

Visual C++

系列丛书之二

Visual C++ 用户指南

郝伟诚 章强 编著



北京大学出版社

TP312
H20

078842

Visual C++ 用户指南

郝伟诚 章 强 编著



北京大学出版社

新登字(京)159号

图书在版编目(CIP)数据

Visual C++用户指南/郝伟诚,章强编, —北京:北京大学出版社, 1994. 7

ISBN 7-301-02511-4

I. V… I. ①郝… ②章… III. C语言-程序系统-手册
IV. TP312C



出版者:北京大学出版社

地址:北京大学校内

邮政编码:100871

排印者:北京科技期刊电脑排印中心照排

北京飞达印刷厂印刷

发行者:北京大学出版社

经销者:新华书店

版本记录:787×1092毫米 16开本 13.125印张 310千字

1994年11月第1版 1994年11月第1次印刷

印数:0001—3500册

定价:25.00元

前 言

Microsoft Visual C++是微软公司于1993年推出的最新版MSC,相当于MSC 8.0。

Visual C++ 与 MSC 7.0 相比,有很大的变动,所以有必要对这个系统—比较详细的介绍。

本书包括了 Microsoft Visual C++ Workbench(第一章至第六章)和 App Studio (第七章至第十四章)两部分完整的使用手册和技术资料,全面而系统地介绍了使用 Microsoft Visual C++的基础知识和基本方法。本书主要包括以下几方面内容:

1. 安装 Microsoft Visual C++。
2. 集成环境 Workbench 入门。
3. Workbench 菜单与选项参考。
4. App Studio 入门。
5. App Studio 的“Visual”(可视化)编辑的实现(包括 6 个编辑器)。

由于 Microsoft Visual C++是在 Windows 环境下运行,故本书对 Windows 也作了较概括的介绍,如果用户希望用它开发 Windows 应用程序,最好与 Windows 技术丛书配套使用。

作 者

1993 年 11 月

目 录

第一章 预备知识	1
1.1 Windows 系统	1
1.2 鼠标器	1
1.3 Windows 的交互风格	2
1.4 Windows 中的控制风格	2
第二章 Visual C++ 安装	4
2.1 安装 Visual C++	4
2.1.1 初始安装	4
2.1.2 添加安装	5
2.2 硬件配置	6
2.3 软件配置	6
2.4 MS-DOS 的配置要求	7
2.5 启动和退出 Visual C++	7
2.5.1 启动	7
2.5.2 退出	7
2.6 进入 Help	7
第三章 Microsoft Visual C++ Workbench 入门	8
3.1 进入 Workbench	8
3.2 组成部分	9
3.2.1 菜单条与菜单	9
3.2.2 工具条	10
3.2.3 窗口和状态条	11
第四章 Workbench 菜单选项参考及使用方法	13
4.1 File 菜单	13
4.1.1 菜单命令	13
4.1.2 具体实现	15
4.2 Edit 菜单	20
4.2.1 菜单命令	20
4.2.2 编辑键	23
4.2.3 具体实现	25
4.3 View 菜单	27
4.3.1 菜单命令	27
4.3.2 具体实现	28
4.4 Project 菜单	29

