



新编 21 世纪高等院校计算机系列规划教材

Visual C++ 6.0

程序设计

北京希望电子出版社 总策划

陈浩杰 主 编

张立华 黄金明 副主编

陈浩杰 董晓华 编 著



新编 21 世纪高等院校计算机系列规划教材

TP312
1966

Visual C++ 6.0

程序设计

北京希望电子出版社 总策划

陈浩杰 主 编

张立华 黄金明 副主编

江苏工业学院图书馆
藏书章

 科学出版社
www.sciencep.com

内 容 简 介

本书详细介绍了利用 Visual C++ 6.0 开发应用程序的各项知识和程序设计的基本原理。

本书全面深入地讲解了 Visual C++ 6.0 的各项功能,并构筑了 Visual C++ 程序设计的几个模块。全书共包括 14 章,内容包括 C++ 语法、Windows C 程序设计、简单应用程序框架及以此为基础的相关类和资源的使用、文档/视图结构等。每章首先介绍基础原理,再结合实例和思考练习题,引领您走进程序设计的殿堂。

全书内容由浅入深、重点突出、实例丰富,既适于初/中级读者按部就班地学习,也适合于作为普通高等院校计算机及相关专业的教材。

需要本书或技术支持的读者,请与北京中关村 083 信箱(邮编 100080)发行部联系,电话:010-82702660 010-82702658 010-62978181 转 103 或者 238,传真 010-82702698, E-mail: tbd@bhp.com.cn。

图书在版编目(CIP)数据

Visual C++ 6.0 程序设计 / 陈浩杰主编. —北京:科学出版社, 2005. 7

ISBN 7-03-015576-9

I. V... II. 陈... III. C 语言—程序设计—教材
IV. TP312

中国版本图书馆 CIP 数据核字 (2005) 第 050826 号

责任编辑: 张海玲 / 责任校对: 孙 红
责任印刷: 双 青 / 封面设计: 梁运丽

科学出版社 出版

北京东黄城根北街16号

邮政编码: 100717

<http://www.sciencep.com>

双青印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2005 年 7 月第 一 版 开本: 787×1092 1/16
2005 年 7 月第一次印刷 印张: 14 1/4
印数: 1—3 000 字数: 326 000

定价: 23.00 元

新编 21 世纪高等院校计算机系列规划教材编委会

- 主 任：** 陈火旺 全国工科院校计算机专业教学指导委员会主任
中国工程院院士
- 副主任：** 李国杰 中国计算机学会理事长
计算技术研究所所长
- 杨芙清 中国计算机学会副理事长
中国科学院院士
- 沈复兴 全国高等师范学校计算机教育研究会副理事长
北京师范大学信息科学学院院长
- 何炎祥 武汉大学计算机学院院长
- 桂卫华 中南大学信息科学与工程学院院长
- 李仁发 湖南大学计算机与通信学院院长
- 陆卫民 中国科学出版集团北京希望电子出版社社长

委 员： （按姓氏笔画为序）

王江晴	王行恒	甘 玲	邓志华	孙中胜	刘晓燕	匡 松
任达森	李华贵	李超锋	李节阳	李新国	李龙澍	李建平
何婷婷	何登旭	张友生	张洪瀚	罗 琳	杨 波	杨宪泽
武兆辉	陈浩杰	陈 庄	郑明红	赵振华	洪汝渝	徐建军
徐 谡	唐光海	唐霁虹	唐 雁	高 丽	阎怀志	曹永存
覃 俊	董玉萍	董晓华	谢秉元	詹国华	戴上平	

秘 书： 徐建军

前 言

Visual C++是Windows程序设计的利器，它深入Windows程序设计的核心，与Windows系列操作系统有天然的兼容性，可以说掌握了Visual C++就掌握了Windows操作系统的核心。

不过，由于Visual C++体系庞大而且具有一定的深度，要顺利掌握Visual C++进行程序设计具有一定的难度。

凭借多年的教学经验，作者编写了《Visual C++ 6.0程序设计》一书，本书的主要特点是：说理透彻，程序简洁，以程说理。

针对Visual C++入门难的特点，本书在叙述上由浅入深，讲解透彻。首先从简单应用程序框架讲起，在简单应用程序框架的基础上分析了MFC类库中几个比较重要的类；为了能顺利理解文档/视图结构，在简单应用程序框架的基础上，分析了菜单资源并对对话框资源做了简单介绍；在学习过文档/视图结构后又深入分析了对话框和控件的使用。这样安排本书的内容，是根据作者多年来的教学心得和学习者对Visual C++的认知程度来安排的。内容的合理安排是本书的精髓。

