

Visual C++程序设计

——学习指导与练习

主编 祁云嵩 主审 张岳新



东南大学出版社

Visual C++ 程序设计

——学习指导与练习

主 编	祁云嵩		
主 审	张岳新		
编 者	吴小俊	王 芳	
	彭新竹	李淑芳	

东南大学出版社

内 容 提 要

本书根据江苏省普通高校非计算机专业学生计算机基础知识和应用能力等级考试二级考试大纲 Visual C++ 考试要求编写而成,主要用于 Visual C++ 程序设计课程的学习参考。全书共分 14 章,每章先给出概念理解,然后给出该章的典型例题讲解,最后给出一定数量的精编习题。所有例题与习题均考虑到读者学习 Visual C++ 程序语言的难点与知识覆盖面,力求使读者在有限的例题与习题中掌握必需的知识点。

本书概念清晰,题量少而精,大部分例题与习题主要根据笔者多年的教学经验、学生学习及等级考试中遇到的问题精心设计而成,其题型均参照江苏省普通高校非计算机专业学生计算机等级考试要求编写而成,因而是一本关于 Visual C++ 程序语言课程的理想的学习参考及等级考试复习用书。

图书在版编目(CIP)数据

Visual C++ 程序设计——学习指导与练习/祁云嵩主编.

—南京:东南大学出版社,2003.4

ISBN 7-81089-226-6

I. V... II. 祁... III. C 语言—程序设计—高等学校—教学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字(2002)第 109260 号

东南大学出版社出版发行

(南京四牌楼 2 号 邮编 210096)

出版人:宋增民

江苏省新华书店经销 扬中邮电印刷厂印刷

开本:787mm×1092mm 1/16 印张:18 字数:450 千字

2003 年 4 月第 1 版 2003 年 4 月第 1 次印刷

印数:1—4000 册 定价:29.00 元

(凡因印装质量问题,可直接向发行科调换。电话:025-3795801)

前 言

VC++ 程序设计语言是 C 语言的扩充,它既具有对传统 C 语言的兼容性,具备 C 语言的结构化程序设计的特征,又具有独特的面向对象的特征,可以为面向对象的程序设计技术提供全面的支持。面向对象的程序设计技术是目前最热门、最先进,同时也是最实用的软件开发手段,因此,越来越多的高校正在改革计算机教学大纲,直接以 VC++ 语言替代传统的 C 语言作为基础计算机语言,使学生尽快掌握面向对象的程序设计方法。

考虑到目前关于 VC++ 的学习参考书较少,大部分关于 C 语言的学习资料还是停留在面向过程设计的传统 C 语言基础上,我们特地组织和编写了该学习指导书,以期对广大读者提供有益的学习参考。

本书概念上的介绍主要引用张岳新教授所著的《Visual C++ 程序设计》(该书是江苏省普通高校计算机等级考试的指定教材),在内容、顺序上也力求与该书配套。全书共分为 14 章,前 8 章为 VC++ 基础,后 6 章为面向对象的程序设计知识。大部分章节例题与习题尽量做到少而精,同时,考虑到其中第 4 章是学生进行完整的程序设计的基础,所以配备了较丰富的习题以使学生尽快熟练掌握 VC++ 程序设计语言的基本语法。

本书由祁云嵩负责全书内容的编写和统稿工作。吴小俊编写第 1、3、9 章;李淑芳编写第 4、6、12 章习题;王芳编写第 7、8 章的例题与习题;彭新竹编写第 1、3、5 章习题。全书最后由张岳新教授主审定稿。

本书在编写过程中得到了华东船舶工业学院教材服务中心主任张莉老师的大力支持以及华东船舶工业学院电子与信息系计算机基础教研室全体老师的热心帮助,在此一并致以衷心的感谢!

