

21 世纪高等学校计算机科学与技术规划教材

C++

程序设计教程

C++ Chengxu Sheji Jiaocheng

主编 方超昆



北京邮电大学出版社
www.buptpress.com

21 世纪高等学校计算机科学与技术规划教材

C++ 程序设计教程

主 编 方超昆

副主编 刘 蔚 邓安远 黄湘霞



北京邮电大学出版社
www.buptpress.com

内容简介

C++语言功能强大,使用灵活、方便,既适用于设计系统程序,也适用于开发应用程序。本书深入浅出地向读者介绍了C++语言的基础知识和程序设计要领,所配备的130多道例题可以帮助读者理解C++的运用方法,配备的近300道习题可以检验读者的学习效果,帮助读者更好地掌握C++语言程序设计方法。为方便读者进一步检验学习效果,光盘中收录了作者精心设计的具有智能化阅卷能力的C++语言自助测试系统及C++试题库,该系统集自助测试及自动阅卷于一体,是读者学习C++语言的良师益友。

本书可作为高等院校计算机科学与技术专业及其他非计算机专业的C++语言程序设计教材,也可作为其他各级各类学校的C++语言程序设计教材或参考书。

图书在版编目(CIP)数据

C++程序设计教程/方超昆主编. —北京:北京邮电大学出版社,2008

ISBN 978-7-5635-1709-1

I. C… II. 方… III. C语言—程序设计—教材 IV. TP312

中国版本图书馆CIP数据核字(2008)第176295号

书 名	C++程序设计教程
主 编	方超昆
责任编辑	沙一飞
出版发行	北京邮电大学出版社
社 址	北京市海淀区西土城路10号(100876)
电话传真	010-62282185(发行部) 010-62283578(传真)
电子信箱	ctrd@buptpress.com
经 销	各地新华书店
印 刷	北京忠信诚胶印厂
开 本	787mm×1 092mm 1/16
印 张	18.75
字 数	430千字
版 次	2009年1月第1版 2009年1月第1次印刷

ISBN 978-7-5635-1709-1

定价:32.00元

如有质量问题请与发行部联系

版权所有 侵权必究

前言

C++语言是一种人性化的语言,它尽量为程序员所想,简化程序的书写方式。例如,它的单行注释、赋值语句特殊格式、复合语句表示方式等都有利于程序书写的简化。它的函数参数默认值和函数模板特性更是把函数的功能发挥得淋漓尽致,使得对函数功能的扩充丝毫不影响已有程序的正确运行,不需要对原来调用这些函数的程序进行任何修改。

程序员使用 C++语言可以最大限度地发挥计算机硬件系统的功能,对于同一个问题,用 C++语言所设计的程序的运行效率接近于用汇编语言设计的程序的运行效率。C++语言是程序员的首选语言,是目前各种计算机语言的领潮者。

本书作者长期从事 C++语言及 VC++程序设计语言的教学工作,积累了丰富的教学经验,对 C++语言的精髓有深刻的领悟。本书深入浅出地向读者介绍了 C++语言的基础知识及程序设计要领,所配备的 130 多道例题可以帮助读者理解 C++的运用方法。本书中设计了近 300 道习题,可以检验读者的学习效果,帮助读者更好地掌握 C++程序设计方法。本书中的习题题型丰富、内容涵盖面宽广,其中,包含 130 道单项选择题,30 道程序运行结果分析题,32 道程序填空题和 83 道程序设计题。所有例题和习题的源程序都在 VC++ 6.0 环境下调试通过。

本书写作方法新颖、独特。一般,每道例题都有设计方法分析和程序说明,并对一些需要重点关注的问题以思考题形式启迪读者思考。内容编排体系也有独到之处,没有将模板、运算符等独立成章,而是根据学习知识的需要,将它们分解到各章之中。因此,读者可以按照顺序阅读本书。

