

21世纪高校计算机系列规划教材

Visual C++ 程序设计解析与实训

张晓如 王 芳 主编 刘永良 潘克勤 华 伟 副主编 张再跃 主审

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

 21 世纪高校计算机系列规划教材

Visual C++程序设计

解析与实训

	张晓如	王 芳	主 编
刘永良	潘克勤	华 伟	副主编
		张再跃	主 审

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

全书共分 10 章, 每章均由知识要点、基本知识、例题解析、练习题、实验 5 个部分组成。知识要点部分以表格的形式直观地展示了每章学习内容的知识点以及相关的知识水平要求; 基本知识部分对章节学习的主要内容进行了简明扼要的概述; 例题解析部分有针对性地解读与分析了具有代表性的典型例题; 练习题部分精选了大量的选择题、填空题和编程题以供读者练习、提高; 实验部分从实践角度帮助学生提高其实践动手能力。

本书自成一体, 既可作为《Visual C++程序设计》一书的配套教材, 也可单独使用。可作为高等院校教师执教和本、专科学生学习 VC++程序设计语言, 参加计算机等级考试等各类资格与水平测试的教材与参考用书。

图书在版编目 (CIP) 数据

Visual C++程序设计解析与实训/张晓如, 王芳主编.
北京: 中国铁道出版社, 2008.1
(21 世纪高校计算机系列规划教材)
ISBN 978-7-113-08548-3
I. V… II. ①张…③王… III. C 语言—程序设计—高等学校—习题 IV. TP312-44
中国版本图书馆 CIP 数据核字 (2008) 第 009995 号

书 名: Visual C++程序设计解析与实训

作 者: 张晓如 王 芳 等

出版发行: 中国铁道出版社 (100054, 北京市宣武区右安门西街 8 号)

策划编辑: 严晓舟 张围伟

责任编辑: 周 欢 郑 楠

封面设计: 付 巍

封面制作: 白 雪

印 刷: 北京鑫正大印刷有限公司

开 本: 787×1092 1/16 印张: 16.5 字数: 387 千

版 本: 2008 年 3 月第 1 版 2008 年 3 月第 1 次印刷

书 号: ISBN 978-7-113-08548-3/TP·2682

定 价: 27.00 元

版权所有 侵权必究

凡购买铁道版的图书, 如有缺页、倒页、脱页者, 请与本社计算机图书批销部调换。

前 言

本书是《Visual C++程序设计》的配套教材，其编写目的是为进一步巩固和加强学生对VC++程序设计基本理论和基本知识的学习、理解与掌握，提高学生对VC++程序设计的认识水平以及在其应用过程中的实践动手能力。

作为计算机程序设计语言，其学习过程大致可分为以下几个阶段：

- 了解熟悉程序设计语言的语法规则及相关语义；
- 解读教材中的经典例题，包括其中的算法和程序；
- 模仿例题编写程序；
- 在模仿的基础上不断提高分析问题、解决问题和编程实现的能力。

其中，前两个阶段主要由教师带领学生来完成，而后两个阶段更多地需要学生主动自觉地完成。针对这一过程的特征，同时结合VC++程序设计语言的特点，我们在施教过程中以记忆语法规则为基本要求，并在此基础上通过有针对性的例题分析，以帮助学生理解程序设计语言的相关语义。为了让学生易于学、乐于学，教师易于教、善于教，我们对传统教材的体系结构进行了调整，在突出重点的同时，将课程学习的难点有机地重新加以分布，较好地体现了循序渐进的教学原则。经过前期讲义试教与使用，取得较好的教学效果。