由于作者水平所限,书中难免存在缺点和不足之处,敬请广大读者指正。

编者(mailqys@xinhuanet.com)

2002 年 10 月

目 录

第 1 章 VC++ 概述	(1)	5.2 作用域和存储类	(55)
1.1 简单的 VC++ 程序介绍	(1)	5.3 编译预处理	(58)
1.2 VC++ 程序的编辑与运行	(2)	5.4 程序的多文件组织	(60)
1.3 例题解析	(4)	5.5 例题解析	(60)
习题 1	(5)	习题 5	(70)
第 2 章 数据类型、运算符和表达式	(7)	第 6 章 数组	(82)
2.1 VC++ 的数据类型	(7)	6.1 数组的定义及应用	(82)
2.2 变量	(7)	6.2 字符型数组与字符串	(86)
2.3 基本数据类型	(7)	6.3 字符串处理函数	(87)
2.4 运算符	(9)	6.4 例题解析	(89)
2.5 运算符的优先级与结合性	(11)	习题 6	(94)
2.6 不同类型数据的混合运算	(12)	第 7 章 结构体、共同体和枚举类型	(111)
2.7 例题解析	(13)	7.1 结构体	(111)
习题 2	(15)	7.2 枚举类型	(113)
第 3 章 简单的输入/输出	(18)	7.3 例题解析	(114)
3.1 cin 输入流	(18)	习题 7	(119)
3.2 cout 输出流	(19)	第 8 章 指针和引用	(123)
习题 3	(20)	8.1 指针和指针变量	(123)
第 4 章 VC++ 的流程控制语句	(22)	8.2 指针和数组	(124)
4.1 VC++ 语句概述	(22)	8.3 指针数组和指向指针的指针变量	(126)
4.2 选择结构语句	(22)	8.4 指针和函数	(126)
4.3 循环结构语句	(24)	8.5 new 和 delete 运算符	(129)
4.4 控制执行顺序的语句	(25)	8.6 引用和 const 类型变量	(130)
4.5 例题解析	(26)	8.7 简单链表及其应用	(132)
习题 4	(31)	8.8 类型定义	(136)
第 5 章 函数和编译预处理	(51)	8.9 例题解析	(139)
5.1 函数的定义和调用	(51)	习题 8	(151)
		第 9 章 类和对象	(164)
		9.1 类与对象的概念	(164)
		9.2 类	(164)

9.3 对象	(166)	第 12 章 类的其他特性	(223)
9.4 成员函数的重载与缺省参数	(168)	12.1 友元函数	(223)
9.5 this 指针	(168)	12.2 虚函数	(226)
9.6 例题解析	(169)	12.3 静态成员	(229)
习题 9	(172)	12.4 例题解析	(231)
第 10 章 构造函数和析构函数	(175)	习题 12	(244)
10.1 构造函数	(175)	第 13 章 运算符重载	(255)
10.2 析构函数	(179)	13.1 运算符重载	(255)
10.3 例题解析	(179)	13.2 几种典型的运算符重载	(258)
习题 10	(189)	13.3 字符串类	(262)
第 11 章 继承和派生类	(197)	13.4 例题解析	(263)
11.1 继承	(197)	习题 13	(267)
11.2 初始化基类成员和对象成员	(199)	第 14 章 VC++ 的文件操作	(272)
11.3 冲突、支配规则和赋值兼容性	(201)	14.1 文件的打开操作	(272)
11.4 虚基类	(204)	14.2 文件的读/写操作	(273)
11.5 例题解析	(205)	14.3 文件关闭操作	(274)
习题 11	(213)	14.4 例题解析	(275)
参考文献	(280)	习题 14	(279)

第 1 章 VC++ 概述

1.1 简单的 VC++ 程序介绍

以下是一个简单的 VC++ 程序,从中可以理解 VC++ 程序的基本结构及其特点。

//一个简单的 VC++ 程序

```
# include <iostream.h>
```

```
void main (void)
```

```
{
```

```
    cout << "我的第一个 VC++ 程序! \n";
```

```
}
```

本程序的运行结果是在屏幕上显示如下一行信息:

我的第一个 VC++ 程序!

下面从本例题出发,来了解一下计算机程序以及 VC++ 的相关知识。

1. 计算机程序及其执行过程

程序是根据一定的语法规则输入计算机的一系列操作指令(程序语句)。计算机执行时依次从主函数的第一条语句开始读入执行,直到最后一条语句,然后正常退出,其间如果有转向语句,程序将改变其执行的自然顺序。

2. VC++ 的程序结构

函数是 VC++ 的基本组成部分。任一 VC++ 程序均由一个或多个函数组成,其中必须有一个主函数 main(),其余函数可有可无。程序的执行就是从主函数的第一条语句开始执行,在主程序中可以调用其他函数。

