

面向21世纪

高等职业技术教育计算机类系列教材

C++ 程序设计

主 编 杨俊清

西安电子科技大学出版社

<http://www.xduph.com>



清华大学出版社

C++ 程序设计

李 强 李 强 编

清华大学出版社
www.tup.tsinghua.edu.cn

面向 21 世纪高等职业技术教育计算机类系列教材

C++ 程序设计

主 编 杨俊清

副主编 陈巧莉

参 编 李培金 石峰 李艳华

西安电子科技大学出版社

2003

内 容 简 介

本书共分为 10 章。第 1 章介绍了 C++语言和 C 语言的关系以及 C++程序的实现和运行环境。第 2~9 章围绕类、对象、类的封装和继承等面向对象程序设计方法中涉及的概念和术语,系统地介绍了 C++语言面向对象的语言成分。第 10 章围绕 Visual C++编程实例,介绍 MFC 编程流程,对 Windows 环境下开发应用程序常见的几个方面的问题分别进行了介绍。

本书语言流畅,附有较多的实例,可以作为高职、高专类院校计算机专业 C++程序课程的教学用书,也可供学过 C 语言并想进一步自修 C++的读者使用。

★本书配有电子教案,需要者可来函向出版社免费索取。

图书在版编目(CIP)数据

C++程序设计 / 杨俊清主编.

—西安:西安电子科技大学出版社,2003.7

(面向 21 世纪高等职业技术教育计算机类系列教材)

ISBN 7-5606-1246-6

I. C… II. 杨… III. C 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(2003)第 037160 号

责任编辑 云立实 王素娟

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)8242885 8201467 邮 编 710071

<http://www.xduph.com>

E-mail: xdupfxb@pub.xaonline.com

经 销 新华书店

印刷单位 陕西光大印务有限责任公司

版 次 2003 年 7 月第 1 版 2003 年 7 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印张 14.875

字 数 345 千字

印 数 1~4 000 册

定 价 16.00 元

ISBN 7-5606-1246-6 / TP·0655(课)

XDUP 1517001-1

*** 如有印装问题可调换 ***

序

进入 21 世纪后,世界已经步入知识经济发展的时期,随着我国社会主义市场经济的快速发展,各行各业越来越需要具有综合职业能力和素质全面的,直接工作在生产、技术、管理和服务第一线的应用型、技能型的高级实用人才。高等职业技术教育的任务就是面向不同岗位,培养具备一定知识和技能,具有一定职业岗位能力和跨职业、跨岗位关键能力,德、智、体全面发展的高级技术和技艺型人才。据权威机构的规划,2005 年,我国高等院校在校生规模将达 1600 万人,其中 50% 是高等职业教育的学生。这说明高等职业技术教育即将和高等教育的本科教育相提并论,在我国高等教育体系中占有相当重要的地位。

高职教育作为我国高等教育的一个重要组成部分,其培养目标是具有必要理论知识和较强实践能力的高等技术应用型专门人才。它的人才培养模式应该是以培养适应生产、建设、管理、服务第一线需要的高等技术应用型专门人才为根本任务;以适应社会需要为目标;以培养技术应用能力为主线;以突出职业性、实践性、适应性和地方性为特点。计算机教学应以传授应用知识为主,强调操作使用,注重培养学生利用计算机开展专业技术分析、解决各种技术问题的意识,培养学生的自学能力和创造性学习的能力。

在我国高等职业技术教育发展的过程中,虽然部分学校已经取得了一些成功经验,并逐渐形成了自己的办学特色,但从总体上来看,高等职业技术教育尚处于起步阶段。高职教材建设明显跟不上高职发展的需要,主要表现在借用本科教材和沿用专科教材的问题上。这类教材多数在编写上以本科教材为蓝本,是“本科压缩型”,尤其在以“应用”为主旨和特征构建课程与教学内容体系上,存在着明显不足,难以符合高等职业技术教育培养目标的要求,对高职人才培养十分不利。因此,做好高职教材改革与建设工作刻不容缓。