本书内容与《Visual C++程序设计》教材内容相辅相成，每章均由知识要点、基本知识、例题解析、练习题、实验5个部分组成。知识要点部分以表格的形式直观地展示了每章学习内容的知识点以及相关的知识水平要求，以便学生从总体上理解和把握所学的知识；基本知识部分对章节学习的主要内容进行了简明扼要的概述，以帮助学生对所学知识进行归纳和总结；在例题解析部分，我们有针对性地选择了具有代表性的典型例题，通过详细的解读与分析，帮助学生学习和了解VC++程序设计的基本思想和基本方法；练习题部分题型丰富，其中选择题与填空题主要是帮助学生进一步巩固学习过程前两个阶段的学习成果，编程题则是为了帮助学生掌握编程的方法与技巧，既为学生模仿编程准备了材料，又为其独立编写程序提供了具体内容；实验部分从实践角度帮助学生把握所学内容，旨在提高他们的实践动手能力。另外，本书还根据江苏省计算机VC++语言等级考试的要求，在题型与题目的选择和内容的选取等方面均作了合理的安排。

编者在编写本书的过程中参考了大量业已出版的教材、书籍和网络资料，在此，我们对这些教材和书籍的作者以及提供网络资料的同仁表示由衷地感谢。此外，编写工作还得到了学校各级领导与相关部门的关心和支持，特别是计算机基础教学部的祁云嵩、邹晓华、於跃成、宋晓宁、段旭、常本勤、邹全、张绛丽、范燕、石亮、潘舒、王勇、黄霞、束鑫、肖寒、严熙等老师积极参与，提出了许多宝贵的意见，在此一并表示感谢！

由于时间仓促，疏漏及不妥之处在所难免，敬请广大读者批评指正。

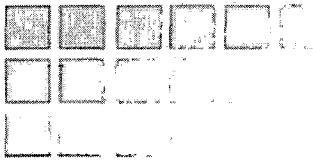
编 者

2007年11月

目 录

第 1 章 Visual C++程序设计入门	1
1.1 知识要点	1
1.2 基本知识	1
1.3 例题解析	8
1.4 练习题	10
1.5 实验	13
第 2 章 流程控制语句	15
2.1 知识要点	15
2.2 基本知识	15
2.3 例题解析	18
2.4 练习题	22
2.5 实验	36
第 3 章 数组	37
3.1 知识要点	37
3.2 基本知识	37
3.3 例题解析	44
3.4 练习题	51
3.5 实验	65
第 4 章 Visual C++语言中的函数	69
4.1 知识要点	69
4.2 基本知识	69
4.3 例题解析	73
4.4 练习题	79
4.5 实验	95
第 5 章 作用域和编译预处理	98
5.1 知识要点	98
5.2 基本知识	98
5.3 例题解析	101
5.4 练习题	105
5.5 实验	116
第 6 章 结构体与简单链表	118
6.1 知识要点	118
6.2 基本知识	118
6.3 例题解析	123

6.4	练习题	131
6.5	实验	143
第 7 章	类和对象	145
7.1	知识要点	145
7.2	基本知识	145
7.3	例题解析	149
7.4	练习题	158
7.5	实验	176
第 8 章	继承和派生	178
8.1	知识要点	178
8.2	基本知识	178
8.3	例题解析	182
8.4	练习题	186
8.5	实验	202
第 9 章	多态性	203
9.1	知识要点	203
9.2	基本知识	203
9.3	例题解析	207
9.4	练习题	218
9.5	实验	239
第 10 章	流类和文件	243
10.1	知识要点	243
10.2	基本知识	243
10.3	例题解析	247
10.4	练习题	249
10.5	实验	254
参考文献		255



第 1 章

Visual C++程序设计入门

1.1 知 识 要 点

知 识 点	主要知识要素	知 识 水 平	学 习 要 求
基本概念	关键字		识记
	自定义标识符	重点	熟练掌握
	库函数		识记
	基本数据类型	重点	识记、熟练掌握
常量与变量	常量		掌握
	符号常量		掌握
	转义字符		熟练掌握
	变量	重点	熟练掌握
	指针变量	难点	熟练掌握
	引用		熟练掌握
运算符与表达式	运算符及优先级		熟练掌握
	表达式运算		熟练掌握

