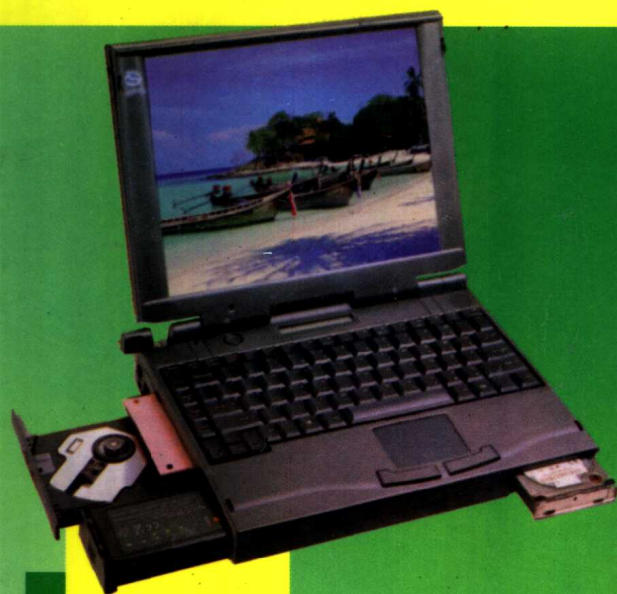


跨世纪中学生计算机学习丛书

王兴国 李献业 编著

# Visual C++ 基础



北京科学技术出版社

跨世纪中学生计算机丛书

# Visual C++ 基础

王兴国 李献业 编著

王兴国

北京科学技术出版社

**图书在版编目(CIP)数据**

Visual C++ 基础/王兴国等编著. - 北京:北京科学技术出版社,2001.5 重印  
(跨世纪中学生计算机学习丛书)

ISBN 7-5304-2196-4

I.V… II.王… III.C语言-程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(98)第 18612 号

**Visual C++ 基础**

王兴国 李献业 编著

\*

**北京科学技术出版社出版**

(北京西直门南大街 16 号)

邮政编码: 100035

---

各地新华书店经销

北京交通印务实业公司印刷

\*

787 毫米×1092 毫米 16 开本 13.75 印张 330 千字

1998 年 8 月第一版 2001 年 5 月第二次印刷

---

**定价: 18.00 元**

(凡购买北京科学技术出版社的图书,如有  
缺页、倒页、脱页者,本社发行科负责调换)

## 编 委 会

**主任委员：** 吕传兴 （中国教育学会中小学计算机教育研究会副理事长）

**副主任委员：** 宋玉升 （中国计算机学会普及工作委员会委员）

**编委：**（以姓氏笔划为序）

毛国平 （吉林省教育学院）

孙殿德 （上海市长宁区教育学院）

姚 期 （上海市第三女子中学）

段 青 （海南省教委教科所）

陈 平 （南京市教学研究室）

高晓红 （西安市教委教研室）

曾庆宇 （北京科学技术出版社）

# 前 言

Visual C++ 是目前比较流行的编程工具,是用于开发面向对象程序的优秀软件包。使用 C++, 可以使复杂的大型应用程序开发过程简单化。

本书介绍的软件环境是 Visual C++ 1.52 版本软件包。它既可以在 Windows 3.2 上运行,也可以在 Windows 95 上使用。本书的内容完全适用 Visual C++ 1.0 版及 2.0 版本软件包。

尽管市面上介绍 C++ 的书籍很多,但是适合于初学者和中学生入门的书很少,起码让初学者感到不亲切,更不要说自学了。

本书的起点很低,只要学习过 C 语言,就可以通过本书掌握 C++ 编程。考虑到中学生的特点,在编写过程中,尽量使用通俗易懂的语言、内容安排上循序渐进,由浅入深,书中通过大量的例子,较直观地将一些难于理解的 C++ 概念表达清楚,非常适合于中学生及初学者使用。

C++ 的语法确实很复杂,初学者只要理解最简单的、能看懂的部分,先编写一些简单的小程序。读者可以从书中的例子学起,循序渐进地掌握 C++。

全书共分 11 章。第 1 章介绍 C++ 的发展历程和 Visual C++ 的安装环境。第 2 章介绍 Visual C++ 1.52 版本软件包的使用,读者对本章可以大概了解即可,在使用时再逐步掌握。第 3 章介绍使用 Visual C++ 编写一个应用程序的基本过程。第 4 章介绍 C++ 中除面向对象特点之外,相对 C 语言新增加的特性。C 语言中有的, C++ 都可以使用,而且 Visual C++ 还有些扩展,学过 C 语言的读者只要找出它们的差别就行了。第 5 章介绍如何定义一个类,以及通过类的实例——对象实现类的操作。从第 5 章起,开始介绍 C++ 面向对象的特点。类和对象是面向对象程序设计中的最基本的概念,读者必须掌握类和对象的概念。第 6、7 章是对第 5 章的补充,进一步讨论 C++ 中的类和对象。第 8 章介绍类的继承性,重点掌握公有派生类。第 9 章介绍虚函数。对于虚函数,读者应从实用的角度出发,掌握虚函数的使用方法,而不要追究概念。第 10 章介绍运算符重载的概念和常用运算符的重载。运算符重载是面向对象的一个很重要的技巧,所以本章也是学习 C++ 的重点之一。第 11 章介绍 C++ 的输入输出系统。C++ 的输入输出系统功能强大,读者只需掌握一些常用的功能。在本书的编写过程中,自始至终得到吉林省教育学院毛国平老师的悉心指导和帮助,全书由毛国平老师审稿,在此表示衷心感谢。

