

SAMS

C++

Primer Plus
(第四版) 中文版

人民邮电出版社
POSTS & TELECOMMUNICATIONS PRESS

[美] Stephen Prata 著
孙建春 韦强 译

C++ Primer Plus (第四版) 中文版

[美] Stephen Prata 著

孙建春 韦强 译

人民邮电出版社

图书在版编目 (CIP) 数据

C++ Primer Plus 中文版: 第4版 / (美) 普拉塔 (Prata, S.) 著; 孙建春, 韦强译. —北京: 人民邮电出版社, 2002.7

ISBN 7-115-10335-6

I. C.... II. ①普...②孙...③韦... III. C语言—程序设计 IV. TP312
中国版本图书馆 CIP 数据核字 (2002) 第 040200 号

版权声明

Stephen Prata: C++ Primer Plus, Fourth Edition

Copyright © 2002 by Sams Publishing.

Authorized translation from the English language edition published by Sams.

All rights reserved.

本书中文简体字版由美国 Sams 出版公司授权人民邮电出版社出版。未经出版者书面许可, 对本书任何部分不得以任何方式复制或抄袭。

版权所有, 侵权必究。

C++ Primer Plus (第四版) 中文版

- ◆ 著 [美] Stephen Prata
译 孙建春 韦强
责任编辑 李际
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
读者热线 010-67180876
北京汉魂图文设计有限公司制作
北京鸿佳印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 48
字数: 1 550 千字 2002 年 7 月第 1 版
印数: 1-4 000 册 2002 年 7 月北京第 1 次印刷

著作权合同登记 图字: 01-2001-5022 号

ISBN 7-115-10335-6/TP·2898

定价: 69.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

内 容 提 要

C++是在 C 语言基础上开发的一种集面向对象编程、通用编程和传统的过程化编程于一体的编程语言，是 C 语言的超集。本书是根据 1998 年的 ISO/ANSI C++标准编写的，通过大量短小精悍的程序详细而全面地阐述了 C++的基本概念和技术。

全书分 17 章和 10 个附录。分别介绍了 C++程序的运行方式、基本数据类型、复合数据类型、循环和关系表达式、分支语句和逻辑操作符、函数重载和函数模板、内存模型和名称空间、类的设计和使用、多态、虚函数、动态内存分配、继承、代码重用、友元、异常处理技术、string 类和标准模板库、输入/输出等内容。

本书针对 C++初学者，书中从 C 语言基础知识开始介绍，然后在此基础上详细阐述 C++新增的特性，因此不要求读者有 C 语言方面的背景知识。本书可作为高等院校教授 C++课程的教材，也可供初学者自学 C++时使用。

MAJ572/52

作者简介

Stephen Prata 在加州肯特菲尔得的马林学院教授天文、物理和计算机科学。他毕业于加利福尼亚技术学院，在加利福尼亚大学伯克利分校获得博士学位。Stephen 本人或与他人合作作为 The Waite Group 撰写了很多书。他撰写过《The Waite Group's New Primer Plus》，这本书获得了计算机出版联合会 1990 年度最佳“**How-to**”计算机图书奖；他所撰写的《The Waite Group's C++ Primer Plus》一书，获得了计算机出版联合会 1991 年度最佳“**How-to**”计算机图书奖的提名。

前 言

第 4 版前言

学习 C++ 是一次探索之旅，因为这种语言容纳了好几种编程模式，其中包括面向对象编程、通用编程和传统的过程化编程。随着新特性的不断添加，C++ 一度成为一个活动目标，不过现在有了 1998 年的 ISO/ANSI C++ 标准后，已经稳定下来了。现代编译器支持该标准要求的多数或全部特性，程序员要花时间来自来习惯这些特性的应用。本书第 4 版是按 ISO/ANSI 标准编写的，将介绍这种成熟的 C++ 版本。