本书可作为未学习过任何计算机语言读者的入门教科书,已经学习过程序设计语言的读者也可以从本书丰富的例题和习题中学习程序设计技术与技巧。

本书共 11 章。第 1 章介绍 3 种程序设计结构中的第 1 种结构即顺序结构程序设计方法,本章重点介绍了 C++语言的基础知识及数据输入、输出方法。第 2 章介绍第 2 种程序设计结构即选择结构程序设计方法,本章重点介绍了条件语句及开关语句的语法结构,为下一章的学习打下基础。第 3 章介绍第 3 种程序设计结构即循环结构程序设计方法,本章重点介绍了 while 循环及 for 循环语句的使用方法。第 4 章介绍数组的基础知识及其应用,其中介绍了 3 种排序算法及二分查找算法等几种典型的常用算法。第 5 章介绍函数和指针的应用,它是本书的核心内容之一。本章重点介绍了函数的定义与调用关系、全局变量与局部变量关系以及指针与引用的应用。第 6 章介绍了结构体、联合体及枚举等各种构造数据类型的基础知识及应用方法。第 7 章介绍了位运算和预处理命令。位运算是 C 系列语言所独有的特性,C++对二进制位的处理能力使得 C++的应用无所不能。第 8 章介绍了运算符重载特性,运算符重载进一步简化了程序设计过程。第 9 章介绍了类和对象的基本概念及其程序设计方法。类和对象是 C++的核心内容,是面向对象语言区别于结构化语言的重要特征,本章花了较大的篇幅来介绍类和对象的程序设计方法。第 10 章介绍了文件 I/O 流的概念,重点介绍了文件的存取

方法。第11章介绍了VC++程序设计基本方法,重点介绍了基于MFC的对话框应用程序设计方法,为今后进一步学习VC++程序设计技术打下良好基础。

本书的配套教材《C++程序设计教程学习与实验指导》给出了教材中的全部习题解答,收录了教材中出现的全部语法,并对VC++集成开发环境的基本操作进行了简要介绍。

为方便读者学习程序设计技术,本书提供了配套光盘。光盘中收录了本书中所有例题的源程序和全部习题的解答,它们全部在VC++ 6.0环境下调试通过。

为了帮助读者检验学习效果,光盘中还收录了作者设计的具有智能化阅卷能力的C++语言自助测试系统,该系统集自助测试及自动阅卷于一体,是读者学习C++语言的良师益友。

本书由方超昆主编,刘蔚、邓安远、黄湘霞任副主编。方超昆编写第2、第3、第9章,并负责全书的修订统稿工作;刘蔚编写第4、第8章;邓安远编写第1章;黄湘霞编写第7、第10章;杨莉萍编写第5章;姚娟编写第11章;林洪编写第6章。

本书编写过程中,北京邮电大学出版社给予了极大的关心和支持,在此表示由衷感谢!

由于作者水平所限,书中错误在所难免,恳请读者提出宝贵意见,作者将不胜感谢。读者可以通过“kcfang@163.com”邮箱地址与作者联系。

编者

2008年9月

目 录

第 1 章 顺序结构程序设计	1
1.1 C++语言简介	1
1.2 C++语言的数据类型	1
1.2.1 变量	2
1.2.2 常量	3
1.3 数据类型转换	5
1.3.1 隐式转换	5
1.3.2 显式转换	6
1.4 算术表达式	7
1.4.1 算术运算符	7
1.4.2 常用数学运算函数	8
1.4.3 算术表达式	10
1.5 程序结构	11
1.6 顺序结构	12
1.6.1 赋值语句和赋值表达式	12
1.6.2 逗号表达式	13
1.6.3 注释语句	13
1.6.4 数据输入和输出	14
1.6.5 数据输出格式控制	18
1.6.6 const 语句	18
1.7 C++语言运算符	19
习题 1	20
第 2 章 选择结构程序设计	23
2.1 关系表达式	23
2.2 逻辑表达式	24
2.3 复合语句与空语句	25
2.4 if 语句	26
2.4.1 单分支结构	26
2.4.2 双分支结构	26
2.4.3 多分支结构	29
2.4.4 条件嵌套语句匹配规则	30
2.4.5 条件表达式	31
2.5 switch 语句	32