4.4.1	工程类型	31
4.4.2	生成一个工程	33
4.4.3	打开和关闭工程	33
4.4.4	删除和增加工程中的文件	35
4.4.5	构造一个工程	35
4.4.6	使用工作区	36
4.4.7	使用外部工程	37
4.5	Browse 菜单	37
4.5.1	菜单命令	37
4.5.2	Browsing (浏览)键	38
4.5.3	使用 Browse 菜单命令和 Browser(浏览器)	38
4.6	Debug 菜单	43
4.6.1	Debug 菜单命令和键盘命令	43
4.6.2	使用调试命令	45
4.7	Tools 菜单	55
4.7.1	菜单功能	55
4.7.2	Tools 的应用和实现	56
4.8	Options 菜单	56
4.8.1	菜单功能	56
4.8.2	具体使用方法	61
4.9	Linker Compiler 和 Resources	66
4.9.1	Compiler 选项	67
4.9.2	Linker 选项	77
4.9.3	Resource Compiler Options	83
4.10	Window 菜单	84
4.10.1	菜单命令	84
4.10.2	窗口管理键	85
4.10.3	使用方法	85
4.11	Help 菜单	86
4.11.1	菜单命令	86
4.11.2	使用 Help	87
第五章	命令行编译器 CL 和链接器 LINK	92
5.1	使用命令行编译器和链接器	92
5.2	CL 命令行选项	92
5.3	LINK 命令行选项	95
第六章	在 Workbench 中调用工具	98
6.1	App Studio	98
6.2	AppWizard	99
6.2.1	打开(Open)和关闭(Close)AppWizard	99
6.2.2	选择工程名和子目录名	99

6.2.3	选择自己开发的应用程序子目录的驱动器(Drive)和目录	100
6.2.4	设置选项(Options)	100
6.2.5	AppWizard 产生的共同文件	103
6.2.6	设置选项后增加的文件	104
6.3	ClassWizard	104
6.4	调用 MS-DOS 和 Windows 工具	105
第七章	App Studio	108
7.1	资源	109
7.1.1	对话框(Dialog Box)	109
7.1.2	菜单(Menu)	109
7.1.3	图标(ICON)	110
7.1.4	位图(Bitmap)	110
7.1.5	加速键表(Accelerator Table)	110
7.1.6	光标(Cursor)	111
7.1.7	字符串表(String Table)	113
7.2	App Studio 概述	113
7.2.1	App Studio 菜单	113
7.2.2	工具条	118
7.2.3	状态条	119
7.2.4	App Studio 的特性(Property)窗	119
7.3	App Studio 中的对话框及其完成的任务	121
7.3.1	New 对话框	121
7.3.2	Open 对话框	121
7.3.3	Install Controls	123
7.3.4	Symbol 对话框	123
7.3.5	Properties 对话框	123
7.3.6	Import 对话框	123
7.4	App Studio 的通用鼠标操作	124
7.5	App Studio 的资源编辑器	125
7.5.1	基本操作	125
7.5.2	编辑器	127
第八章	对话框编辑器	132
8.1	预备知识	132
8.1.1	对话框类型	132
8.1.2	App Studio Controls	133
8.1.3	Dialog 编辑器工具条	133
8.1.4	如何进入 Dialog 编辑器	134
8.2	一个例子: 创建一个对话框	134
8.2.1	创建一个对话框边框(Frame)	135
8.2.2	选择对话框 Dialog 需要的各种控制	135

8.2.3	调整 Dialog 对话框中包含的控制	138
8.2.4	重新安排各控制(框)的大小	138
8.2.5	安排 Dialog 控制的位置	141
8.2.6	定义各控制的特性(Property)	143
8.2.7	生成 Dialog 模板	143
8.2.8	定义 OPEN 框中各控制的 Tab 顺序	144
8.2.9	为各控制定义简捷键	145
8.2.10	为 OPEN 对话框产生相应的代码	146
8.2.11	测试 OPEN 对话框	146
第九章	菜单编辑器	147
9.1	预备知识	147
9.1.1	有关菜单的术语	147
9.1.2	菜单的组成	148
9.1.3	开发菜单的全过程	148
9.2	一个菜单例子	148
9.2.1	创建一个 Editor Menu 资源	148
9.2.2	产生菜单条上的菜单选项	149
9.2.3	为 Tools 和 Edit 菜单产生下拉式子菜单	149
9.2.4	将 Tools 及其子菜单拷贝到 Edit 的子菜单中	152
9.2.5	为 Edit 菜单各子菜单命令设置加速键(Accelerator Key)	154
9.2.6	一点建议	154
第十章	加速键表编辑器	155
10.1	产生一个新的 Accelerator 表	155
10.2	向 Accelerator 表中加入 Editor 资源描述文件的加速键	156
10.3	为 Accelerator 表中的加速键改变对应的值	156
10.4	Accelerator 表中值有效值的有效范围	157
10.5	一点建议	157
第十一章	字符串编辑器	158
11.1	创建一个字符串表	158
11.2	向字符串表加入字符串	158
11.3	字符串编辑器	159
11.4	一点建议	160
第十二章	图形编辑器	161
12.1	位图图形编辑器概述	161
12.1.1	Image 编辑窗	161
12.1.2	图形板(Graphics Palette)	161
12.1.3	状态条(Status Bar)	163
12.1.4	Image Menu (映象菜单)	163
12.2	产生一个位图	163
12.2.1	预备操作	163

