

National Computer Rank Examination

全国计算机等级考试专用辅导丛书

全国计算机等级考试 专用辅导教程

二级 C++

——2012版——

李佐勇 宋文慧 编著



考情回顾——纵观历年真题，了解试题分布，总结重点内容，提炼核心考点。
题链接——穿插最新真题，强化考试内容，了解最新动态，把握命题规律。
题分析——挑选经典考题，熟悉解题技巧，吃透考试题目，掌握解题方法。
关必备——精选过关试题，及时自我检测，提升学习效果，瞬间巩固提高。



本书配有立体化辅导光盘，
包括**考试模拟系统**与**上机操作视频演示**等资源，帮助考生轻松过关！



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

National Computer Rank Examination

全国计算机等级考试专用辅导丛书

全国计算机等级考试
专用辅导教程

二级 C++

——2012版——

李佐勇 宋文慧 编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内容简介

本书紧扣最新版考试大纲,结合编者多年从事命题、阅卷及培训辅导的实际经验编写而成。本书章节安排与官方教程同步,主要包括:C++语言概述、数据类型、运算符和表达式、C++的基本语句和控制结构、数组、指针与引用、函数、类和对象、继承和派生、运算符重载、模板、C++流。章节前插入了近几年考试分布,精讲考试重点与难点,讲解过程中链接最近2年的考试真题,方便读者了解最新考试动态。章节末安排了典型题讲解和适量过关习题。书末附有3套样题和解析,供考生考前实战演练。

本书配有上机光盘,盘中提供全真的模拟考试系统,考试环境、题型与真实考试一致。

本书以全国计算机等级考试考生为主要读者对象,适合于考生在等考前复习使用,也可作为相关考试培训班的辅导教材。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

全国计算机等级考试专用辅导教程:C++ / 李佐勇,宋文慧编著. —北京:电子工业出版社,2012.1

(全国计算机等级考试专用辅导丛书)

ISBN 978-7-121-15337-2

I. ①全… II. ①李… ②宋… III. ①电子计算机—水平考试—自学参考资料②C语言—程序设计—水平考试—自学参考资料 IV. ①TP3

中国版本图书馆CIP数据核字(2011)第246973号

责任编辑:牛勇

特约编辑:顾慧芳

印刷:

装订:北京京科印刷有限公司

出版发行:电子工业出版社

北京市海淀区万寿路173信箱

邮编:100036

开本:860×1092 1/16

印张:17.75

字数:475千字

印次:2012年1月第1次印刷

定价:39.80元(含光盘1张)

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zltts@phei.com.cn, 盗版侵权举报请发邮件到 dbqq@phei.com.cn。

服务热线:(010) 88258888。

前 言

全国计算机等级考试是目前我国规模最大、参加人数最多的全国性计算机类水平考试，因其具有权威性、公平性和广泛性而在社会上享有良好的声誉，很多单位都把获取计算机等级证书作为单位录用、职称评定的标准之一。

应广大考生的需要，我们按教育部考试中心指定教材的篇章结构，组织从事全国计算机等级考试试题研究的老师和专家精心编写了“全国计算机等级考试专用辅导丛书”，为读者打造最简单、最实用的考试教程，目的是让考生在短时间内快速过关，知己知彼，百战百胜！