1.2 基 本 知 识

1. VC++程序中源代码所用符号集

大、小写英文字母：A~Z，a~z

数字字符：0~9

其他字符（共 32 个）：~、!、#、%、^、&、*、(、)、-、_（下画线）、+、=、\、\、{、}、

[、]、:、;、"、'、<、>、,、.、?、/、空格、制表符（Tab 键产生的字符）、换行符（Enter 键所产生的字符）

特别提示：键盘上除去 3 个字符：\、@、\$，其余的可显示字符在程序代码中均能使用。

2. 关键字

关键字是指 VC++ 中用于表示特定含义的固定词汇。主要关键字见表 1-1。

表 1-1 VC++ 主要关键字表

关 键 字	类 型	关 键 字	类 型
void	类型说明符	return	语句
char	类型说明符	case	标号
int	类型说明符	default	标号
float	类型说明符	auto	说明符
double	类型说明符	static	说明符
long	类型说明符	register	说明符
short	类型说明符	extern	说明符
const	类型说明符	typedef	说明符
struct	类型说明符	signed	说明符
union	类型说明符	unsigned	说明符
enum	类型说明符	this	说明符
class	类型说明符	inline	说明符
if	语句	operator	说明符
else	语句	virtual	说明符
switch	语句	private	访问说明符
while	语句	protected	访问说明符
do	语句	public	访问说明符
for	语句	friend	访问说明符
break	语句	delete	运算符
continue	语句	new	运算符

3. 自定义标识符的命名规则

- (1) 由字母、数字、下画线组成的字符序列，第 1 个字符不能是数字。
- (2) 不能是关键字。

4. 编写程序时的注意事项

- (1) 程序中可含注释符：/*与*/为多行注释；//为单行注释，从标识起至本行结束。
- (2) 程序一般包含输入输出编译预处理命令：
#include <iostream.h>
- (3) VC++ 的程序有且只有一个主函数 main。

(4) 程序的书写规则：对于 VC++ 的编译器而言，一条语句可以写成若干行，一行内也可以写若干条语句。

特别提示：VC++ 编译器严格区分大、小写字母。

5. 库函数

库函数是指 VC++ 已提供的函数，它们包含在各个头文件中，使用时必须用预处理指令将相应的头文件包含进来。库函数及相应的头文件见表 1-2。

表 1-2 常用库函数

函 数 原 型	功 能	要包含的头文件
int abs(int x)	求整数的绝对值	math.h
float fabs(float x)	求实数的绝对值	math.h
double sqrt(double x)	求平方根	math.h
double exp(double x)	求 e^x	math.h
double pow(double x, double y)	求 x^y	math.h
char *strcpy(char *p1, char *p2)	字符串拷贝	string.h
char *strcat(char *p1, char *p2)	字符串连接	string.h
int strcmp(char *p1, char *p2)	字符串比较	string.h
int strlen(const char *p)	求字符串长度	string.h
void abort(void)	终止程序的执行	stdlib.h
void exit(int)	终止程序的执行	stdlib.h
int rand(void)	产生一个随机数	stdlib.h
setw(int)	设置输出项宽度	iomanip.h
cin.get(char)	输入一个字符（包括空格）	iostream.h
cin.getline(char *p, int)	输入一行字符	iostream.h

6. 数据类型

数据类型包括基本数据类型与自定义数据类型，基本数据类型见表 1-3，自定义数据类型包括结构体、类等，见表 1-4。

在 VC++ 中为了更加准确地描述数据类型，提供了 4 个关键字来修饰基本的数据类型：长型 long、短型 short、有符号型 signed、无符号型 unsigned。这 4 个修饰符用来修饰 int 时，int 可以省略。无修饰符的 int 和 char，编译器认为是有符号的，相当于加修饰符 signed。

7. 变量的说明（定义）与使用

变量说明的语法格式为：

数据类型 变量名 1, 变量名 2, ..., 变量名 n;