习题 2	35
第 3 章 循环结构程序设计	38
3.1 while 语句	38
3.2 do... while 语句	40
3.3 for 语句	41
3.4 循环控制	43
3.4.1 break 语句	43
3.4.2 continue 语句	44
3.5 循环语句嵌套	45
3.6 goto 语句	46
习题 3	50
第 4 章 数组	56
4.1 一维数组	56
4.1.1 数组定义	56
4.1.2 数组元素初始化	57
4.1.3 字符数组和字符串	59
4.2 二维数组	64
4.2.1 数组定义	64
4.2.2 数组元素初始化	65
4.2.3 字符串数组	67
4.3 常用算法	68
4.3.1 冒泡排序算法	68
4.3.2 选择排序算法	71
4.3.3 插入排序算法	73
4.3.4 顺序查找算法	74
4.3.5 二分查找算法	75
习题 4	78
第 5 章 函数和指针	85
5.1 函数基础知识	85
5.1.1 函数定义与原型声明	85
5.1.2 变量作用域与生存期	92
5.2 函数递归	98
5.3 函数默认参数值	102
5.4 函数重载	102
5.5 函数模板	105
5.5.1 函数模板概念	105
5.5.2 函数模板定义	105
5.5.3 函数模板使用	106
5.6 指针与引用	107
5.6.1 指针基础知识	108

5.6.2	指针运算	109
5.6.3	指针与数组	112
5.6.4	指针与函数	119
5.6.5	二级指针	119
5.6.6	引用	121
5.7	动态内存分配	123
5.7.1	new 运算符	123
5.7.2	delete 运算符	123
5.8	const 与指针	125
5.8.1	指向常量的指针	125
5.8.2	指向变量的常量指针	125
5.8.3	指向常量的常量指针	126
	习题 5	126
第 6 章	构造数据类型	132
6.1	结构体	132
6.1.1	结构体类型定义	132
6.1.2	结构体变量定义和初始化	134
6.1.3	结构体成员访问	135
6.1.4	结构体数组	137
6.1.5	结构体参数	140
6.1.6	结构体指针	142
6.2	联合体	145
6.2.1	联合体类型定义	145
6.2.2	联合体变量定义和初始化	146
6.2.3	联合体成员访问	147
6.3	枚举	150
6.3.1	枚举类型定义	150
6.3.2	枚举变量定义	150
6.4	用 typedef 定义类型	152
6.4.1	基本数据类型别名定义	152
6.4.2	构造数据类型别名定义	152
	习题 6	153
第 7 章	预处理和位运算	162
7.1	文件包含	162
7.1.1	固定路径查找	162
7.1.2	当前路径查找	163
7.1.3	指定路径查找	163
7.2	宏	163
7.2.1	无参宏	163
7.2.2	有参宏	165

7.2.3	宏终止	168
7.3	条件编译	168
7.3.1	#ifdef	168
7.3.2	#ifndef	169
7.3.3	#if	169
7.4	位运算	170
7.4.1	按位或运算	171
7.4.2	按位与运算	171
7.4.3	按位异或运算	172
7.4.4	按位取反运算	172
7.4.5	按位左移运算	173
7.4.6	按位右移运算	173
7.4.7	位运算符优先级	173
	习题 7	174
第 8 章	运算符重载	180
8.1	运算符重载基本概念	180
8.2	重载单目运算符	181
8.2.1	重载前缀++、-- 运算符	182
8.2.2	重载后缀++、-- 运算符	183
8.2.3	重载正负号+、- 运算符	185
8.3	重载双目运算符	186
8.3.1	重载四则运算符	186
8.3.2	重载输入运算符	189
8.3.3	重载输出运算符	190
8.3.4	重载关系运算符	191
	习题 8	194
第 9 章	类	197
9.1	类的基础知识	197
9.1.1	类与对象	197
9.1.2	成员访问属性	199
9.1.3	成员访问	199
9.1.4	对象与指针	200
9.1.5	this 指针	202
9.2	构造函数	202
9.2.1	自定义构造函数	202
9.2.2	缺省构造函数	207
9.2.3	拷贝构造函数	208
9.2.4	构造函数初始化表	214
9.3	析构函数	215
9.3.1	缺省析构函数	215