本书在介绍 C++ 特性的同时，讨论了基本 C 语言，使二者成为有机的整体。书中介绍了 C++ 的基本概念，并通过短小精悍的程序来阐明，这些程序都很容易复制和试验。还介绍了输入和输出、如何让程序执行重复性任务、如何让程序做出选择、处理数据的多种方式以及如何使用函数等内容。另外，还讲述了 C++ 在 C 语言的基础上新增的许多特性，其中包括：

- 类和对象。
- 继承。
- 多态、虚函数和 RTTI（运行阶段类型识别）。
- 函数重载。
- 引用变量。
- 通用（或独立于类型）的编程，这种技术是由模板和标准模板库（STL）提供的。
- 处理错误条件的异常机制。
- 管理函数、类和变量名的名称空间。

本书在传授知识方面有几个优点。大约 20 年前，《C Primer Plus》一书开创了优良的初级教程传统，本书建立在这样的基础之上，吸收了其中很多成功的理念：

- 初级教程应当是友好的、便于使用的指南。
- 初级教程不要求读者已经熟悉相关的编程概念。
- 初级教程强调的是“实践性”学习，通过简短、容易输入的范例同时阐述一个或多个概念。
- 初级教程用示意图来解释概念。
- 初级教程提供问题和练习来检验读者对知识的理解，从而适于自学或课堂教学。

本书可以帮助读者理解这种用途广泛的语言，并学习如何使用它。

- 对何时使用某些特性，例如何时使用公共继承来建立 is-a 关系，提供了概念方面的指导。
- 阐释了常用的 C++ 编程理念和技术。
- 提供了大量的附注，如提示、警告、注意等。

我们尽最大的努力使本书简单、明了、生动有趣。我们的目标是：读者阅读本书后，能够编写出可靠、高效的程序，并且觉得这是一种享受。

与前几版一样，本书讲述通用的 C++，并不针对任何特殊类型的计算机、操作系统或编译器。所有的程序都经过 CodeWarrior Pro 6（Macintosh 和 Windows）和 Microsoft Visual C++ 6.0 的测试，多数程序经过 Borland C++ Command-Line Compiler 5.5、Gnu g++ 2.9.5（针对 DOS）、Comeau C/C++ 4.42.5（针对 Linux）的测试。只有少数程序会受到编译器不一致问题的影响。

C++ 为程序员提供了丰富多彩的内容。祝您学习愉快！

对教师的提示

本书第 4 版的宗旨之一是：提供一本既可以用作自学书也可以用作教材的书籍。下面是本书在支持教

学方面的一些特征:

- 本书介绍的是通用 C++，不依赖于特定的实现。
- 本书内容记录了 ISO/ANSI C++ 标准委员会的工作，并讨论了模板、标准模板库、string 类、异常、RTTI 和名称空间。
- 本书不要求学生了解 C 语言，但如果有一定的编程经验则更好。
- 本书内容经过了精心安排，前几章可以作为对 C 预备知识的复习章节一带而过。
- 各章都有复习题和编程练习。
- 本书介绍的一些主题很适于计算机科学课程，包括抽象数据类型、堆栈、队列、简单链表、模拟、通用编程以及使用递归来实现分而治之的策略。
- 各章都非常简短，用一周甚至更短的时间就可以学完。
- 本书讨论了何时使用具体的特性以及如何使用它们。例如，把 is-a 关系的公有继承同组合、has-a 关系的私有继承联系起来，讨论了何时应使用虚函数以及何时不应使用。

本书的内容结构

本书分为 17 章和 10 个附录。

第 1 章：预备知识

本章介绍了 Bjarne Stroustrup 如何通过向 C 语言的基础上添加对面向对象编程的支持，来创造 C++ 编程语言的。讨论了过程化语言（如 C 语言）与面向对象语言（如 C++）之间的区别。读者将了解 ANSI/ISO 在制定 C++ 标准方面所做的工作。本章还讨论了创建 C++ 程序的技巧，介绍了当前几种 C++ 编译器使用的方法。最后，本章介绍了本书的一些约定。

第 2 章：开始学习 C++

本章介绍创建简单 C++ 程序的步骤。读者可以学习到 main() 函数扮演的角色以及 C++ 程序使用的一些语句。读者将使用预定义的 cout 和 cin 对象来实现程序输出和输入，学习如何创建和使用变量。最后，本章还将介绍函数——C++ 的编程模块。

