21世纪 高等院校计算机系列教材

C++程序设计简明教程

（第二版）

艾德才　迟丽华　等编著

中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书之所以称其为简明教程，一是内容简洁，二是重点突出，三是实用性强。本书内容虽仅八章，但概括了C++的全部主要内容，使读者免去了阅读那些烦琐的冗长教科书的烦恼。本书把C++的精华全部概括其中，书中所有程序都是作者教学经验之结晶，是作者得意之作并都在计算机上调试过的。其中不乏指导学生参加C++程序设计比赛的优秀作品。

全书共分八章，分别介绍了C++的基本概念、数据类型、语句、函数和程序的基本构成，类的定义、派生与继承，函数和操作符的重载，数据流与文件等内容。

在每一章的后面都附有适量的习题，读者可通过习题巩固已学的知识。上机练习是每章学习过程中必做的工作，通过上机练习把理论知识与实际应用结合起来，加深对C++程序设计技术的理解并掌握程序设计的技巧。

本书可作为高等院校本专科相关专业学习程序设计课程的教学用书，也可作为自学C++程序设计的参考书目。

图书在版编目（CIP）数据

C++程序设计简明教程 / 艾德才等编著. —2版. —北京：中国水利水电出版社，2004

（21世纪高等院校计算机系列教材）

ISBN 7-5084-2297-X

Ⅰ.C… Ⅱ.艾… Ⅲ.C语言—程序设计—高等学校—教材 Ⅳ.TP312

中国版本图书馆CIP数据核字（2004）第076501号

书　　名	C++程序设计简明教程（第二版）
作　　者	艾德才　迟丽华　等编著
出版　发行	中国水利水电出版社（北京市三里河路6号　100044）
	网址：www.waterpub.com.cn
	E-mail：mchannel@263.net（万水）
	sales@waterpub.com.cn
	电话：（010）63202266（总机）、68331835（营销中心）、82562819（万水）
经　　售	全国各地新华书店和相关出版物销售网点
排　　版	北京万水电子信息有限公司
印　　刷	北京蓝空印刷厂
规　　格	787mm×1092mm　16开本　14印张　314千字
版　　次	2001年1月第1版
	2004年8月第2版　2004年8月第4次印刷
印　　数	16001—21000册
定　　价	20.00元

前　言

《C++程序设计简明教程》（第二版），是在2001年出版的《C++程序设计简明教程》（修订版）的基础上修改而成的。

较之第一版，本教材教学体系根据国人的认知习惯和C++的特点进行了较大的改进，教学内容实例进行了重新按排。在简明的基础上，尽力做到了使教学内容由简到繁、由易到难、循序渐进，同时又增加了具有实用性和代表性的程序实例。

面向对象的程序设计技术，在近年来有了突飞猛进的发展，亦成为最热门、最实用的软件开发手段。

C++程序设计语言是一门高效、实用的程序设计语言。C++语言保持了与C语言的兼容，一方面是为了广大的熟悉C语言的程序设计人员，他们只需学习C++语言新增加的功能部分；另一方面也是为了保护广大用户在C语言上所做的大量的人力物力上的投资。作为一种程序设计语言，C++有它自己的抽象和结构特征。

在面向对象的程序设计语言中，C++是最常用的语言之一。C++是一种新型的程序设计语言，虽然在结构和规则上是C语言的增强与扩充，但由于C++引入了面向对象的程序设计思想，已不再是C语言的高版本，而成为一种新的面向对象的程序设计语言。C++允许程序设计人员把传统的结构化的程序设计技术与面向对象的程序设计技术结合起来，以这种方式程序设计人员取两种程序设计手段之所长，从而可开发出大型软件系统。程序设计人员利用C++面向对象的程序设计技术，派生的类（derived classes）可被定义，这些类从一个或多个基类（base classes）继承所有特性，并可进行重新定义和增加新特性。应用结构化程序设计技术的函数定义了每种方法，以确保执行这些方法的一系列步骤是正确的。一旦实际应用中的对象的类被定义或确定，用结构化程序设计技术编写的应用程序，再一次保证程序运行期间的正确性。