任一函数的函数体部分必须用一对“{}”括起来。

3. 包含文件或编译预处理指令

“#”开头的行称为编译预处理指令。一个 VC++ 程序如果要进行输入/输出,必须要包含文件“iostream.h”。在这个系统提供的文件中定义了要用到的输入/输出函数。

4. 注解或说明信息

注解或说明信息只是为了阅读程序的方便,程序并不执行它。

注解或说明的方法有两种:第一种是在注解行前面加上双斜杠“//”;第二种方法是将注解信息置于“/*”和“*/”之间。

5. 一个完整的程序语句必须以分号“;”结束

如果遇到编译预处理命令及注解信息不属于 VC++ 的执行语句,没必要以“;”结束;函数包含语句,但函数本身不必以分号结束。

6. 程序的书写规则

从理论上讲,VC++ 的程序书写格式非常自由。一个语句可写成若干行,一行内也可以写若干语句,但为了程序的阅读和交流,程序的书写应符合以下规则:

(1) 对齐规则:同一层次的语句应从同一列开始。

(2) 缩进规则:属于内一层次的语句,应缩进几个字符。

(3) 任一函数的定义均从第一列开始书写。

(4) 在 VC++ 中,字母的大小写是严格区分的。如“A”和“a”表示不同的字母。这一点,在用到 c,k,o,p,s,u,v,w,x,z 等大小写区别不明显的字母时应特别注意。

1.2 VC++ 程序的编辑与运行

前面书写的程序,计算机并不能直接执行,以字符方式书写的源程序还必须经过编译系统转换成计算机能识别的机器代码文件,即目标文件,然后再将这些目标文件与系统函数等进行连接,产生一个可执行文件方可在计算机上运行。不过,程序的编译及连接均可由 VC 集成环境自动完成。程序编写者所需做的是编辑完成其所需功能的源程序。VC++ 源程序的文件的缺省扩展名为 .cpp。在 VC 集成环境下,如果一个源程序的扩展名为 .c,系统将认为它是一个 C 源程序,VC++ 是对 C 的扩展。

VC++ 程序具体上机过程如下:

(1) 在操作系统环境下启动 VC++ 集成环境(Microsoft Visual C++ 6.0,其中 6.0 为版本号,也可为其他版本),则产生如图 1.1 所示界面。

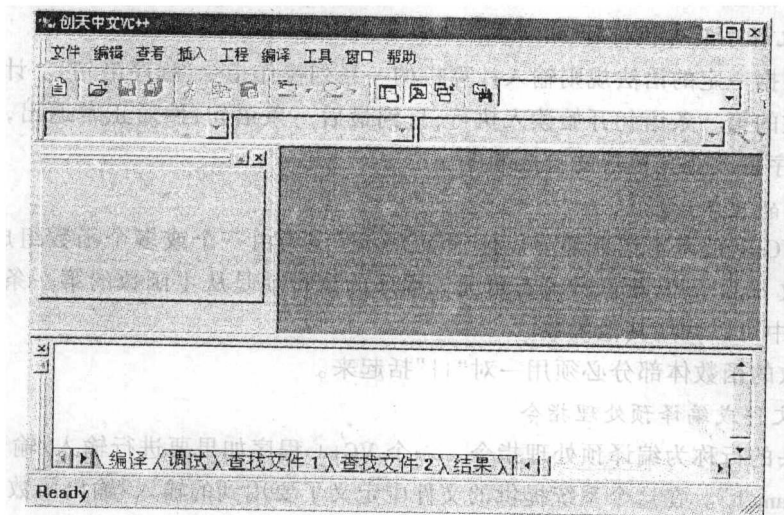


图 1.1 VC++ 集成环境界面

(2) 选择“文件”菜单下的“新建”命令,出现如图 1.2 所示界面(不可直接按“新建”按钮,此按钮是新建一个文本文件)。此界面缺省标签是要为新程序设定工程项目,但对初学者来说,编辑小的源程序不必建立项目,可以直接选择其左上角的“文件”标签,产生如图 1.3 所示界面。

(3) 在图 1.3 所示的界面中左边选定文件类型为“VC++ Source File”,右边填好文件名并选定文件存放目录,然后单击“确定”按钮,出现如图 1.4 所示编程界面,开始输入程序。

(4) 输入完源程序后,按“编译”菜单下的编译命令,对源程序进行编译。系统将在下方的窗口中显示编译信息。如果无此窗口,可按“Alt + 2”键或执行“查看”菜单下的“输出”命令。