相对而言, C++ 是比较难于掌握的,但愿本书能成为你学习 C++ 的良师益友。由于作者的水平有限,写作中的疏漏在所难免,敬请使用本书的读者批评指正。

作 者

1998 年 4 月

# 目 录

|                                            |      |                             |      |
|--------------------------------------------|------|-----------------------------|------|
| 第1章 C++ 语言概述 .....                         | (1)  | 第4章 C++ 对 C 的增强 .....       | (23) |
| § 1.1 C++ 语言的发展历程 .....                    | (1)  | § 4.1 C++ 注释 .....          | (23) |
| 一、C++ 语言的起源 .....                          | (1)  | § 4.2 C++ 的输入输出简介 .....     | (23) |
| 二、C++ 的特点 .....                            | (1)  | § 4.3 C++ 标准数据类型 .....      | (25) |
| § 1.2 Visual C++ 1.52 的运行环境及软<br>件安装 ..... | (1)  | § 4.4 变量说明的位置 .....         | (26) |
| 一、Visual C++ 1.52 版软件包的运行环境 ..             | (1)  | § 4.5 内联函数 .....            | (27) |
| 二、Visual C++ 1.52 版软件包的安装 .....            | (2)  | § 4.6 函数原型 .....            | (29) |
| 第2章 Visual C++ 1.52 简介 .....               | (3)  | § 4.7 缺省函数参数 .....          | (30) |
| § 2.1 Visual C++ 软件包的组成 .....              | (3)  | § 4.8 const 修饰符 .....       | (31) |
| § 2.2 启动 Visual C++ 的工作平台 ..               | (3)  | § 4.9 volatile 修饰符 .....    | (32) |
| 一、使用 Windows95 启动 Visual C++ .....         | (3)  | § 4.10 void .....           | (33) |
| 二、使用 Windows3.2 启动 Visual C++ .....        | (5)  | 一、在函数中使用 void .....         | (33) |
| § 2.3 Visual C++ 的主菜单 .....                | (5)  | 二、void 类型的指针 .....          | (33) |
| 一、File(文件)菜单 .....                         | (5)  | § 4.11 域分辨操作符:: .....       | (34) |
| 二、Edit(编辑)菜单 .....                         | (6)  | § 4.12 引用类型 .....           | (35) |
| 三、View(查看)菜单 .....                         | (7)  | 一、引用的基本概念 .....             | (35) |
| 四、Project(项目)菜单 .....                      | (7)  | 二、引用参数 .....                | (36) |
| 五、Browse(浏览)菜单 .....                       | (9)  | 三、返回引用 .....                | (37) |
| 六、Debug(调试)菜单 .....                        | (9)  | § 4.13 结构 .....             | (38) |
| 七、Tools(工具)菜单 .....                        | (10) | § 4.14 枚举类型 .....           | (39) |
| 八、Options(选项)菜单 .....                      | (10) | § 4.15 new 和 delete .....   | (40) |
| 九、Windows(窗口)菜单 .....                      | (12) | § 4.16 函数重载 .....           | (42) |
| 十、Help(帮助)菜单 .....                         | (12) | § 4.17 链接说明 .....           | (43) |
| § 2.4 Visual C++ 工具栏 .....                 | (14) | 练习题 .....                   | (44) |
| 第3章 使用 Visual C++ 编写应用<br>程序 .....         | (16) | 第5章 类 .....                 | (45) |
| § 3.1 构造一个 QuickWin 应用<br>程序 .....         | (16) | § 5.1 在 C++ 中创建新的数据类型 ..... | (45) |
| 一、创建一个新项目 .....                            | (16) | 一、类的说明 .....                | (45) |
| 二、编辑源文件 .....                              | (18) | 二、类的实现 .....                | (47) |
| 三、编译和链接 .....                              | (18) | 三、类的使用(对象) .....            | (48) |
| 四、运行程序 .....                               | (19) | 四、成员函数(包括内联成员函数) .....      | (51) |
| § 3.2 分析应用程序 .....                         | (21) | § 5.2 构造函数和析构函数 .....       | (54) |
| 练习题 .....                                  | (22) | 一、构造函数 .....                | (54) |
|                                            |      | 二、重载构造函数 .....              | (56) |
|                                            |      | 三、缺省构造函数 .....              | (58) |