为了促进高等职业技术教育教材建设,西安电子科技大学出版社组织陕西省高职院校的骨干教师共同策划编写了高职教育非计算机专业和计算机专业系列教材,现已出版。本系列教材以适应社会需要为目标,以培养技术应用能力为主线来设计学生的知识、能力、素质结构和培养方案。编写上本着重能力、严实践、求创新的总体思路;体现科学性、思维性、启发性、先进性和教学的适用性;以培养能力为主,基础理论适度,适当反映科学技术领域内的新成果来优化课程内容。本套教材突出了高职教材的特色,适合高等职业学校、高等专科学校、成人高校等高等职业技术教育和五年制高等职业技术教育以及部分中等职业技术教育的需要。

系列教材编委会
2002 年 8 月

高等职业技术教育计算机类系列教材

编委会名单

主任委员：翟 轰 （陕西工业职业技术学院院长）

副主任委员：冉崇善 （陕西省职业技术教育学会计算机委员会主任）

张晓云 （西安航空技术高等专科学校计算机工程系主任）

王 津 （陕西工业职业技术学院信息工程系主任）

李荣才 （西安电子科技大学出版社总编辑）

委员：（按姓氏笔划排列）

丁春莉 （陕西交通职业技术学院）

王存祥 （陕西财经职业技术学院）

白景让 （杨凌职业技术学院）

刘敏涵 （陕西国防工业职业技术学院）

刘晓云 （西安铁路职业技术学院）

赵生智 （陕西能源职业技术学院）

秦兴文 （西安航空职业技术学院）

雷育春 （陕西省邮电学校）

项目负责人 马乐惠

策 划 云立实 马武装 马晓娟

电子教案 马武装

前 言

面向对象技术是近几年来在计算机应用领域迅速发展起来的软件开发技术,与传统的结构化程序设计思想相比有许多优点。面向对象的程序设计方法易于编程、修改和维护,代码的可重用性好,可以大幅度提高软件开发的效率,几乎在所有的计算机研究和应用领域都能看到C++的影子。

C++是一门高效实用的程序设计语言,它既可进行过程化程序设计,也可进行面向对象程序设计。C++语言强调对高级抽象的支持,实现了类的封装、数据隐藏、继承及多态,使得其代码容易维护且可高度重用。

C++语言是C语言的超集,是在C语言的基础上扩充了面向对象的语言成分后形成的。在C语言课程的基础上再通过C++语言的学习可以掌握面向对象的基本思想和程序设计方法。

本书共分为10章。第1章介绍了C++语言和C语言的关系以及C++程序的实现和运行环境。第2~9章围绕类、对象、类的封装和继承等面向对象程序设计方法中涉及的概念和术语,系统地介绍了C++语言面向对象的语言成分。第10章围绕Visual C++编程实例,介绍MFC编程流程,对Windows环境下开发应用程序常见的几个方面的问题分别进行了介绍。

本书由西安航空技术高等专科学校的杨俊清老师担任主编,陕西国防职业技术学院的陈巧莉老师担任副主编。西安航空职业技术学院的李艳华老师编写第1章和第10章,陈巧莉编写第2~4章,杨俊清编写第5~7章;西安航空技术高等专科学校的石峰老师编写第8章;陕西工业职业技术学院的李培金老师编写第9章。本书在编写过程中得到了西安电子科技大学出版社的大力支持,在此谨表谢意。

由于编著者水平有限,书中难免存在一些缺点和错误,殷切希望广大读者批评指正。

编著者

2003年3月

目 录

第 1 章 最简单的 C++ 程序

1.1 C 和 C++	1	1.3 C++ 程序的实现与运行环境	4
1.1.1 C++ 的发展史	1	1.3.1 Visual C++ 6.0 常用功能介绍	5
1.1.2 面向对象的程序设计	1	1.3.2 VC++ 6.0 控制台应用程序	6
1.2 一个简单的 C++ 程序	2	1.3.3 VC++ 6.0 MFC 应用程序	8
1.2.1 一个简单的 C++ 程序	2	习题	11
1.2.2 C++ 程序结构特点	3		