9.3.2	自定义析构函数	215
9.4	对象运算重载	218
9.4.1	重载单目运算符	218
9.4.2	重载双目运算符	218
9.5	常成员、常对象及静态成员	225
9.5.1	常成员与常对象	225
9.5.2	静态成员	227
9.6	类继承	230
9.6.1	类继承基础知识	231
9.6.2	类继承属性	232
9.6.3	基类与继承类对象初始化	235
	习题 9	237
第 10 章	文件	241
10.1	I/O 流	241
10.1.1	输入流	241
10.1.2	输出流	241
10.1.3	标准 I/O 流与文件 I/O 流	243
10.2	文件概述	243
10.2.1	文件概念	243
10.2.2	文件分类	243
10.3	文件流	244
10.3.1	文件流对象	244
10.3.2	文件的打开和关闭	244
10.3.3	顺序文件	247
10.3.4	随机文件	260
	习题 10	263
第 11 章	VC++ 程序设计基础	265
11.1	VC++ 基础知识	265
11.2	对话框常用控件	270
11.2.1	静态控件	270
11.2.2	编辑框控件	271
11.2.3	按钮控件	272
11.2.4	列表控件	278
11.2.5	组合框控件	282
11.3	文档与视图	287
11.3.1	文档与视图概念	287
11.3.2	文档类型	287
	习题 11	289
附录	字符与 ASCII 码对照表	290

第 1 章 顺序结构程序设计

C++语言源程序可以分为 3 种基本结构,即顺序结构、选择结构和循环结构,这 3 种基本结构可以组成各种复杂的程序。本章主要介绍顺序结构程序设计基本方法,为后续章节的学习打下基础。

1.1 C++ 语言简介

C 语言于 1972 年由美国的 Dennis Ritchie 设计,它由早期的 BCPL 语言(Basic Combind Programming Language)发展演变而来。当 C 语言发展到巅峰时,出现了一个名为“C with Class”的 C 语言版本,它是 C++的雏形。

1979 年 4 月,Bjarne 博士等人试图去分析 UNIX 的内核,但当时没有合适的工具来有效地分析由于内核分布而造成的网络流量。同年 10 月,Bjarne 博士完成了一个可以运行的预处理程序——Cpre,它为 C 语言加上了类似 Simula 的类机制。当时,Bjarne 博士开始思考是不是可以开发一种新的语言,贝尔实验室对这个想法很感兴趣,就让 Bjarne 博士组织了一个开发小组专门进行研究。由于当时 C 语言在编程界居于主导地位,要想发展一种新的语言,最强大的竞争对手就是 C 语言。所以,当时最关切的问题是这种新语言要在运行时间、代码紧凑性和数据紧凑性方面能够与 C 语言相媲美,且要尽量避免在语言应用领域的限制。在这种情况下,一个很自然的想法就是让“C++”从 C 语言继承过来,而 Bjarne 博士更具有先见之明,为了使“C++”成为一种新的当之无愧的语言,他吸收了很多语言的优点,如从 Simula 继承了类的概念,从 Algol68 继承了运算符重载、引用以及在任何地方声明变量的特性,从 BCPL 获得了简捷的双斜杠注释,从 Ada 吸收了模板等。

刚诞生的“C++”和现在的 C++版本有很大的区别。首先,没有 C++编译器,早期的“C++”代码都是先将其转换为 C 语言代码,然后用 C 语言编译器进行编译。其次,那时的“C++”没有类的继承,没有成员访问属性,也没有虚函数的概念,并且这种改良的新语言还没有一个合适的名字。随着“C++”语言功能的不断完善,C 语言标准委员会为这个版本的 C 语言取了一个响亮的名字——C++,以 C++语言中的“++”运算符来体现它是 C 语言的进步,并随之成立了 C++标准委员会。

1.2 C++ 语言的数据类型

和数学一样,计算机程序中要处理的数据也有类型之分。例如,人的年龄和体重应当用数值来表示,在 C++中称为数值型数据。其中,年龄是整数,称为整型数据;体重可以包含小数,

称为实数型数据。而人的姓名和性别是文字,在 C++中称为字符型数据。

C++语言提供的基本数据类型包括整型、单精度型、双精度型、字符型和逻辑型。其中,整型又分为短整型、长整型。单精度型、双精度型统称为实数型。另外,数据还有无符号和有符号之分。