一、本书特色

1. 突出标准性与严谨性

本丛书由从事全国计算机等级考试试题研究人员及在等级考试第一线从事命题研究、教学、辅导和培训的老师合作编写，层次清晰，结构严谨，导向准确。

2. 注重典型考题的分析

紧扣新大纲要求，精讲考点、重点与难点，深入分析典型范例，抓住等级考试题眼，并提供实战训练。

3. 突出实用性和高效性

各书的章名、节名与教育部考试中心指定教程同步，每章还精心设计以下板块。

考情回顾：总结本章需要掌握的重点内容，分析最近几年的考试真题，整理出试题分布、试题题型和分值，提炼出本章的考核要点。

真题链接：整理出最近两年的考试真题，穿插在知识点的讲解中，帮助考生理解知识点，也让考生了解最新考试动态和命题规律。

典型考题分析：精选出常考的典型题型，并针对每一个典型题进行详细解析，引导考生掌握重点内容，吃透重点考试题目，并且让考生了解解题方法和解题思路。

过关必备：针对每章知识点安排适当的过关题目，便于考生及时将知识点运用于具体题目中，方便考生检查学习效果，巩固提高所学知识。

4. 全面模拟，实战提高

根据新大纲、新考点、新题型进行最新命题，书末提供3套上机样题，供考生考前实战演练。

5. 注重上机考试的辅导

针对上机考试的特点，本书在深入研究上机真题库的基础上，将上机真题进行分类，提炼出题

型，按类型进行解析，便于考生专项攻克难关，提高复习效率。

6. 书盘结合，立体化辅导

本书为考生应考提供立体化辅导光盘。光盘中不仅包括考试模拟系统，还包括上机操作视频演示，帮助考生轻松过关。

二、读者对象

本书可供参加全国计算机等级考试二级 C++（2012 版）的考生复习使用，也可以作为相关等级考试培训班的辅导教材。

三、关于作者

本书由李佐勇、宋文慧任主编。参与本书编写工作的人员还有：李赛红、吕伟、严惠、宋永生、张艳、施俊飞、吴海涛、俞露、李胜、姚昌顺、朱贵喜、杨万扣、张华明。在此对诸位作者表示衷心的感谢。

由于作者学识有限，书中难免有错误和疏漏之处，恳请专家和广大读者指正。

编 者

目 录

第 1 章 C++语言概述	1
1.1 C++语言的发展	1
1.2 C++语言的特点	2
1.3 面向对象程序设计	2
1.4 C++语言的基本符号	2
1.5 C++语言的词汇	3
1.5.1 关键字	3
1.5.2 标识符	3
1.5.3 字面常量	4
1.5.4 运算符	4
1.5.5 标点符号	4
1.6 C++程序的基本框架	5
1.6.1 结构化程序设计框架	5
1.6.2 面向对象程序设计框架	5
1.7 C++程序的开发过程	6
1.7.1 编辑	6
1.7.2 编译	6
1.7.3 链接	6
1.7.4 运行和调试	6
1.8 典型考题分析	7
1.9 过关必备	8
第 2 章 数据类型、运算符和表达式	10
2.1 C++语言的数据类型	10
2.1.1 基本类型	10
2.1.2 基本类型的派生类型	11
2.2 常量	11
2.2.1 逻辑常量	12
2.2.2 字符常量	12
2.2.3 整型常量	13

2.2.4	实型常量	13
2.2.5	枚举常量	13
2.3	变量	14
2.3.1	变量的定义	14
2.3.2	变量的使用方式	15
2.3.3	符号常量声明语句	16
2.4	运算符和表达式	16
2.4.1	运算符和表达式的概念	16
2.4.2	运算类型与运算符	16
2.4.3	赋值运算	17
2.4.4	算术运算符和算术表达式	17
2.4.5	关系运算符和关系表达式	17
2.4.6	逻辑运算符和逻辑表达式	18
2.4.7	位运算	18
2.4.8	其他运算	19
2.4.9	优先级和结合性	20
2.5	典型考题分析	21
2.6	过关必备	23
第 3 章	C++的基本语句和控制结构	24
3.1	C++语句	24
3.2	顺序结构	25
3.2.1	声明语句	25
3.2.2	表达式语句	25
3.2.3	基本输入/输出语句	25
3.2.4	复合语句和空语句	26
3.3	选择结构	27
3.3.1	if 语句	27
3.3.2	switch 语句	28
3.4	循环结构	29
3.4.1	for 语句	29
3.4.2	while 语句	30
3.4.3	do...while 语句	30
3.4.4	循环的嵌套	31
3.5	跳转语句	32
3.5.1	break 语句	32
3.5.2	continue 语句	33
3.5.3	return 语句	33

