

Visual C++程序设计

解析与训练

祁云嵩 刘永良 华伟 编著



华东理工大学出版社

VC++程序设计解析与训练

祁云嵩 刘永良 华 伟 编著

 华东理工大学出版社

图书在版编目(CIP)数据

VC++程序设计解析与训练/祁云嵩 刘永良 华伟
编著—上海:华东理工大学出版社,2005.8

ISBN 7-5628-1769-3

I. V... II. ①祁... ②刘... ③华... 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(2005)第 083447 号

VC++程序设计解析与训练

.....

编 著 / 祁云嵩 刘永良 华 伟

责任编辑 / 周永斌

封面设计 / 王晓迪

责任校对 / 许 春

出版发行 / 华东理工大学出版社

地 址:上海市梅陇路 130 号,200237

电 话:(021)64250306(营销部)

传 真:(021)64252707

网 址:www.hdlgpress.com.cn

印 刷 / 江苏省通州市印刷总厂有限公司

开 本 / 787×960 1/16

印 张 / 17.5

字 数 / 319 千字

版 次 / 2005 年 8 月第 1 版

印 次 / 2006 年 8 月第 2 次

印 数 / 4051—7080 册

书 号 / ISBN 7-5628-1769-3/TP·139

定 价 / 23.00 元

内 容 提 要

本书是一册 VC++ 程序设计语言的学习和上机实践的参考书,在内容和章节安排上主要参照张岳新教授编著的《Visual C++ 程序设计》。全书共分为 14 章,每章内容分三部分:第一部分为基础知识概要,第二部分是为巩固基础知识而准备的习题,第三部分为实验指导。

本书内容丰富,概念清晰,习题典型,所安排的实验也具有一定的代表性,能较好地使学生从整体上掌握相应知识。

前 言

VC++语言程序设计和其他语言的程序设计一样,是一门实践性很强的课程,若仅通过课堂教学和阅读教科书,学生很难熟练掌握该门课程,更难提高程序设计能力。在日常的教学实践中,广大教师们普遍感到学生对课堂内容易懂、难记,实际编程时无从下手。其主要原因还是学习方法不当。本书的主要思想是使学生在应用中学习知识,在习题中巩固知识。

本书全部内容共分为14章,每章内容又分为三部分,第一部分为基础知识概要,第二部分是为巩固基础知识而准备的习题,第三部分为实验指导。编程实验不但可以培养、训练学生的应用能力,更重要的是可以使学生在应用中学习知识。学生要解决应用中的问题,必定要掌握相关的知识,这种内在的需求比常规的被动学习效果要好得多。所以,各章节的第三部分的实验是本书的主线,其内容的安排也力求由浅入深、由易到难。在每章的实验中,都安排了一个或几个实验程序,对一些有难度的实验内容,书中还给出了类似的可供参考的例题,用来帮助学习者加深理解。建议学生在完成第三部分实验的基础上再通过第二部分的习题巩固基础知识,细化知识点。

本书由江苏科技大学教材中心资助编写,由祁云嵩主编,刘永良编写1~8章的习题及相关内容,华伟编写9~14章的习题及相关内容,张晓如、潘克勤、王逊、王芳、段旭、邹晓华、范燕、石亮、潘舒、宋晓宁等老师分别负责了各章节习题的补充与校对工作。此外,本书的出版还得到了江苏科技大学教材中心全体教师的大力支持,在此一并表示感谢。

由于编者水平有限,书中疏漏错误之处,恳请读者批评指正。

目 录

第 1 章 VC++概述	1
1.1 基础知识	1
1.2 基础知识练习	5
1.3 实验	6
第 2 章 数据类型、运算符与表达式	8
2.1 基础知识	8
2.2 基础知识练习	14
2.3 实验	18
第 3 章 简单的输入/输出	20
3.1 基础知识	20
3.2 基础知识练习	22
3.3 实验	24
第 4 章 C++的流程控制语句	25
4.1 基础知识	25
4.2 基础知识练习	30
4.3 实验	51
第 5 章 函数和编译预处理	57
5.1 基础知识	57
5.2 基础知识练习	68
5.3 实验	86
第 6 章 数组	89
6.1 基础知识	89
6.2 基础知识练习	97
6.3 实验	111



第 7 章 结构体和枚举类型	116
7.1 基础知识	116
7.2 基础知识练习	120
7.3 实验	124
第 8 章 指针和引用	127
8.1 基础知识	127
8.2 基础知识练习	143
8.3 实验	153
第 9 章 类和对象	156
9.1 基础知识	156
9.2 基础知识练习	163
第 10 章 构造函数和析构函数	168
10.1 基础知识	168
10.2 基础知识练习	173
10.3 实验	182
第 11 章 继承和派生类	186
11.1 基础知识	186
11.2 基础知识练习	195
11.3 实验	204
第 12 章 类的其他特性	207
12.1 基础知识	207
12.2 基础知识练习	216
12.3 实验	231
第 13 章 运算符重载	236
13.1 基础知识	236
13.2 基础知识练习	245
13.3 实验	253