|                           |       |                                  |       |
|---------------------------|-------|----------------------------------|-------|
| 四、析构函数 .....              | (59)  | § 8.2 派生类 .....                  | (112) |
| § 5.3 this 指针 .....       | (61)  | 一、公有派生类 .....                    | (113) |
| § 5.4 举例 .....            | (63)  | 二、私有派生 .....                     | (115) |
| 练习题 .....                 | (68)  | 三、在派生类的成员函数中访问基类成员 .....         | (116) |
| <b>第 6 章 对象</b> .....     | (69)  | 四、保护派生 .....                     | (119) |
| § 6.1 拷贝构造函数 .....        | (69)  | 五、类的友员没有继承性 .....                | (120) |
| 一、缺省拷贝构造函数 .....          | (69)  | § 8.3 公有派生类与基类的转换 .....          | (120) |
| 二、自定义拷贝构造函数 .....         | (70)  | § 8.4 继承下派生类构造函数的设计 .....        | (122) |
| § 6.2 对象初始化和对象赋值 .....    | (72)  | 一、派生类构造函数的初始化 .....              | (122) |
| 一、对象初始化 .....             | (72)  | 二、对派生类构造函数的进一步讨论 .....           | (124) |
| 二、对象赋值 .....              | (72)  | § 8.5 成员覆盖 .....                 | (128) |
| 三、对象初始化和对象赋值的区别 .....     | (75)  | § 8.6 举例 .....                   | (131) |
| § 6.3 在函数中使用对象 .....      | (75)  | 练习题 .....                        | (136) |
| 一、以传值方式传递对象 .....         | (75)  | <b>第 9 章 虚函数与多态性</b> .....       | (137) |
| 二、以传地址方式传递对象 .....        | (77)  | § 9.1 动态连接与静态连接 .....            | (137) |
| 三、以传引用方式传递对象 .....        | (78)  | § 9.2 虚函数的概念 .....               | (139) |
| 四、从函数返回对象 .....           | (79)  | § 9.3 调用虚函数 .....                | (141) |
| § 6.4 对象数组 .....          | (81)  | 一、以静态连接方式调用虚函数 .....             | (141) |
| 一、对象数组概念 .....            | (81)  | 二、以动态连接方式调用虚函数 .....             | (143) |
| 二、初始化对象数组 .....           | (83)  | § 9.4 虚函数与继承的关系 .....            | (145) |
| 三、对象数组和缺省构造函数 .....       | (83)  | § 9.5 覆盖函数与虚函数 .....             | (146) |
| 四、对象数组和 new, delete ..... | (84)  | § 9.6 对虚函数的进一步说明 .....           | (147) |
| § 6.5 对象指针 .....          | (85)  | 一、显式访问 .....                     | (147) |
| § 6.6 对象成员 .....          | (87)  | 二、纯虚函数和抽象类 .....                 | (148) |
| § 6.7 举例 .....            | (89)  | 三、虚析构函数 .....                    | (149) |
| 练习题 .....                 | (92)  | § 9.7 举例 .....                   | (152) |
| <b>第 7 章 类的其它特性</b> ..... | (93)  | 练习题 .....                        | (160) |
| § 7.1 静态成员 .....          | (93)  | <b>第 10 章 运算符重载与类型转换函数</b> ..... | (162) |
| 一、静态数据成员 .....            | (93)  | § 10.1 运算符重载的概念 .....            | (162) |
| 二、静态成员函数 .....            | (95)  | § 10.2 实现运算符重载 .....             | (163) |
| § 7.2 const 成员 .....      | (97)  | 一、运算符重载的时机 .....                 | (163) |
| 一、const 数据成员 .....        | (97)  | 二、成员运算符函数 .....                  | (165) |
| 二、const 成员函数 .....        | (101) | 三、友元运算符函数 .....                  | (168) |
| § 7.3 友员 .....            | (103) | § 10.3 重载双目运算符 .....             | (170) |
| 一、友员函数 .....              | (103) | 一、使用成员运算符函数 .....                | (170) |
| 二、友员类 .....               | (105) | 二、使用友元运算符函数 .....                | (174) |
| § 7.4 举例 .....            | (106) | § 10.4 重载单目运算符 .....             | (175) |
| 练习题 .....                 | (109) |                                  |       |
| <b>第 8 章 继承</b> .....     | (110) |                                  |       |
| § 8.1 继承的概念 .....         | (110) |                                  |       |

|                                  |       |                          |       |
|----------------------------------|-------|--------------------------|-------|
| 一、使用成员运算符函数 .....                | (175) | 三、成员函数 flags() .....     | (192) |
| 二、使用友元运算符函数 .....                | (177) | 四、其他的成员函数 .....          | (193) |
| § 10.5 运算符重载小结 .....             | (178) | § 11.3 操纵算子 .....        | (195) |
| § 10.6 数据类型的转换 .....             | (179) | 一、使用操纵算子 .....           | (195) |
| 一、内建数据类型转换成类类型 .....             | (179) | 二、操纵算子与类 ios 成员函数的比较 ... | (197) |
| 二、类型转换函数 .....                   | (182) | § 11.4 用于输出的运算符<< .....  | (198) |
| 三、数据类型转换的不确定性 .....              | (183) | 一、预定义的插入类型 .....         | (198) |
| § 10.7 举例 .....                  | (184) | 二、重载插入符<< .....          | (199) |
| 练习题 .....                        | (186) | § 11.5 用于输入的运算符>> .....  | (201) |
| <b>第 11 章 再论C++ 的输入/输出</b> ..... | (187) | 一、预定义的提取类型 .....         | (201) |
| § 11.1 输入/输出流类库 .....            | (187) | 二、重载提取符>> .....          | (203) |
| 一、流的概念 .....                     | (187) | § 11.6 文件输入/输出流 .....    | (205) |
| 二、流库的基本结构 .....                  | (188) | 一、打开一个文件流 .....          | (205) |
| 三、Visual C++ 中预定义的流 .....        | (188) | 二、文件数据流的操作 .....         | (206) |
| § 11.2 格式化输入/输出 .....            | (188) | 三、举例 .....               | (207) |
| 一、格式标志 .....                     | (189) | 练习题 .....                | (210) |
| 二、成员函数 setf() .....              | (190) |                          |       |



# 第 1 章 C++ 语言概述

## § 1.1 C++ 语言的发展历程

### 一、C++ 语言的起源

C 语言是世界上最受喜爱和应用最广的程序设计语言之一，它有许多突出的优点，如使用灵活、可移植性强等。但是，C 语言仍然是一种结构化的程序设计语言，不适合开发大型的应用程序。

C++ 是为了适应开发和维护复杂的应用软件的需要而研制的。1980 年，Bjarne Stroustrup 在贝尔实验室里首先提出对 C 进行改进和扩充，最初把这种语言称为“含类的 C”，到 1983 年才改名为 C++。

C++ 语言是 C 语言的扩充，它是 C 的一个超集（即 C 中有的 C++ 中都有）。C++ 对 C 中固有的特性做了一些简单的增强，另外还增加了 C 中没有的一些新的重要特性。