第 2 章 类 与 对 象

2.1 类与对象的概念	13	2.3.3 重载成员函数	23
2.2 类的定义	15	2.3.4 调用成员函数	24
2.2.1 类的概念	15	2.4 保护成员	28
2.2.2 类的定义格式	16	2.5 编程实例	30
2.3 函数成员的实现	20	小结	34
2.3.1 函数成员的定义	20	习题	34
2.3.2 内联函数和外联函数	21		

第 3 章 构造 函 数

3.1 对象的定义	38	3.3.3 拷贝初始化构造函数	53
3.1.1 const 类型声明	38	3.4 带参数的构造函数	56
3.1.2 动态地分配和撤销内存空间	40	3.5 构造对象的顺序	57
3.1.3 对象的定义	42	3.6 构造函数重载	60
3.2 构造函数的使用	47	3.7 编程实例	63
3.3 析构函数	50	小结	65
3.3.1 析构函数和构造函数的比较	50	习题	65
3.3.2 缺省构造函数和缺省析构函数	52		

第 4 章 引 用

4.1 引用的概念	68	4.3.2 返回多个值	74
4.2 引用的操作	69	4.4 编程实例	75
4.3 引用作为函数的参数	71	小结	77
4.3.1 用引用传递函数参数	72	习题	77

第5章 静态成员与友员

5.1 静态成员的需要性	80	5.5.2 友元函数	99
5.2 静态数据成员	81	5.5.3 友元类	103
5.3 静态函数成员	85	5.5.4 友元成员	105
5.4 静态成员的使用	90	5.6 编程实例	107
5.5 友元的使用	95	小结	111
5.5.1 需要友元的原因	95	习题	112

第6章 继 承

6.1 继承的概念	114	6.4 虚函数	130
6.1.1 继承的概念	114	6.4.1 虚函数的使用	130
6.1.2 派生类的定义格式	115	6.4.2 不恰当的虚函数	133
6.1.3 继承方式	115	6.4.3 虚函数的限制	135
6.1.4 公有继承下的赋值兼容规则	123	6.5 抽象类	136
6.2 派生类的构造和析构	123	6.5.1 纯虚函数	136
6.2.1 派生类构造函数	123	6.5.2 抽象类的概念和定义	136
6.2.2 派生类析构函数	126	6.6 编程实例	139
6.2.3 派生类构造函数和析构函数的使用 中应注意的问题	127	小结	144
6.3 多态性	128	习题	145

第7章 重 载

7.1 重载的概念	148	7.2.5 重载赋值运算符	156
7.2 运算符重载	149	7.2.6 函数调用运算符()的重载	160
7.2.1 运算符重载的概念	149	7.2.7 编程实例	162
7.2.2 类成员运算符重载	149	小结	163
7.2.3 友元运算符重载	151	习题	163
7.2.4 重载增量运算符	153		

第8章 I/O 流

8.1 C++流的概念	166	8.3.1 标准库文件函数输入/输出	170
8.2 C++的输入/输出	167	8.3.2 文件输入/输出流	173
8.2.1 标准输入/输出	168	8.4 重载插入运算符和提取运算符	177
8.2.2 标准错误输出 cerr	168	8.5 编程实例	178
8.2.3 输入/输出格式控制	168	小结	180
8.3 文件流与文件操作	170	习题	180

第9章 模 板

9.1 模板的概念	183	9.3 类模板	186
9.1.1 模板的含义	183	9.4 编程实例	190
9.1.2 模板的基本语法	183	小结	195
9.2 函数模板	184	习题	196

第10章 Visual C++编程实例

10.1 MFC 编程流程	198	10.3 对话框的应用	209
10.2 常用 MFC 类和消息处理	202	10.4 菜单的应用	216
10.2.1 常用 MFC 类	202	10.5 工具栏应用	224
10.2.2 绘制图形	206	小结	225
10.2.3 消息处理	208	习题	226
参考文献	227		