如果源程序有语法错误,系统将显示出错误所在的行数并给出提示信息,此时可双击相应的错误提示,光标将自动移至错误所在的行,但这仅仅表示错误可能由这一行引起,具体错误在哪里可根据系统提示进行判断。一般从第一个错误开始修改。最好是每修改一处错误就编译一次。

如果编译后已无提示错误,则可按“编译”菜单下的“构件”命令来生成相应的可执行文件,随后可按“编译”菜单下的“执行”命令运行程序。

(5) 如果要想编辑第二个源程序,则应该通过“文件”菜单下的“关闭工作区”命令来关闭当前工作区(注意:文件菜单下的“结束”命令只能关闭正在编辑的文件而不能关闭当前工作区),然后重复以上步骤。否则,即使关闭了原来的文件,新编辑的源程序还将与原来的程序相互影响而难以编译运行。

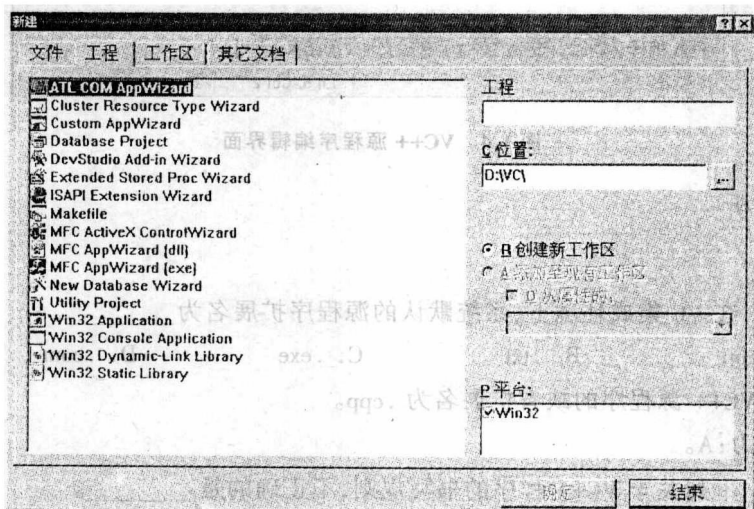


图 1.2 新建 VC++ 工程项目界面

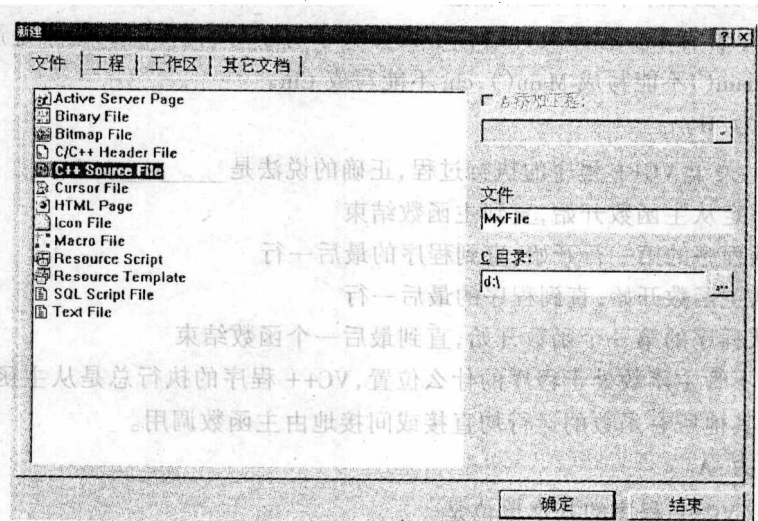


图 1.3 新建 VC++ 源程序文件界面

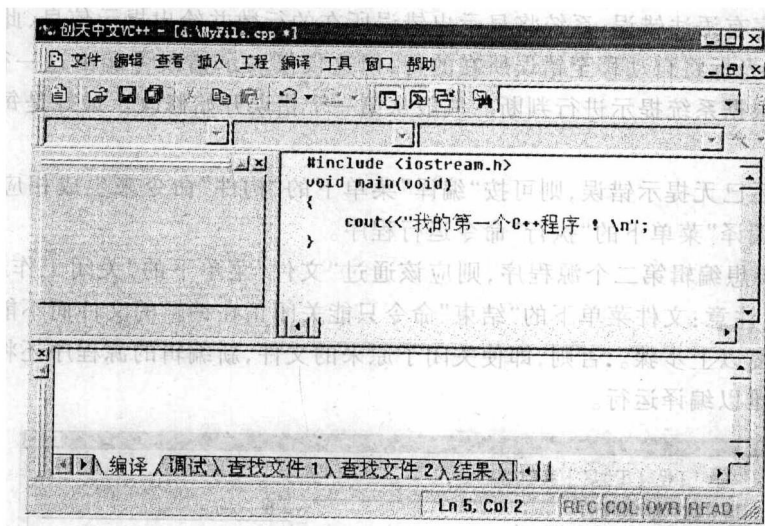


图 1.4 VC++ 源程序编辑界面

1.3 例题解析

1.3.1 选择题

【例 1.1】 在 VC 集成环境下,系统默认的源程序扩展名为_____。

- A. .cpp B. .txt C. .exe D. .com

【解答】 VC++ 源程序的缺省扩展名为 .cpp。

正确选项为:A。

【例 1.2】 下列关于 VC++ 程序的书写规则,不正确的是_____。

- A. 一行可以写若干条语句 B. 一条语句可以写成若干行
C. 可以在程序中插入注释信息 D. VC++ 程序不区分大小写字母

【解答】 尽管有不少程序设计语言不区分大小写字母,但 VC++ 程序是严格区分大小写字母的,如 main()不能写成 Main(),cin 不能写成 Cin。

正确选项为:D。

【例 1.3】 关于 VC++ 程序的执行过程,正确的说法是_____。

- A. 一定从主函数开始,直到主函数结束
B. 从程序的第一行开始,直到程序的最后一行
C. 从主函数开始,直到程序的最后一行
D. 从程序的第一个函数开始,直到最后一个函数结束

【解答】 不管主函数处于程序的什么位置,VC++ 程序的执行总是从主函数开始,直到主函数结束。其他所有函数的运行均直接或间接地由主函数调用。

正确选项为:A。

【例 1.4】 VC++ 程序的基本单位是_____。