第 14 章 VC++ 的文件操作	256
14.1 基础知识	256
14.2 基础知识练习	261
14.3 实验	265
参考文献	269

第 1 章 VC++ 概述

1.1 基础知识

1.1.1 VC++ 源程序格式

下面是一个简单的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)          // 主函数
{
    cout <<"我的第一个C++程序! \n";/* 程序中的双引号、分号为西
    文符号 */
}
```

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C++程序!

由上例可以看出:VC++的程序结构是由编译预处理、注释和程序体组成。

1. 编译预处理指令

“#”开头的行称为编译预处理指令。上例中的第二行编译预处理指令的功能是将头文件“iostream. h”包含进来。一个VC++程序如果要进行输入/输出操作,必须要包含头文件“iostream. h”。在这个系统提供的头文件中定义了常用的输入/输出功能。

2. 程序体

函数是VC++的基本组成部分。任一VC++程序均由一个或多个主函数组成,其中必须有且只能有一个主函数,上例中的“void main (void)”就是函数



的头部。程序的执行就是从主函数的第一条语句开始执行,直至主函数结束。在主程序中可以调用其他函数。函数体要用一对花括号“{}”括起来,C++程序中的每个函数体都必须以“{”开始,以“}”结束;在main()函数体中,cout<<"我的第一个C++程序!\n";是一行输出语句,它的功能是将双引号内的内容在屏幕上显示出来。注意,一个完整的VC++功能语句必须以分号结束(有些说明性质的程序行不是功能语句,如上例中的前三行)。语句中的双引号、分号为西文符号。

3. 注释或说明信息

注释或说明信息只是为了阅读程序的方便,程序并不执行它。

注释或说明的方法有两种,第一种是从双斜杠“//”起直至该行结束均为程序的注释,如上例的第一行和第三行;第二种方法是将注释信息置于“/*”和“*/”之间,如上例中的第五行。

4. VC++程序书写格式

虽然VC++程序的书写格式比较自由,但如果遵循以下原则将有利于程序的阅读和理解:

- (1) 对齐规则:同一层次的语句应从同一列开始。
- (2) 缩进规则:属于下一层次的语句,应缩进几个字符。
- (3) 注释:适当地使用注释,说明程序块所完成的功能、变量所起的作用,提高程序的可读性。

1.1.2 VC++程序上机过程

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。

(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“C++ 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。