第 3 章：处理数据

C++ 提供了内置类型来存储两种数据：整数（没有小数的数字）和浮点数（带小数的数字）。为满足程序员的各种需求，C++ 为每一种数据都提供了几个类型。本章将要讨论这些类型，包括创建变量和编写各种类型的常量。另外，还将讨论 C++ 是如何处理不同类型之间的隐式和显式转换的。

第 4 章：复合类型

C++ 允许程序员使用基本的内置类型来创建更复杂的类型。最高级的形式是类，这将在第 10、11、12、13 和 14 章讨论。本章讨论其他形式，包括数组（存储多个同类型的值）、结构（存储多个不同类型的值）、指针（标识内存位置）。读者还将学习如何创建和存储文本字符串及如何处理文本输入和输出。最后，还将学习 C++ 处理内存分配的一些方法，其中包括用于显式地管理内存的 new 和 delete 操作符。

第 5 章：循环和关系表达式

程序经常需要执行重复性操作，为此 C++ 提供了 3 种循环结构：for 循环、while 循环和 do while 循环。这些循环必须知道何时终止，C++ 的关系操作符使程序员能够创建测试来引导循环。本章还将介绍如何创建逐字符地读取和处理输入的循环。最后，读者将学习如何创建二维数组以及如何使用嵌套循环来处理它们。

第 6 章：分支语句和逻辑操作符

如果程序可以根据实际情况调整执行，我们就说程序能够智能地行动。在本章，读者将了解到如何使用 if、if else 和 switch 语句及条件操作符来控制程序流程，学习如何使用逻辑操作符来表达决策测试。另外，本章还将介绍确定字符关系（如测试字符是数字还是非打印字符）的函数库 ctype。

第 7 章：函数——C++ 的编程模块

函数是 C++ 的基本编程部件。本章重点介绍 C++ 函数与 C 函数共同的特性。具体地说，读者将复习函数定义的通用格式，了解函数原型是如何提高程序可靠性的。同时，还将学习如何编写函数来处理数组、

字符串和结构。还要学习有关递归的知识（即函数在什么情况下调用自身）以及如何用它来实现分而治之的策略。最后将介绍函数指针，它使程序员能够通过函数参数来命令函数使用另一个函数。

第 8 章：函数探幽

本章将探索 C++ 中函数新增的特性。读者将学习内联函数，它可以提高程序的执行速度，但会增加程序的长度；还将使用引用变量，它们提供了另一种将信息传递给函数的方式。默认参数使函数能够自动为函数调用中省略的函数参数提供值。函数重载使程序员能够创建多个参数列表不同的同名函数。类设计中经常使用这些特性。另外，读者还将学习函数模板，它们使程序员能够指定相关函数族的设计。

第 9 章：内存模型和名称空间

本章讨论如何创建多文件程序，介绍分配内存的各种方式、管理内存的各种方式以及作用域、链接、名称空间，这些内容决定了变量在程序的哪些部分是可见的。

第 10 章：对象和类

类是用户定义的类型，对象是类的实例，如变量。本章介绍面向对象编程和类设计。对象声明描述的是存储在对象中的信息以及可对对象执行的操作（类方法）。对象的某些组成部分对于外界来说是可见的（公有部分），而某些部分却是隐藏的（私有部分）。特殊的类方法（构造函数和析构函数）在对象创建和释放时发挥作用。在本章中，读者将学习所有这些内容以及其他类知识，了解如何使用类来实现抽象数据类型（ADT），如堆栈。

第 11 章：使用类

在本章中，读者将深入了解类。首先了解操作符重载，它使程序员能够定义与类对象一起使用的操作符，如+。还将学习友元函数，这些函数可以访问外部世界不可访问的类数据。同时还将了解一些构造函数和重载操作符成员函数是如何被用来管理类类型转换的。