12.2.2	创建一个位图图形资源	163
12.2.3	绘出位图前的选择操作	163
12.2.4	绘出位图图形	164
12.2.5	整理位图图形	164
12.2.6	往剪接板(Clipboard)上拷贝位图中的椭圆	167
12.3	产生一个图标(ICON)	167
12.3.1	一个例子:产生一个图标(ICON)	168
12.3.2	创建一个图标资源	169
12.3.3	绘图前的准备操作	169
12.3.4	编辑编辑器的图形	170
12.4	光标(Cursor)编辑器	171
12.4.1	创建一个光标资源	171
12.4.2	绘图前的准备操作	172
12.4.3	绘图	172
12.4.4	定义“热点”(Hotspot)	172
12.5	产生一个分列式工具图形条	172
12.5.1	设置分列图形的大小	172
12.5.2	绘出各按钮图形	173
12.6	一些说明	173
12.6.1	图形编辑器的特性页	173
12.6.2	用户自己定义一个绘图刷	174
12.6.3	用鼠标器改变位图(图标、光标)的大小	174
12.6.4	ICON 和 Cursor 的映象设置	174
12.6.5	创建自己的调色板	175
第十三章	二进制码编辑器	176
13.1	进入二进制码编辑器	176
13.2	二进制码编辑器内编辑数据的方法	176
第十四章	Class Wizard	177
14.1	App Studio、Workbench 及 Class Wizard 工具的关系	177
14.2	添加新类或调入已存在的类	179
14.2.1	添加新类	179
14.2.2	使用 Class Wizard 添加一个新类	179
14.2.3	调入一个已存在的类	179
14.3	将消息映射到函数	181
14.3.1	定义消息句柄	182
14.3.2	删除一个消息处理函数	182
14.3.3	从 Class Wizard 中跳到源代码中	183
14.4	Dialog-Box 数据(Data)	183
14.4.1	对话数据交换(DDX)	184
14.4.2	为对话框数据交换定义数据成员	184

14.4.3	对话数据的有效性(DDV).....	186
14.5	一个例子：用 ClassWizard 构造一个对话框.....	187
14.5.1	为一个对话框控制定义一个消息句柄.....	187
14.5.2	使用 DDX/DDV.....	188
14.6	当代码改变时如何更新 Class Wizard.....	189
14.6.1	删除类.....	189
14.6.2	给类重新命名或移动一个类.....	190
14.6.3	重新构造 ClassWizard (.CLW)文件.....	190
14.7	更新已存在的程序代码.....	190
附录一	构造一个 Windows 应用程序样例.....	192
附录二	构造一个 QuickWin 程序样例.....	192
附录三	App Studio 资源编辑加速键.....	192
附录四	图形编辑器的键盘命令.....	193
附录五	特性(Property)页.....	194
附录六	CL 程序的命令选项与 Workbench Option 中 Compiler 对话框中各选项的对照.....	196