### 二、C++ 的特点

C++ 语言是一种灵活、高效、可移植的面向对象语言，它的主要特点有：

1. C++ 语言保持了与 C 完全兼容。现有的用 C 语言编写的代码，几乎不经任何修改就可以为 C++ 所用。所以，C 程序员仅需要学习其新特性。
2. C++ 语言融合了面向对象的能力，它在 C 语言的基础上增加了数据抽象、继承、多态性以及其它一些改善 C 程序设计结构的机制。
3. C++ 语言与其他面向对象程序设计语言相比，最重要的优点是程序的执行效率高。

## § 1.2 Visual C++ 1.52 的运行环境及软件安装

迄今为止，Visual C++ 软件包已经发行许多版本，包括 1.0 版、1.5X 版、2.0 版、4.X 版、5.0 版等。前两种版本的软件包用于开发 16 位的应用程序，可以在 Windows 3.2 或 Windows 95 上运行。从 2.0 版以后的软件包，主要用于开发 32 位的应用程序，只能在 Windows 95 上运行。

本书介绍的内容主要适用于 Visual C++ 1.52 版软件包，也适用于 Visual C++ 1.0 版和 2.0 版。Visual C++ 1.52 版软件包的使用方法，与 Turbo C 2.0 版及 Borland C 3.1 版的风格很接近，所以，学习过 C 语言的读者，很容易掌握 Visual C++ 1.52 版软件包的使用。

### 一、Visual C++ 1.52 版软件包的运行环境

使用 Visual C++ 1.52 版软件包，需要有一定的硬件环境与软件环境的支持，具体说

明如下：

1. 硬件环境

- PC386 以上兼容机。
- 内存至少 4MB，100MB 以上硬盘。
- VGA 显示器。
- 鼠标器。

2. 软件环境

- DOS 3.0 版本以上。
- Windows 3.2 或 Windows 95。

## 二、Visual C++ 1.52 版软件包的安装

1. 首先安装 Windows 3.2 或 Windows 95。

2. 然后在 Windows 中，执行 Visual C++ 1.52 版软件包系统盘上的 SETUP 命令，开始安装。

进入安装程序以后，屏幕上会显示安装提示信息，按提示进行操作即可完成安装工作。安装完毕，Windows 会自动为 Visual C++ 建立一个程序组。

## 第 2 章 Visual C ++ 1.52 简介

### § 2.1 Visual C ++ 软件包的组成

Visual C ++ 1.52 软件包的主要组成部分包括：

#### 1. Visual C ++ 工作平台

工作平台是 Visual C ++ 的集成开发环境，用以创建、调试程序和浏览程序源代码。

#### 2. Microsoft C/C ++ 编译器

用以对源程序进行编译连接，生成可执行程序。

#### 3. Microsoft 基本类库

Microsoft 基本类库（简称 MFC）用来创建 Windows 应用程序。

#### 4. 资源编辑器：AppStudio

AppStudio 用来为 Windows 应用程序创建资源。例如菜单、对话框和图标等。

#### 5. 运行库

运行库包含大量的标准库函数。例如 `abs()`、`printf()` 函数等。

#### 6. 范例程序

Visual C ++ 软件包中包含大量的例子，供我们学习使用。

#### 7. 联机帮助

Visual C ++ 软件包提供了丰富的联机帮助信息，当遇到疑难问题时，可以随时解答。

### § 2.2 启动 Visual C ++ 的工作平台

本书主要介绍如何在 Windows 95 上使用 Visual C ++ 1.52 软件包。

无论是在 Windows 95 或 Windows 3.2 上，对 Visual C ++ 1.52 软件包的操作方法是一样的，只是启动 Visual C ++ 的工作平台的步骤稍有不同。

