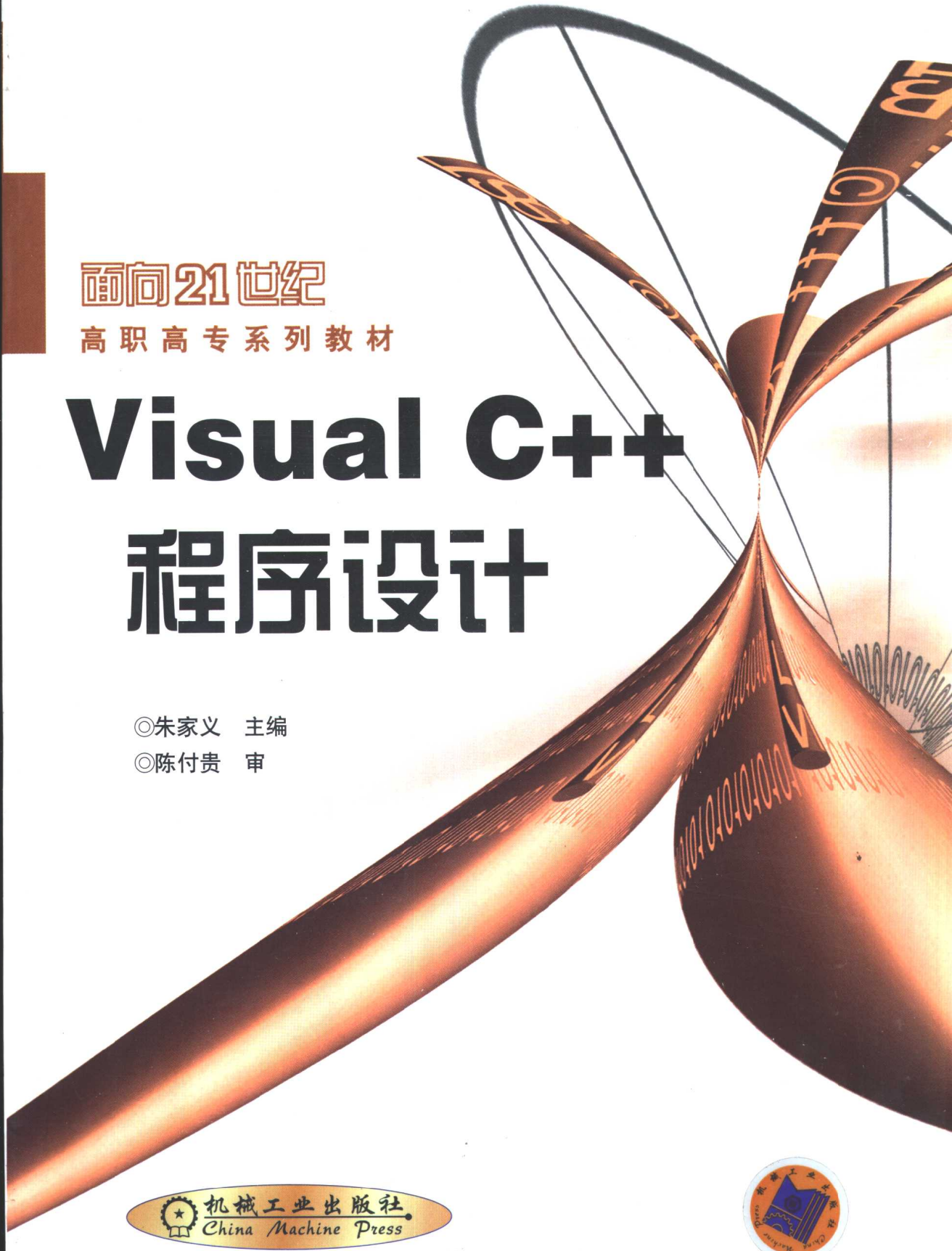
面向21世纪

高职高专系列教材

Visual C++ 程序设计

◎朱家义　主编

◎陈付贵　审



机械工业出版社
China Machine Press

面向21世纪高职高专系列教材

Visual C++程序设计

朱家义　主编
陈付贵　审

机械工业出版社

全书共分 11 章，系统介绍了 Visual C++6.0 的程序设计方法。内容包括：Visual C++6.0 简介，MFC 的层次结构，AppWizard 的使用，菜单、状态栏和工具栏，响应用户命令，制作普通帮助文件，文档/视图结构，输入输出及打印，数据库编程、ActiveX 控件的设计与使用等。