3.5.4 goto 语句	33
3.6 典型考题分析	34
3.7 过关必备	35
第4章 数组、指针与引用	39
4.1 数组	39
4.1.1 一维数组	39
4.1.2 多维数组	40
4.1.3 字符数组	41
4.2 指针	43
4.2.1 指针和地址	43
4.2.2 指针和数组	44
4.3 引用	46
4.4 动态存储分配	46
4.5 典型考题分析	48
4.6 过关必备	51
第5章 函数	54
5.1 函数定义	54
5.2 函数调用	55
5.3 函数原型	57
5.4 函数返回类型	57
5.5 函数参数	59
5.5.1 参数的传递方式	59
5.5.2 默认参数	59
5.6 函数重载	60
5.7 内联函数	60
5.8 递归函数	61
5.9 变量的生存周期	61
5.10 典型考题分析	63
5.11 过关必备	65
第6章 类和对象	69
6.1 类的定义	69
6.1.1 类的定义	69
6.1.2 类成员的访问控制	70
6.1.3 类的数据成员	70
6.1.4 类的成员函数	71
6.2 对象的定义	72

6.2.1	对象的定义	73
6.2.2	对象的成员	73
6.3	构造函数和析构函数	74
6.3.1	构造函数和析构函数的定义	74
6.3.2	默认构造函数和默认析构函数	75
6.3.3	拷贝构造函数	75
6.4	自由存储对象	76
6.5	this 指针	77
6.6	静态成员	79
6.6.1	静态数据成员	79
6.6.2	静态成员函数	79
6.7	常成员	81
6.7.1	常对象	81
6.7.2	常成员函数	81
6.7.3	常数据成员	81
6.8	友元	82
6.8.1	友元函数	82
6.8.2	友元类	82
6.9	对象数组	83
6.10	成员对象	84
6.11	典型考题分析	85
6.12	过关必备	93
第 7 章	继承和派生	104
7.1	继承与派生	104
7.1.1	基本概念	104
7.1.2	派生类的定义与构成	105
7.2	派生类对基类成员的访问	106
7.3	派生类的构造函数和析构函数	108
7.3.1	派生类的构造函数	109
7.3.2	派生类的析构函数	109
7.4	多继承与虚基类	112
7.4.1	多继承中的二义性问题	112
7.4.2	虚基类的定义	113
7.4.3	虚基类的构造函数	113
7.5	子类型关系	113
7.6	虚函数与多态性	114
7.6.1	多态性的概念	114

7.6.2	虚函数	114
7.6.3	虚析构造函数	115
7.6.4	纯虚函数与抽象类	115
7.7	典型考题分析	116
7.8	过关必备	125
第 8 章	运算符重载	136
8.1	运算符函数与运算符重载	136
8.1.1	运算符重载的概念	136
8.1.2	运算符重载的两种形式	137
8.2	典型运算符的重载	138
8.2.1	关于分数类 fraction	138
8.2.2	重载取负运算符“-”	139
8.2.3	重载加法运算符“+”	139
8.2.4	重载自加运算符“++”	139
8.2.5	重载类型转换符“long”	140
8.2.6	重载赋值运算符“=”	140
8.2.7	重载复合赋值运算符“+=”	140
8.2.8	重载关系操作符“>”	140
8.2.9	重载下标访问运算符“[]”	141
8.2.10	重载 C++ 流运算符“>>”和“<<”	141
8.3	运算符重载应注意的几个问题	142
8.4	典型考题分析	143
8.5	过关必备	145
第 9 章	模板	151
9.1	函数模板	151
9.1.1	函数模板的概念和定义	151
9.1.2	模板函数的实例化	152
9.1.3	模板实参的省略	152
9.2	类模板	154
9.2.1	类模板的概念和定义	154
9.2.2	类模板的实例化	155
9.3	典型考题分析	156
9.4	过关必备	159
第 10 章	C++ 流	163
10.1	C++ 流的概念	163
10.1.1	C++ 流的体系结构	163

10.1.2	预定义流对象	164
10.1.3	提取运算符>>和插入运算符<<	164
10.1.4	有格式输入/输出和无格式的输入/输出	165
10.1.5	操作符	165
10.2	输入/输出的格式控制	165
10.2.1	默认的输出/输入格式	165
10.2.2	格式标志与格式控制	166
10.2.3	输入输出宽度的控制	167
10.2.4	浮点数输出方式的控制	168
10.2.5	输出精度的控制	168
10.2.6	对齐方式的控制	168
10.2.7	小数点处理方式的控制	169
10.2.8	填充字符的控制	169
10.2.9	插入换行符	169
10.2.10	输入/输出数制状态的控制	169
10.3	文件流	171
10.3.1	文件流的建立	171
10.3.2	文件流的关闭	172
10.3.3	文件流状态的判别	172
10.3.4	文件流的定位	172
10.3.5	有格式的输入/输出	173
10.3.6	无格式输入/输出	173
10.4	典型考题分析	174
10.5	过关必备	176
第 11 章	上机指导	180
11.1	数据类型的定义和使用	180
11.2	数组、指针的定义和使用	182
11.3	函数的定义和常用算法的函数实现	185
11.4	类和对象	193
11.5	继承和派生	199
11.6	运算符重载	205
11.7	模板的定义和使用	209
11.8	C++的流操作	212
附录 A	本书过关必备答案	214
附录 B	样题	217
附录 C	笔试样题答案及分析	256