使用变量时要注意以下几点：

(1) 变量要先定义，后使用。

表 1-3 基本数据类型

类 型	名 称	占用字节数	取 值 范 围
void	无值型	0	
char	字符型	1	-128 ~ 127
signed char	有符号字符型	1	-128 ~ 127
unsigned char	无符号字符型	1	0 ~ 255
short int	短整型	2	-32 768 ~ 32 767
signed short int	有符号短整型	2	-32 768 ~ 32 767
unsigned short int	无符号短整型	2	0 ~ 65535
int	整型	4	$-2^{31} \sim (2^{31} - 1)$
signed int	有符号整型	4	$-2^{31} \sim (2^{31} - 1)$
unsigned int	无符号整型	4	$0 \sim (2^{32} - 1)$
long int	长整型	4	$-2^{31} \sim (2^{31} - 1)$
signed long int	有符号长整型	4	$-2^{31} \sim (2^{31} - 1)$
unsigned long int	无符号长整型	4	$0 \sim (2^{32} - 1)$
float	单精度实型	4	$-10^{38} \sim 10^{38}$
double	双精度实型	8	$-10^{308} \sim 10^{308}$
long double	双精度长实型	8	$-10^{308} \sim 10^{308}$

表 1-4 自定义数据类型

类 型	名 称	示 例
type[]	数组	int a[10]={1,2,3};
type*	指针	float x=5, *p = &x;
type&	引用	int a=3; int &m=a;
void	空类型	void main(void) { }
struct	结构	struct stu{int num; char name[10];};
union	联合	union stu{int num; double score};
enum	枚举	enum stu{one,two,three};
class	类	class stu{int num; char name[10];public:sut();};

(2) 数据从键盘输入时, 多个数据之间用分隔符分隔。分隔符有: 空格、制表符、换行符。

(3) 输出时, 字符串照原样输出, 变量输出其值。

(4) 可用语句 `cout.setf(ios::fixed);` 设置小数位输出格式。

8. 常量表示

(1) 八进制、十进制、十六进制、指数法表示数的输入输出格式, 见表 1-5。

表 1-5 不同进制数输入输出格式

格 式 (设有 int i,j; float x;)	含 义
<code>cin>>i>>dec>>j;</code> <code>cout<<i<<'t'<<j<<endl;</code>	输入、输出的整型变量 i 和 j 为十进制数

续上表

格式 (设有 int i,j; float x;)	含 义
cin>>hex>>i>>j; cout<<hex<<i<<'t'<<j<<endl;	输入、输出的整型变量 i 和 j 为十六进制数
cin>>oct>>i>>j; cout<<oct<<i<<'t'<<j<<endl;	输入、输出的整型变量 i 和 j 为八进制数
cout.setf(ios::scientific,ios::fixed); cout<<x<<endl;	按指数格式输出实型变量 x 的值

特别提示：用指数法表示的数，在 E 或 e 的前面必须有数字，且在其后必须是整数。

(2) 字符型常量。字符型常量是用单引号引起来的单个字符，如：‘a’、‘5’、‘+’等。在计算机内部是以它们的 ASCII 码表示的。

(3) 转义字符。转义字符一般用来表示一些不能直接从键盘输入的符号，也可以表示一般字符。例如：‘\123’用八进制数表示一个 ASCII 码字符‘S’；‘\x78’用十六进制数表示一个 ASCII 码字符‘x’，其对应十进制值均要求在 0~255 之间；‘\’、‘\42’、‘\x22’都表示双引号字符。常用转义字符见表 1-6。

表 1-6 常用转义字符

字符形式	功能或用途
\a	响铃
\b	退格
\f	换页
\n	换行
\r	回车符
\t	水平制表符 (Tab 键)
\v	纵向制表符
\\	输出反斜杠符
\'	输出单引号符
\"	输出单双号符
\ddd	输出 1~3 位八进制数所代表的字符
\xdd	输出 1~2 位十六进制数所代表的字符