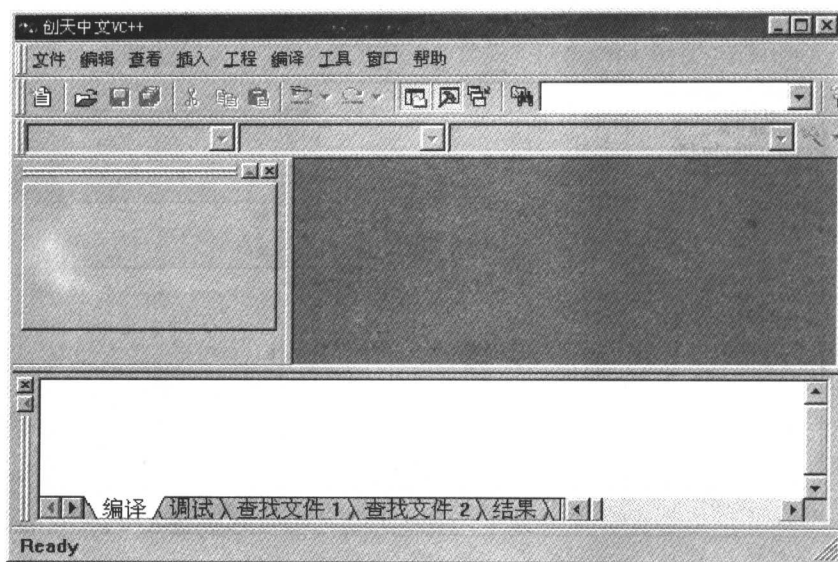


图 1-1 VC++集成环境界面

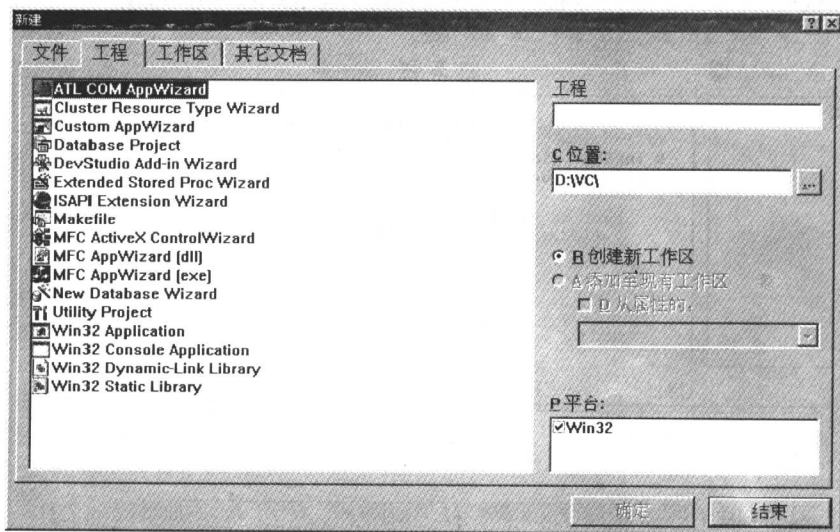


图 1-2 新建VC++工程项目界面

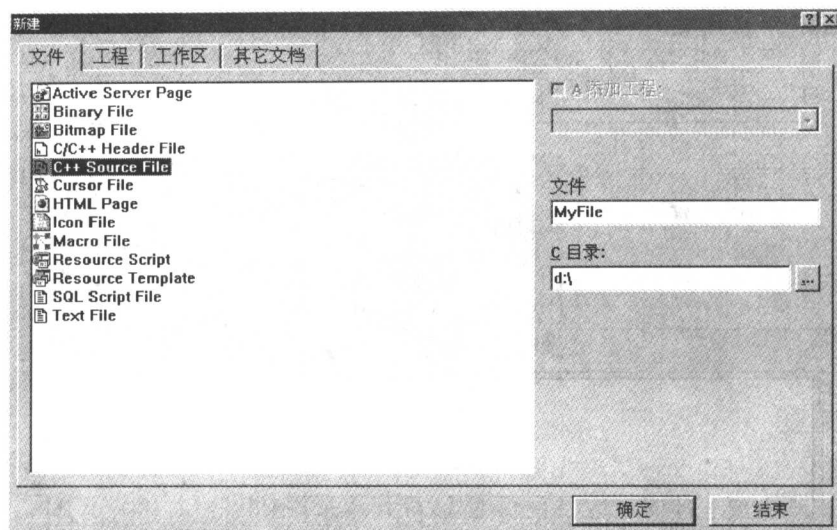


图 1-3 新建VC++源程序文件界面

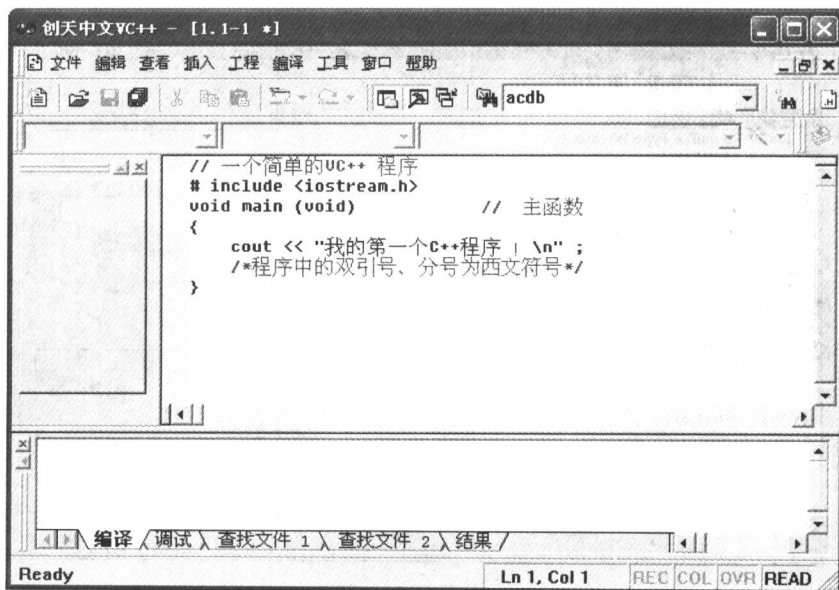


图 1-4 VC++源程序编辑界面



(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。

如果编译后已无错误提示,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。

1.2 基础知识练习

1.2.1 选择题

- 关于C++程序的执行过程,正确的说法是_____。
 - 一定从主函数开始,直到主函数结束
 - 从程序的第一行开始,直到程序的最后一行结束
 - 从主函数开始,直到程序的最后一行结束
 - 从程序的第一个函数开始,直到程序的最后一个函数结束
- 下列关于C++程序的书写规则,不正确的是_____。
 - 一行可以写若干条语句
 - 一条语句可以写成若干行
 - 可以在程序中插入注释信息
 - C++程序不区分大小写字母
- 在VC集成环境下,系统默认的源程序扩展名为_____。
 - .cpp
 - .txt
 - .exe
 - .obj
- 下列可用于标识VC++源程序注解的符号为:_____。
 - #
 - //
 - ;
 - { }
- 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



1.2.2 填空题

1. 一个C++程序必须有且只能有一个_____函数。
2. 在C++程序中,要使用库函数,必须用编译预处理指令将相应的头文件包含进来;如要进行标准输入/输出,则该编译预处理指令为_____。
3. VC++源程序编辑好后,还必须经过_____和_____才能得到可执行的文件。
4. VC++源程序中,函数体应置于_____之内。
5. 一个完整的VC++功能语句应以_____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_____,再连接生成的可执行文件扩展名为_____。

1.3 实验

1.3.1 实验目的

- (1) 熟练地使用VC++的编程环境。
- (2) 初步了解 VC++程序的编译、连接和运行的过程。
- (3) 掌握和理解 VC++程序的结构。
- (4) 熟悉 VC++程序数据的输入、输出。

1.3.2 例题

在屏幕上显示几行文字信息。

[程序]