最简单的 C++ 程序

1.1 C 和 C++

1.1.1 C++的发展史

C 语言是 20 世纪 80 年代以来被迅速推广使用的一种程序设计语言。它既具有高级语言的特点，又具有低级语言的一些特点，能够很方便地实现汇编级的操作，目标程序效率较高。然而，C 语言也存在一些缺陷，例如类型检查机制相对较弱，缺少支持代码重用的语言结构等，使得用 C 语言开发大程序比较困难。

为了克服 C 语言存在的缺点，并保持 C 语言简洁、高效的特点，贝尔实验室的 Bjarne Stroustrup 博士及其同事于 1983 年开始对 C 语言进行改进和扩充，将“类”的概念引入了 C 语言，构成了最早的 C++ 语言。发明 C++ 语言的重要目标就是在保留 C 语言原有精华的基础上提供全面的面向对象的编程支持，使得程序的结构更加清晰，更容易维护和扩充，同时又不丧失其高效性。后来 Stroustrup 博士及其同事们又为 C++ 引进了运算符重载、引用、虚函数等许多特性，并使之更加精练。

C++ 支持面向对象的程序设计方法，很适合于大、中型软件的开发，从开发时间、费用到软件的可重用性、可扩充性、可维护性和可靠性等方面，都具有很大的优越性。

1.1.2 面向对象的程序设计

1. 面向对象的程序设计的基本概念

C 语言、PASCAL、FORTRAN 等都是一种面向过程的程序设计。面向过程的程序设计的基本思想是将问题分成独立的功能模块，用过程或函数来实现，达到最终解决问题的目的。它采用函数来描述对数据结构的操作，但又将函数与其所操作的数据分离。作为对现实世界的抽象，函数和它所操作的数据是密切联系、相互依赖的。如果数据结构发生改变，则必须改写相应的函数。某个函数的改动经常引起许多其它函数的改动，因此维护较为困难，特别是难以适应大型复杂软件系统开发的需要。

面向对象程序设计从所处理的数据入手，并以数据为中心。它把现实世界的问题抽象为“类”的概念。类是对现实世界中的客观事物的抽象，是对具有共同属性和行为的一类事物的抽象描述，其中，共同属性被描述为类中的数据成员，共同行为被描述为类中的成员函数。例如：汽车是对各种汽车(如小轿车、大卡车、公共汽车等)的一个抽象，汽车的

颜色、型号、发动机等是属性，而开动和转弯等是它的成员函数即它的方法。

面向对象的程序设计方法最基本的思想就是把客观世界看成一个一个相对独立而又相互联系的实体，这些实体称为对象。例如，一个桌子、一个气球都是一个对象。类是对象集合的抽象，规定了这些对象的公共属性和方法，对象是类的一个实例。例如，钢笔是一个类，具体到一支钢笔就是一个对象。

2. 面向对象程序设计的基本特征

面向对象程序设计具有封装、抽象、继承和多态性 4 种基本特征。

1) 封装

封装是把函数和数据封藏起来，把它看成一个有机整体。封装是面向对象的重要特征，首先，它实现了数据隐藏，保护了对象的数据不被外界随意更改；其次，它使对象成了相对独立的功能模块。

2) 抽象

抽象就是忽略一个主题中与当前目标无关的那些方面，以便充分地注意与当前目标有关的方面。对象是具有特定属性的一个抽象。

3) 继承

自然界中的大部分事物之间都有很多共性，但也有不同。比如，四边形是一个类，而矩形与四边形有相同的性质，也有自己的属性。我们可以将事物之间的共性保留下来也就是继承，如矩形继承四边形这个类的公共属性，将不同的特性再定义。面向对象程序设计方法允许一个类继承其它类(称为基类)的属性和方法，该类称为派生类。

继承是类的层次结构之间共享数据和方法的机制，允许和鼓励类的重用。

4) 多态性

多态性是指不同类的对象对同一消息作出不同的响应。例如，同样是加法，把两个时间加在一起和把两个整数加在一起的内涵是完全不同的。再比如，十字路口的交通灯，同样是灯，当红灯亮时，人们知道该停止；当绿灯亮时，人们知道该通行。多态性包括参数多态性和包含多态性，具有灵活、抽象、行为共享和代码共享的优势。

