21st CENTURY
规划教材

面向21世纪高职高专计算机系列规划教材

COURSES FOR VOCATIONAL HIGHER EDUCATION: COMPUTER

Visual C++
程序设计教程

肖 力 编 著

科学出版社
www.sciencep.com

规划教材

面向21世纪高职高专计算机系列规划教材

COURSES FOR VOCATIONAL HIGHER EDUCATION: COMPUTER

Visual C++程序设计教程

肖　力　编著

科学出版社

北京

内 容 简 介

本书在介绍 Visual C++ 6.0 的基本操作基础上，系统地分析了基于 MFC 的 Windows 应用程序所需的基本编程技术。这些技术主要包括：Windows 应用程序框架的创建与消息处理、框架、文档与视图、菜单、工具栏与状态栏、对话框、控件、文件操作、多媒体编程和数据库编程等。

本书选材新颖，实例丰富，通俗易懂，可操作性强。全书注重培养读者的应用能力和良好的程序设计风格，并力求理论联系实际。

本书可作为高职高专院校计算机专业学生的 Visual C++ 程序设计教材，也可作为从事计算机应用的科技人员的自学教材和培训教材。

图书在版编目(CIP)数据

Visual C++ 程序设计教程/肖力编著. —北京：科学出版社，2004
（面向 21 世纪高职高专计算机系列规划教材）
ISBN 7-03-013855-4

Ⅰ.V … Ⅱ.肖… 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（2004）第 068120 号

责任编辑：李昱颉 陈砺川/责任校对：柏连海
责任印制：吕春珉/封面设计：飞天创意

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
*
2004 年 8 月第 一 版 开本：782×1092 1/16
2004 年 8 月第一次印刷 印张：14 1/4
印数：1－3 000 字数：320 000

定价：20.00 元

（如有印装质量问题，我社负责调换〈双青〉）

面向21世纪高职高专规划教材专家委员会

主　任　李宗尧

副主任（按姓氏笔画排序）

丁桂芝　叶小明　张和平　林　鹏
黄　藤　谢培苏

委　员　略

信息技术系列教材编委会

主　任　丁桂芝

副主任（按姓氏笔画排序）

万金保　方风波　徐　红　鲍　泓

委　员（按姓氏笔画排序）

于晓平　马国光　仁英才　王东红　王正洪
王　玉　王兴宝　王金库　王海春　王爱梅
邓　凯　付百文　史宝会　本柏忠　田　原
申　勇　任益夫　刘成章　刘克敏　刘甫迎
刘经玮　刘海军　刘敏涵　安志远　许殿生
何瑞麟　佘少华　吴春英　吴家碧　吴瑞萍
宋士银　宋锦河　张红斌　张环中　张海鹏
张蒲生　张德实　李云程　李文森　李　洛
李德家　杨永生　杨　闯　杨得新　肖石明
肖洪生　陈　愚　周子亮　周云静　胡秀琴
赵从军　赵长旭　赵动庆　郝　梅　唐铸文
徐洪祥　徐晓明　袁德明　郭庚麒　高延武
高爱国　康桂花　戚长政　曹文济　黄小鸥
彭丽英　董振珂　蒋金丹　韩银峰　魏雪英

出版前言

随着世界经济的发展，人们越来越深刻地认识到经济发展需要的人才是多元化、多层次的，既需要大批优秀的理论性、研究性的人才，也需要大批应用性人才。然而，我国传统的教育模式主要是培养理论性、研究性的人才。教育界在社会对应用性人才需求的推动下，专门研究了国外应用性人才教育的成功经验，结合国情大力度地改革我国的“高等职业教育”，制定了一系列的方针政策。联合国教科文组织1997年公布的教育分类中将这种教育称之为“高等技术与职业教育”，也就是我们通常所说的“高职高专”教育。

我国经济建设需要大批应用性人才，呼唤高职高专教育的崛起和成熟，寄希望于高职高专教育尽快向国家输送高质量的紧缺人才。近几年，高职高专教育发展迅速。目前，各类高职高专学校已占全国高等院校的近1/2，约有600所之多。教育部针对高职高专教育出台的一系列政策和改革方案主要体现在以下几个方面：

