Windows 的可视化 C++ 语言版本，它是一个集 C++ 程序编辑、编译、连接、调试运行等功能以及可视化软件开发为一体的开发工具。相比其他的编程工具而言，Visual C++ 在提供可视化的编程方法的同时，也适应编写直接对系统进行底层操作的程序。随 Visual C++ 一起提供的 Microsoft 基础类库 (Microsoft Foundation Class Library, MFC) 对 Windows 9X/NT 所用的 Win32 应用程序接口 (Win32 Application Programming Interface) 进行了十分彻底的封装，这使得 Windows 9X/NT 应用程序的开发可以使用完全的面向对象的方法进行，从而能够大大地缩短应用程序的开发周期，降低开发成本，也使得 Windows 程序员从大量的繁杂的劳动中解脱出来。

VC 6.0 是 Microsoft Visual Studio 套件的一个有机组成部分。Visual C++ 软件包含有许多单独的组件，如编辑器、编译器、连接器、生成实用程序和调试器等，以及各种各样为开发 Windows 下的 C/C++ 程序而设计的工具。Visual Studio 把所有的 Visual C++ 工具结合在一起，集成为一个整体，通过一个由窗口、对话框、菜单、工具栏、快捷键等组成的完整系统，可以观察和控制整个开发过程。

1.1.1 Visual C++ 6.0 的安装与启动

在使用 VC 6.0 之前，首先要安装 VC 6.0。VC 6.0 的安装方法和其他 Windows 应用程序的安装方法类似。将 VC 6.0 系统安装盘放入光驱，一般情况下系统能自动运行安装程序，否则运行安装盘中的 setup.exe 文件。启动安装程序后，根据屏幕提示依次进行安装操作，便可完成系统的安装。

启动 VC 6.0 的过程十分简单。常用的方法是：在 Windows 桌面选择“开始”|“程序”|“Microsoft Visual Studio 6.0”|“Microsoft Visual C++ 6.0”命令，即可启动 VC 6.0，屏幕上将显示如图 1-1 所示的 VC 6.0 主界面。

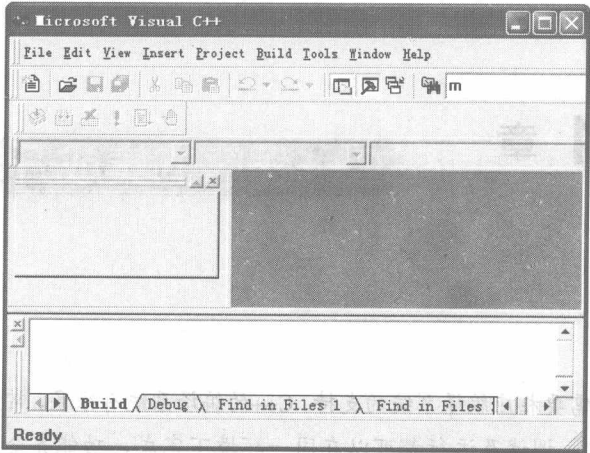


图 1-1 VC 6.0 主界面

1.1.2 Visual C++ 6.0 主界面的组成

和其他 Windows 的应用程序一样，VC 6.0 主界面也具有标题栏、菜单栏和工具栏。标题栏的内容是“Microsoft Visual C++”。菜单栏提供了编辑、运行和调试 C/C++ 程序所需要的菜单命令。工具栏是一些菜单命令的快捷按钮，单击工具栏上的按钮，即可执行该按钮所代表的操作。

在 VC 6.0 主界面的左侧是项目工作区（Workspace）窗口，右侧是程序编辑窗口，下方是输出（Output）窗口。项目工作区窗口用于显示所设定的工作区的信息，程序编辑窗口用于输入和修改源程序，输出窗口用于显示程序编译、运行和调试过程中出现的状态信息。

1. 菜单栏

VC 6.0 的菜单栏共有 9 个菜单项：File、Edit、View、Insert、Project、Build、Tools、Window 和 Help，每个菜单项都有下拉菜单，用鼠标单击菜单项可弹出其下拉菜单，通过下拉菜单中的每项菜单命令可执行不同的功能。下面对各菜单项进行详细介绍。