本书注重应用方法和可视化程序设计的介绍，概念表达准确，语言精练，通俗易懂，实用性强，便于教学和自学。

本书适于作为高职高专计算机专业的专业课程教材，同时可作为科技人员和高校学生的自学用书，也可供从事应用软件开发的技术人员参考。

图书在版编目(CIP)数据

Visual C++程序设计/朱家义主编.—北京：机械工业出版社，2003.1
面向 21 世纪高职高专系列教材
ISBN 7-111-11097-8

Ⅰ.V… Ⅱ.朱… 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(2002)第 082541 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)
策　　划：胡毓坚
责任编辑：王冰飞
责任印制：闫　焱

北京交通印务实业公司印刷·新华书店北京发行所发行

2003 年 1 月第 1 版·第 1 次印刷
1000mm×1400mm　B5·8.875 印张·343 千字
0001-5000 册
定价：22.00 元

面向21世纪高职高专
计算机专业系列教材编委会成员名单

顾问	曾玉崑　王文斌　陈瑞藻　李　奇　凌林海 林　东
主任委员	周智文
副主任委员	周岳山(常务副主任)　詹红军　陈付贵 穆天保　赵佩华　黄甘洲　武文侠　吕何新
委员	郭曙光　王德年　刘瑞新　陈丽敏　孔令瑜 李　玲　鲁　辉　陶书中　赵增敏　马　伟 孙心义　翟社平　廖常武　于恩普　王春红 王娟萍　屈　圭　汤新广　谢　川　姜国忠 汪赵强　董　勇　梁国浚　张晓婷
秘书长	胡毓坚
副秘书长	陈丽敏(兼)

出版说明

积极发展高职高专教育，完善职业教育体系，是我国职业教育改革和发展的一项重要任务。为了深化职业教育的改革，推进高职高专教育的发展，培养21世纪与我国现代化建设要求相适应的，并在生产、管理、服务第一线从事技术应用、经营管理、高新技术设备运作的高级职业技术应用型人才，尽快组织一批适应高职高专教学特色的教材，已成为各高职高专院校的迫切要求。为此，机械工业出版社与高职高专计算机专业、电子技术专业和机电专业教材编委会联合组织了全国40多所院校的骨干教师，共同研究开发了一批计算机专业、电子技术专业和机电专业的高职高专系列教材。

各编委会确立了“根据高职高专学生的培养目标，强化实践能力和创新意识的培养，反映现代职业教育思想、教育方法和教育手段，造就技术实用型人才为立足点”的编写原则。力求使教材体现“定位准确、注重能力、内容创新、结构合理和叙述通俗”的编写特色。

本套系列教材是由高职高专计算机专业、电子技术专业、机电专业教材编委会分别会同各院校第一线专业教师针对高职高专计算机、电子技术和机电各专业的教学现状和教材存在的问题开展研讨，尤其针对目前高职高专教学改革的新情况，分别拟定各专业的课程设置计划和教材选题计划。在教材的编制中，将教学改革力度比较大、内容新颖、有创新精神、比较适合教学、需要修编的教材以及院校急需、适合社会经济发展的新选题优先列入选题规划。在广泛征集意见及充分讨论的基础上，由各编委会确定每个选题的编写大纲和编审人员，实行主编负责制，编委会通过责任编委和主审对教材进行质量监控。

担任本套教材编写的老师们都是来自各高职高专院校教育第一线的教师，他们以高度的责任感和使命感，经过近一年的努力，终于将本套教材呈现在广大读者面前。由于高职高专教育还处于起步阶段，加上我们的水平和经验有限，在教材的选题和编审中可能出现这样那样的问题，希望使用这套教材的教师和学生提出宝贵的意见和建议，以利我们今后不断改进，为我国的高职高专教育事业的繁荣而共同努力。