本书在编写过程中，采用了大量的程序实例，由浅入深，系统地介绍了C++的特点和用C++进行面向对象程序设计的技术手段，帮助读者真正掌握C++语言和面向对象的程序设计方法。全书共分八章，分别介绍了C++的基本概念、数据类型、语句、函数和程序的基本构成，类的定义、派生与继承，函数和操作符的重载，数据流与文件等内容。在每一章的后面都附有适量的习题，读者可通过习题巩固已学的知识。上机练习是每章学习过程中必做的工作，通过上机练习把理论知识与实际应用结合起来，加深对C++程序设计技术的理解并掌握程序设计的技巧。

本书是作者多年教学经验和实际程序设计经验相结合的成果，选材从实际出发，深入浅出，语言通俗易懂，并兼顾了深度和广度。本书中所使用的实例均在计算机上调试通过，希望读者进一步上机去体会C++的程序设计技巧和面向对象程序设计的精髓。

参加本书编写的还有龚涛、李俊生、杨宏宇、姚嘉康等。艾德才教授审较了全部书稿。

编写本教材，是在C++程序设计教学上从繁到简的一次尝试，虽力图做好，但由于作者水平有限，不足之处在所难免，殷切希望广大同仁和读者不吝赐教，以便使本教材质量进一步得到提高。

编　者

2004年6月于天津大学

目　　录

第 1 章　C++概述

1.1　C++面向对象基础

C++程序设计语言源于 C 语言，C 语言是在 20 世纪 70 年代推出的。由于 C 语言的规则简单，它不仅拥有高级语言的数据表示和运算功能，而且可以直接对存储器内的数据进行操作，用 C 语言编写的程序在运行时的效率也比较高，所以 C 语言被广大的程序员所青睐，成为一种世界流行的程序设计语言。

C++程序设计语言诞生于 20 世纪 80 年代初，由贝尔实验室推出。最初 C++称为带类的 C，后来命名为 C++。

C++是 C 语言的扩充。C++的名字强调了从 C 语言的演化特性。C++在 C 语言的基础上增加了面向对象程序设计的特征，C 语言在 C++中作为子集保留下来。由于 C++具有面向对象、高效、安全可靠、灵活以及良好的继承性和前瞻性等特点，深受软件设计人员的普遍欢迎。

C++是面向对象的程序设计语言。面向对象程序设计以更接近于人类思维的方式分析问题和解决问题，面向对象程序设计的基本概念是对象和类。自然界中的任何事物都可以看作对象，也就是说，自然界是由千千万万个对象构成，它们之间通过一定方式相互联系。例如，学校是一个对象，学校中的每个班级都是不同的对象。每个对象都有它的特性和行为，例如，一个班级的人数、专业、学制等都是它的特性。班级要进行活动，如上课、开会、文娱等，这些都是它的行为。在面向对象程序设计中，将同一种类型的对象归属为一个类。例如，声明一个班级类，那么 1 班、2 班、3 班都是班级类的对象。

C++包含面向对象的 4 个重要特征：封装、数据隐蔽、继承和多态。

（1）封装和数据隐蔽。通过类将对象的特性和功能组合成一个整体，称为封装。建立类就是建立封装的实体，类作为一个整体使用，隐藏了内部构造。对用户来说，只要知道如何使用类的功能，而不必知道它是如何工作的。

（2）继承与重用。在原有类的基础上，经过适当的扩充和完善而派生出的新类，称之为子类或派生类，原有的类称为子类的父类或基类，也可称为超类。子类和父类形成类的层次和类的继承关系。由于子类是在父类的基础上建立的，因此子类继承了父类的特性和功能。通过类的继承机制可以实现程序代码的重用，提高程序设计效率。

（3）多态。多态性是指对象改变形式的能力，一个多态性对象可以有多种形式。C++通过函数多态与类多态支持“不同类型有各自的响应”的思想。

1.2 C++程序简介

1.2.1 C++程序举例

用C++可以编写由函数组成的程序，也可以编写带类的程序。本节给出几个简单的C++程序，通过这些程序初步了解C++程序。

下面是一个完整的C++程序，命名为Hello.cpp，程序的功能是在屏幕上显示一行文字。

【程序1.1】简单C++程序举例。

```
#include<iostream.h>
void main()
{ cout<<"Hello world.\n"; }
```