本书的程序相当简洁，摒弃了大多数Visual C++书籍程序庞大的缺点；另外，程序与原理密不可分，尤其是对比较难的原理能做到通过例程来说明。

本书包括下面章节的内容：

- (1) 第1章介绍了Visual C++的开发环境和工程。
- (2) 第2、3章介绍了C++的语法。
- (3) 第4章通过C语言设计Windows程序，介绍了Windows程序设计的几个重要概念。
- (4) 第5、6、7、8章介绍了简单应用程序框架及几个比较重要的MFC类和菜单资源，并简单介绍了对话框资源，另外在第8章还分析了几个Windows公用对话框，MFC的文件类和文件公用对话框等内容。
- (5) 第9、10、11章首先介绍了文档/视图结构，并在其中介绍了AppWizard 和ClassWizard这两个辅助工具的使用；然后深入介绍了对话框及控件的使用。
- (6) 第12、13、14章可作为提高部分。

本书配套有便于教学用的电子教案，并配有四套模拟试题，以及习题参考答案和书中实例程序源代码。由于篇幅有限，书中没有列出这些内容，以上所有内容可到<http://www.bhp.com.cn>网站下载，或向TextBooks@126.com发邮件索要以上内容。

本书除署名作者外，宋贞海老师协助编写了部分章节，衣进韬老师协助翻译了部分外文资料，在此一并表示衷心的感谢！

由于本书编写时间仓促，书中存在错误在所难免，恳请读者批评指正。

编 者

目 录

绪言	1	7.3 使用快捷键	76
第 1 章 Visual C++6.0 概述	5	7.4 字符串表和状态栏	78
1.1 Visual C++6.0 的工作环境	5	7.5 工具栏的使用	80
1.2 工程的建立与编译、连接	6	7.6 思考与练习	86
1.3 思考与练习	10	第 8 章 对话框初步和公用对话框的使用	87
第 2 章 C++对 C 的补充	11	8.1 对话框初步	88
2.1 C++的输出和输入	11	8.2 色彩对话框	94
2.2 函数重载、默认函数参数和引用	12	8.3 字体对话框	96
2.3 思考与练习	16	8.4 文件对话框与 CFile 类	101
第 3 章 C++的类	17	8.5 思考与练习	110
3.1 类的定义	17	第 9 章 文档/视图结构的应用程序框架	111
3.2 类的继承性	21	9.1 分工合作的文档/视图结构	113
3.3 虚拟函数与多态性	24	9.2 单文档和多文档的文档/视图结构	115
3.4 思考与练习	26	9.3 单文档应用程序框架的建立与分析	117
第 4 章 Windows API 程序设计	27	9.4 文档/视图结构应用程序执行流程	126
4.1 Windows 程序设计与 DOS 程序 设计的区别	27	9.5 多文档应用程序框架简介	138
4.2 API 应用程序举例	30	9.6 思考与练习	141
4.3 思考与练习	35	第 10 章 对话框、数据交换与验证	142
第 5 章 简单应用程序框架	36	10.1 对话框模板与对话框类的连接	142
5.1 MFC 的基本组成	36	10.2 控件与控件类的连接	146
5.2 简单的 MFC 应用程序分析	38	10.3 数据交换 (DDX) 与验证 (DDV)	152
5.3 消息映射与消息处理	40	10.4 微调控件的使用	160
5.4 默认的消息映射和消息处理函数	42	10.5 用滑动控件代替编辑控件	162
5.5 应用程序举例	45	10.6 用滚动控件代替编辑控件	165
5.6 思考与练习	49	10.7 思考与练习	169
第 6 章 CDC 类与 CGdiObject 类	50	第 11 章 常用控件的使用	170
6.1 CDC 类	50	11.1 单选按钮控件	170
6.2 CGdiObject 类	54	11.2 复选框	175
6.3 CFont 类与 LOGFONT 结构	57	11.3 列表框	177
6.4 定时器的使用	59	11.4 组合框	182
6.5 思考与练习	61	11.5 思考与练习	185
第 7 章 菜单和相关资源的使用	62	第 12 章 自定义消息与非模式对话框的使用	186
7.1 菜单消息与消息映射	62	12.1 自定义消息	186
7.2 用图标美化程序	74	12.2 非模式对话框简介	187
		12.3 非模式对话框使用举例	189