- “就业导向”成为高职高专教育的共识。高职高专院校在办学过程中充分考虑市场需求，用“就业导向”的思想制定招生和培养计划。
- 加快“双师型”教师队伍建设。已建立12个国家高职高专学生和教师的实训基地。
- 对学生实行“双认证”教育。学历文凭和职业资格“双认证”教育是高职高专教育特色之一。
- 高职高专教育以2年学制为主。从学制入手，加快高职高专教学方向的改革，充分办出高职高专教育特色，尽快完成紧缺人才的培养。
- 开展精品专业和精品教材建设。已建立科学的高职高专教育评估体系和评估专家队伍，指导、敦促不同层次、不同类型的学校办出一流的教育。

在教育部关于“高职高专”教育思想和方针指导下，科学出版社积极参与到高职高专教材的建设中来。在组织教材过程中采取了“请进来，走出去”的工作方法。即：由教育界的专家、领导和一线的教师，以及企事业从事人力资源工作的人员组成顾问班子，充分分析我国各地区的经济发展、产业结构以及人才需求现状，研究培养国家紧缺人才的关键要素，寻求切实可行的教学方法、手段和途径。

通过研讨认识到，我国幅员辽阔，各地区的产业结构有明显的差异，经济发展也不平衡，各地区对人才的实际需求也有所不同。相应地，相同专业和相近专业，不同地区的教学单位在培养目标和培养内容上也各有自己的定位。鉴此，适应教育现状的教材建设应该具有多层次的设计。

为了使教材的编写能针对受教育者的培养目标，出版社的编辑分不同地区逐所学校拜访校长、系主任和老师，深入到高职高专学校及相关企事业，广泛、深入地和教学第

一线的老师、用人单位交流，掌握了不同地区、不同类型的高职高专院校的教师、学生和教学设施情况，清楚了各学校所设专业的培养目标和办学特点，明确了用人单位的需求条件。各区域编辑对采集的数据进行统计分析，在相互交流的基础上找出各地区、各学校之间的共性和个性，有的放矢地制定选题项目，并进一步向老师、教育管理者征询意见，在获得明确指导性意见后完成“高职高专规划教材”策划及教材的组织工作：

- 第一批“高职高专规划教材”包括三个学科大系：经济管理、信息技术、建筑。
- 第一批“高职高专规划教材”在注意学科建设完整性的同时，十分关注具有区域人才培养特色的教材出版。
- 第一批“高职高专规划教材”组织过程中，正值高职高专学制从 3 年制向 2 年制转轨，教材编写将其作为考虑因素，要求提示不同学制的讲授内容。
- 第一批“高职高专规划教材”编写
  - 强调以就业岗位对知识和技能需求下的教材体系的系统性、科学性和实用性。
  - 强调教材以实例为先，应用为目的；围绕应用讲理论，取舍适度，不追求理论的完整性。
  - 强调提出问题→解决问题→归纳问题的教、学法，培养学生触类旁通的实际工作能力。
  - 强调课后作业和练习（或实训）真正具有培养学生实践能力的作用。

在“高职高专规划教材”编委的总体指导下，第一批各科教材基本是由系主任，或从教学一线中遴选的骨干教师执笔撰写。在每本书主编的严格审读及监控下，在各位老师的辛勤编撰下，这套凝聚了所有作者及参与研讨的老师们的经验、智慧和资源，涉及三个大的学科近 200 种的高职高专教材即将面世。我们希望经过近一年的努力，我们奉献给读者的是他们渴望已久的适用教材。同时，我们也清醒地认识到，“高职高专”是正在探索中的教育，加之我们的水平和经验有限，教材的选题和编辑出版会存在许多不尽人意的地方，真诚地希望得到老师和学生的批评建议，以利今后改进，为繁荣我国的高职高专教育不懈努力。

科学出版社

2004 年 6 月 1 日

前　言

Visual C++ 6.0 是 Microsoft 公司推出的面向对象的、功能强大的、目前广为流行的新一代可视化软件开发工具。它提供了一个高效的 Windows 编程环境，将程序和资源的编辑、编译、调试和运行融为一体，将程序设计方法与可视的软件开发环境完美地结合在一起，具有优越的性能和强大的功能，受到众多程序设计人员的喜爱。

本书注重培养读者的实际操作能力，在介绍编程实现技术时，尽量做到浅显易懂，不过分追求知识的系统性和完整性，对于一些过于难懂的理论尽量省略，而主要介绍应用程序的实现方法，注重对读者实际编写应用程序能力的训练和培养，并且从开发实际应用程序的需要出发，重点培养读者分析问题和运用 Visual C++编程知识解决问题的能力。