第 12 章：类和动态内存分配

一般来说，让类成员指向动态分配的内存很有用。如果程序员在类构造函数中使用 new 来分配动态内存，就有责任提供适当的析构函数，定义显式复制构造函数和显式赋值操作符。本章介绍了在程序员没有提供显式定义时，将如何隐式地生成成员函数以及这些成员函数的行为。读者还将通过使用对象指针，了解队列模拟问题，扩充类方面的知识。

第 13 章：类继承

在面向对象编程中，继承是功能最强大的特性之一，通过继承，派生类可以继承基类的特性，可重用基类代码。本章讨论公有继承，这种继承模拟了 is-a 关系，即派生对象是基对象的特例。例如，物理学家是科学家的特例。有些继承关系是多态的，这意味着相同的方法名称可能导致依赖于对象类型的行为。要实现这种行为，需要使用一种新的成员函数——虚函数。有时，使用抽象基类是实现继承关系的最佳方式。本章讨论了这些问题，说明了公有继承在什么情况下合适，在什么情况下不合适。

第 14 章：在 C++ 中的代码重用

公有继承只是代码重用的方式之一。本章将介绍其他几种方式。如果一个类包含了另一个类的对象，则称为包含。包含可以用来模拟 has-a 关系，其中一个类包含另一个类的对象。例如，汽车有马达。也可以使用私有继承和保护继承来模拟这种关系。本章说明了各种方法之间的区别。同时，读者还将学习类模板，它使程序员能够使用通用类型来定义类，然后使用模板来根据具体类型创建特定的类。例如，堆栈模板使程序员能够创建整数堆栈或字符串堆栈。最后，本章还将介绍多重公有继承，使用这种方式，一个类可以从多个类派生而来。

第 15 章：友元、异常和其他

本章扩展了对友元的讨论，讨论了友元类和友元成员函数。然后从异常开始介绍了 C++ 的几项新特性。异常为处理程序异常提供了一种机制，如函数参数值不正确或内存耗尽等。读者还将学习 RTTI（运行阶段类型信息），这种机制用来确定对象类型。最后，本章还将介绍一种更安全的方法来替代不受限制的强制类型转换。

第 16 章：string 类和标准模板库

本章将讨论 C++ 语言中新增的一些类库。对于传统的 C-风格字符串来说，string 类是一种方便且功能

强大的替代方式。auto_ptr 类帮助管理动态分配的内存。标准模板库 (STL) 提供了几种类容器 (包括数组、队列、链表、集合和映射) 的模板表示。它还提供了高效的通用算法库, 这些算法可用于 STL 容器, 也可用于常规数组。

第 17 章: 输入、输出和文件

本章复习 C++ I/O, 并讨论如何格式化输出。读者将要学习如何使用类方法来确定输入或输出流的状态, 了解输入类型是否不匹配或是否检测到了文件尾。C++ 使用继承来派生用于管理文件输入和输出的类。读者将学习如何打开文件, 以进行输入和输出, 如何在文件中追加数据, 如何使用二进制文件, 如何获得对文件的随机访问权。最后, 还将学习如何使用标准的 I/O 方法来读取和写入字符串。