(1) File 菜单

File 菜单中包含了用于对文件进行各种操作的命令选项，其快捷键及功能如表 1-1 所示。

表 1-1 File 菜单命令

菜 单 命 令	快 捷 键	功 能 说 明
New	Ctrl+N	创建一个新的文件、项目或工作区
Open	Ctrl+O	打开一个已存在的文件
Close		关闭当前被打开的文件
Open Workspace		打开一个已存在的工作区
Save Workspace		保存当前被打开的工作区
Close Workspace		关闭当前被打开的工作区

续上表

菜单命令	快捷键	功能说明
Save	Ctrl+S	保存当前打开的文件
Save As		将当前文件另存为一新的文件
Save All		保存所有打开的文件
Page Setup		对页面的布局进行设置
Print	Ctrl+P	打印当前被打开的文件
Recent Files		最近使用的文件列表
Recent Workspaces		最近使用的工作区
Exit		退出集成开发环境

(2) Edit 菜单

Edit 菜单包含所有与文件编辑操作有关的命令选项，其快捷键及功能如表 1-2 所示。

表 1-2 Edit 菜单命令

菜单命令		快捷键	功能说明
Undo		Ctrl+Z	撤销上一次的操作
Redo		Ctrl+Y	恢复被撤销的操作
Cut		Ctrl+X	将所选中的内容剪切掉并送至剪贴板中
Copy		Ctrl+C	将所选内容复制至剪贴板中
Paste		Ctrl+V	将当前剪贴板中的内容粘贴到当前插入点
Delete		Del	删除所选中的内容
Select All		Ctrl+A	选定当前窗口中全部内容
Find		Ctrl+F	查找指定的字符串
Find in Files			在多个文件中查找指定字符串
Replace		Ctrl+H	替换指定字符串
Go To		Ctrl+G	光标自动转移到指定位置
Advanced	Bookmarks	Alt+F2	设置书签或书签导航
	Incremental Search	Ctrl+I	开始向前搜索
	Format Selection	Alt+F8	对选中对象进行快速缩排
	Tabify Selection		在选中对象中用跳格代替空格
	Untabify Selection		在选中对象中用空格代替跳格
	Make Selection Uppercase	Ctrl+Shift+U	把选中部分改成大写
	Make Selection Lowercase	Ctrl+U	把选中部分改成小写
a-b View Whitespace		Ctrl+Shift+8	显示或隐藏空格点
Breakpoints		Alt+F9	编辑程序中的断点
List Member		Ctrl+Alt+T	显示出全部关键字

续上表

菜单命令	快捷键	功能说明
Type Info	Ctrl+T	显示变量、函数或方法的语法
Parameter Info	Ctrl+Shift+Space	显示函数的参数
Complete Word	Ctrl+Space	给出相关关键字的全称

(3) View 菜单

View 菜单中包含了用于检查源代码和调试信息的各种命令选项，其快捷键及功能如表 1-3 所示。

表 1-3 View 菜单命令

菜单命令	快捷键	功能说明
ClassWizard	Ctrl+W	编辑应用程序的类
Resource Symbols		浏览和编辑资源文件中的资源标识符（ID 号）
Resource Includes		编辑修改资源文件名及预处理命令
Full Screen		切换到全屏显示方式
Workspace	Alt+0	激活项目工作区（Workspace）窗口
Output	Alt+2	激活输出（Output）窗口
Debug Windows	Watch	Alt+3 激活监视（Watch）窗口
	Call Stack	Alt+7 激活调用栈（Call Stack）窗口
	Memory	Alt+6 激活内存（Memory）窗口
	Variables	Alt+4 激活变量（Variables）窗口
	Registers	Alt+5 激活寄存器（Registers）窗口
	Disassembly	Alt+8 激活反汇编（Disassembly）窗口
Refresh		更新选中区域
Properties	Alt+Enter	打开源文件属性窗口

(4) Insert 菜单

使用 Insert 菜单，用户可以向当前项目中插入新类、新资源等，其快捷键及功能如表 1-4 所示。

表 1-4 Insert 菜单命令

菜单命令	快捷键	功能说明
New Class		在项目中添加一个新类
New Form		在项目中添加一个新表单
Resource	Ctrl+R	创建各种新资源
Resource Copy		对选定的资源进行复制
File As Text		可以将一个已存在的文件插入到当前焦点中
New ATL Object		在项目中添加一个新的 ATL 对象