- A. 子程序 B. 函数 C. 语句 D. 行

【解答】 VC++ 程序是由一个或多个函数组成的。

正确选项为:B。

1.3.2 填空题

【例 1.5】 一个 VC++ 语言程序必有一个_____函数。

【解答】 正确答案为:主函数(或 main())。

【例 1.6】 在 VC++ 语言程序中,要进行标准输入/输出,则程序中必须用编译预处理指令将相应的头文件包含进来,该编译预处理指令为_____。

【解答】 正确答案为: # include <iostream.h>。

习 题 1

一、选择题

- 下列不属于 VC++ 程序的特点是_____。
 - 结构化程序
 - 程序执行效率高
 - 程序的可移植性好
 - 语法结构严密
- 下列可用于标识 VC++ 源程序注解的符号为_____。
 - #
 - //
 - ;
 - {}|
- 下列可用于标识编译预处理命令的是_____。
 - #
 - //
 - ||
 - ""
- 一个完整的 VC++ 语言的源程序中_____。
 - 必须有一个主函数
 - 可以有多个主函数
 - 必须有主函数和其他函数
 - 可以没有主函数
- VC++ 程序的语句一行写不下时,可以_____。
 - 用逗号换行
 - 在任意空格处换行
 - 在任意位置均可换行
 - 用分号换行
- 下列 VC++ 语言的源程序中,书写风格较好的是_____。
 - # include <iostream.h>
void main (void)
{cout <<"This is a VC++ program! \n";cout <<"The end! \n";}
 - # include <iostream.h>
void main (void) {cout <<"This is a VC++ program! \n";}
 - # include <iostream.h>
void main (void)
{cout <<"This is a VC++ program! \n";}
 - # include <iostream.h>
void main (void)
{cout <<"This is a VC++ program! \n"; cout <<"The end. \n";}

二、填空题

- VC++ 源程序编辑好后,还必须经过 编译 和 链接 才能得到可执行的文件。
- VC++ 源程序中要进行输入/输出,必须包含头文件_____。
- 完整的 VC++ 源程序中必须包含的主函数名为_____。