1.2.1 变量

在程序中,使用变量来保存数据,在程序运行过程中可以随时改变变量的值。

1. 变量命名规则

每个变量都有一个名称,其命名要遵循以下规则:

- ①变量名由字母、数字和下划线组成。
- ②变量名的第 1 个符号必须是英文字母或者下划线。
- ③变量名中的英文字母区分大小写。

例如,a、Book、book、_Make_Cipher 都是合法的变量名,且 Book 和 book 是两个不同的变量名,而 123A、x+y 都不是合法的变量名。

在 C++语言中,各种名称的命名规则都和变量名的命名规则相同,且统称为标识符。

2. 变量数据类型

变量是用来保存数据的,具有类型之分,可以分为整型变量、字符型变量、指针变量等。C++语言中的常用数据类型及其特性如表 1-1 所示。

表 1-1 C++语言的常用数据类型

数据类型名	数据类型描述	数据类型的长度(字节)	数据取值范围
char	字符型	1	0~255
int	有符号整型	4	-2147483648~2147483647
unsigned int	无符号整型	4	0~4294967295
short int	短整型	2	-32768~32767
long int	长整型	4	-2147483648~2147483647
unsigned long	无符号长整型	4	0~4294967295
float	单精度实数	4	-3.4E+38~3.4E+38
double	双精度实数	8	-1.7E+308~1.7E+308
long double	长双精度实数	10	-3.4E+4932~3.4E+4932
bool	逻辑型(或布尔型)	1	true、false

数据占据内存的大小和所使用的编译器有关。在 Borland C++中一个 int 型数据占用两字节内存空间,但在 Visual C++(以下简称 VC++)中占用 4 字节内存空间。在 VC++中,int 型和 long 型数据的作用完全相同。本书中将以 VC++为背景进行介绍。

3. 变量定义

每个变量在使用前都必须先进行定义,说明其数据类型。定义变量数据类型的语法格式如下:

数据类型符 变量名 1,变量名 2,...,变量名 n;

例如：

```
int height,weight; // 定义身高和体重为整型
char name[20]; // 定义姓名为字符串
```

4. 变量初始化

变量初始化的作用是给变量赋初值,可以在定义变量时直接给它赋值。变量初始化的语法格式如下：

(1)语法格式 1

数据类型符 变量名=初值;

例如：

```
int x,age=20,score=100;
```

(2)语法格式 2

数据类型符 变量名(初值);

例如：

```
int x,age(20),score(100);
```

在上面的定义中,变量 x 未赋初值,而变量 age 和 score 的初值分别为 20 和 100。在程序中,变量未赋值之前不允许使用,即要遵循“先赋值后使用”的原则。

1.2.2 常量

常量的值在程序运行过程中不发生变化,如圆周率 3.14159 就是一个常量。常量也有类型之分,可以分为整型常量、实型常量、字符常量、字符串常量和符号常量。常量所占据的内存空间及取值范围与变量的规定相同。

1. 整型常量

整型常量用来表示整数。整型数据可以用不同进制形式表示,不同的进制有其不同的表示方式,如表 1-2 所示。

表 1-2 整型常量的表示方式

进制	表示方式	示 例
十进制	一般整数的表示方法	0,-22,55
八进制	在八进制整数前加数字 0	00,-072,+0331
十六进制	在十六进制整数前加数字 0 和字母 x	0x0,0x1B5,-0xb3

2. 实型常量

实型常量只有十进制数表示方式,且没有单精度常量,所有带小数点的常量都视为双精度常量。实型常量具有定点数和浮点数两种类别,具体表示方式如表 1-3 所示。

表 1-3 实型常量表示方式

类别	表示方式	示 例
定点数	整数部分. 小数部分	0.0, 1.34, -34.0
浮点数	尾数 E(或 e)指数	3.57E10, -5.6e-9

说明:

①浮点数表示方式相当于数学中的科学计数法。与科学计数法:

$$\text{尾数} \times 10^{\text{指数}}$$

对应的书写格式为:

$$\text{尾数 } e \text{ 指数}$$

或

$$\text{尾数 } E \text{ 指数}$$

②浮点数中的指数部分只能是整型常量,尾数可以是任何实型或整型常量。

例如,定点数 -123.567 用浮点数表示法可以表示为 -123.567e0、-12.3567e1、-1.23567e2、-0.123567e3、-1235.67e-1、-12356.7e-2 等。

3. 字符常量

字符常量用来表示字符,必须把字符常量用两个单引号引起来,如 'A'、'*'、'!' 等。使用字符常量的规则如下:

①空格也是字符,表示为 ' '。

②单引号中必须恰好有一个字符,不能空缺,如 " 是错误的字符常量。

在 C++ 语言中有一种特殊的字符常量,称为转义字符,转义字符用来表示特殊符号或控制代码。常用的转义字符及其作用如表 1-4 所示。

表 1-4 常用转义字符表

转义字符	意义	转义字符	意义
\n	回车、换行符	\a	响铃
\t	水平制表符	\"	双引号
\v	垂直制表符	\'	单引号
\b	左退一格	\\	反斜杠
\r	回车符	\ddd	1~3 位八进制数 ddd 对应的字符
\f	换页符	\xhh	1~2 位十六进制数 hh 对应的字符

注意:如果字符本身就是一个单引号,不可表示为 ''。因为 ' 是字符的界定符,要把它的作用临时改为一个字符,必须使用转义字符,应当表示为 \'。同样的道理,对于反斜杠字符 \, 不可表示为 \。因为反斜杠 \ 的作用是转义字符,而不是反斜杠字符,所以,要临时把它当作一个反斜杠字符时,应当使用转义字符来恢复它本身的字符,应当表示为 \\。

在 C++ 语言中,每一个字符都有一个唯一的代码,简称为 ASCII 码。字符与其对应的

ASCII 码是等价的,在程序中可以混用它们。例如,对于大写字母 A,在程序中既可以用'A'来表示,也可以用它的 ASCII 码“65”来表示。字符的 ASCII 码对应表可参见附录。

4. 字符串常量

字符串是若干个字符的集合。必须把字符串常量用两个双引号引起来,如"hei!"、"12345"、“a”。

"12345"和 12345 是两个不同类型的常量,前者是一个字符串,后者是一个整数。

字符串可以不含任何字符,称为空串,表示为""。

如果字符串中含有双引号字符,不能直接表示为",应当使用转义字符\"。

字符串中所含的字符个数称为字符串的长度,如"okabc123"、“a”、“”的长度分别为 8、1、0。计算字符串长度时应注意以下两点:

①对于含有转义字符的字符串,应将转义字符计算为 1 个字符。例如,"abc\\12\\n"的长度为 7,而不是 9;"abc\\\\12\\n"和"abc\\\\123\\n"的长度均为 6。

②在字符串中,反斜杠表示转义字符的开始,如果其后面没有表 1-4 中所列出的转义字符,则该反斜杠被忽略,并不参与长度计算。例如,"\\a"的长度为 1,而"\"是非法的。

5. 符号常量

上面所介绍的常量都是具体数据,在程序中,为了方便对程序的理解以及程序前后的一致性,可以用特定符号来表示一个特殊的常量,一般称这个特定符号为符号常量。

符号常量的语法格式如下:

```
#define 符号常量名 常量
```

例如:

```
#define PAI 3.14159
```

这里,PAI 是一个符号常量,可以在程序中使用 PAI 来代替圆周率 3.14159。

在程序中使用符号常量为修改程序提供了很大的方便。例如,当不需要太高的数据精度时,只需将符号常量的定义修改为:

```
#define PAI 3.14
```

这样可以避免在程序中去修改每一处的圆周率。

1.3 数据类型转换

在 C++ 中,不同类型的数据可以进行混合运算,但在计算机内部运算时,要求参与运算的对象的数据类型相同(运算后得到的运算结果的类型与运算对象也相同)。因此,在运算过程中常常需要对变量或常量的数据类型进行转换。转换的方法有两种,一种是系统自动转换(又称为隐式转换),一种是由程序员在程序中主动转换(又称为显式转换)。