(5) Project 菜单

Project 菜单包含用于管理项目和工作区的一系列菜单项，其快捷键及功能如表 1-5 所示。

表 1-5 Project 菜单命令

菜单命令		快捷键	功能说明
Set Active Project			选择指定项目为当前工作区中的活动项目
Add To Project	New		在项目中增加新文件
	New Folder		在项目中增加新文件夹
Add To Project	Files		在项目中插入已存在的文件
	Data Connection		在当前项目中增加数据连接
	Components and Controls		在当前项目中插入一个部件或 ActiveX 控件
Dependencies			编辑项目组件
Settings		Alt+F7	编译及调试的设置
Export Makefile			以制作文件 (.mak) 形式输出可编译项目
Insert Project into Workspace			将项目插入到项目工作区窗口中

(6) Build 菜单

Build 菜单中包含的命令选项用于创建、编译、调试及运行应用程序，其快捷键及功能如表 1-6 所示。

表 1-6 Build 菜单命令

菜单命令		快捷键	功能说明
Compile		Ctrl+F7	编译当前编辑窗口中打开的文件
Build		F7	生成一个可执行文件，即编译一个项目
ReBuild All			编译和连接多个项目文件
Batch Build			一次编译和连接多个项目文件
Clean			删除当前项目中所有中间文件及输出文件
Start Debug	Go	F5	开始或继续调试程序
	Step Into	F11	单步运行调试
	Run to Cursor	Ctrl+F10	运行程序到光标所在处
	Attach to Process		连接正在运行的进程
Debugger Remote Connection			用于编辑远程调试连接设置
Excute		Ctrl+F5	运行可执行文件
Set Active Configuration			选择激活的项目及配置
Configurations			编辑项目配置
Profile			选中该菜单项，用户可以检查代码的执行情况

(7) Tools 菜单

Tools 菜单中包含 VC 6.0 中提供的各种工具，用户可以直接从菜单中调用它们，其快捷键及功能如表 1-7 所示。

表 1-7 Tools 菜单命令

菜 单 命 令	快 捷 键	功 能 说 明
Source Browser	Alt+F12	浏览对指定对象的查询及相关信息
Close Source Browser File		关闭信息浏览文件
Visual Component Manager		激活组件管理器
Register Control		激活注册控件
Error Lookup		激活错误查找器
ActiveX Control Text Container		激活 ActiveX 控件测试器
OLE/COM Object Viewer		激活 OLE/COM 对象查看器
Spy++		激活 Spy++工具包
MFC Tracer		激活 MFC 跟踪器
Customize		定制 Tool 菜单和工具栏
Options		改变集成开发环境的各项设置
Macro		创建和编辑宏
Record Quick Macro	Ctrl+Shift+R	记录宏
Play Quick Macro	Ctrl+Shift+P	运行宏

(8) Window 菜单

Window 菜单用于设置 VC 6.0 开发环境中窗口的属性，其快捷键及功能如表 1-8 所示。

表 1-8 Window 菜单命令

菜 单 命 令	快 捷 键	功 能 说 明
New Window		为当前文档打开另一窗口
Split		将窗口拆分为多个窗口
Docking View	Alt+F6	启动或关闭 Docking View 模式
Close		关闭当前窗口
Close All		关闭所有打开的窗口
Next		激活下一个窗口
Previous		激活上一个窗口
Cascade		将所有打开的窗口重叠排列
Tile Horizontally		将工作区中所有打开的窗口进行纵向平铺
Tile Vertically		将工作区中所有打开的窗口进行横向平铺
Windows		管理当前打开的窗口

(9) Help 菜单

Help 菜单提供了详细的帮助信息，其快捷键及功能如表 1-9 所示。

表 1-9 Help 菜单命令

菜 单 命 令	功 能 说 明
Contents	显示所有帮助信息的内容列表
Search	利用在线查询获得帮助信息
Index	显示在线文件的索引
Use Extension Help	开启或关闭 Extension Help
Keyboard Map	显示所有键盘命令
Tip of the Day	显示 Tip of the Day
Technical Support	显示 Visual Studio 的支持信息
Microsoft on the Web	有关 Microsoft 的网站或网页
About Visual C++	显示版本的有关信息