(4) 字符串常量。用双引号引起来的若干个字符称为字符串常量。字符串常量在内存中按顺序逐个存放字符的 ASCII 码值，并在最后自动存放一个字符‘\0’，‘\0’是字符串的结束标志。

(5) 符号常量。符号常量有如下两种定义方法：

用 const 定义。例如，const int MAX = 666；则 MAX 是常量，值为 666。

用#define 说明一个符号常量。例如，#define PI 3.1415926

9. 指针与指针变量的说明、初始化、运算

指针是一个地址常量，指针变量是存储地址的变量，其说明方法是在变量名前加“*”号。例如：

```
int *p; //说明了一个指向整型变量的指针变量 p。
```

其中变量名 p 前的“*”为一标志。

可用地址运算符“&”求一个变量的地址。“&”是一个单目运算符，例如：

```
int a=5; int *p =&a;
```

还可用取内容运算符“*”求一个内存地址中存储的值，称为指针的内容。“*”也是一个单目运算符，例如：

```
cout<<a<<'\t'<<*p;           //输出为 5 5
```

10. 引用

引用是给一个已定义的变量起一个别名，两变量共享同一个内存空间。例如：

```
int x=5;int &m=x;
```

其中“&”为引用运算符。执行语句 `cout<<m<<endl;`后，输出的值为 5。

11. 运算符及其优先级

(1) 自增(++)与自减(--)运算符。它们的运算又分为前置与后置两种。

前置运算：先将变量的值加 1 (或减 1)，再将变量参与运算。

后置运算：先将变量参与运算，再将变量的值加 1 (或减 1)。

(2) 除法(/)运算符。若两个操作数都是整型，则结果也是整型。若有一个是实型，则结果是实型。

(3) 求模(%)运算符。要求操作数必须都是整型数，结果是两个整型数相除后的余数。

(4) 关系(<、<=、>、>=、==)运算符。关系运算符中除大于、小于单个符号表示外，其余都是用两个连在一起的符号表示。尤其是等于关系运算符(==)一定不要用错。

(5) 逻辑(!=、&&、||)运算符。逻辑表达式运算时编译器会自动优化，但有时会产生副作用。

(6) 复合赋值运算符。包括：+=、-=、*=、/=、%=等。

(7) 逗号(,)运算符。逗号表达式的计算方法是：按先后顺序依次计算各个表达式的值，最后一个表达式的值作为整个逗号表达式的值。逗号运算符在所有运算符中优先级最低。

(8) sizeof()运算符。sizeof()运算符是一元运算符，它用于计算某一操作数类型在内存中所占的字节数。

(9) ?:运算符。这是一个三目运算符，用于条件运算。

特别提示：除法与求模运算符的第 2 个操作数不能为零。

12. 表达式

表达式是由变量、常量、运算符、函数、圆括号等按一定规则组成的式子。一个变量、一个常量、一次函数调用都是表达式。

表达式的求值要根据运算符的意义、优先级、结合性以及类型转换约定共同决定。见表 1-7。

表 1-7 运算符的意义、优先级、结合性

优 先 级	运 算 符	含 义	目 数	结 合 性
1	::	类域	2	左结合
	()	圆括号		
	[]	下标运算符		
	->	指向成员运算符		

续上表

优 先 级	运 算 符	含 义	目 数	结 合 性
1	.	成员运算符	2	左结合
	&	引用运算符	1	左结合
2	!	逻辑非运算符	1	右结合
	++	自增运算符		
	--	自减运算符		
	-	取负运算符		
	+	取正运算符		
	(类型)	类型转换运算符		
	*	指针运算符		
	&	地址运算符		
	sizeof	数据类型长度运算符		
	new	分配存储单元		
	delete	释放存储单元		
3	*	乘法运算符	2	左结合
	/	除法运算符		
	%	求模运算符		
4	+	加法运算符	2	左结合
	-	减法运算符		
6	<, <=, >, >=	小于、小于等于、大于、大于等于运算符	2	左结合
7	==	相等运算符	2	左结合
	!=	不相等运算符		
11	&&	逻辑与运算符	2	左结合
12		逻辑或运算符	2	左结合
13	?:	条件运算符	3	右结合
14	=, +=, -=, *=, /=, %=	赋值运算符	2	右结合
15	,	逗号运算符	2	左结合