高职高专系列教材编委会
机 械 工 业 出 版 社

前　言

Visual C++是运行于 Windows 95/98/NT 环境下的可视化编程工具中最重要的成员之一。它把 Windows 统一直观的界面风格、面向对象的程序设计方法和面向资源的环境结合在一起，形成一个功能强大的 C++编译器。它提供了简单高效的操作方式、高效的内存管理、与设备无关的图形接口、数据共享和多任务运行机制，同时又提供了一系列功能强大的开发工具和内容丰富的开发资源。因此，使用 Visual C++来进行 Windows 应用程序的开发具有很强的优势，学习 Visual C++也成为广大程序设计和开发人员的迫切需要。

为满足我国高职高专教育的需要，根据高职高专院校的培养目标、培养规格、教学计划和课程教学大纲的基本要求，我们会同有关专业教师认真研究了《Visual C++程序设计》一书的编写大纲，在总结长期教学实践经验，集思广益，博采众长的基础上编写了此教材。

全书共分 11 章。在第 1 章介绍了 C++语言的基本概念和基础知识，第 2 章至第 11 章系统介绍了 Visual C++6.0 的程序设计方法。内容包括：Visual C++6.0 简介，MFC 的层次结构，AppWizard 的使用，菜单、状态栏和工具栏，响应用户命令，制作普通帮助文件，文档/视图结构，输入输出及打印，数据库编程，ActiveX控件的设计与使用等内容。

本书图文并茂，概念表达准确清晰，语言精炼，通俗易懂。

本书不仅适合于教学也非常适合于用 Visual C 编程和开发应用程序的用户学习和参考。认真阅读本书，并结合上机操作进行练习，就能在较短的时间内基本掌握 Visual C 及其应用技术。

本书由朱家义主编，陈付贵审稿。其中朱家义编写第 1～2 章；李克普编写第 3～4 章；付俊辉编写第 5～7 章；高国红编写第 8～11 章。在编写出版过程中，得到了周智文、周岳山先生及机械工业出版社的大力支持，在此一并致谢。

由于水平所限，书中不妥及错误之处，殷切期望专家、同仁和广大读者指正。

编　者

目　录

第1章　C++语言基础知识

Visual C++6.0继承了C/C++语言的简单、高效、易学易用的特点，代码结构清晰，可读性好，并且融入了面向对象、过程可视化、事件驱动等软件开发的最新技术，使C/C++语言编程技术发展到了一个新的高度。

1.1　编程概述

生成一个程序的四个基本步骤是：设计算法、编写程序、编译和调试程序。

- 设计程序是指分析问题，确定解决问题的策略，选择优良的算法。
- 编写程序是指通过一种计算机语言来实现算法的过程。用高级语言编写的程序称为源程序。
- 编译是把由高级语言编写的源程序转换为计算机可识别的目标程序。一个程序被编译后，计算机可以运行它了。
- 调试指的是查找和修改源程序错误的过程，是对源程序的完善。

1.1.1　编程基础

程序由许多命令组成，C++程序的起始点称为主函数，当程序开始执行时，主函数中的第一个语句被执行。然后，依次执行所有的后续语句。

C++程序通常包括源文件和头文件两大部分。源文件称为CPP文件（表示C Plus Plus文件），它包含了程序的主要部分。在源文件中，可输入例程、定义数据并决定程序的流程。

源文件中可以使用其他源文件或库中的例程，其方法是包含一个头文件。头文件为编译器指明了那些从其他源文件或库中调用的例程的名称和特征。

如果在一个文件中生成了例程，并且把这些例程运用于另一个文件中，就要生成一个头文件来描述这些例程。通过在其他源文件中读头文件，就可以访问在那些源文件或外部库中的例程。

1.源文件

源文件由语句组成。语句用来确定计算机要执行什么操作。例如，下面这一行语句完成了计算半径为10 cm的圆的面积：

```
area = 3.14 * 10 * 10;
```

每一个语句都要以分号结尾。

用“{”和“}”把一组语句括起来,就形成了复合语句,一个复合语句被看作一个整体。

用“{”和“}” 也可以定义函数的开始与结束。

在向一个函数传递参数时,用“(”和“)”把这些参数括起来。

2.main 函数

main 函数不返回数值,main 函数中的第一行将被计算机最先执行。例如:

```
#include<iostream.h>
void main()
{
cout<<"hello!"<<endl;
}
```