1.2 一个简单的 C++ 程序

本节以一个简单的 C++ 程序为例，介绍 C++ 程序的结构特点。

1.2.1 一个简单的 C++ 程序

例 编程，求从键盘上输入的两个浮点数的和。

```
#include<iostream.h>

void main()
{
    double x,y;
    cout<<"Enter two double number:";
```

```

        cin>>x>>y;

        double  z=x+y;

        cout<<"x+y="<<z<<endl;

    }

```

执行该程序，屏幕上出现如下提示信息：

```
Enter two double number:3.456 71.3✓
```

运行结果为：

```
x+y=74.7
```

1.2.2 C++程序结构特点

1. C++程序的组成

C++程序由若干个文件组成，每个文件又由若干个函数组成。C++程序的若干个函数是相对独立的程序段。组成一个程序的若干个函数中，有且仅有一个是主函数，其函数名为main，它可以有参数，也可以无参数。主函数只能存在于组成该程序的若干个文件中的一个中。

程序是用计算机语言对程序要完成任务(即功能)的描述。程序必须存储在文本文件中，称为源程序文件。对于C++，源程序文件约定的扩展名是.CPP。文件名最好有一定的提示作用，能使人联想到程序的内容或功能。

利用 Visual C++(简称 VC++)进行 Windows 编程时，源程序代码主要可以分为 3 类：

- (1) 头文件(*.h)。头文件用于定义函数原型、类的声明等。
- (2) CPP 文件(*.CPP)。CPP 文件用于定义函数或类成员函数的实现。
- (3) 资源文件(*.RC)。资源文件中定义的是应用程序中所遇见的菜单、对话框、位图等资源，它是一个文本文件，可以被 VC++集成编译环境可视化显示。

另外，VC++还会产生一些工程管理文件，由于应用程序通常由许多源程序组成，因此，为了进行有效管理，VC++将它们以一种形式组织在一起，这种组织形式称为工程(Project)，其扩展名为.dsp。

2. 预处理命令#include

预处理命令是包含命令，其功能是把一个文本文件的内容包含到该命令处。1.2.1 节的程序的第 1 行以#打头，表明该行是一个预处理命令；<iostream.h>是命令的参数，它指出了要包含的那个文件的文件名是 iostream.h。

3. 头文件

以.h 作为扩展名，包含有对象和函数说明的文件称为头文件。在 1.2.1 节的程序中，用 C++流完成输入、输出的操作，其中，cout 是输出流对象，cin 是输入流对象；<<和 endl 都是特殊函数。这些对象和函数是系统提供的，但必须先说明后使用，它们的说明就在系统提供的头文件 iostream.h 中。

4. 输入和输出

程序要求用户提供信息的过程称为程序的输入。程序向用户发出一些信息的过程称为

程序的输出。C++程序的输入操作是由系统提供的标准输入流对象 `cin` 来完成的。它的格式为

```
cin>>v1>>v2>>...>>vn;
```

其中，“>>”的功能是从输入流中提取数据赋值给一个变量，称为提取操作符；`v1`，`v2`，…，`vn` 都是变量。该语句的意思是，程序暂时中止执行，等待用户从键盘上输入数据。用户输入了所有的数据后，按回车键表示输入结束，程序将用户键入的数据存入各变量中，并继续执行下面的语句。

C++的输出操作是由系统提供的标准输出流对象 `cout` 来完成的。它的格式为

```
cout<<"x="<<x<<";"<<"y="<<y<<endl;
```

其中，“<<”称为插入运算符，`x`，`y` 都是表达式，`endl` 是换行符。这个语句的意思是，将表达式的值输出到屏幕上当前光标位置处。在输出时可适当使用字符串作为提示信息。