13. 数据类型转换

(1) 自动转换。在算术运算和关系运算中如果参与运算的操作数类型不一致, 编译系统会自动对数据进行转换(即隐含转换), 基本原则是将低精度类型数据转换为高精度类型数据。如图 1-1 所示。

(2) 强制转换。当需要人为地改变一个数据的类型时, 就需要用到强制类型转换。

语法格式为:

(数据类型) 表达式

或

数据类型 (表达式)

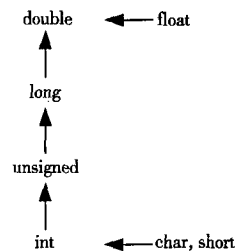


图 1-1 类型自动转换

1.3 例题解析

【例 1】设有 `int a=0, m=5, n=10`; 执行语句 `a = (m-n, m+++n)`; 后, `a`、`m`、`n` 的值分别为 (1)、(2)、(3)。

解答: (1) 15 (2) 6 (3) 10

分析:

(1) 小括号 `(m-n, m+++n)` 中, 是一个逗号表达式, 后一项 `m+++n` 的值作为整个括号的值赋给左边的变量 `a`。

(2) 根据运算符的优先级, `++` 运算符为 2 级, `+` 运算符为 4 级, 故 `++` 运算符高于 `+` 运算符。因此, `m+++n` 相当于 `m++ + n`。而 `m++` 是后置运算, 故先将 `m+n` 的值 15 赋给变量 `a`, 然后再将 `m` 的值加 1, 使 `m` 的值为 6。整个运算 `n` 的值没有变化, 仍为 10。

可将题目中给出的代码写入主函数中, 再增加输出部分, 就可以验证结果。例如:

```
#include <iostream.h>
void main()
{
    int a=0, m=5, n=10;
    a = (m-n, m+++n);           //A 行
    cout << a << '\t' << m << '\t' << n << endl;
}
```

理解与思考:

(1) 若将 A 行小括号去掉, 改为: `a = m-n, m+++n`; 则输出结果是什么?

(2) 若再将 A 行改为: `a = (m-n, m+ ++n)`; 则输出结果又是什么? 请给出相应解释。

【例 2】以下是从键盘输入直角三角形的两条直角边边长, 求三角形的斜边长与面积的程序。

现程序中有若干错误, 请改正。

要求: 可以修改程序中的一部分语句, 增加少量的变量说明、函数原型说明或编译预处理命令, 但不能增加其他语句, 也不能删去整条语句。

```
#include <iostream.h>           //1
                                //2
void main(void) {               //3
    float a, b, c, area;        //4
    cout << "输入两条直角边 a, b 的值: " << endl; //5
    cin >> a >> b;              //6
    c = sqrt(a*a-b*b);          //7
    area = 1/2*a*b;             //8
    cout << "三角形斜边长 c 的值为: " << c << endl; //9
    cout << "三角形面积为: " << aree << endl;      //10
}                                //11
```

解答:

第 2 行改为: `#include <math.h>`

第 7 行改为: `c = sqrt(a*a+b*b);`

第 8 行改为: `area = a*b/2;`

第 10 行改为: `cout << "三角形面积为: " << area << endl;`

分析: 这类题目一般可先将含有错误的程序代码输入到计算机中, 再在计算机上调试。

若是语法错, 对含错源程序进行编译, 每次先修改第 1 个错误。再编译, 再修改。

若是语义错，可通过程序的运行结果，推断出程序的可能出错位置。

以本题为例，其调试改错过程如下：

(1) 编译出现的第1个错误提示是：

```
error C2065:'sqrt':undeclared identifier
```

sqrt 是库函数名，而在此系统不识此函数名，说明缺少库函数的头文件，故只要加上编译预处理命令 #include <math.h> 即可。

(2) 再编译，出现的第1个错误提示是：

```
error C2065:'aree':undeclared identifier
```

双击出错提示，编译窗口光标停在第10行，说明这一行或其前后位置存在错误。再看出错提示中 'aree' 系统不能识别，到第4行处查看变量说明语句，原来说明的变量名为 area。变量名前后不一致，故可将程序第10行中的 aree 改为 area。