12.4 思考与练习	194	14.1 Windows 多任务的概念	201
第 13 章 动态链接库的使用	195	14.2 多线程程序设计的基本概念	202
13.1 静态链接和动态链接	195	14.3 多线程程序设计举例	203
13.2 动态链接库设计	196	14.4 思考与练习	209
13.3 在应用程序中使用动态链接库	199	附录 习题参考答案	210
13.4 思考与练习	200	参考文献	221
第 14 章 多线程程序设计	201		

绪 言

面向对象程序设计导论

C语言是世界上应用最广泛的几种计算机语言之一,《C语言程序设计》作为一门程序设计语言为广大的计算机专业和非专业人员所喜欢,传统的《C语言程序设计》的主要内容是用C语言进行DOS程序设计,比较流行的开发环境是DOS操作系统下的Turbo C。

或许读者已经听说过面向对象程序设计(Object-Oriented Programming)或OOP,面向对象程序设计的确是一种全新的程序设计思想,它的出现主要是为了应对日益严重的软件危机。

60年代为了应对软件危机,诞生了结构化的编程思想,C语言就是一种结构化的编程语言,结构化的语言能比较容易地写出中等复杂的程序,然而,随着计算机技术的发展,现在的应用软件变得越来越庞大,许多软件采用结构化的程序设计方法已经无能为力,结构化程序设计方法存在的最主要的问题是:代码的重用性差。代码重用是提高生产率的关键,采用传统的面向过程的程序设计,每次进行软件开发,除了一些标准的库函数可以调用外,程序员几乎总是从零做起。

面向对象的程序设计方法是以人们通常描述现实世界的方法来描述软件问题的,现实世界是由各种各样的对象构成的,比如计算机、汽车等,每个对象都有自己独特的属性和功能(方法),比如汽车的属性有汽车的颜色、型号等,其功能有前进、后退等。在软件世界里把对象的属性抽象为数据,把对象的功能抽象为函数,这样根据现实世界的对象就可以构造出软件世界的对象,如图1所示。

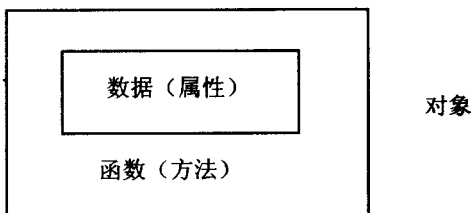


图1 对象的概念

面向对象程序设计最重要的工作是构造各种现实世界对象的软件模型。这样的模型一次构造,可以多次使用。因此,面向对象的程序设计思想提高了程序代码的重用性,比如计算机的CPU是一个对象,则CPU做好后就可以在各种计算机上使用。在面向对象程序设计中,程序员所做的工作就是像搭积木一样把各种对象组合起来,构成一个功能完备的程序。

在这里要引入类的概念,类是一类对象的抽象,或者反过来说类的实例化就是对象。比如前面讲到的汽车对象,一辆红色的汽车,一辆蓝色的汽车,是否需要分别构造这两辆

汽车对象呢？这样做显然是不科学的，可以先构造一个汽车类，这个汽车类都具备颜色这一属性，在实例化时指明具体对象的颜色即可，如图2所示。

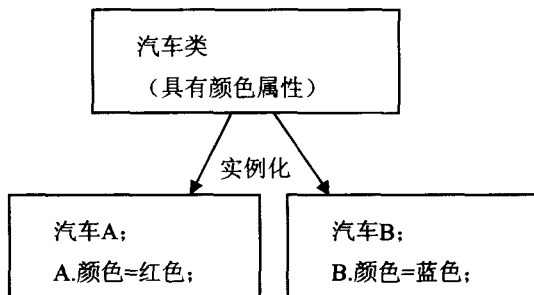


图2 类与对象的关系

可见类与对象的关系就如同C语言中数据类型与具体的变量之间的关系。

在程序设计中可以自己构造对象，也可以直接使用别人已构造好的对象，还可以在别人所构造的对象的基础上进行修改、补充产生新的对象（类的继承性），Visual C++ 6.0程序设计就是在MFC（微软基础类库）的基础上派生自己的类，正是因为类具有继承性，程序设计中代码的重用性得到大大的提高，本书在第3章将讲述有关类与对象的基本概念。