程序说明：

（1）程序中的#include<iostream.h>是一个声明，作用是指示编译程序将iostream.h文件包含到程序中。iostream.h文件中包括对输入输出的声明。有了这些声明，就可以向标准输出流cout（屏幕）输出信息。

（2）main()定义了名为main的函数，每一个C++程序都必须有main函数。main函数的内容写在大括号{ }之内。void表示main函数没有返回值。

（3）语句cout<<"Hello world.\n"是main函数的内容。作用是将字符串"Hello world.\n"输出到标准输出流cout（屏幕）上。字符串是用双引号括起来的字符序列。\n表示换行字符。

【程序1.2】带注释的程序。

```
#include<iostream.h>
void main()
{/*This is a comment and it extends until
  the closing star-slash comment mark .*/
cout<<"Hello,world!.\n";
//This comment ends at the of the line.
}
```

该程序的运行结果与程序1.1相同。

注释是对程序的说明，不影响程序的运行结果。为增加程序的可读性，便于日后维护和修改，需要为程序添加注释。

C++的注释有两种类型，一种是行注释，另一种是块注释。

行注释以//为标志，从//开始到本行末尾的所有字符都是对程序的说明。

块注释以/*和*/为标志，其中/*表示注释的开始，*/表示注释的结束。/*和*/之间的所有字符都是对程序的说明，可跨越多行。

注意：每一个/*都必须有另一个*/与之相匹配。

【程序 1.3】由两个函数组成的程序。

```
#include<iostream.h>
int add(int x,int y)
{ cout<<"In add() received "<<x<<" and "<<y<<"\n";
 return(x+y);
}
void main()
{cout<<"I'm in main()!\n";
    int a,b,c;
    cout<<"Enter two number:";
    cin>>a;
    cin>>b;
    cout<<"\nCalling add()\n";
    c=add(a,b);
    cout<<"\nBack in main().\n";
    cout<<"c was "<<c;
}
```

程序运行情况：

```
I'm in main()!
Enter two numbers: 2 4↙
Calling add()
In add() received 2 and 4
Back in main()
c was 6
```

本程序由 main()和 add()两个函数组成。程序从 main()开始执行，从 c=add(a,b);行程序转向执行函数 add()，函数 add()接受两个整数参数并返回一个整数，语句 return(x+y);表示结束函数调用，并返回结果值。cin 的功能是将两个数值读入到变量 a 和变量 b 中。cout 用于向屏幕输出数据。

main()主函数是一个特殊的函数，是程序的入口。操作系统通过调用它开始程序。其他函数是在程序的运行过程中，由 main()函数或另外一些函数调用。程序是按照源代码中的次序逐行执行，直到调用一个函数，此时程序转向去执行这个函数。当函数调用结束后，它把控制权交给调用函数语句的下一行语句。在后面的几章中，还要对函数进行深入讨论。

【程序 1.4】带类的 C++程序。程序功能是计算矩形面积。

```
class rect                    //建立计算矩形面积的 rect 类
{ public:
  int area(int a, int b )
   { return a*b; }
};
void main()
```

```
{ rect s ;                    //建立 rect 类的对象 s
  cout<<s.area(3,9)<<endl ; //通过对象 s 计算矩形面积
}
```

1.2.2 C++程序的结构

通过以上几个程序可以看出，C++程序中可以包括头文件、主函数、其他函数和类等。头文件由#include 语句开始，这些语句通知编译程序，将给定的头文件包含到程序中。头文件中包含各种运算的定义、I/O 操作的定义和操作系统服务等。对程序而言，头文件的内容与程序体的内容起着相同的作用。一个 C++程序总是从主函数 main()开始执行，每一个 C++程序必须有一个且只允许有一个 main()函数，它是运行程序的起始点。随着程序量的增加，可以把程序划分成若干个函数，每个函数完成一个专门的任务。程序中的各个函数必须有不同的函数名，不能重名。每个函数由函数名和函数体组成，函数体由大括号{}之内的一系列语句组成。由于 C++是面向对象的高级语言，因此程序中也可以包括类的定义和使用。