第 1 章 C++语言概述

考情回顾

通过对最近 5 次考试真题的分析, 可知本章题量和分值比较稳定, 大约共 2 道选择题和 1 道填空题, 约占试卷分值的 3%。表 1-1 统计了最近 5 次考试中本章考点的分布。

表 1-1 历年考题知识点分布统计表

年份	试题分布	题型	分值	考核要点
2011.09	6, 17	选择题	4	C++语言基本概念、算法的特性、结构化程序的三种基本结构（顺序结构、选择结构、循环结构）和模块化结构等
2011.03	4, 5, 17	选择题	6	C++语言基本概念、程序设计、结构化程序的三种基本结构（顺序结构、选择结构、循环结构）和模块化结构等
	3	填空题	2	
2010.09	3, 5, 6	选择题	6	程序设计、算法的特性、算法的描述方法、结构化程序的三种基本结构（顺序结构、选择结构、循环结构）和模块化结构等
2010.03	6	选择题	2	C++语言基本概念、结构化程序的三种基本结构（顺序结构、选择结构、循环结构）和模块化结构等
2009.09	6	选择题	2	程序设计、算法的特性、结构化程序的三种基本结构（顺序结构、选择结构、循环结构）和模块化结构等
	3	填空题	2	

1.1 C++语言的发展



考核说明: 本节主要介绍 C++语言的发展、特点及程序的基本要求和程序的开发步骤; 了解面向对象的程序设计思想及四个要素。

C++是一种静态数据类型检查的, 支持多重编程范式的通用程序设计语言。它支持过程化程序设计、数据抽象、面向对象程序设计、制作图标等泛型程序设计等多种程序设计风格。

C++语言起源于 C 语言。1980 年, 美国贝尔实验室的 Bjarne Stroustrup 和他的同事们在 C 语言的基础上, 从 Simula 67 中引入面向对象的特征, 开发出一种过程性与对象性相结合的程序设计语言。最初称为“带类的 C”, 1983 年取名为 C++。以后又经过不断的完善和发展, 成为目前的 C++ 语言。

1.2 C++语言的特点

C++是一种面向对象的程序设计语言，其特点是：（1）抽象数据类型。（2）封装和信息隐藏。（3）以继承和派生方式实现程序的重用。（4）以运算符重载和虚函数来实现多态性。（5）以模板来实现类型的参数化。

C++是程序员和软件开发者在实践中创造的。C++往往从编写实际程序的角度出发，为程序员提供了各种实用、灵活、高效的语言特性。

C++是 C 语言的超集，能够很好地兼容 C 语言正是 C++取得成功的原因之一，这是因为：C++继承了 C 语言简明、高效、灵活等众多优点；以前使用 C 语言编写的大批软件可以不加任何修改，直接在 C++开发环境下维护；C 语言程序员只需要学习 C++扩充的新特性，就可以很快地使用 C++编写程序。

- （1）C++语言是一种面向对象的程序设计语言。
- （2）C++是程序员和软件开发者在实践中创造的。
- （3）C++是 C 语言的超集。

1.3 面向对象程序设计

考核说明：本节主要介绍面向对象程序设计，C++是一种面向对象的程序设计语言。

它充分支持面向对象思想中的三个主要特征。

1. 封装性

封装性是指将数据和算法捆绑成一个整体，这个整体就是对象，描述对象的数据被封装在其内部。如果需要存取数据，可以通过对象提供的算法来进行操作，而无须知道对象内部的数据是如何表示和存储的。这种思想被称为信息隐藏。