4. VC++ 源程序中,函数体应置于_____之内。
5. VC++ 源程序的语句通常以_____结束。
6. VC++ 源程序缺省扩展名为_____,经编译后生成的目标文件扩展名为_____,连接后生成的可执行文件扩展名为_____。
7. VC++ 程序编译环境除了可以运行扩展名为 .cpp 的 VC++ 源程序外还可以编译运行_____源程序。
8. VC++ 程序注释的两种书写方式分别是_____和_____。

三、编程题

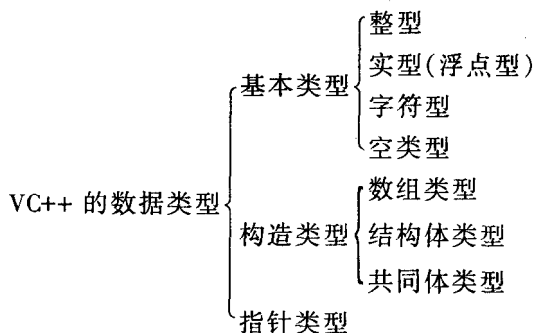
编制一个简单的 VC++ 程序,在屏幕上显示自己的姓名和学号。

第 2 章 数据类型、运算符和表达式

本章内容对初学者来说较为抽象,刚开始同学们可以先了解基本数据类型以及基本运算符的概念及其使用等,对于一些细节问题,可以在对 VC++ 源程序较为熟悉之后再进一步理解,但至少应在学习第 5 章之前掌握本章内容。

2.1 VC++ 的数据类型

编制程序的目的是为了处理各种各样的数据,在 VC++ 中,系统将各种各样的数据分成如下几种类型:



2.2 变量

在 VC++ 中,常常用一个“名字”来代表要处理的一些数据,这个“名字”称之为标识符。特别地,当标识符所指代的数据在程序运行过程中会发生值的变化时,称之为变量。当然,标识符除了可指代要处理的数据外,还可以指代函数名、文件名等。

标识符的命名必须符合 VC++ 的规定:

- (1) 标识符只能由字母、数字和下划线 3 种字符组成。
- (2) 标识符必须由字母或下划线开头,不能以数字开头。
- (3) 在 VC++ 中,有些标识符已被系统运用,这些标识符称之为关键字或保留字,在程序中不能另作他用。如 `cin`、`cout`、`void`、`main` 等。
- (4) 推荐使用的标识符命名法:为了增加程序的可阅读性,可采用 Windows 的标准命名法,即标识符字母大小写可混用;一个标识符可由多个英文单词组成(也可由汉语拼音组成);每一个单词的第一个字母均须大写,其余的为小写字母;标识最后可加上若干字符标识其所代表的数据类型。

此外,VC++ 还要求对用到的变量作强制定义,即“先定义,后使用”。

2.3 基本数据类型

2.3.1 整型数据

1. 整型数据的分类

整型数据可分为基本型、短整型、长整型和无符号型 4 种,根据所要表示的数据值的范围来确定选择哪种具体类型。

基本型:以 int 表示。

短整型:以 short int 表示或 short 表示。

长整型:以 long int 表示或 long 表示。

无符号型:在其他类型前加以 unsigned 表示,存储单元中的全部二进制位用作有效数本身,不留符号位,因而其可表示的数值范围多一个二进制位,但不可表示负数。如整型变量在内存中占 4 个字节,则其可表示的整数的范围为 $-2^{31} \sim 2^{31} - 1$,而 unsigned int 类型可表示的整数的范围为 $0 \sim 2^{32}$ 。

2. 各种整型数据可表示数的大小

各种整型数据可表示数的大小,根据其在内存中所占的字节数及其符号位来确定。VC++ 的各种数据类型在内存中所占字节数如表 2.1 所示。

表 2.1 各种数据类型所占字节数

类 型	字节数
char, unsigned char, signed char	1
short, unsigned short	2
int, unsigned int	4
long, unsigned long	4
float	4
double	8
long double	8

注:字节数可能随所使用的机器的字长不同而不同。

3. 整型变量的定义

VC++ 是一种强类型数据程序语言,所有的变量都必须先指定其所表示的数据类型(先定义)后再使用。如:

```
int a;
```

```
long b;
```

```
short c;
```

```
unsigned long d, e;
```