附录 A: 计数系统

本附录讨论八进制数、十六进制数和二进制数。

附录 B: C++ 关键字

本附录列出了 C++ 关键字。

附录 C: ASCII 字符集

本附录列出了 ASCII 字符集及其十进制、八进制、十六进制和二进制表示。

附录 D: 操作符优先级

本附录按优先级从高到低的顺序列出了 C++ 的操作符。

附录 E: 其他操作符

本附录总结了正文中没有介绍的其他 C++ 操作符, 如按位操作符等。

附录 F: string 模板类

本附录总结了 string 类方法和函数。

附录 G: STL 方法和函数

本附录总结了 STL 容器方法和通用的 STL 算法函数。

附录 H: 精选读物

本附录列出一些参考书, 帮助读者深入了解 C++。

附录 I: 转换为 ANSI/ISO 标准 C++

本附录提供了从 C 和老式 C++ 实现到标准 C++ 的转换指南。

附录 J: 复习题答案

本附录提供每章结尾的复习题的答案。

目 录

第 1 章 预备知识	1
1.1 学习 C++	2
1.2 C++ 简史	2
1.2.1 C 语言	2
1.2.2 C 语言编程原理	3
1.2.3 面向对象编程	4
1.2.4 通用编程	4
1.2.5 C++	5
1.3 可移植性和标准	6
1.4 程序创建的技巧	6
1.4.1 创建源代码	7
1.4.2 编译和链接	8
1.5 本书约定	11
1.6 我们的系统	12
第 2 章 开始学习 C++	13
2.1 进入 C++	14
2.1.1 main () 函数	15
2.1.2 C++ 注释	17
2.1.3 C++ 预处理器和 iostream 文件	17
2.1.4 头文件名	18
2.1.5 名称空间	18
2.1.6 使用 cout 进行 C++ 输出	19
2.1.7 C++ 源代码的格式	20
2.2 有关 C++ 语句的其他知识	22
2.2.1 声明语句和变量	22
2.2.2 赋值语句	23
2.2.3 cout 的新花样	23
2.3 其他 C++ 语句	24
2.3.1 使用 cin	25
2.3.2 再谈 cout	25
2.3.3 类简介	25
2.4 函数	26
2.4.1 使用有返回值的函数	26
2.4.2 函数变体	29
2.4.3 用户定义的函数	29
2.4.4 用户定义的有返回值的函数	31
2.4.5 语句小结	33
2.5 总结	33
2.6 复习题	34
2.7 编程练习	34

第 3 章 处理数据	36
3.1 简单变量	37
3.1.1 变量名	37
3.1.2 整型	38
3.1.3 short、int 和 long	38
3.1.4 无符号类型	42
3.1.5 应使用哪种类型	43
3.1.6 整型常量	44
3.1.7 C++ 如何确定常量的类型	45
3.1.8 char 类型: 字符和小整数	45
3.1.9 新的 bool 类型	50
3.2 const 限定符	51
3.3 浮点数	51
3.3.1 书写浮点数	52
3.3.2 浮点类型	53
3.3.3 浮点常量	54
3.3.4 浮点数的优缺点	55
3.4 C++ 算术操作符	55
3.4.1 什么顺序: 操作符优先级和结合性	56
3.4.2 除法分支	57
3.4.3 求模操作符	58
3.4.4 类型转换	59
3.5 总结	62
3.6 复习题	62
3.7 编程练习	63
第 4 章 复合类型	64
4.1 数组	65
4.2 字符串	68
4.2.1 字符串拼接	69
4.2.2 在数组中使用字符串	69
4.2.3 字符串输入	71
4.2.4 混合输入字符串和数字	74
4.3 结构	75
4.3.1 程序说明	77
4.3.2 其他结构属性	78
4.3.3 结构数组	79
4.3.4 位字段	80
4.4 共用体	80
4.5 枚举	81
4.5.1 设置枚举量的值	82
4.5.2 枚举的取值范围	83
4.6 指针和自由存储空间	83
4.6.1 声明和初始化指针	85

4.6.2	指针和数字	87
4.6.3	使用 new 来分配内存	87
4.6.4	使用 delete 来释放内存	89
4.6.5	使用 new 来创建动态数组	90
4.7	指针、数组和指针算术	91
4.7.1	指针和字符串	95
4.7.2	使用 new 创建动态结构	98
4.7.3	自动存储、静态存储和动态存储	100
4.7.4	自动变量	100
4.8	总结	101
4.9	复习题	102
4.10	编程练习	103
第 5 章	循环和关系表达式	104
5.1	for 循环	105
5.1.1	for 循环的组成部分	106
5.1.2	回到 for 循环	110
5.1.3	修改步长	111
5.1.4	使用 for 循环访问字符串	111
5.1.5	递增操作符 (++) 和递减操作符 (--)	112
5.1.6	组合赋值操作符	113
5.1.7	复合语句 (语句块)	114
5.1.8	逗号操作符 (或其他句法技巧)	115
5.2	关系表达式	117
5.2.1	可能犯的错误	118
5.2.2	字符串比较	119
5.3	while 循环	121
5.3.1	for 与 while	122
5.3.2	等待一段时间	124
5.4	do while 循环	125
5.5	循环和文本输入	127
5.5.1	使用原始的 cin 进行输入	127
5.5.2	使用 cin.get (char) 进行补救	128
5.5.3	使用哪一个 cin.get ()	129
5.5.4	文件尾条件	129
5.5.5	另一个 cin.get ()	131
5.6	嵌套循环和二维数组	133
5.6.1	初始化二维数组	134
5.7	总结	136
5.8	复习题	136
5.9	编程练习	137
第 6 章	分支语句和逻辑操作符	139
6.1	if 语句	140
6.1.1	if else 语句	141