2. 继承性

继承性是指一种事物保留了另一种事物的全部特征，并且具有自身的独有特征。C++语言采用继承来支持重用，程序可以在现有类型的基础上扩展功能来定义新类型。新类型是从现有类型中派生出来的，因此被称为派生类。

3. 多态性

多态性是指当多种事物继承自一种事物时，同一种操作在它们之间表现出不同的行为。C++语言中使用函数重载、模板、虚函数等概念来支持多态性。C++语言主要包括面向过程和面向对象两部分内容。学习 C++语言时，应该先学习其面向过程部分，再学习面向对象部分。

1.4 C++语言的基本符号

考核说明：本节主要介绍 C++的基本符号，以及五种合法词汇，其中标识符的命名和

常量变量的命名法则是重点。

C++语言由基本符号组成，其基本符号可以分为3类。

(1) 字母：包括大小写英文字母，共52个符号。

(2) 数字：包括0~9，共10个符号。

(3) 特殊符号：包括 + - * / = , . _ : ; ? \ " ' ~ | ! # % & () [] { } ^ < > 和 “空格”共30个符号。

1.5 C++语言的词汇

词汇是C++语言的基础。只有五种合法词汇：关键字、标识符、字面常量、运算符和标点符号。

1.5.1 关键字

关键字也称为保留字，它是由C++语言本身预先定义好的一类单词。关键字一共有61个，包括ANSI C标准规定的32个关键字（如表1-2所示）和ANSI C++标准补充的另外29个关键字（如表1-3所示）。

表 1-2 ANSI C 标准规定的关键字

auto	break	case	char
const	continue	default	do
double	else	enum	extern
float	for	goto	if
int	long	register	return
short	signed	sizeof	static
struct	switch	typedef	union
unsigned	void	volatile	while

表 1-3 ANSI C++标准补充的关键字

bool	catch	class	const_cast
delete	dynamic_cast	explicit	false
friend	inline	mutable	namespace
new	operator	private	protected
public	reinterpret_cast	static_cast	template
this	throw	true	try
typeid	typename	using	virtual
wchar_t			

1.5.2 标识符

标识符是用户为程序中各种需要命名的“元素”所起的名字。这些“元素”包括：变量、符号

常量、函数、函数的参数、结构、类、对象等。标识符的组成要符合一定的规则：

- (1) 标识符是一个以字母或下画线开头的，由字母、数字、下画线组成的字符串；
- (2) 标识符不能与任意一个关键字同名；
- (3) 标识符中的字母区分大小写；

(4) 标识符不宜过长。C++语言一般设定一个标识符的有效长度为 32 个字符。如果程序中的标识符超过这个长度，超出的部分被忽略不计。

1.5.3 字面常量

常量是指在程序中固定不变的值。常量一般有两种表示形式，即符号常量和字面常量。C++语言中符号常量的名字就是一个标识符，而字面常量却是一类特殊的单词。字面常量分为整型、浮点型、字符型和字符串型四类。

1.5.4 运算符

运算符是对程序中的数据进行操作的一类单词。C++语言中运算符有以下 4 种：

- (1) 单字符组成的运算符，例如：+，-，*，/ 等；
- (2) 双字符组成的运算符，例如：<<，<=，&&，*=，-> 等；
- (3) 三个字符组成的运算符，例如：<<=，>>= 等；
- (4) 关键字运算符：new，delete，sizeof 等。

1.5.5 标点符号

标点符号是在程序中起分隔内容和界定范围作用的一类单词。C++语言的标点符号如表 1-4 所示。

表 1-4 C++语言的标点符号

标点符号	描述
(空格)	语句中各成分之间的分隔符
； (分号)	语句的结束符
' (单引号)	字符常量的起止标记符
" (双引号)	字符串常量的起止标记符
# (井字号)	预处理命令的开始标记符
{ (左花括号)	复合语句的开始标记符
} (右花括号)	复合语句的结束标记符
// (双斜杠)	行注释的开始标记符
/* (斜杠和星号)	块注释的开始标记符
*/ (星号和斜杠)	块注释的结束标记符



真题链接

【例 1-1】定义无符号整数类为 UInt，下面可以作为类 UInt 实例化值的是_____。(2011.03)