同一变量说明语句可以说明几个变量,但各变量之间必须以逗号隔开,整个说明语句以分号结束。

2.3.2 实型数据

1. 实型数据又称浮点数,实数有两种表示形式

(1) 十进制小数形式,它由数字和小数点组成,如果是一个纯小数,小数点前面的零可以省略,如果要指定一个整数为实型数据形式,则小数点不可省略,如 9.0。

(2) 指数形式,即所说的科学记数法,它以 10 的多少次方表示,如 1.23E45 表示 1.23×10^{45} ,其中 E 可以大写,也可以小写,但 E 的前面必须有数字,且 E 的后面应为一个整数。

2. 实型数据的舍入误差

实型数据是用有限的存储单元存储的,因此提供的实数的位数也是有限的,超出计算机所能表示的位数以外的数字将被舍去,由此可能会产生一些误差。所以实型数据在计算机中的表示是近似的。

2.3.3 字符数据

1. 字符

字符型数据只能表示一个字符,它可以把一个字符用单引号括起来表示,也可以用转义序列表示,即用一单引号将一个反斜杠“\”加一个控制字符或一个 ASCII 码值来表示,其中的 ASCII 码值必须是八进制或十六进制整数,其取值范围(转换成十进制)在 0~255 之间,八进制数可以以 0 开头,也可以不以 0 开头,而十六进制数必须以 0x 或 0X 开头。例如: 'A', '\101' 都是合法的字符,都表示字母 'A'。但用双引号括起来的 "A" 不是字符,而是字符串。

2. 字符串

将一个或多个字符用双引号括起来就成为字符串,在 VC++ 中,字符串类型的变量必须用字符型数组或指针才能表示。

当双引号要作为一个字符串中的一个字符时,必须采用转义序列表示法,如 "This is a \"string\"".

2.4 运算符

2.4.1 算术运算符

VC++ 的算术运算符有 + (正号)、- (负号)、+、-、*、/、%。关于其运算功能,需注意以下几点:

- (1) + (正号)通常可以省略。
- (2) * (乘法运算)不能以 × 表示,并且两个变量相乘,乘号不能省略。
- (3) 关于 / (除法运算),若两边的操作数均为整数时,则作整数运算(其结果只取运算结果的整数部分,没有四舍五入的概念),否则,其结果保留小数。如 5/2 结果为 2,而 5.0/2 或 5/2.0 或 5.0/2.0 结果为 2.5。
- (4) % (求模运算符)两边的操作数必须为整数,其结果是两数相除后的余数。
- (5) 如果运算的结果超过了 VC++ 的基本数据类型所能表示的范围,则其结果将不正确,系统也不报错,这种情况一般出现在加法、减法和乘法运算中。

2.4.2 关系运算符

VC++ 的关系运算符有 <、<=、>、>=、==、!=、=。

- (1) 关系运算符的“==”不同于赋值运算符的“=”,使用时必须注意。
- (2) 关系运算符的结果为一整数:当关系成立时,结果为 1;关系不成立时,结果为 0,运算的结果可以参加其他运算。
- (3) 对于数学上的区间表达式,如 $a < x < b$,应分别限定 x 的关系为 $a < x$ 且 $x < b$ (VC++ 中表示为 " $a < x \&\& x < b$ ").
- (4) 由于实数在计算机内的存储是近似的,因而比较两个实数是否相等时,不能采用精确比较,而应判断其差的绝对值是否小于一个足够小的数。

2.4.3 自增自减运算符

自增、自减运算符分别为:

- ++ 自增运算符,使变量的值加 1。
- 自减运算符,使变量的值减 1。

这两个运算符都只有一个操作数,称为单目运算符。但其运算对象仅限于变量,不能对一个常量或表达式进行自增或自减运算。

自增运算符和自减运算符可以放在变量前面(前置),也可以放在变量的后边(后置)。如果在一个表达式中自增(减)运算符前置,则表示变量先自增(减)然后再将变量的值参加下一步运算。例如:

```
int k = 1, s;
s = ++k + ++k + ++k + ++k;
cout << s << ' \ t' << k << endl;
```

在求 s 的值时,先对第一个加号两边的 k 自增(两次)后 k 的值为 3,然后作 $3 + 3 = 6$ 的运算,再对第二个加号两边的 k 自增,但前面的 k 已自增了,故只对后面的 k 自增,此时 k 是在 3 的基础上自增后值为 4,然后作 $6 + 4 = 10$ 的运算,再对最后的加号作类似运算, k 在 4 的基础上自增后值为 5,最终表达式的结果为 $10 + 5 = 15$, k 的值为 5。