5. 注释符

在 C++中，注解的形式有两种：一种以两个斜杠“//”起头，直到行末；一种是用斜线星号组合“/*”和“*/”括起的任意文字。后一种多用于注解篇幅多于一行的情况。

编译系统不理睬注释文字，因此注释文字可以是任意的。注释可使程序更容易理解，在编写程序时随时添加注释是一种良好的习惯。

1.3 C++程序的实现与运行环境

Visual C++是美国 Microsoft 公司开发的 Microsoft Visual Studio 的一部分，是一个使用广泛的 C/C++集成化开发环境。它成功地将面向对象和事件驱动编程概念联系起来，并得到了很好的配合，使得编写 Windows 应用程序的过程变得简单、方便且代码量小。

VC++ 6.0 集程序的代码编辑、编译、连接、调试于一体，给编程人员提供了一个完整、方便的开发界面和许多有效的辅助开发工具。VC++ 6.0 的编辑环境包含了许多独立的组件，它们是文本编辑器、资源编辑器、C/C++编译器、连接器、调试器、AppWizard、ClassWizard、源程序浏览器以及联机帮助。所有这些构件的功能都隐藏在 VC++ 6.0 的菜单和工具条中。通过该集成环境，程序员可以观察和控制整个开发进程。

VC++ 6.0 的主窗口可以分为图 1-1 所示的几个部分。

典型的 Windows 应用程序结构有控制台应用程序，基于框架窗口的应用程序，基于对话框的应用程序和基于文档 / 视图结构的应用程序 4 种。

VC++既可用于管理基于 Windows 的应用项目，也可用于管理基于 DOS 的应用项目。基于 DOS 的应用系统也称为控制台应用系统，本书前 9 章的应用程序均为控制台应用程序。控制台应用程序结构简单，可以不使用 MFC 类库。本书的第 10 章将介绍基于 Windows 的编程，基于框架窗口以及基于对话框和单文档的应用程序。

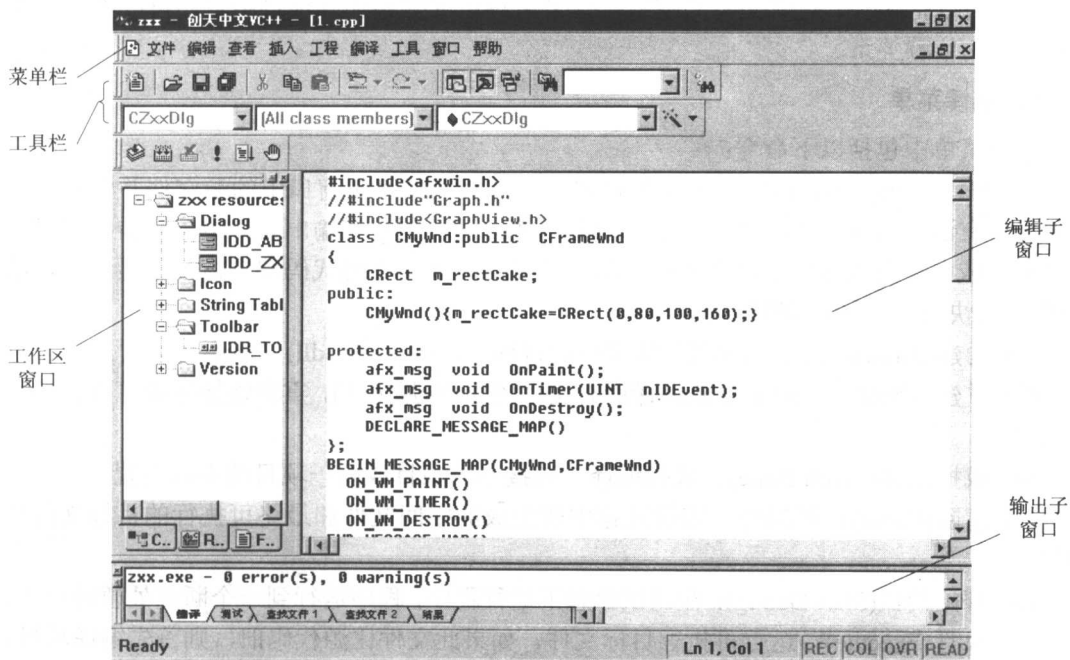


图 1-1

1.3.1 Visual C++ 6.0 常用功能介绍

1. 文件菜单

文件菜单中共有 14 个菜单项，分成 6 组，组与组之间通过凹下去的横线分割开。第一组是基本文件操作；第二组是工作区操作，工作区是 VC++ 6.0 中最大的一个处理单位，每个工作区包括多个工程，每个工程又包含若干个文件；第三组用于文件保存；第四组用于文件打印。