- A. -369 B. 369 C. 0.369 D. 整数集合{1,2,3,4,5}

分析：本题考查的知识点是无符号整数。这种类型不存放负值，且如果一个表达式中同时出现普通的整数和无符号类型的整数，普通的整数就会被转换成无符号类型。

答案：B

1.6 C++程序的基本框架

考核说明：由于 C++ 语言既支持结构化程序设计，又支持面向对象程序设计，所以它同时具有结构化程序设计和面向对象程序设计两种基本框架。本节主要介绍这两种框架。

1.6.1 结构化程序设计框架

在 C++ 的结构化程序设计框架中，函数是程序的基本组成单元。程序的主体通常由若干函数定义构成。函数是程序中完成一定功能的模块。各个函数之间通过参数、返回值和全局变量来进行数据通信。

使用结构化程序设计方法编写出的 C++ 程序包括一个主函数和若干用户定义的函数。函数与函数之间是相对独立的并且是并行的，函数之间可以相互调用。同一个函数可以被一个或多个函数调用多次。主函数由系统调用，在主函数中调用其他函数。

一般地，C++ 程序的结构化程序设计框架可以表示为：

```
<函数 1 的声明>
<函数 2 的声明>
...
<函数 n 的声明>
<主函数 main()>
<函数 1 的定义>
<函数 2 的定义>
...
<函数 n 的定义>
```

1.6.2 面向对象程序设计框架

20 世纪 80 年代，Novell 公司和 Banyan 公司进入 NOS 市场。进入 20 世纪 90 年代，IBM 公司和 Microsoft 公司也成为 NOS 的主要厂商，网络环境呈现出多姿多彩的开放局面。

在 C++ 的面向对象程序设计框架中，类是程序的基本组成单元。类可以把数据和函数封装在一起，用以描述事物的属性和对事物的操作。类与类之间一方面通过封装而具有明确的独立性；另一方面又通过成员、友元、参数、继承和派生等关系，达到相互通信和联络的目的，从而形成一个有机的整体。

在结构化程序设计中，除了主函数之外，所有函数之间一律“平等”，没有层次关系可言，当程序规模稍大时，就会使整体结构变得相当混乱。相反，在面向对象程序设计中，类与类之间能够按照逻辑关系组成有条理的层次结构，从而使一个复杂程序变得有“纲”可循。

一般地，一个 C++ 程序的面向对象程序设计框架可以表示为：

```
<类 1 的定义>
<类 2 的定义>
...
<类 n 的定义>
<主函数 main()>
<类 1 的成员函数定义>
<类 2 的成员函数定义>
...
<类 n 的成员函数定义>
```



真题链接

【例 1-2】结构化程序所要求的基本结构不包括_____。（2011.03）

- | | |
|-------------|-------------|
| A. 顺序结构 | B. GOTO 跳转 |
| C. 选择（分支）结构 | D. 重复（循环）结构 |

分析：本题考查的知识点是程序结构。按照结构化程序设计的观点，任何算法功能都可以通过由程序模块组成的三种基本程序结构的组合：顺序结构、选择结构和循环结构来实现。

答案：B

1.7 C++程序的开发过程



考核说明：本节主要介绍如何使用 Visual C++ 6.0 集成开发环境编写和调试程序。开发一个 C++ 程序的过程通常包括编辑、编译、链接、运行和调试等步骤。

1.7.1 编辑

编辑是 C++ 程序开发过程的第 1 步，主要包括文本的输入和修改。在 Visual C++ 6.0 集成开发环境中，用户可以使用编辑窗口来进行 C++ 程序的编辑工作。当用户完成了 C++ 程序的编辑时，应将输入的程序文本保存为以 .cpp 为扩展名的文件（保存 C++ 头文时应以 .h 为扩展名）。

1.7.2 编译

编译是由源程序文件转换到目标文件的过程。在 Visual C++ 6.0 集成开发环境中，可以使用编译（Compile）命令将一个 .cpp 源文件转换成一个 .obj 的目标文件。

1.7.3 链接

链接是将目标代码变成可执行程序（.exe 文件）的过程。在 Visual C++ 6.0 集成开发环境中，可以使用生成（Build）命令进行链接。

1.7.4 运行和调试

运行可执行程序，查看运行结果。在 Visual C++ 6.0 集成开发环境中，用户可以使用执行（Execute）