C++程序结构如下所示：

```
#include<…>       //包含头文件
#include<…>
void main()       //主函数
{…
}
func1(参数列表)   //函数 func1
{…
}
fun2(参数列表)    //函数 func2
{…
}
class myclass     //myclass 类
{…
};
```

1.2.3 C++的字符集

通过以上几个 C++源程序例子可以看出，C++源程序是由一系列规定的字符、C++专用的英语词汇，再按照 C++规定的语法组合而成。C++与其他的语言一样，也是由字符集和规则组成。

字符是语言中不可分解的最基本的语法单位，规则则是 C++所规定的构成 C++词汇、表达式、语句以及构成函数和程序等的行为准则。

C++语言的字符集是 ASCII 码的子集，其中包括：

（1）英文的大写 26 个字母（A～Z），英文的小写 26 个字母（a～z）。

（2）十进制的数字 10 个（0～9）。

（3）标点符号 8 个，它们分别是：

	空格	语句中各成分之间的分隔符
,	逗号	数据之间的分隔符
;	分号	简单语句结束符
'	单引号	字符常量起、止标记符
"	双引号	字符串常量起、止标记符
:	冒号	语句标号结束符或条件运算符
{	左花括号	复合语句的开始标记
}	右花括号	复合语句的结束标记

（4）单字符运算符 19 个，它们分别是：

[	左中括号	与右中括号配对使用，用于数组元素的访问
]	右中括号	与左中括号配对使用，用于数组元素的访问
(	左圆括号	与右圆括号配对使用，用于表达式或函数运算
)	右圆括号	与左圆括号配对使用，用于表达式或函数运算
+	加号或正号	
-	减号或负号	
*	乘号或间接访问运算符	
/	除号	
%	取整余数	
.	小数点或结构成员访问符	
<	小于号或左尖括号	
>	大于号或右尖括号	
=	赋值号	
!	感叹号	逻辑非运算符
~	波折号	按位取反运算符
&		按地址或按位与运算符，也是引用说明符
|	竖线	按位或运算符
^	尖字符	按位异或运算符
?	问号	条件表达式运算符

（5）特殊用途符号 3 个，它们分别是：

#	井号符	预处理命令行的开始标记
\	反斜线	转义字符序列的开始标记
_	下划线	只在标识符中使用

另外，需说明的是，在 C++的字符串中可以使用 ASCII 码中的任何字符，即使是汉字

也可以在C++的字符串中使用。

1.2.4 C++程序的执行

一个C++程序要经过编写、编译、连接和执行等几个环节，如图1.1所示。利用C++集成开发环境可以很方便地实现这些环节。常见的C++集成开发环境有Turbo C++，Borland C++，MS C++，GNU C++以及Visual C++等。

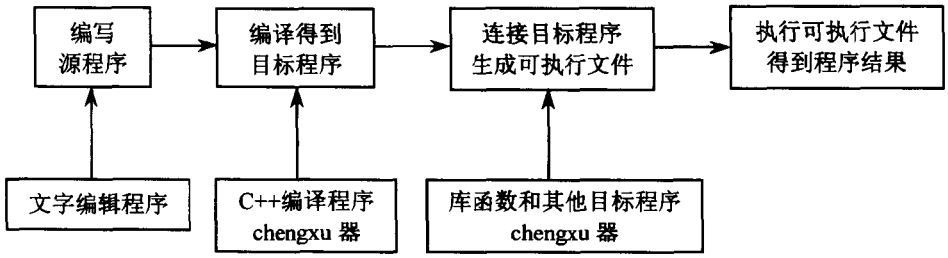

图1.1 编写、执行C++程序的步骤

程序编写完后，要得到C++程序的运行结果，需要使用C++编译程序对程序进行编译，得到目标程序，再把目标程序连接成可执行程序。运行可执行程序，便能得到程序的运行结果。在C++集成开发环境中，只要执行某个菜单命令就可以完成这些工作。也可以自行输入命令完成这些工作。例如，将hello.cpp程序转变成可执行程序hello.exe，不同的开发环境有不同的命令。