第一章 预备知识

Visual C++是一个在 Windows 环境下运行的程序设计语言和环境,本章简要介绍了 Windows 系统及其使用方法,以利于从未接触过 Windows 的用户能顺利地使用 Visual C++,对于能熟练使用 Windows 的用户来说,可以直接跳过本章,从第二章开始学习。

1.1 Windows 系统

Microsoft Windows 是一个多任务操作环境,它的思想和 DOS 操作系统的思想有很大的不同。它提供了图形用户界面(GUI)、高级应用程序编程接口(API)和软件开发工具箱 SDK,实现了动态数据交换、模块动态链接、自动内存管理等功能,具有标准 DOS 环境不具有的许多特征。特别是其高级应用程序编程接口的设备无关性,保证了应用程序与各种不同的显示器、打印机和其它设备的兼容性,使得开发人员能够集中精力进行软件开发,而不用过多考虑相应的设备驱动程序。

Microsoft Windows 还有一个特点,它采用了面向对象(Object-Oriented)的程序设计思想,可以对开发任务进行自动处理。此外,它还提供了一套直观优美的公共用户界面,可充分发挥微型计算机的潜力。

1.2 鼠标器

鼠标是一种被 Windows 程序设计普遍采用的输入设备,下面介绍鼠标的使用方法。由于鼠标的型号、品牌很多,这里只介绍各种鼠标都能完成的功能。

鼠标一般有两个键(或三个键),按其位置分别称为左、右(或左、中、右)键。鼠标与系统(如 Windows)连通以后,当用户移动鼠标使鼠标器光标进入或通过窗口时(或者当光标在窗口里按下或放开鼠标按钮时),系统将向有关应用程序发送消息,报告来自鼠标输入的控制,选定消息。对于鼠标器上的各个按钮(或称为键)而言,都只有两个状态:按下(Down)和放开(Up),正是这几个键的两种状态不同组合,完成了所有的菜单选择、窗口选择等一系列控制、选择功能,下面列出几个通用的操作方法:

(1) 选定(Click)操作:将鼠标指示器(Mouse Point)移动到要选择的选项上,按左(或右)键后放开。

(2) 拖(Drag):先选定选项,然后按住左(或右)键移动鼠标,直到鼠标器光标指向选项应该移到的位置,放开鼠标键。

(3) 两次选定(Double-Click): 先将鼠标指示器移到待选项上,在系统定义的时间内完成两次按键,然后放开。

1.3 Windows 的交互风格

最后介绍进入 Microsoft Windows 的方法及 Windows 的控制风格。

在 Windows 目录(或 Windows 的 BIN 子目录)里,直接键入 WIN,就可直接进入 Windows 环境,此时如图 1.1 所示。

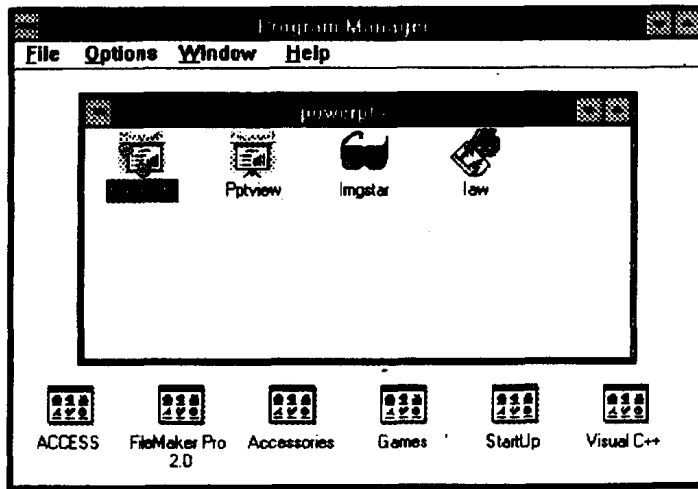


图 1.1 Windows 系统

如果 Visual C++ 已经装入机器内(见上图右下角),可以把鼠标移到 Visual C++ 图标上,然后两次选定,就可进入 Visual C++ 环境。

1.4 Windows 中的控制风格

Windows 采用了图形用户界面(GUI)与用户交互,对于习惯于 DOS 环境的用户来说,这种交互风格比较陌生,但更直观、方便,表 1.1 列出了在对话框等交互式接口中一些控制的统一风格。