2. 编辑菜单

编辑菜单包含用于编辑和搜索的命令，这些命令与其它 Windows 应用程序中的同名命令具有相似的功能。

3. 查看菜单

查看菜单包含用于检查源代码和调试信息的命令项，可以用来设置全屏方式显示窗口，或者打开工作区窗口、输出窗口和各种调试窗口。

4. 插入菜单

插入菜单可以创建新的类、资源、窗体并将它们插入到文档中，可以将文件作为文本插入到文件中，也可以添加新的 ATL 对象到项目中。

5. 工程菜单

工程菜单中的命令用于管理项目和工作区。可以选择指定的项目作为工作区中的活动

项目，也可以把文件、文件夹、数据链接以及可再用部件添加到项目中，还可以编辑和修改项目间的依赖关系。

6. 编译菜单

编译菜单中包括以下命令：

(1) 编译(Compile)：此命令用于编译显示在源代码的编辑窗口中的源文件并检查源文件中是否有语法错误。在编译过程中若有警告或错误，则将在输出窗口中显示错误信息。可以向前或向后浏览输出窗口中的错误信息，然后按 F4 键在源代码编辑窗口中显示相应的代码行。(快捷键：【Ctrl+F7】。)

(2) 构建(Build)：用于创建项目的可执行目标文件(.exe 或 .dll)，但不运行它。

(3) 重建全部(Build all)：重新编译所有文件(包括资源文件)，重新连接生成可执行的目标文件。

(4) 成批编译(Batch Buile)：成批编译、连接不同项目或同一项目的不同设置。

(5) 清除(Clean)：把编译、连接过程中所生成的中间文件和最终可执行的目标文件删除掉。

(6) 开始调试(Start Debug)：在调试状态下运行程序，程序运行到一个断点处暂时停止。

(7) 执行(Execute)：运行可执行目标文件，如果此文件比源代码旧，则首先编译项目，再运行新产生的可执行目标文件。(快捷键：【Ctrl+F5】。)

1.3.2 VC++ 6.0 控制台应用程序

1. 应用项目的建立

一个应用项目(Project)由若干个编译单元组成，而每个编译单元由一个程序文件(扩展名是.CPP)及与之相关的头文件(扩展名是.H)组成。在组成项目的所有单元中，必须有一个(也只能有一个)单元包含函数 main()的定义，这个单元称为主单元，相应的程序文件称为主程序文件。一个简单的控制台应用系统可以只有一个单元，即主单元。通过编译，每个单元生成一个浮动程序文件(也称为目标程序文件，扩展名是.OBJ)。通过链接这些浮动程序文件可使整个系统生成一个惟一的可执行文件，扩展名是.EXE。

由若干个关系密切的项目构成一个工作区，工作区在建立时自动生成扩展名为.DSW的工作区文件以及其它文件。

建立一个控制台应用项目(名为 li)的过程分为三步。

1) 建立项目及工作区

(1) 启动 Visual C++后，选择菜单命令“文件”→“新建”，屏幕上出现新建对话框，其中包括文件、工程、工作区和其它文档 4 个卡片。一般当前卡片是工程，如果不是，则应点击“工程”标签，使之成为当前卡片，如图 1-2 所示。

(2) 选择 Win32 Console Application(32 位控制台应用程序)。在窗口的右上方的“工程”处，输入要建立的项目的名称，在“位置”处输入工程所在的路径，然后按“确定”按钮。屏幕上出现图 1-3 所示的 Win32 Console Application-Step 1 of 1 界面。