本书共分 10 章。第 1 章是 C++语言基础，对编程语言进行了介绍。第 2 章介绍 Visual C++集成开发环境。第 3 章介绍 Windows 应用程序框架的创建与 Windows 应用程序中的消息处理。第 4 章是框架、文档和视图，对框架窗口进行概述，介绍了文档和视图及其关系。第 5 章介绍菜单、工具栏和状态栏的相关知识及使用方法。第 6 章介绍对话框的相关概念。第 7 章介绍常用控件的属性、操作和消息。第 8 章介绍文件操作的有关概念。第 9 章介绍多媒体编程技术。第 10 章是数据库编程，介绍 ODBC、DAO 及 ADO 的概念，以及在 Visual C++中如何使用 ODBC、DAO 和 ADO 创建数据库应用程序等。在每章结束部分精选大量习题，有助于读者复习和训练编程能力。

本书所有例题中的程序都在 Visual C++ 6.0 下编译调试通过。

由于作者水平有限，加之时间仓促，书中错误和不当之处在所难免，敬请读者批评指正。

编　者

2004 年 5 月

目　录

第 1 章　C++ 语言基础

知识点

- 类、对象、构造函数、析构函数
- 派生类与继承方式
- 重载、虚函数

难点

- 友元的概念及应用
- 内联函数
- 虚函数

要求

掌握：

- 构造函数、析构函数
- 派生类的定义与继承方式
- C++的 I/O 流

了解：

- C++的发展历史

1.1　C++ 的起源和特点

1.1.1　C++的起源

C++是美国贝尔实验室的 Bjarne Stroustrup 博士在 C 语言基础上，弥补了 C 语言存在的一些缺陷，增加了面向对象的特征，于 1980 年开发出来的一种既面向对象又面向过程的混合型程序设计语言。它提出了把数据和在数据之上的操作封装在一起的类、对象和方法的机制，并通过派生、继承、重载和多态性等特征，从而使软件，特别是大型复杂软件的构造和维护变得更加有效和容易，并使软件开发能更自然地反映事物的本质，从而大大提高了软件的开发效率和质量。

C++继承了 C 语言的原有精髓，例如高效率、灵活性；扩充增加了对开发大型软件极为有效的面向对象机制；弥补了 C 语言不支持代码重用、不适合开发大型软件的不足，成为一种既适合于作为系统描述语言，也适合于编写应用软件的，既可用于表现过程模型，又可用于表现对象模型的优秀的程序设计语言。

许多软件公司都为C++设计了编译系统，如AT&T、Apple、Sun、Borland和Microsoft等，其中最为流行的是Borland公司的Borland C++和Microsoft公司的Visual C++。与此同时，许多大学和公司也为C++编写了各种不同的类库，其中Borland公司的OWL（Object Windows Library）和Microsoft公司的MFC（Microsoft Foundation Class）就是比较优秀的代表，尤其是Microsoft公司的MFC，在国内外得到了广泛的应用。

1.1.2 C++的特点

C++现在得到了越来越广泛的应用，其主要特点是：

- C++保持了与C语言的兼容。C程序中的代码不经修改就可被C++使用，同时用C语言编写的大量的库函数也可用于C++中。
- 用C++编写的程序可读性更好，代码的结构更为合理，可直接在程序中映射问题空间的结构。
- 生成代码的质量较高，运行效率仅比汇编语言代码慢10%到20%。
- 从开发时间、费用到形成的软件的可重用性、可扩充性、可维护性和可靠性等方面都有了很大的提高，使得大型复杂软件的开发变得更加容易。
- 支持面向对象的机制，可方便地构造出模拟现实问题的实体和操作。

1.2 一个简单的C++程序

下面是一个C++程序的例子，其功能是在屏幕上显示“Hello World!”，代码如下：

【例1.1】 C++程序示例。

```
# include <iostream.h>
void main()
{
   char str_string[]="Hello World!";
   cout<<str_string<<endl;
}
```