为了适应新的面向对象编程思想，C语言也在改进，发展到C++语言，C++是由Bjarne Stroustrup在新泽西州的贝尔实验室发明的，起初把这种新语言称之为“带类的C”，1983年正式更名为C++。有人把C++语言称之为C的超集，因为C++语言包括了C语言的全部内容，并增加了“类”以适应面向对象的程序设计。但这并不意味着C++语言只能进行面向对象程序设计，它也可以进行面向过程的程序设计。

Windows程序设计导论

随着Windows操作系统的出现，现在的应用软件大都是基于Windows操作系统的，因此开发Windows操作系统下的应用软件成为必须。

基于Dos操作系统的程序设计，其特点是顺序过程驱动的程序设计方法，一个程序会有一个明显的开始，明显的执行过程和明显的结束。

Windows程序设计是基于事件（消息）驱动的，事件的含义很广泛，比如最常见的有鼠标事件。当单击鼠标时，产生鼠标单击事件；当右击时，产生鼠标右击事件。Windows应用程序运行时都会打开一个窗口，并随时检测有无消息产生，程序员所要做的工作就是对产生的各种消息进行处理，如图3所示。

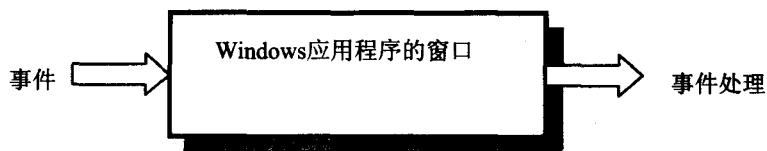


图3 Windows 应用程序的执行思路

由于消息的产生是不可预测的,因此Windows程序设计是一种非顺序的消息驱动的程序设计方法。

可以用C语言直接进行Windows程序设计,被称为Windows API程序设计,所谓API就是Application Interface(应用程序接口),也就是大量的Windows函数。用C语言进行Windows程序设计就是对API函数的直接调用,这种程序设计方法思路清晰、原理简洁,但由于API函数数量众多且杂乱无章,因此掌握起来比较困难。而且更重要的一点是,没有采用面向对象程序设计方法,代码没有重用性。本书第4章将简单介绍API程序设计的一些基本知识。

因此有必要采用C++进行Windows程序设计,称之为Visual C++ 6.0程序设计。Visual C++ 6.0的类库MFC把API函数进行了合理的分类封装,程序员可以从MFC类库派生自己的类,使Windows应用程序开发过程大大简化。

综合面向对象的开发思想和Windows程序设计的消息驱动模型,把图1和图3综合,形成如图4所示的Visual C++ 6.0面向对象的Windows程序设计。从图4可以看到,对象之间通过消息进行通信联系,而且对象内增加了消息处理函数。

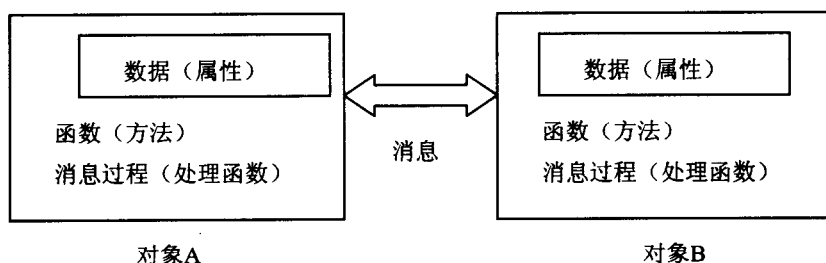


图4 Visual C++ 6.0面向对象的Windows程序设计

因此,C语言家族可以进行四种类型的应用程序开发。

- 1) 用C语言开发基于Dos的应用程序;
- 2) 用C++开发基于Dos的应用程序;
- 3) 用C语言开发基于Windows应用程序;
- 4) 用C++开发基于Windows的应用程序。