在这个程序中有一个称作 main 的函数,程序就从这里开始。用“{”和“}”来表示 main 函数的开始和结束。

函数是自包含的例程,这些例程用来完成各种各样的功能,如数据的输入、处理或输出等。有些函数能够返回数值,例如,sin 函数可以返回一个数的正弦值。

void 表示不返回任何值。

3.数据的输出

当程序中出现 cout << 时,就表示要输出它后面的内容,如果要输出文本内容,可以用双引号把要输出的文本括起来,例如:

```
cout<<"Hello!";
cout<<3.14*10*10;
```

也可以同时输出几个数值。例如:

```
cout<<"area is"<<3.14*10*10;
```

从上面这个例子可以看到,文本和数字可以很容易地结合在一起同时输出。

输出数据时,可以使用下面的特殊字符:

\n	换行
\t	Tab
\b	回车
\f	开始新的一页
\\	输出 \ 字符
\'	输出'字符
\"	输出"字符

其中“\n”也称作换行符,例如:

```
cout << "area is \ n" << 3.14 * 10 * 10;
```

输出结果为：

```
area is
314
```

也可写成如下形式：

```
cout << "area is" << " \ n" << 3.14 * 10 * 10;
```

C++提供了一个称作 endl 的函数，也可以用它来实现换行。上面的例子可用 endl 改写成如下形式：

```
cout << "area is" << endl;
cout << 3.14 * 10 * 10;
```

与" \ n"不同，endl 不能遮字符串的中间或末尾使用，而是必须把字符串分成段。如：

```
cout << "My name" << endl << "is Bob" << endl;
```

4.数据的输入

使用 cin 和 >> 可实现数据的输入，例如：

```
cin >> a;
```

用户输入数值后，这个数值被存入变量 a 中。

5.预处理

include 预处理指令告诉编译器装入一个 include 文件。例如 cin 和 cout 被定义在称为 iostream.h 的文件中(其中.h 是头文件的标准扩展名)。为了装入这些定义，必须在程序的开始加上如下一行：

```
# include〈iostream.h〉
```

它装入了 cin、cout 和许多 iostream 库中的其他例程的定义。

预处理指令后面不加分号，并且只能写在一行上。

1.1.2 数据类型

C++提供了许多预先定义的数据类型，用户可以把不同的数据类型组合起来构造更复杂的数据类型。下面是四种最常用的数据类型：

char——表示字符。如'a'、'b'和'*'都是字符。该类型数据占一个字节的空间。

float——表示浮点数。即带有小数点的数字，例如 3.14、-1.50 和 20.0 等等，浮点数有时也称作实数，浮点数的取值范围是 $\pm 3.14 \times 10^{-38} \sim \pm 3.4 \times 10^{38}$，该类型

数据占四个字节的空间。

double——表示双精度数。双精度数的取值范围是 $\pm 1.7 \times 10^{-308} \sim \pm 1.7 \times 10^{308}$，该类型数据占八个字节的空间。

int——表示整数。整数是不包含小数点的数，例如，0、3、30 和 - 59 都是整数，但是 1.0 不是整数，整数的取值范围是 - 32768 ~ 32767。该类型数据占两个字节的空间。

除上述一些常见的数据类型外，还有如下一些数据类型：

long——表示长整型数。取值范围是 - 2147483648 ~ 2147483647，该类型数据占四个字节的空间。

short——表示短整型数。取值范围是 - 32768 ~ 32767，该类型数据占两个字节的空间。

unsigned——表示无符号整型数。它表示数值总是正的整数，例如 unsigned int 类型的取值范围是 0 ~ 65535。

void——无类型数，用来表明函数不返回值，如：void main()。

1.类型安全性

C++ 对数据类型有严格的要求，如果把一个变量声明为一种类型，而在实际使用时又试图把它用作另外一种类型，编译器就会产生错误。这种对数据类型的严格要求称作类型安全性，例如：

```
int r;
r = "Hello!";
```

运行此程序时将会出错，因为 r 是整型，所以不能给它赋一个字符串，因此，应该明确告诉编译器把一种数据类型转换成另一种数据类型，这称作强制类型转换。要做到这一点，只需在被转换的项前面加入要转换的数据类型名，例如：

```
int   r;
float inp;
r = (int)inp;
```

此时，编译器把浮点型变量 inp 的值转换成整型并赋给变量 r。例如，如果 inp 的值为 3.141592653，那么 r 将被赋值为 3。

2.常量

常量指的是在程序执行过程中不变的数据。它有两种形式，其一是数值形式，如：3.14、135、0.132e + 02 等；其二是标识符形式，称为符号常量。要使一个标识符成为符号常量，只需在声明前面加上 const 即可，例如：

```
const in   Length = 3;
```