（1）在DOS操作系统下使用Turbo C++开发环境，可以输入命令：`tcc hello.cpp`。

（2）在DOS操作系统下使用Borland C++开发环境，可以输入命令：`bcc hello.cpp`或`bc hello.cpp`。

（3）在UNIX（如Linux）操作系统下，可以输入命令：`g++ hello.cpp`或`cc hello.cpp`。

1.3 数据和运算

程序设计离不开数据运算。明确C++对数据和运算的规定是程序设计的基础。本节介绍C++对标识符、保留字、数据类型、变量、常量、运算符和表达式等的具体规定。

1.3.1 标识符

程序中需要使用各种对象，如变量、符号常量、用户定义的数据类型、函数等，均需用户定义。每个对象都要有一个名字，这个名字就是标识符。标识符可以由编程者自由指定，但是需要遵循一定的语法规定。C++对标识符的定义有如下规定：

（1）标识符只能由字母、数字和下划线组成，且标识符必须以字母或下划线开头。

（2）标识符严格区分大小写字母，如a1和A1是两个不同的标识符。

（3）不能用C++的保留字作标识符。保留字（关键字）是已被C++占用的标识符，

它们有专门的意义和用途，如 main，public，new 等。C++的关键字如表 1.1 所列。

表 1.1　C++的关键字

asm	auto	break	case	catch	char	class	const	continue
default	delete	do	double	else	enum	extern	float	for
friend	goto	if	inline	int	long	new	operator	private
protected	public	register	return	short	signed	sizeof	static	struct
switch	template	this	throw	try	typedef	union	unsigned	virtual
void	volatile	while						

（4）应当使标识符能够具有一定的含义。为便于记忆和阅读方便，最好使用英文或汉语拼音来表示它代表的对象，如用标识符 myphone 表示电话号码。表 1.2 列出了一些合法和不合法的标识符。

表 1.2　合法与不合法标识符举例

合法标识符	非法标识符	错误原因
myclass2	2myclass	数字开头
myphone	public	用保留字作标识符
_name	-name	含非法字符-

1.3.2　数据类型

数据类型是指程序中能够表示和处理哪些类型的数据。表 1.3 列出了 C++提供的数据类型，包括数据类型的名称、占用的空间、取值范围和缺省值等。

表 1.3　数据类型

数据类型	关键字	占用字节数	取值范围
字符型	char	1	-128 ~ +127 内的整数
有符号字符型	char，signed char	1	-128 ~ +127 内的整数
无符号字符型	unsigned char	1	0~255 内的整数
短整型	short，short int	2	-32768~32767，$-2^{15} \sim 2^{15}-1$ 内的整数
有符号短整型	short，short int，signed short int	2	-32768~32767，$-2^{15} \sim 2^{15}-1$ 内的整数
无符号短整型	unsigned short unsigned sort int	2	0~65535，　$0 \sim 2^{16}-1$ 内的整数
整型	int	4	-2147483648~2147483647
有符号整型	int，signed int	4	-2147483648~2147483647
无符号整型	unsigned，unsigned int	4	0~4294967295
长整型	long	4	-2147483648~2147483647
有符号长整型	long int，signed long int	4	-2147483648~2147483647

续表

数据类型	关键字	占用字节数	取值范围
无符号长整型	unsigned long，unsigned long int	4	0~4294967295
单精度数	float	4	$-3.4\times10^{\pm38}\sim3.4\times10^{\pm38}$
双精度数	double	8	$-1.7\times10^{\pm308}\sim1.7\times10^{\pm308}$
长双精度数	long double	10	$-3.4\times10^{\pm4932}\sim1.1\times10^{\pm4932}$