本书的重点内容是介绍用C++开发基于Windows的应用程序,即Visual C++ 6.0面向对象的Windows程序设计,但Visual C++ 6.0的开发环境同样可以开发其他类型的应用程序。

Visual C++ 6.0的发展历史与技术特点

Visual C++ 6.0这套功能强大的Windows应用程序开发系统是从Microsoft C/C++演化而来的,从Microsoft C/C++8.0开始改称为Visual C++,一直发展到今天的Visual C++ 6.0。

Visual C++ 6.0有一套集成开发工具,包括各种编辑器、编译工具、集成调试器等,其主要技术特点有两个。

一是Visual C++ 6.0支持面向对象编程技术,这一技术包装了Windows内在的复杂的运

行机制，使Windows编程变得简单易学。

提到面向对象编程技术，就不得不提MFC，MFC是由微软的AFX小组开发的，MFC提供了一整套用于Windows应用程序开发的类。1992年3月，MFC的第一个版本随同Microsoft C/C++7.0诞生了，可见MFC比Visual C++还要早，也许正是MFC的诞生促使Microsoft C/C++演变成Visual C++。

二是Visual C++ 6.0支持可视化编程，这正是Visual名字的由来。Visual C++ 6.0提供了AppWizard和ClassWizard这两个功能强大的可视化编程向导，AppWizard负责建立应用程序的框架，ClassWizard则在框架的基础上负责给应用程序添加代码。

同Microsoft Visual Studio家族中的Visual Basic相比，Visual C++功能更加强大，开发人员对程序的控制更强，生成的代码效率更高，程序运行的速度更快，但要求开发人员所做的工作也就更多。从使用语言的角度来说，Visual C++比Visual Basic低级，正如同C语言与Basic语言的关系，因此Visual C++一般来说适合于专业的软件开发人员使用。

第1章 Visual C++6.0概述

本章要点:

本章主要讲述Visual C++6.0的工作环境，工程的概念，并以Win32 Console Application工程为例讲述了工程的建立、源文件的添加、工程的编译、连接与执行。

本章的主要内容有:

- Visual C++6.0的工作环境。
- 工程的概念。
- 工程的建立、源文件的添加、工程的编译、连接与执行。

1.1 Visual C++6.0的工作环境

本书选择中文版Visual C++6.0作为工作环境，打开Visual C++，其工作环境如图1-1所示。

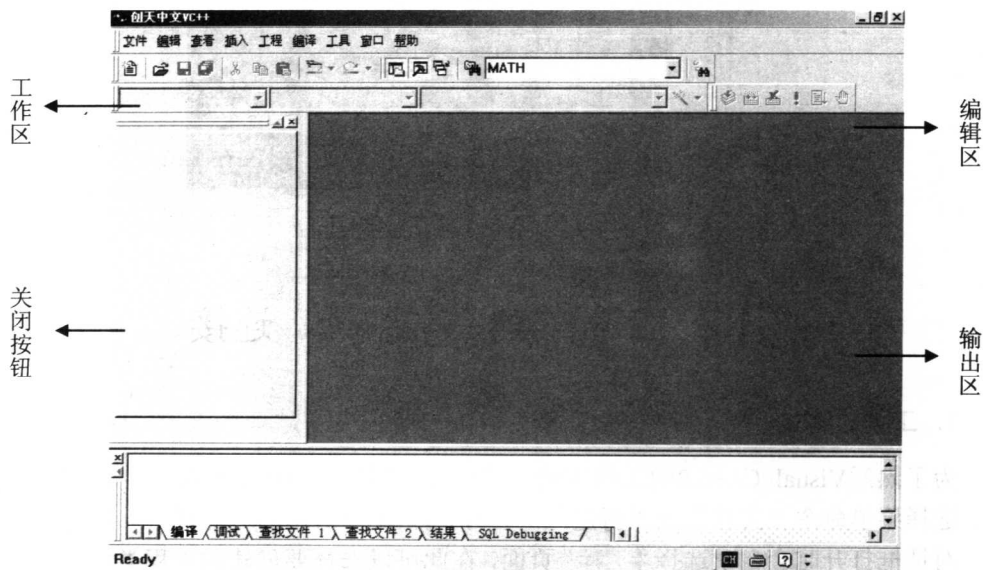


图 1-1 Visual C++6.0 的工作环境

Visual C++6.0的工作环境可以划分为三块区域，最左边的区域是工作区，最下面的区域是输出区，最右面的区域是编辑区。

编辑区用来对源文件进行编辑，现在的编辑区是灰色的，表示还没有源文件在进行编辑。

输出区（Output）的作用是：在对工程进行编译和连接后，如果程序有错误或警告，

则显示在输出区，可以对照错误或警告提示进行程序修改。

这里要讲一下工程（Project）的概念：Visual C++6.0把要完成的某个程序设计任务称之为工程，要进行程序设计先要建立一个工程，当然刚建立的工程还只是个空架子，要完成具体的程序设计任务还必须往工程里添加源文件（如：.cpp文件、.c文件、.h文件等）。工程管理添加在工程下的各种源文件，这些源文件有机结合，通过编译和连接生成可执行文件。

因此工程就相当于盖房子的总体图纸，要把房子盖起来还必须购买具体的原材料（钢筋、水泥等）。

工作区（Workspace）的作用就是用来管理工程及工程下的各种源文件，在它的管理下，可以有条不紊地进行各种源文件的编辑。

可以单击工作区或输出区的关闭按钮关闭这两个区域，图1-1示意了输出区的关闭按钮。这两个区域关闭后能增加编辑区的面积，当对某个源程序进行编辑时，可以关闭这两个区域，需要的时候再打开。

打开方法：选择菜单命令“查看”→“工作区”或“查看”→“输出”，如图1-2所示，分别打开这两个区域。

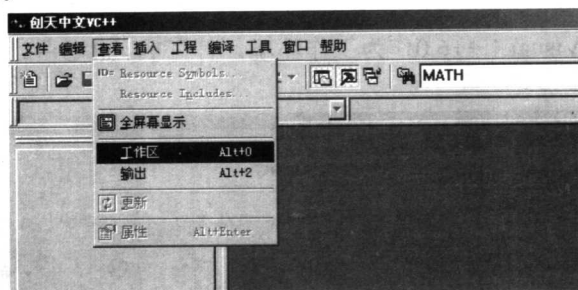


图 1-2 工作区和输出区的打开

1.2 工程的建立与编译、连接

1. 工程的建立

为了熟悉Visual C++6.0的工作环境，首先选择用C语言编写DOS程序来掌握工程的建立。选择菜单命令“文件”→“新建”，会打开如图1-3所示的对话框。

对话框打开时会自动选择“工程”页面，在此可以选择要创建的工程类型。要编写DOS程序，应选择“Win32 Console Application”（Win32 控制台应用）工程。

在“位置”栏里选择新建的工程所存放的路径，为了便于对工程进行管理，建议新建一个文件夹集中管理工程，在“工程”栏里输入工程的名称，单击“确定”按钮，完成工程的建立。

工程建立完成后，在工作区会出现工程的名字，如图1-4所示。可以看到工程名下连接了3个文件夹，分别为源文件、头文件和资源文件文件夹，每个文件夹下面可以添加多个源文件。

需要说明的是，在工作区下面出现了两个页面，一个是“Class View”页面（对类进行管理），另一个是“File View”页面（管理工程下的各种文件），图1-4为“File View”页面中的内容。