如果你已经安装了 Windows 及 Visual C ++ 软件包，下面就可以启动 Visual C ++ 的工作平台。

#### 一、使用 Windows 95 启动 Visual C ++

1. 开机后即可进入 Windows 95。

2. 使用鼠标单击“开始”，弹出“开始菜单”。

3. 将鼠标移动到开始菜单中的“程序”选项上，弹出“程序菜单”。

4. 将鼠标移动到程序菜单中的 Microsoft Visual C ++ 选项上，弹出“Microsoft Visual C ++ 菜单”。

以上几个步骤可参看图 2.1。

5. 在 Microsoft Visual C ++ 菜单中，使用鼠标单击 Visual C ++ 选项。

这时，Windows 将打开 Visual C ++ 工作平台，如图 2.2 所示。

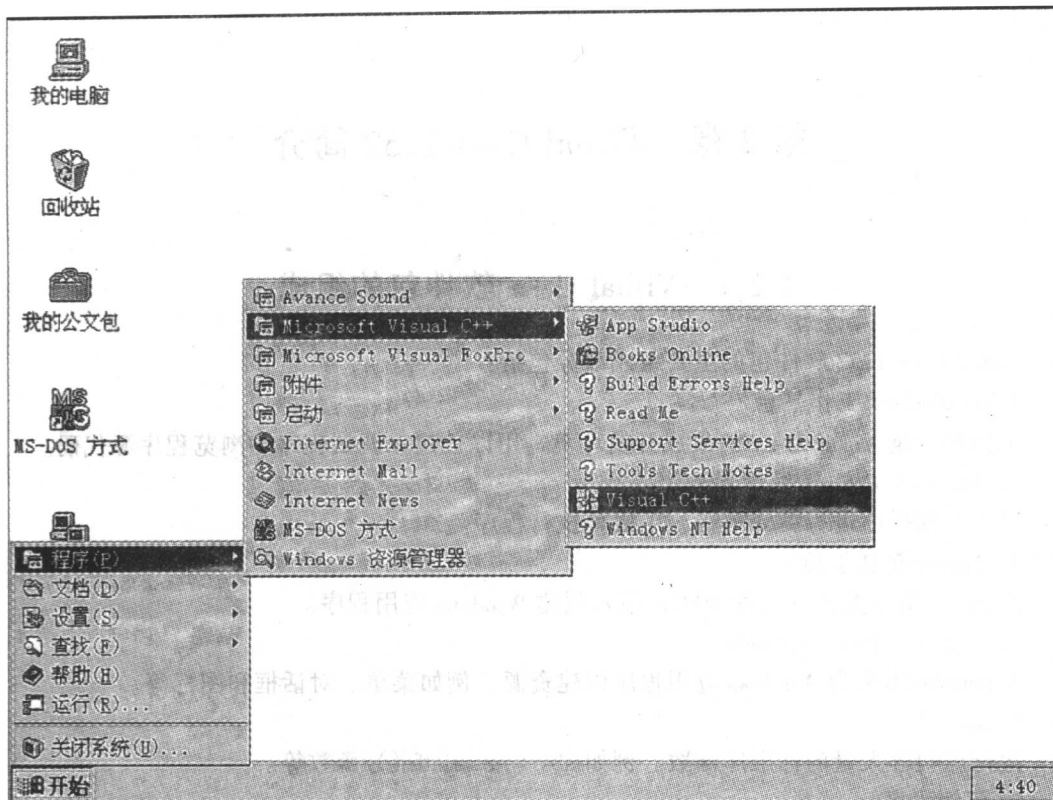


图 2.1 启动 Visual C++ 工作平台的步骤

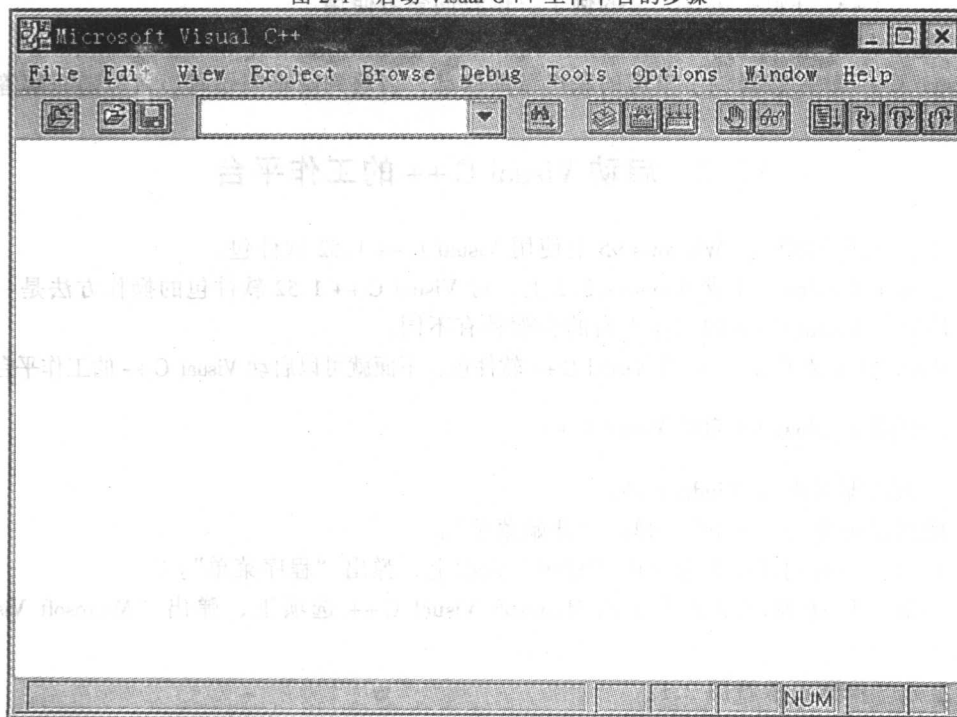


图 2.2 Visual C++ 工作平台

可以看到，Visual C++ 工作平台分为四部分：  
主菜单栏，工具按钮栏，编辑窗口区，状态信息行。

**二、使用 Windows 3.2 启动 Visual C++**

- 1. 键入 win，进入 Windows 3.2。
  - 2. 使用鼠标双击 Visual C++ 程序组的图标打开C++ 程序组。
- 这时，Windows 打开 Visual C++ 程序组窗口，如图 2.3 所示。