(3) 再编译，程序没有错误。接着运行程序，输入3与4，输出结果如下：

输入两条直角边 a,b 的值：

3 4

三角形斜边长 c 的值为：-1. #IND

三角形面积为：0

程序没有语法错误，但却存在语义错误。

从输出结果可以看出，斜边长为负数，说明计算公式有误。检查程序发现第7行有误，求斜边长应该是两直角边平方和的平方根，但程序中变成平方差的平方根了。改正如下：

```
c=sqrt(a*a+b*b);
```

(4) 再编译后运行程序，输入3与4，输出结果如下：

输入两条直角边 a,b 的值：

3 4

三角形斜边长 c 的值为：5

三角形面积为：0

按照勾股定理，直角三角形斜边长是正确的，但面积不正确。检查程序中求面积的公式：

```
area=1/2*a*b;
```

从数学角度看似乎没有问题，但在程序第8行中，按照表达式计算，应先计算 1/2，此时结果为 0。0 乘以任何数都为 0，所以结果为 0。故表达式应改为：

```
area=a*b/2;
```

再编译、运行，输出结果正确。

【例3】 已知 int a=3,*pa=&a; 输出指针 pa 十进制的地址值的方法是_____。

A. cout<<pa;

B. cout<<*pa;

C. cout<<&pa;

D. cout<<long(&pa);

解答：D

分析：答案 A 输出了指针 pa 本身的值，即变量 a 的地址；答案 B 输出了指针 pa 的内容，即变量 a 的值 3；答案 C 以十六进制形式输出了指针 pa 的地址值。因为插入运算符 << 输出指针类型对象的地址值时，默认为十六进制形式。如果要输出十进制形式的地址值，必须用 long 类型进行强制转换。

理解与思考：

能否用 cout<<int(&pa); 输出 pa 十进制的地址值？

1.4 练 习 题

一、选择题

- 关于 VC++ 程序的执行过程, 正确的说法是 ()。
 - 一定是从主函数开始, 直到主函数结束
 - 从程序的第 1 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 1 个函数开始, 直到程序的最后一个函数结束
- 下列关于 VC++ 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
 - VC++ 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 ()。
 - #
 - //
 - ;
 - { }
- 在一个完整的 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
- 下列符号中, 能用做 VC++ 标识符的是 ()。
 - 5A
 - int
 - a+b
 - F4
- 下列的常数表示中有一个不正确, 不正确的是 ()。
 - 25
 - 1.2E-5
 - 354
 - 6e2.5
- 设有 `char c1;` 则以下赋值正确的是 ()。
 - `c1 = "2"`
 - `c1 = "\x255"`
 - `c1 = "\255"`
 - `c1 = "A"`
- 对于变量说明 `int a1=10, a2=3; float a3=8.0;` 表达式 `a1/a2*a2*a3` 的值为 ()。
 - 80
 - 80.0
 - 72
 - 72.0
- 数学式 $\sqrt{\frac{x^2 + y^2}{xy}}$, 在 VC++ 程序中正确的表达方式 ()。
 - `sqrt(x*x+y*y)/x*y`
 - `SQRT(x*x+y*y)/x*y`
 - `sqrt((x*x+y*y)/(x*y))`
 - `SQRT((x*x+y*y)/(x*y))`
- 不正确的变量定义方法是 ()。
 - `int a, &b=a;`
 - `float a, *b=&a;`
 - `int a(8), b=0;`
 - `int m=n=10;`
- 对于变量说明 `int k=3;` 表达式 `5ll(k++)+(k++)` 的值为 ()。
 - 6
 - 9
 - 1
 - 0
- 设有变量说明 `int x;` 则表达式 `(x=4*5, x*5), x+25` 的值为 ()。
 - 20
 - 45
 - 100
 - 125