如果自增(减)运算符后置,则先将变量的值参加表达式的运算,然后再对变量自增。例如:

```
int k = 1, s;
s = k++ + k++ + k++ + k++;
cout << s << ' \ t' << k << endl;
```

运算结果 s 的值为 4, k 的值为 5。

2.4.4 逻辑运算符

VC++ 的逻辑运算符有三种:

! 逻辑非(取反运算)

&& 逻辑与(并且运算)

|| 逻辑或(或者运算)

逻辑运算的结果只有真和假两种,分别以整数 1 和 0 表示。在 VC++ 中还规定,任何操作数,当其值不为 0 时,若将此操作数参加逻辑运算,均视作逻辑真(即整数 1)处理,否则,其逻辑值为假。表 2.2 给出了三种逻辑运算的规则。

表 2.2 逻辑运算表

a	b	! a	! b	a&& b	a b
0	0	1	1	0	0
0	1	1	0	0	1
1	0	0	1	0	1
1	1	0	0	1	1

关于逻辑运算,需要特别强调的是:在 VC++ 中,在求逻辑表达式的结果的过程中,一旦能确定逻辑表达式的值时,系统将不再对逻辑表达式剩下的部分进行运算。这称之为逻辑表达式的优化。例如:

```
int k = 0, m = 5, n1 = 1, n2 = 2, a, b, c;
a = k++ && k++ && k++ ;
b = 0 && m++;
```

```
c = 0 && n1++ || n2++ ;
```

在求 a 的值的過程中,由于 k++ 是后置自增,先将 k=0 的值参加运算,由于 0 和任何数相与的结果都为 0,系统认为表达式已有确定的结果,因而不再进行后面的自增运算。其结果是 a 的值为 0, k 的值为 1。

而在求 b 的值的過程中,由于 0 和任何数相与其结果均为 0,所以系统不做 m++ 就能得到 b 的值而将 m++ 这一运算忽略。最终结果是 b 的值为 0, m 的值仍为 5。

在求 c 的值时,由于“&&”的优先级比“||”高,所以先作 0&&n1++ 运算, n1++ 被优化,此时表达式转化为 0||n2++, 在此表达式中, n2++ 不能被优化。最终结果是 c 的值为 1, n1 的值为 1, n2 的值为 3。

综上所述,应尽量避免在逻辑表达式中改变变量的值。

2.4.5 复合赋值运算符

复合赋值运算是针对二目运算符而言的。所谓二目运算符,是指参加运算的操作数有两个。与此类似,还有一目运算符和三目运算符。

在赋值运算符之前加上其他二目运算符,就构成了复合赋值运算符。其含义是指复合运算符左边的数与右边的数进行二目运算后将结果赋给左边的操作数。如:

```
a+ = b; 等同于 a = a + b;
```

```
a* = b; 等同于 a = a * b;
```

```
a/ = b; 等同于 a = a / b;
```

2.4.6 三目运算符

三目运算符即条件运算符,其参加运算的是三个操作数(表达式),其语法规则如下:

表达式 1? 表达式 2: 表达式 3;

该语句的功能是:当表达式 1 的值为真时,结果为表达式 2,否则,结果为表达式 3。如:对于 int a=5, b=6, c; c=a<b? a:b; 在求变量 c 的值时,先计算出表达式 a<b 的结果为真,则 c 取值为 a,即 c 的值为 5。

2.4.7 sizeof() 运算符

sizeof() 运算符的功能是计算某一类型数据在计算机中所占的字节数。它属于单目运算符,操作数可以是 C++ 的类型名或变量名,结果为一整数。如:

```
int i = sizeof(int);
```

```
int j = sizeof(i);
```

2.4.8 逗号运算符

用逗号将一系列操作数(或表达式)分隔开就构成了 C++ 的逗号表达式,其结果取最后一个操作数(表达式)的值,但每一个分表达式均计算一遍。如:

```
int a=0, b=0, c=0, d;
```

```
d = a++ , b++ , c++ ;
```

则运行之后 d, a, b, c 的值分别为 0, 1, 1, 1。

2.5 运算符的优先级与结合性

在学习算术的四则运算时,就知道先乘除后加减,这表示乘、除的优先级比加、减高。在 VC++ 的各种运算符之间,也存在着优先级的問題。