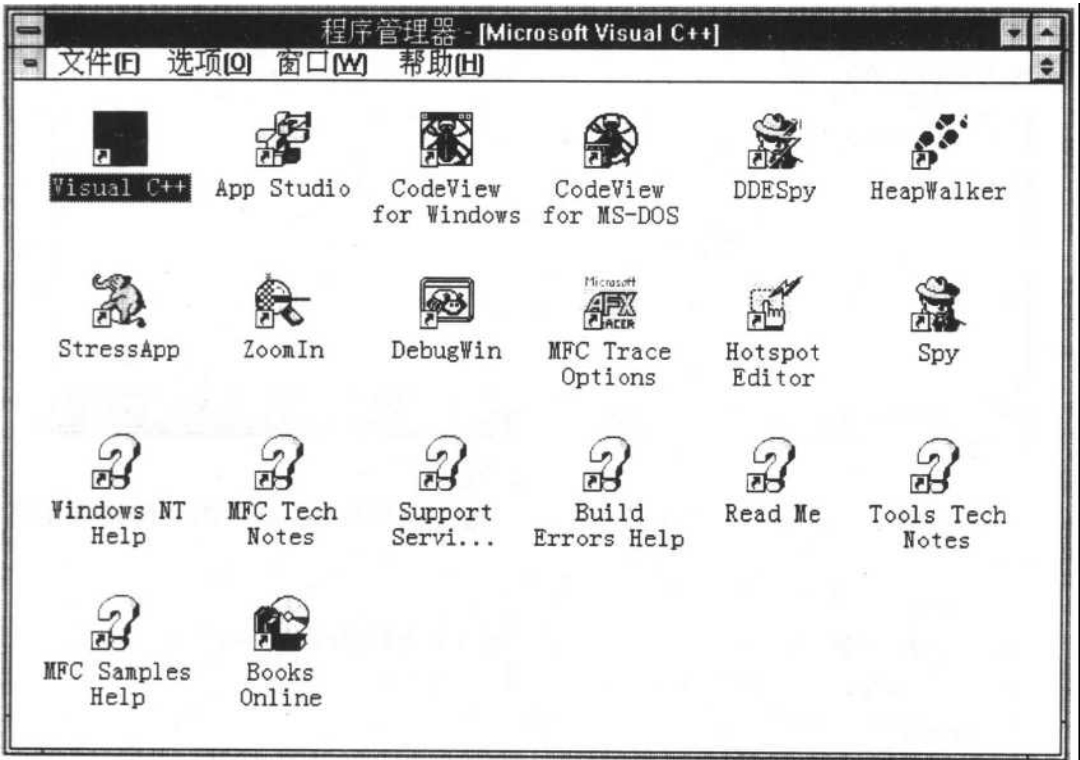


图 2.3 Visual C++ 程序窗口

注意，双击是指快速连续地按两下鼠标左钮，单击是指按一下鼠标左钮。

- 3. 在 Visual C++ 程序组窗口中双击 Visual C++ 图标。

这时，Windows 打开 Visual C++ 工作平台，如图 2.2 所示。

**§ 2.3 Visual C++ 的主菜单**

**一、File（文件）菜单**

File 菜单用来管理文件。该菜单包含九个命令，如图 2.4 所示。

- 1. New 命令，创建一个新文件。
- 2. Open 命令，打开一个已存在的文件。

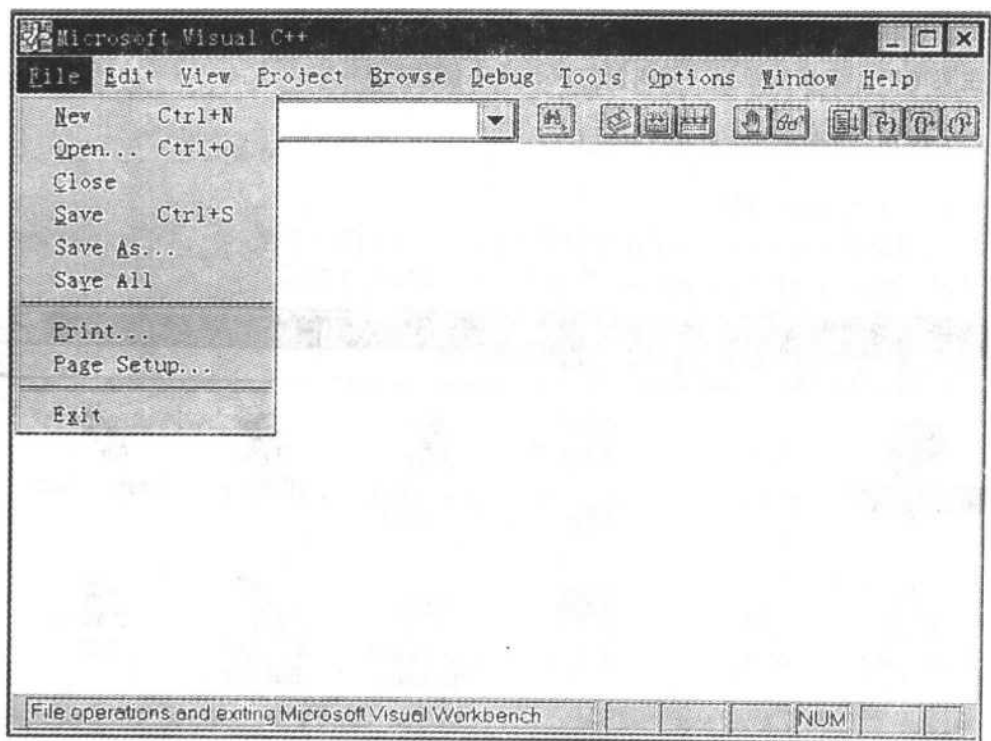


图 2.4 File 菜单

3. Close 命令，关闭当前活动的文件窗口。当前活动窗口的特征是它的标题栏被加亮成深色。

4. Save 命令，保存文件。

5. Save As 命令，以一个新文件保存当前活动文件窗口中的文件内容。

6. Save All 命令，保存当前所有打开的文件。

7. Print 命令，打印活动窗口中的文件内容。

8. Page Setup 命令，设置打印输出结果。

9. Exit 命令，退出 Visual C++ 的工作平台。

## 二、Edit (编辑) 菜单

Edit 菜单用来编辑文件、查找、替换文字串。Edit 菜单包括十个命令。如图 2.5 所示。

1. Undo 命令，使最近一次的编辑操作无效。

2. Redo 命令，取消最近一次的 Undo 命令操作。