2. 工具栏

默认状态下，VC 6.0 提供了 11 个工具栏，但只显示 3 个工具栏。用户可以通过鼠标右键单击工具栏，在弹出的快捷菜单中选择需要显示的工具栏，如图 1-2 所示。在图 1-2 中显示了系统提供的工具栏，其中具有复选标记的菜单项，表示在开发环境中显示的工具栏。用户可以通过单击菜单项来控制工具栏是否显示。

实际上，工具栏中的命令按钮多数是与菜单栏中的菜单项相对应的。因此这里只介绍在开发环境中显示的工具栏。对于其他工具栏，可以参考菜单操作。

(1) 标准工具栏 (Standard)

标准工具栏主要帮助用户维护和编辑在工作区中的文本和文件，如图 1-3 所示。

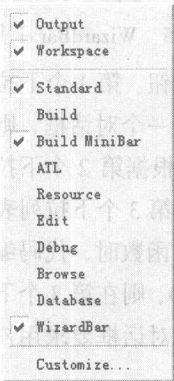


图 1-2 工具栏快捷菜单

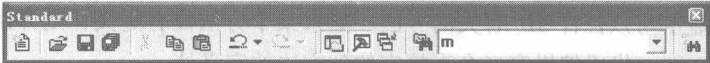


图 1-3 Standard 工具栏

标准工具栏对应的菜单命令如表 1-10 所示。

表 1-10 标准工具栏中各功能按钮所对应的菜单项

名 称	相应菜单项	名 称	相应菜单项
New	File New	Redo	Edit Redo
Open	File Open	Workspace	View Workspace

续上表

名 称	相应菜单项	名 称	相应菜单项
Save	File Save	Output	View Output
Save All	File Save All	Window List	View Window List
Cut	Edit Cut	Find in Files	Edit Find in Files
Copy	Edit Copy	Find	Edit Find
Paste	Edit Paste	Search	Help Search
Undo	Edit Undo		

(2) 编译工具栏 (Build MiniBar)

编译工具栏是 Build 工具栏的子集。Build Minibar 工具栏如图 1-4 所示。



图 1-4 Build MiniBar 工具栏

编译工具栏按钮对应的菜单项如表 1-11 所示。

表 1-11 编译工具栏中各功能按钮所对应的菜单项

名 称	相应菜单项	名 称	相应菜单项
Compile	Build Compile	Execute Program	Build Execute
Build	Build Build	Go	Build Start Debug Go
Stop Build	Build Stop Build	Insert/Remove Breakpoint	Edit BreakPoints

(3) 向导工具栏 (WizardBar)

向导工具栏主要用于向类中添加成员变量、方法，以及查看类中某个方法的声明和定义。向导工具栏如图 1-5 所示。

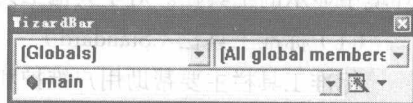


图 1-5 WizardBar 工具栏

在向导工具栏中包含 3 个下拉列表框和 1 个 WizardBar Actions 控制按钮。第 1 个下拉列表框中列出了当前项目中定义的类。如果第 1 个下拉列表框中当前显示的类关联一个对话框，则第 2 个下拉列表框中还包含对话框中控件的 ID 号。第 3 个下拉列表框中的内容会根据第 2 个下拉列表框的内容变化。如果在第 2 个下拉列表框中选择 “All class members” 选项，则第 3 个下拉列表框中会显示当前类的所有成员及成员函数，当在第 3 个下拉列表框中选择某个成员函数时，代码编辑窗口会直接定位到该函数的定义处。如果在第 2 个下拉列表框中选择一个控件 ID，则第 3 个下拉列表框中会显示与该控件 ID 有关的方法。如果某个方法没有定义，则弹出一个对话框要求用户添加该方法，否则，代码编辑窗口会直接定位到方法的定义处。

此外，向导工具栏中还包含一个 “Actions” 按钮，单击 WizardBar Actions 按钮，会弹出一个快捷菜单，从中可以选择要执行的菜单命令。

(4) 工具栏的定制

VC 6.0 为用户提供了 11 个工具栏，用户还可以根据需要对工具栏进行一些自定义的设置。操作步骤如下：

① 鼠标右键单击工具栏，在弹出的快捷菜单中选择 “Customize” 菜单项，打开 “Customize” 对话框，如图 1-6 所示，选择 “Toolbars” 选项卡。