```
#include<iostream.h>
void main(void)
{
    cout<<"我设计的第一个C++程序!"<<endl;
    cout<<"从中我了解了C++程序的基本组成."<<endl;
}
```

该程序经编译、连接后运行时在屏幕上显示以下两行文字:

我设计的第一个C++程序!
从中我了解了C++程序的基本组成。

程序的第一行表示程序编译时要将系统提供的文件“iostream.h”包含进来,因为在该文件中定义了标准输入输出的方法。在以后我们的C++程序设



计中如果要用到标准设备(键盘、显示屏)的输入和输出时,均应通过该指令将“iostream.h”文件包含进来。

任一个可执行的C++程序均应包含一个且只能包含一个主函数,一个C++程序总是从主函数开始执行的。程序中的第二行是主函数的头部,下面用一对大括号括起来的部分是主函数的内容,我们只要修改这部分的内容就可得到不同的运行结果。

在设计上述程序时我们还应注意以下几点:

- (1) 主函数的内容必须包含在一对大括号内。
- (2) 主函数中各语句必须以分号结束。
- (3) 引号中的内容就是我们要在屏幕上显示的内容。
- (4) 程序中的输出的关键字“endl”表示换行信息,如果不输出换行信息,则第二行输出将接在第一行后面输出。
- (5) 编程时应注意分号和引号均应是西文字符。

1.3.3 实验内容

编程在屏幕上以右对齐的方式分行显示自己的姓名、学号、通讯地址等信息。

1.3.4 问题讨论

总结实验中在编辑、编译、连接、运行等各环节中所出现的问题及解决方法。

第 2 章 数据类型、运算符与表达式

2.1 基础知识

2.1.1 标识符及变量

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”称为标识符。特别是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, 称其为变量。当然, 标识符除了可指代要处理的数据外, 还可以指代函数、文件等。

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, 这些标识符称为关键字或保留字, 在程序中不能再定义为用户标识符。如 cin、cout、void、main 等。
- (4) 推荐使用标识符命名法: 为了增加程序的可阅读性, 可采用 Windows 的标准命名法: 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 (对我们来说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。
- (5) 标识符应先定义, 后使用。

对于变量的定义, 其基本格式为:

<数据类型> <变量名 1>, [变量名 2], ……,[变量名 n];

2.1.2 基本数据类型

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, 我们根据所要表示的数据的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 unsigned 标识,存储单元中的全部二进制位都用作有效数本身,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,则其可表示的数的范围为 $-2^{31} \sim (2^{31}-1)$,而 unsigned int 类型可表示的数的范围为 $0 \sim (2^{32}-1)$ 。

(2) 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),然后再使用。

例如: 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,但各变量必须以逗号隔开,整个说明语句以分号结束。

2. 实型数据

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,例如 9.0。

(2) 指数形式,即我们所说的科学记数法,它以 10 的幂次方表示,例如 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。

实型数据是用有限的存储单元存储的,因此提供的有效数字也是有限的,在有效位以外的数字将被舍去,由此可能会产生一些误差。因而实型数据在计算机中的表示是近似的。

3. 字符数据

VC++ 中字符类型的数据分字符和字符串两种,其中字符串是字符的集合。

(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,即单引号内一个反斜杠“\”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。八进制数可以以 0 开头,也可以不以 0 开头,取值范围为 $0 \sim 0377$ 而十六进制数必须以 0x 或 0X 开头,取值范围为 $0 \sim 0XFF$ 。

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