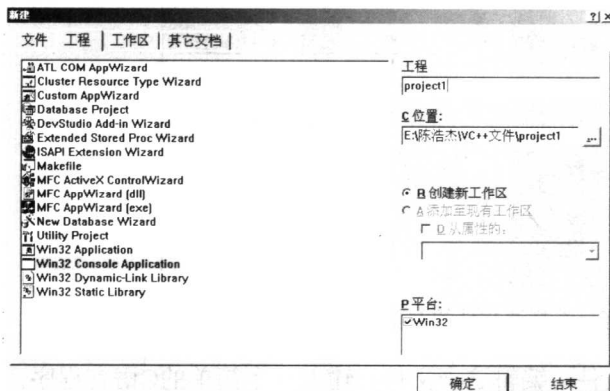


图 1-3 工程类型的选择和建立

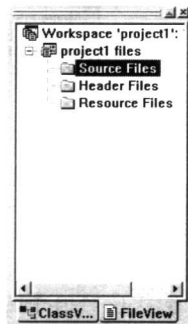


图 1-4 工作区出现工程名

2. 源文件的加入与工程的编译、连接

新建的工程还只是个空架子，三个文件夹下面还没有任何源文件，选择菜单命令“工程”→“添加工程”→“新建”，给工程添加源文件，如图1-5所示。

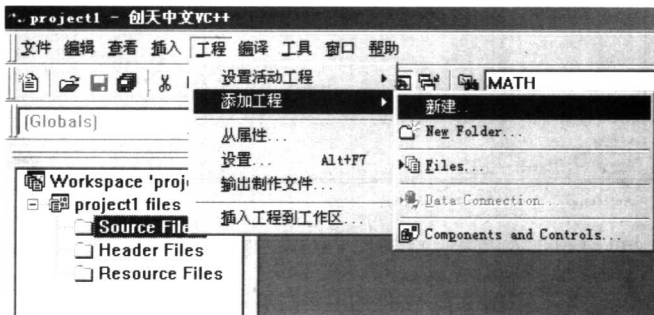


图 1-5 给工程添加源文件

这时会打开如图1-6所示的对话框，对话框打开时会自动选择“文件”页面。在“文件”页面里可以选择源文件的类型，现在选择“C++ Source File”，表示要往工程里添加一个C++源文件，在“文件”栏里输入要新建的源文件名，单击“确定”按钮完成源文件的加入。

另外，在图1-6的“添加工程”栏里可以看到刚建立的工程名，这表示新建的源文件将自动添加到刚建立的工程中。系统会按照源文件的类型，把建立的文件添加到工程中适当的文件夹里，现在选择的是“C++ Source File”，则新建的源文件以.cpp为后缀，将自动添加到工程中的源文件夹里。如果选择的源文件类型为“C/C++ Header File”，则将建立一个以.h为后缀的头文件，头文件将自动添加到工程中的头文件夹里。

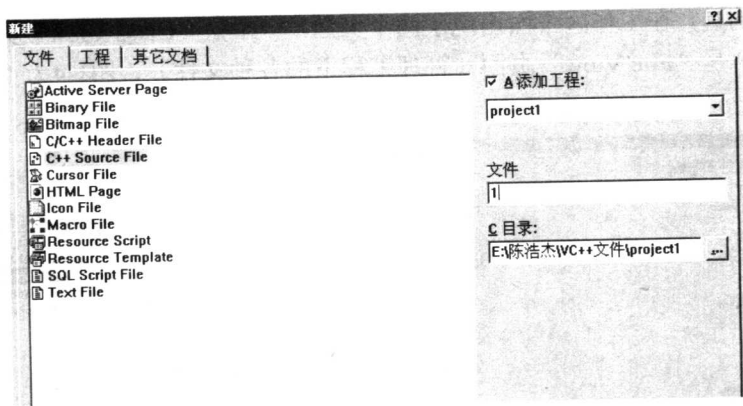


图 1-6 选择源文件的类型

现在编辑区会出现新建的.cpp源文件，编辑这一文件，输入一个简单的C语言程序，如图1-7所示。

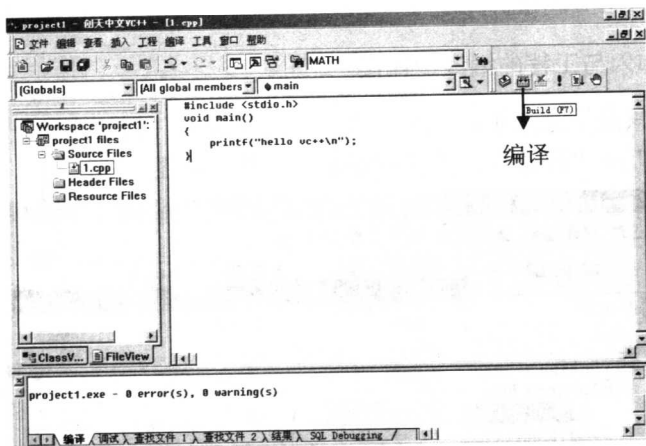


图 1-7 文件的编辑与编译、连接

编辑完成后，先选择菜单命令“编译”→“编译”对工程进行编译，编译完成后会生成以.obj为后缀的目标文件，然后选择菜单命令“编译”→“构件”对工程进行连接，连接完成后生成可执行文件。这两个步骤可以合为一个，即单击图1-7所示的Visual C++工具栏上的build工具，一次性完成编译和连接。