从上例可以看出，用C++编写的程序与用C语言编写的程序在程序结构上基本相同。例如：都是以main函数作为程序的入口，两者都是以一对{}把函数中的语句括起来等。但两者也有一些不同之处，C++中是以iostream.h文件作为标准输入输出头文件，C语言中是以stdio.h作为标准输入输出头文件；C++中采用符号“<<”作为标准输出，而不是通过printf函数来实现。

由此可知，C++语言和C语言两者之间既有紧密的联系，又各有自己的特点。

1.3　C++的基本概念

1.3.1　注释行

程序员给程序加上适当的注释能有效地提高程序的可读性，对以后程序的维护有很大的帮助。尤其对于大型的程序，好的注释更加必不可少。

C++编译器支持两种形式的注释：

1）C 语言中使用的用“/*”及“*/”作为注释分界符号，在C++中仍然可以使用。如

```
/* This is a Test     */
```

2）以“//”开始的单行注释方式。在这种注释中，编译器一遇到“//”就会认为从它以后直到本行尾的所有内容都是注释。如

```
x=y+z  //This is a comment
```

由此可见，第一种注释方式一般用于多行注释；第二种注释方式用于单行注释。

1.3.2　新的 I/O 流

在C++中，除了可以继续使用 C 语言函数库 stdio.h 提供的输入输出函数 scanf ( )和 printf ( )外，还可以使用自己定义的输入输出系统来进行 I/O 操作，即标准输出流 cout 和标准输入流 cin，其定义在标准头文件 iostream.h 中。具体使用方法如下：

1）标准输出流 cout 与插入运算符“<<”结合使用，可以代替 C 语言中的 printf ( ) 函数，并且不需要任何格式符。例如：

```
cout<<"I like C++! "<<endl;
```

表示把字符串“I like C++!”插入到输出流 cout 中，其中，endl 表示换行。

2）标准输入流 cin 与抽取运算符“>>”结合使用，可以代替 C 语言中的 scanf ( ) 函数，也不需要任何格式符，如

```
cin>>x;
```

表示从输入流 cin 中抽取 x 的值。

【例 1.2】　使用 cout、cin 输出输入数据。

```
# include<iostream.h>
void main ()
{
   cout<<" I like C++! ";        //输出一个字符串
   cout<<2003;                   //输出一个整数
   cout<<"\ n ";                 //换行
   cout<<20.1;                   //输出一个实数
   cout<<endl;                   //换行
   cout<<"I am "<<20<<"years old student.";      //连续输出
   char name[30];
   int age;
   cout<<"Please give your name : ";
   cin>>name;                    //从键盘输入字符串到数组 name 中
   cout<<"Please tell me how old are you?";
```

```
cin>>age;                    //表示键盘输入整数到变量 age 中
cout<<" Your name is"<<name<<endl;
cout<<"Your are "<<age<<" years old.";
}
```

注意：

- 用 cout 和<<可以输出包含字符串在内的任何基本数据类型，且不需要格式符。
- 在输出语句中，可以通过输出 endl 来换行。
- 用 cin 和>>可以输入包含字符串在内的任何基本数据类型，且不需要格式符。
- 在一个 cout（cin）语句中，可以连续使用多个输出运算符<<（输入运算符>>）来输出（输入）多个变量。

1.3.3 灵活的变量声明

下面代码段在 C++中是正确的，编译时不会出错。

```
f()
{
    int i ;
    i=10;
    int j;
    j=25;
   ……
}
```

此外，在 C++中允许在 for 循环语句中声明变量，如

```
for ( int k=5; k>=0; k--)
```

由此可见，在 C++中允许在代码中的任意位置声明局部变量，它所声明变量的作用范围为声明点到该变量所在的最小分程序末。

1.3.4 内联函数

在函数声明前，加上关键字“inline”后，该函数就被声明为内联函数。而引入内联函数的目的是为了解决程序中函数调用的效率问题。

在程序中，当出现对内联函数的调用时，C++编译器不会像一般函数那样要转去执行被调用函数的函数体，执行完成后再转回调用函数中，而是直接使用函数体中的代码替代函数调用表达式，这样就可以加快代码的执行，减少调用开销，从而提高了运行速度。

但要注意的是，内联函数在调用之前必须进行完整的定义，否则编译器将无法知道应插入什么代码，它通常声明在主函数前。而且内联函数无法递归调用。

【例 1.3】 内联函数的使用。

```
# include<iostream.h>
inline int doub ( int x )
{ return x*2 }
int main ()
{
```