3. Cut 命令，把选定的文字移送到剪贴板中。所谓选定的文字，是指用深色加亮的文字块。

4. Copy 命令，把选定的文字块复制到剪贴板中。

5. Paste 命令，把剪贴板中的文字粘贴到光标所在位置。

6. Delete 命令，删除选定的文字。

7. Find 命令，查找文字串。

8. Replace 命令，替换文字串。

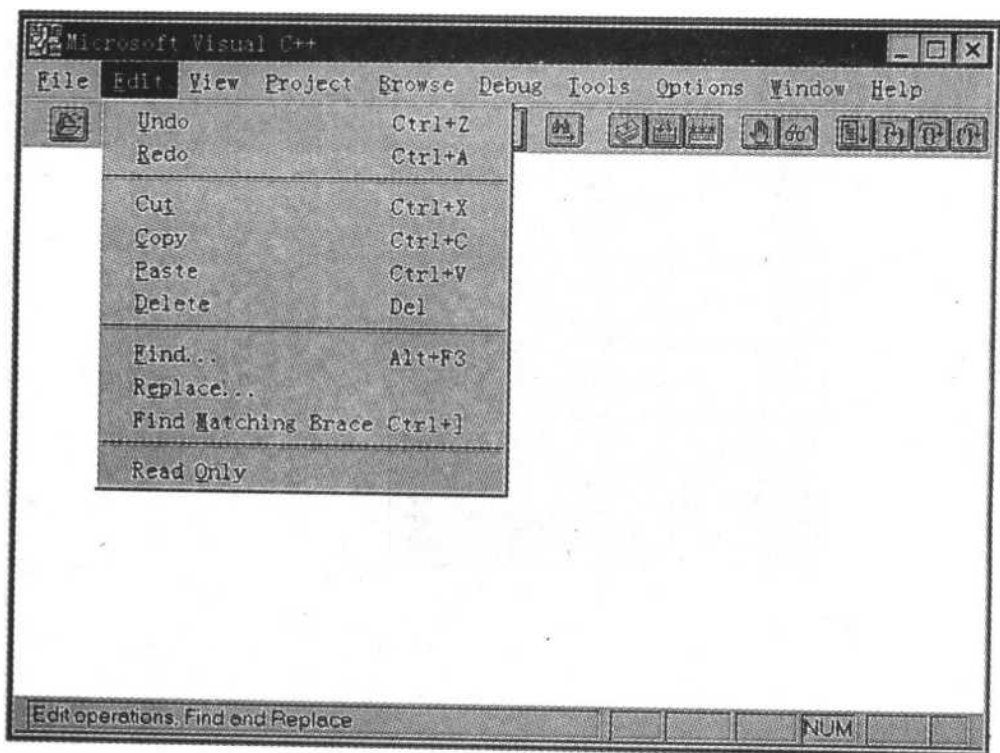


图 2.5 Edit 菜单

9. Find Matching Brace 命令，查找匹配的括号。
10. Read Only 命令，把活动窗口设置成只读状态。

### 三、View (查看) 菜单

View 菜单提供了 11 个命令，为我们观察源代码提供了方便条件，如图 2.6 所示。

1. Line 命令，把光标移动到指定的行，并显示该行。
2. MixedSource/Asm 命令，设置以源代码与汇编代码混合方式显示代码。
3. Next Error 命令，把光标移动到源文件中下一条程序编译错误发生的地方。
4. Previous Error 命令，把光标移动到源文件中前一条程序编译错误发生的地方。
5. Toggle Bookmark 命令，设置或清除书签。
6. Next Bookmark 命令，把光标移动到下一个书签处。
7. Previous Bookmark 命令，把光标移动到上一个书签处。
8. Clear All Bookmark 命令，清除所有书签。
9. Tool Bar 命令，显示或清除工具栏。
10. Status Bar 命令，显示或清除状态栏。
11. Syntax Coloring 命令，用来指定以何种方式对源代码进行语法着色。