当输出区显示“0 errors, 0 warnings”时，表示没有错误和警告，反之则会按序号列出错误和警告。双击错误或警告，编辑标志会出现在源文件可能出错的位置，当然有时候提示的位置不一定很准确。

3. 程序的执行

可以使用菜单，也可以单击工具栏上的“红色感叹号”工具，Win32控制台工程执行后会出现一个DOS界面，如图1-8所示。

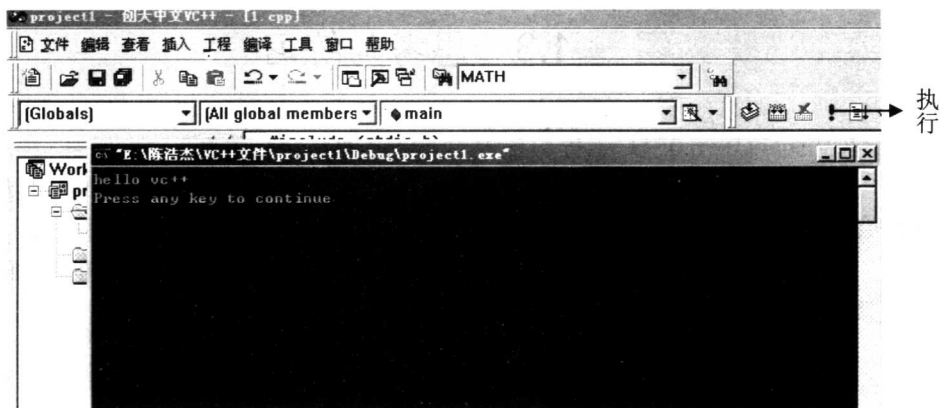


图 1-8 程序的执行

4. 工作区的关闭

本次工程完成后，可以关闭工程，进行下一次工程的建立。选择菜单命令“文件”→“关闭工作区”，即可完成工程的关闭。

可以根据工程建立时的路径，在磁盘中找到刚建立的工程，如图1-9所示，其中文件1.cpp就是刚编辑的源文件。应当注意的是project1.dsw文件，双击这一文件可以把整个工程打开，双击1.cpp文件则仅仅打开源文件。project1工程目录下还有一个“Debug”文件夹，在里面有刚生成的可执行文件。

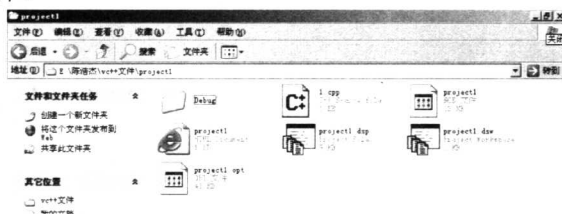


图 1-9 工程目录下的各种文件

5. 工程的打开

选择菜单命令“文件”→“打开工作区”，找到刚建立的工程，选择工程目录下的“project1.dsw”文件，单击“打开”按钮，即可打开工程，如图1-10所示。

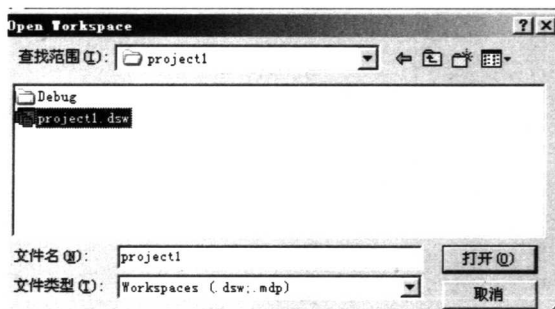


图 1-10 工程的打开

1.3 思考与练习

1. 在Visual C++6.0中，工程与各种源文件之间是什么关系？
2. 一个Visual C++6.0工程下会有3个文件夹，这3个文件夹分别是什么？以.cpp为后缀的文件会添加到哪个文件夹中？以.h为后缀的文件会添加到哪个文件夹中？
3. 以Win32 Console Application工程为例，简述工程的建立、文件的添加、工程的编译和连接及工程的执行？