1.3.1 隐式转换

不同类型的数据进行混合运算时,系统会自动对其进行转换。转换规则如下:

①若参与运算的对象的数据类型不同,则先转换成相同的数据类型,然后进行运算。

②将低类型数据转换成高类型数据后进行运算。例如, char 型和 int 型运算时,先把 char 型转换成 int 型后再进行运算。

类型的高低是根据它们所占内存空间的字节数按从大到小的顺序排列的,顺序如下:

double, float, long, int, short int, char

③所有实数运算都是按照双精度进行运算的,即使仅含 float 型数运算的表达式,也要先转换成 double 型,再作运算。

例如:

```
float PAI=3.14;
```

```
int s,r=7;
```

```
s=r*r*PAI;
```

在执行“s=r*r*PAI;”语句时,因为 r 和 r 的类型相同,先直接计算 r*r,其结果为 int 型数 49;再计算 49*PAI,此时,将 49 和 PAI 都转换成 double 型后再进行计算,运算结果也为 double 型,运算结果为 153.86;但由于 s 为 int 型,故应将“=”右边的运算结果转换成 int 型(舍去小数部分),最后得到 s 的值为 153。

1.3.2 显式转换

显式转换是通过类型转换运算来实现的,其语法格式如下:

类型说明符 (表达式)

或

(类型说明符) (表达式)

两种转换格式的功能都是把表达式的运算结果转换成类型说明符所表示的类型。建议尽量使用第 1 种格式。

例如, (float)(a+1) 和 float(a+1) 都是把 a+1 转换为实型; (int)(x+y) 和 int(x+y) 都是把 x+y 的结果转换为整型。

说明:

如果使用第 2 种格式,类型说明符和表达式都必须加圆括号(单个变量或常量可以不加圆括号)。如果把 (int)(x+y) 写成 (int)x+y,则只是把 x 转换成整数之后再与 y 相加。

对一个变量进行类型转换后,变量本身的数据类型和值并不发生改变。

例如:

```
float f=12.8;
```

```
int x;
```

```
x=int(f);
```

将 f 的值转换成整数 12,因此 x=12,而 f 本身的数据类型并未改变,其值仍为 12.8。

1.4 算术表达式

用来对各种类型的数据进行运算的符号称为运算符,运算符是表达式的重要组成部分。C++语言中主要有算术运算符、关系运算符、逻辑运算符和位运算符。

1.4.1 算术运算符

C++语言中的算术运算符的作用类似于数学中的算术运算。算术运算符中运算对象的数据类型、运算意义和结合性如表 1-5 所示。

表 1-5 算术运算符及其运算功能

算术运算符	运算符名称	运算对象个数	运算符意义	对象数据类型	运算符结合性	
+	正	1 个	取原值	整型或实型	自右向左	
-	负	1 个	取相反数			
+	加	2 个	加法运算		自左向右	
-	减	2 个	减法运算			
*	乘	2 个	乘法运算			
/	除	2 个	除法运算			
%	求余数	2 个	求余数	整型	自右向左	
++	增 1	1 个	变量增 1	整型或实型		
--	减 1	1 个	变量减 1			

1. 运算符的目

在 C++语言中,参加运算的对象个数称为运算符的目。C++语言中的运算符分为单目、双目和三目 3 种。

(1)单目运算符

只有 1 个运算对象的运算符称为单目运算符。常用的单目运算符有+、-、++、--、&、*等,如-120、+63212、x++。

(2)双目运算符

双目运算符是指参加运算的对象有两个,大多数运算符都是双目运算符,如2+3、7%3。

(3)三目运算符

在 C++语言中,只有一个三目运算符,它是“?:”,将在 2.4.5 节中进行介绍。

2. 运算符优先级

和数学一样,C++语言中的运算符也有优先级,也遵循“先乘、除,后加、减,括号优先”的运算规则。%的优先级与*、/的优先级相同,它们的优先级低于++、--的优先级。

(1)+、-、*

加法、减法、乘法运算符与数学中的运算意义相同。例如,3.5+4.7 结果是 8.2;3.5-4.7