### 四、Project (项目) 菜单

Project 菜单用来创建和管理应用程序。包括 14 个命令，如图 2.7 所示。

1. App Wizard 命令，创建一个 Windows 应用程序的框架，该框架使用了 Microsoft 基本类库。

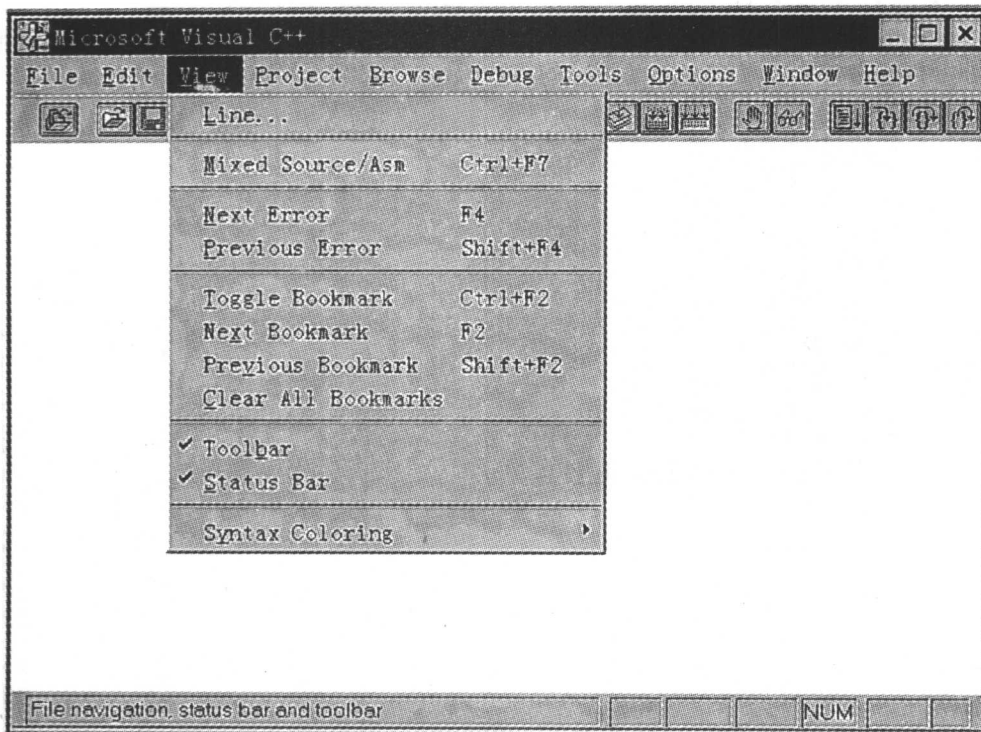


图 2.6 View 菜单

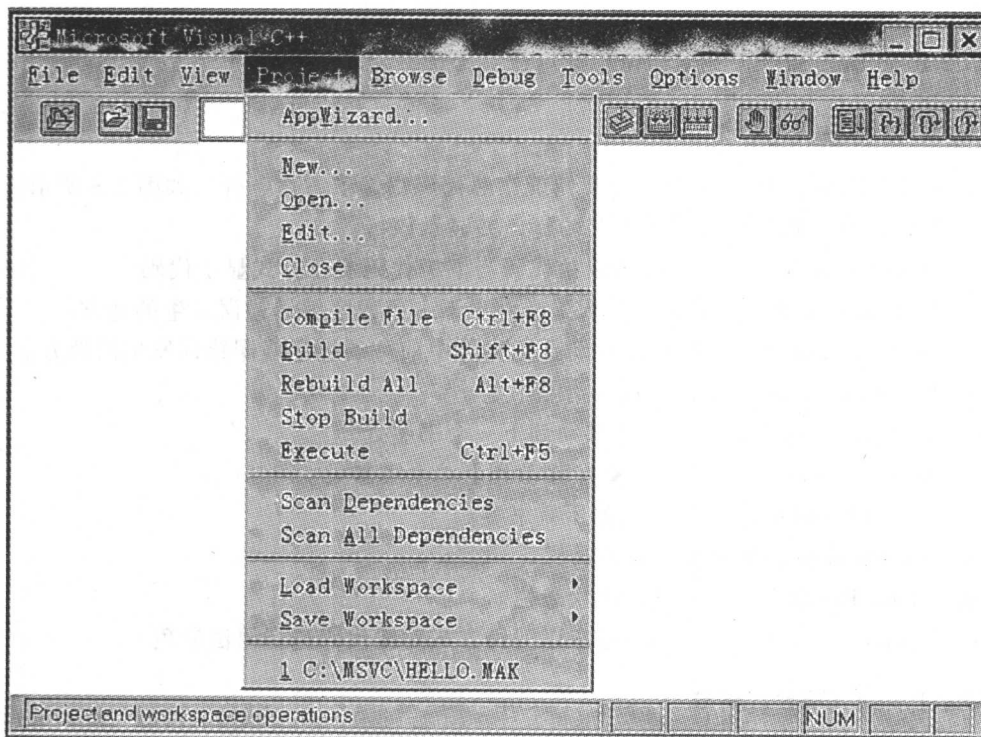


图 2.7 Project 菜单



2. New 命令, 创建一个新项目。
3. Open 命令, 打开一个已存在的项目。
4. Edit 命令, 向当前的项目中添加文件或删除文件。
5. Close 命令, 关闭当前的项目。
6. Comppile File 命令, 编译活动窗口中的源文件。
7. Build 命令, 依据文件之间的依赖关系构造可执行文件。
8. Rebuild All 命令, 重新编译项目中的所有文件, 生成可执行文件。
9. Stop Build 命令, 停止当前生成可执行文件的过程。
10. Execute 命令, 运行当前的应用程序。
11. Scan Dependencies 命令, 为活动窗口中的文件更新依赖关系。
12. Scan All Dependencies 命令, 为项目中的所有文件更新依赖关系。
13. Load Workspace 命令, 装入以前保存的、与当前项目相联系的三个工作空间中的一个。
14. Save Workspace 命令, 保存与当前项目相联系的三个工作空间中的一个。

## 五、Browse (浏览) 菜单

Browse 菜单中提供了若干命令, 可以对程序中的符号进行浏览, 这里的符号是指程序中的变量名、函数名、类型名等等。

Browse 菜单包括 14 个命令, 如图 2.8 所示。

1. Go to Definition 命令, 跳到符号被定义的地方。
2. Go to Reference 命令, 把光标移动到符号第一次被引用的地方。
3. Next 命令, 把光标移动到符号的下一次定义处或引用的地方。
4. Previous 命令, 把光标移动到符号的前一次定义处或引用的地方。
5. Pop Context 命令, 把光标移动到 Go to Definition 或 Go to Reference 命令作用之前所选定的符号处。
6. Open 命令, 用来打开 Browse 窗口, 并打开当前项目的 Browser 数据库。
7. Class Wizard 命令, 用于创建新的基于 Microsoft 基本类库 (MFC) 的类, 或者向已经存在的基于 MFC 的类加入新的消息, 映射给成员函数。

## 六、Debug (调试) 菜单

Debug 菜单用来调试应用程序, 包括 10 个命令, 如图 2.9 所示。

1. Go 命令, 用来运行程序, 直到遇到断点或到达程序终点。
2. Restart 命令, 重新运行程序, 直到遇到断点或到达程序终点。
3. Stop Debugging 命令, 终止应用程序调试过程。
4. Step Into 命令, 单步调试当前应用程序行。如果程序行中有函数调用, 将跟踪进入该函数的函数体。
5. Step Over 命令, 也是用来单步调试当前应用程序行。但是, 该命令不跟踪函数调用。
6. Step Out 命令, 把程序运行到该函数调用完成后的下一条语句。
7. Stepto Cursor 命令, 把程序运行到光标所在位置处。