表 1.1 Windows 中的一些控制风格

控制名	说明
按钮	按钮可用于直接选择一个操作,如删除(Del)、增加(Add)、取消(Cancel)、Ok 等。
圆按钮	圆按钮主要用于选择,组合操作,用户一次仅能在一组中选择一个圆按钮。

控制名	说 明
检测盒	多个检测盒经常排列在一起,以供用户选择,在小方框中出现一个叉表示这个检测盒被选中,但是它们彼此独立,用户在任何时刻可打开或关闭若干个检测盒。
编辑框	可显示数字和正文,用户也可以输入数字和正文。
列表框	显示一组字符串,例如文件名或目录名。
组合框	将一组其它的控制组合在一起(如一组选中格,一组按钮等)。

第二章 Visual C++ 安装

Visual C++ 是高级语言程序设计环境 Microsoft C 7.0 的升级版本,它不但吸收了以前版本的所有优点,而且加入了完全集成化的、基于 Windows 的开发工具,并且在传统的 C/C++ 程序设计环境中增添了目前流行的“可视化(Visual)”用户接口驱动(User Interface Drive),免去了用户开发各种图形界面之苦。

Microsoft Visual Workbench 是 Visual C++ 的集成开发环境,是 Visual C++ 开发平台的基石。第三章到第五章将重点介绍 Workbench,第六章介绍命令行程程序的使用方法,此后各章节将重点介绍 APP Studio。

本章描述了 Visual C++ 的安装、启动、退出及本系统对软、硬件的配置要求,本章介绍下列内容:

- 初始安装;
- 添加安装;
- 硬件配置要求;
- 软件配置要求;
- MS-DOS 的配置要求;
- 启动和退出;
- 进入帮助。

2.1 安装 Visual C++

本节介绍两种安装 Visual C++ 的方法,第一种方法是初次安装 Visual C++,第二种方法是在安装了 Visual C++ 之后,再装入以前没有装入的文件、应用程序等。

2.1.1 初始安装

整个初始安装过程如下:

- ① 将 Visual C++ 系统的第一张盘插在 A 驱动器里。
- ② 如果 Windows 操作系统已经装入机器内,先进入 Windows 环境中,如图 2.1。选定 File 菜单的 Run 命令,此时弹出一个 Run 对话框,如图 2.2。

在图 2.2 的 File 正文框中,用户键入 A:\SETUP 即可开始安装 Visual C++;如果机器里没有 Windows,先装入 Windows,然后重复第②步。

- ③ 在安装的屏幕上显示一个安装选择项对话框,用户可根据自己的需要设置这些选项,我们建议用户在初次安装 Visual C++ 时,采用系统的缺省选项,有些没有安装的应