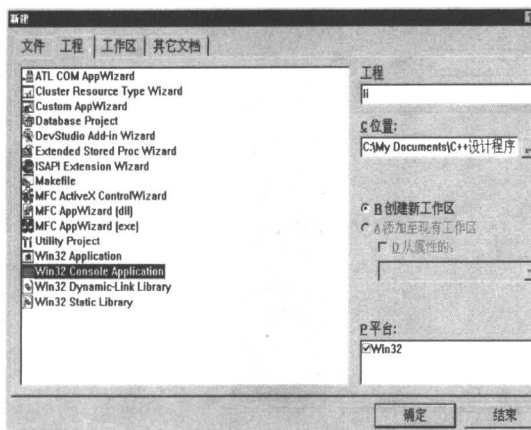


图 1-2

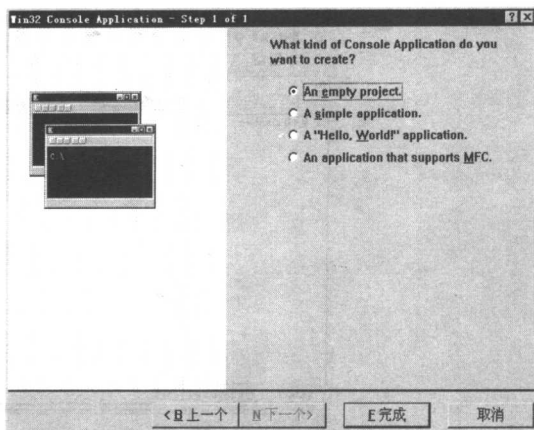


图 1-3

(3) 图 1-3 是应用程序生成向导，提问要生成的项目类型。选择“An empty project(空项目)”，然后按“完成”按钮。此时，屏幕上会出现图 1-4 所示的“新建工程信息”窗口，检查无误后按“确定”按钮。

2) 建立主程序文件 lyh.cpp

再次选择“文件”菜单的“新建”选项，弹出图 1-5 所示的窗口。选择“文件”标签，从窗口中选择 C++ Source File(C++源程序)，在窗口右侧“文件”处填写文件的名字 lyh，在“目录”处输入该文件存放的路径，然后按“确定”按钮，弹出图 1-6 所示的窗口。

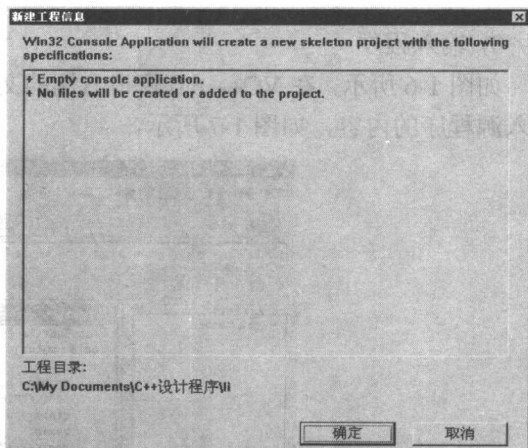


图 1-4

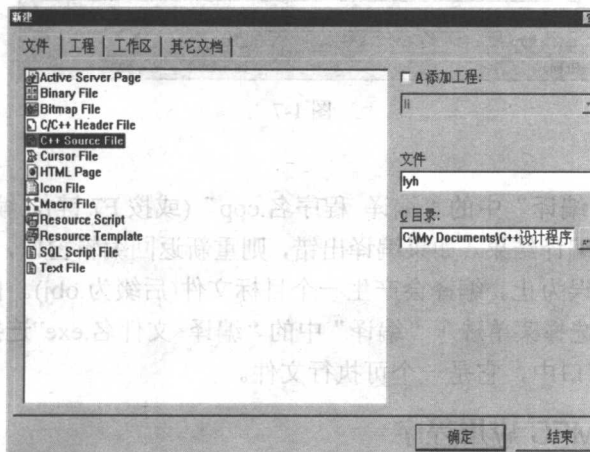


图 1-5