【程序 1.5】用 sizeof()运算符测试 C++各种数据类型占用的字节数。

```
#include<iostream.h>
void main()
{ cout<<"The size of an int is:"<<sizeof(int)<<"bytes.\n";
  cout<<"The size of a short int is: "<<sizeof(short)<<"bytes.\n";
  cout<<"The size of a long int is: "<<sizeof(long)<<"bytes.\n";
  cout<<"The size of a char is: "<<sizeof(char)<<"bytes.\n";
  cout<<"The size of a float is: "<<sizeof(float)<<"bytes.\n";
  cout<<"The size of a double is: "<<sizeof(double)<<"bytes.\n";
}
```

程序输出结果：

```
The size of an int is: 2bytes.
The size of a short int is: 2bytes.
The size of a long int is: 4bytes.
The size of a char is: 1bytes.
The size of a float is: 4bytes.
The size of a double is: 8bytes.
```

1.3.3 常量

程序中涉及的数据可以用常量和变量表示。常量表示值不变的量，变量表示值可以变化的量。常量有不同的类型，如 12、-23 为整型常量，12.3 为浮点常量，'A'为字符常量，"ABCD"为字符串常量。另外还可以用符号表示常量。

1. 整型常量

整型常量可以采用十进制、八进制和十六进制三种形式表示。如 789，-789 是十进制整数，十进制整数的第一位不能为 0。八进制整数以数字 0 开头，如 012 表示十进制整数 10，-017 表示十进制整数-15。十六进制整数以 0x 或 OX 开头（数字 0，字母 x），如 0x64 表示十进制整数 100，-OX2A 表示十进制整数-42。

在整型常量末尾添加一个大写字母 L 或小写字母 l，表示长整型数，如 123L、-123L，027L。在整型常量末尾添加字母 U 或 u 表示无符号整型数，如 123U。

2. 实型常量

实型常量表示带小数的数值常量。实型常量又称为浮点常量或实数，只有十进制形式，

例如 0.123，12.3，-123.0 和-12.3 等都是实型常量。

实型常量还可以用科学计数法（指数形式）表示，如 1.23e2 或 1.23E2，它们代表 1.23×10^2，再如-1.23e2 或-1.23E2，它们代表-1.23×10^2。

3. 字符常量

字符常量用于表示单个字符，要求用单引号将字符括起来，如'A'，'a'，'2'。可以直接在单引号中写出某个字符，或是用转义字符代表一些特殊字符，如回车、换行等，还可以用字符的编码表示字符，字符编码要写成三位八进制数形式或两位十六进制数形式，如八进制数'\101'和十六进制数'\x41'都代表字符 A。

表 1.4 列出了 C++转义字符的表示方式和代表的含义。所有的转义字符都以反斜线 \ 开始。

表 1.4　转义字符

转义字符	含义	对应 ASCII 码
\b	退格	\x08
\t	水平制表符 tab	\x09
\n	换行	\x0a
\f	换页	\x0c
\r	回车	\x0d
\"	双引号	\x22
\'	单引号	\x27
\\	反斜线	\x5c
\ddd	用 3 位 8 进制数表示字符	
\xff	用 2 位 16 进制数表示字符	

4. 字符串常量

字符串常量是用双引号括起来的字符序列，如"Hello"，"How are you ?"等。字符串中可以包含转义字符，如"C:\\MY.CPP"。可以用连接符（+）将多个字符串连接起来，组成更长的字符串，如"ABC"+"DE"，"ABC"+"DE"+"FG"等。

5. 符号常量

符号常量用一个标识符表示数据。定义符号常量常用两种方法，一种是用#define 定义，另外一种是用关键字 const 定义。

（1）用#define 定义常量。用#define 定义常量的格式如下：

#define 常量名　字符串

#define 的作用是通知编译预处理程序，将程序中出现的常量名替换为对应的字符串。

例如，下面定义的含义是，用符号 students 表示字符串 36。编译预处理程序会把程序中出现的符号 students 替换为字符串 36。这里，36 是字符串，不要理解为数值。

```
#define students 36
```

（2）用关键字 const 定义常量。在 C++中，用关键字 const 定义常量是一种更新、更