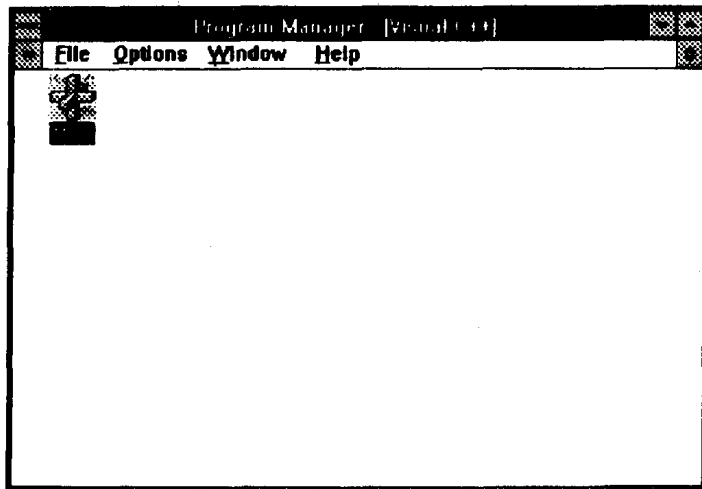


图 2.1 Windows Program Manager

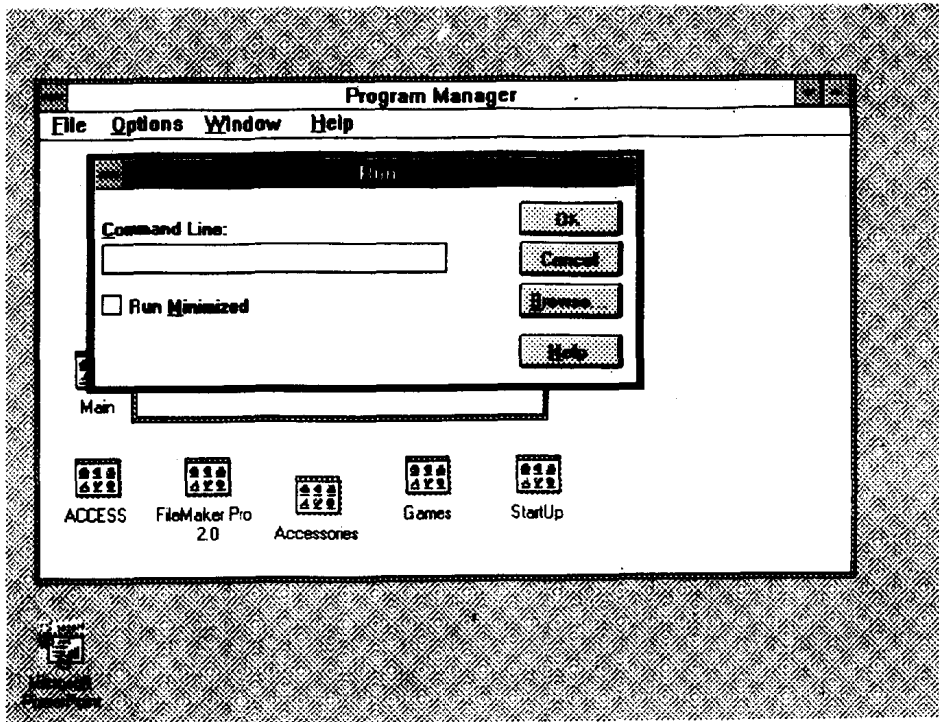


图 2.2 Run 对话框

用程序等工具,等用户熟悉这个系统且需要某个没有装入的工具后,再添加安装进系统,安装完第一盘之后,直接选择 Continue 选择项,直到安装完最后一张盘。

2.1.2 添加安装

- ① 和初次安装一样运行 SETUP。
- ② 在安装选择项对话框中,选定需要安装的种类,清除其余的检测框(见图 2.2),并

在同一组合框里,选择与这个种类相对应的命令按钮。

③ 选择命令按钮后,屏幕上将显示相应的对话框,这时清除所有的缺省选择,设置用户自己的选择项。

用户可以在 Directories, Libraries, Samples, Help Files, Tools 中设置自己需要的选择项,下面列出各对话框中各选项具体意义。