Length 就是一个符号常量,其值为 3。

1.1.3 变量

变量指的是在程序执行过程中其值可以变化的量,任何一个变量必须有确定的名称。只需通过变量的名称,便可以访问存储在变量内的信息。

1.命名变量

变量名由字母、数字和下划线组成,但有以下限制:

- 变量名不能以数字开头;
- 变量名中不能有空格;
- 变量名中除了能使用 26 个英文字母外,只能使用"_";
- 变量名不能与 C++ 语言中的关键词同名,表 1-1 列出了 Visual C++ 的关键词;
- 变量名不能与 C++ 中的库函数名相同。

表 1-1 Visual C++ 关键词

关 键 词	关 键 词	关 键 词	关 键 词
_ asm	_ based	_ cdecl	_ declspec
_ except	__ fastcall	_ finally	_ inline
_ int16	_ int32	int64	_ int8
_ leave	_ multipleinheritance	_ single _ inheritance	_ virtual _ inheritance
auto	bool	break	case
catch	char	class	const
const _ cast	continue	default	delete
dllexport	dllimport	do	double
dynamic _ cast	else	enum	explicit
extern	false	float	for
friend	goto	if	inline
ntlong	main	multable	naked
namespace	new	operator	private
protected	public	register	reinterpret _ cast
return	short	signed	sizeof
static	static _ cast	struct	switch
template	this	thread	throw
true	try	typedef	typeid
typename	union	unsigned	using
uuid	virtual	void	volatile
while	wmain	xiloc	

例如 way123、abc、a _ bc 是合法的变量名，而 if、5ab、x + y 是一些错误的变量名。变量名是区分大小写的，例如 abc、Abc 和 aBc 都是不同的变量。

2.定义变量

定义变量时，先写变量的类型名称，再写出变量名称，例如：

```
int r;
float Len;
long Johns;
```

如果所定义变量的类型都相同，可以把同类型的变量写在一行内。例如，若要定义三个名为 a，b 和 c 的整型变量，可写成：

```
int  a,b,c;
```

3.变量初始化

在定义一个变量的同时，给变量一个初始值，这个过程称为变量的初始化。初始值是这个变量第一次使用时的值，它应写在变量名和“ = ”的后面，例如：

```
int Num = 3;
float Len = 3.5;
long Johns = 32700;
```

1.1.4 结构体

把一系列相关的变量组织成一个单一实体的过程称作生成结构体。

1.声明结构体

声明结构体的一般方法是：

```
class 结构体名
{
public:
类型 1  成员 1;
类型 2  成员 2;
类型 3  成员 3;
…
类型 n  成员 n;
};
```

例如：

```
class str {
```

```
public:
int Num;
int Time;
};
```

在这个例子中,结构体的名称是 str,它包含一个称作 Num 的整型成员和一个称作 Time 的整型成员。

把一个变量定义为一个结构体类型的变量,只需在结构体名后面写出变量名,例如:

```
str a,b;
```

2.访问结构体的成员

通过结构体类型的变量,就可以访问它的成员。方法是:

结构体变量名.成员名

例如:

```
cin >> a.Num;
```

1.1.5 运算符与表达式

由运算符及其运算对象所组成的式子,称为表达式。

1.常用运算符

表 1-2 列出了常用的运算符及其说明。

表 1-2 运算符及其说明

运算符	使用方法	说 明
*	a*b	表示两个数相乘。例如,6*3 等于 18
/	a/b	表示两个数相除。例如,18/3 等于 6
+	a+b	表示两个数相加。例如,6+3 等于 9
-	a-b	表示两个数相减。例如,9-3 等于 6
%	a%b	表示取模。返回两个数相除的余数。例如,10%3 等于 1,因为 10/3 等于 3,而余数是 1
++	a++、++a	增量运算符。等价于 a=a+1
--	a--、--a	减量运算符。等价于 a=a-1
>>	a>>b	右移位运算符。如果执行 a>>b,那么它等价于 $a/2^b$ 的整数结果
<<	a<<b	左移位运算符。如果执行 a<<b,那么它等价于 $a*2^b$ 的结果

(1)%运算符