14. 设有说明 `int x=1,z=1,y=1,k`; 则执行语句 `k=x++||++y&&++z`;后, 变量 `y` 的值为()。
- A. 1 B. 2 C. 3 D. 4
15. 设有说明 `int a,b`; 则执行语句 `b=(a=3*5,a*4),a+15`;后, `b` 的值为()。
- A. 15 B. 30 C. 60 D. 90
16. 设有变量说明 `int m=4`; 则执行语句 `m+=m*=m-=m/=m`;后, `m` 的值为()。
- A. 12 B. 0 C. 16 D. 8
17. 下列表达式中, 错误的是()。
- A. `4.0%2.0` B. `k+++j` C. `a+b>c+d?a:b` D. `x*=y+25`
18. 设有说明 `int x=2,y=3`; 则表达式 `++x>y--?x:y` 的值为()。
- A. 1 B. 2 C. 3 D. 4
19. 设有说明 `int x=1,y=1,z=1,c`; 则执行语句 `c=--x&&--y||--z`;后, `x`、`y`、`z` 的值分别为()。
- A. 0、1、1 B. 0、1、0 C. 1、0、1 D. 0、0、1
20. 执行 `int k=11;k=1/3*k++`;后, `k` 的值是()。
- A. 0 B. 3 C. 4 D. 1
21. 设有 `int a`; 则以下不能正确表达数学关系 $10 < a < 15$ 的表达式是()。
- A. `10 < a < 15` B. `a==11||a==12||a==13||a==14`
C. `a>10&& a<15` D. `!(a<=10)&&!(a>=15)`
22. 下列程序段的输出结果是()。
- ```
int a=0,b=4,k=2;k=a+1>0|| (++a<0)&&!(b--<=0);
cout<<k<<','<<a<<','<<b;
```
- A. 1, 1, 3              B. 1, 1, 4              C. 1, 0, 3              D. 1, 0, 4
23. 设有定义 `float y=3.45678;int x`; 则以下表达式中, 能实现将 `y` 中数值保留小数点后 2 位四舍五入的表达式是( )。
- A. `y=(y*100+0.5)/100.0`              B. `x=y*100+0.5,y=x/100.0`  
C. `y=y*100+0.5/100.0`              D. `y=(y/100+0.5)*100.0`
24. 设有说明 `int j=10;float x`; 则执行语句 `x=5/j`;后, `x` 的值为( )。
- A. 0.5                      B. 0.0                      C. 0                      D. 1.0
25. 设有说明 `int a,b,c;a=b=c=5`; 则执行语句 `b+=++a>b&&++c>b`;后, `a`、`b`、`c` 的值分别为( )。
- A. 6、7、6              B. 6、6、6              C. 6、6、5              D. 6、1、6
26. 设 `int n=10,j=4`; 则执行赋值运算 `n%=j+1` 后, `n` 的值是( )。
- A. 0                      B. 3                      C. 2                      D. 1
27. `sizeof(double)` 的值是( )。
- A. 2                      B. 4                      C. 8                      D. 16
28. 设有定义 `int a;float f;double j`; 则表达式 `10+'a'+j*f` 值的数据类型是( )。
- A. `int`                      B. `double`                  C. `float`                      D. 不确定
29. 在 VC++ 语言中, 080 是( )。
- A. 八进制数              B. 十进制数              C. 十六进制数              D. 非法数
30. 设有变量定义 `int x=5,n,*p1=&x,*p2`; 则下列表达式正确的是( )。
- A. `p2=&p1`                  B. `*p1=*p2`              C. `p2=p1`                  D. `n=*p2`