- Directories: 在 Directories 中设置 SETUP 文件所在的驱动器及根目录名。
 - Libraries: 在 Library 中设置文件所在的库名。
 - Memory Models: 可选项有 Small/Tiny, Mediam, Compact, Large/Huge。
 - Math Support: 可选项有 Emulation, 80X87, Alternate。
 - Target: 可选择项有 Windows. EXE, Windows. DLL, Quickwin. EXE 以及 MS-DOS. EXE。用户根据自己的需要可在以下三个组合框中选择,并清除不需要的选项。
 - Samples: 可从 User's Quide Samples, Windows C++ (MFC) Samples 及 Windows C(SDK) Sample 中选择需装入的样例程序种类。
 - Help Files: 在一般情况下,当用户选择 Libraries Directories 等各选择项时,系统全自动装入有关的帮助文件,不需要选择这个命令按钮单独装入帮助文件。
 - Tools: 选择 Tools 命令按钮以后,Tool Options 对话框将提供一系列 Windows MS-DOS 工具的检测盒,供用户选择即将装入的工具名。
- ④ 定义好以上对话框后,选择 OK 按钮关闭 Tool Option 对话框。
- ⑤ 选择 Continue 按钮开始安装。

2.2 硬件配置

要使 Visual C++ 得以顺利运行,必须至少具备以下配置:

- ① 能运行 MS-DOS 5.0 及其更高版本的 IBM PC 机及其兼容机。
- ② 一个 VGA 监测器(Monitor)。
- ③ 一个 80386 或更高的处理器。
- ④ 至少有 4M 可用内存(最好有 6M)。
- ⑤ 一个足以存储用户安装 Visual C++ 系统的硬盘(至少需要 40M 存储空间)。
- ⑥ 至少有一个 1.2M、5.25 英寸的磁盘驱动器,或者一个 1.44M、3.5 英寸的磁盘驱动器。

2.3 软件配置

需要机器里安装有在增强模式下运行的 Microsoft Windows 或 Microsoft Windows for WorkGroup 3.1 系统。

2.4 MS-DOS 的配置要求

在不改变安装目录名的情况下,为了保证 Visual C++ 正常运行,需要对 AUTOEXEC.BAT 程序作以下改动:

在 PATH 变量(如果不存在,加上 PATH=)中加入 C:\MSVC\BIN;

在 INCLUDE 变量(如果不存在,则加上 SET INCLUDE=)后加上 C:\MSVC\INCLUDE;C:\MSVC\MFC\INCLUDE;

在 LIB 变量(如果不存在,则加上 SET LIB=)后加上 C:\MSVC\LIB;C:\MSVC\MFC\LIB;

另外,为了能在 DOS 状态下或 Windows 环境里正常运行 CL, LINK, NMAKE, CV-PACK 等应用程序,还必须对 SYSTEM.INI 作如下改动:

在 SYSTEM.INI 中加入一行: DEVICE=C:\MSVC\BIN\DOSEXNT.386。

注意:以上对程序 SYSTEM.INI 和 AUTOEXEC.BAT 作改动时,都是假设 Visual C++ 安装在 C 盘,如果不在 C 盘,只需改变驱动器号即可。

2.5 启动和退出 Visual C++

2.5.1 启动

当安装了 Visual C++ 之后,在 Windows 的 Program manager 程序组窗口中会出现 Visual C++ 的图标,只需两次选定(Double-Click)该图标,就可以进入 Visual C++ 环境。

2.5.2 退出

当 Workbench(或别的工具,如 App Studio)操作进行完毕后,选择 File 菜单的 Exit 命令或两次选定 Workbench 窗口左上角的控制菜单图标,就可以退出 Workbench,然后再两次选定 Microsoft Visual C++ 窗口左上角的控制菜单图标,可以完全退出 Visual C++ 环境。

2.6 进入 Help

在 Visual C++ 的任何应用程序中都可以访问帮助菜单(Help), Visual C++ 丰富的帮助信息是它的一大特色,获得帮助主要有以下两种方法:

- ① 通过 Help 菜单得到帮助信息。
- ② 按 F1 键直接得到屏幕上有关信息的联机帮助。

若想得到某一菜单或屏幕区域的进一步帮助信息,可按 Shift+F1 键,这时屏幕上出现一个带“+”的问号光标,将光标移动到菜单或区域处并按下鼠标左按钮即可。