6.1.2	格式化 if else 语句	142
6.1.3	if else if else 结构	143
6.2	逻辑表达式	144
6.2.1	逻辑 OR 操作符:	144
6.2.2	逻辑 AND 操作符: &&	146
6.2.3	逻辑 NOT 操作符: !	149
6.2.4	逻辑操作符细节	150
6.3	cctype 字符函数库	151
6.4	?: 操作符	152
6.5	switch 语句	153
6.5.1	将枚举量用作标签	156
6.5.2	switch 和 if else	157
6.6	break 和 continue 语句	157
6.7	读取数字的循环	159
6.8	总结	162
6.9	复习题	162
6.10	编程练习	164
第 7 章	函数——C++的编程模块	166
7.1	复习有关函数的基本知识	167
7.1.1	定义函数	167
7.1.2	函数原型和函数调用	169
7.2	函数参数和按值传递	171
7.2.1	多个参数	172
7.2.2	另外一个接受两个参数的函数	174
7.3	函数和数组	175
7.3.1	函数如何使用指针来处理数组	176
7.3.2	将数组作为参数意味着什么	177
7.3.3	其他数组函数范例	179
7.3.4	使用数组区间的函数	182
7.3.5	指针和 const	184
7.4	函数和二维数组	186
7.5	函数和 C-风格字符串	187
7.5.1	返回字符串的函数	188
7.6	函数和结构	189
7.6.1	传递和返回结构	190
7.6.2	另一个范例	191
7.6.3	传递结构的地址	194
7.7	递归	196
7.8	函数指针	198
7.8.1	函数指针的基础知识	198
7.9	总结	201
7.10	复习题	201
7.11	编程练习	202

第 8 章 函数探幽	205
8.1 内联函数	206
8.2 引用变量	208
8.2.1 创建引用变量	208
8.2.2 将引用用作函数参数	210
8.2.3 引用的属性和特别的地方	212
8.2.4 将引用用于结构	215
8.3 默认参数	218
8.4 函数重载（多态）	220
8.4.1 重载范例	221
8.4.2 何时使用函数重载	223
8.5 函数模板	224
8.5.1 重载的模板	226
8.5.2 显式具体化	228
8.5.3 实例化和具体化	231
8.5.4 使用哪一个函数	232
8.6 总结	235
8.7 复习题	236
8.8 编程练习	237
第 9 章 内存模型和名称空间	239
9.1 单独编译	240
9.2 存储持续、作用域和链接性	243
9.2.1 作用域和链接	244
9.2.2 自动存储持续	244
9.2.3 静态持续变量	248
9.2.4 说明符和限定符	255
9.2.5 函数和链接性	256
9.2.6 语言链接	257
9.2.7 存储方案和动态分配	257
9.3 名称空间	258
9.3.1 传统的 C++ 名称空间	258
9.3.2 新的名称空间特性	259
9.3.3 名称空间范例	264
9.3.4 名称空间及其前途	267
9.4 总结	267
9.5 复习题	268
9.6 编程练习	270
第 10 章 对象和类	272
10.1 过程性编程和面向对象编程	273
10.2 抽象和类	274
10.2.1 类型是什么	274
10.2.2 类	274

10.2.3	实现类成员函数	277
10.2.4	使用类	281
10.2.5	小结	283
10.3	类的构造函数和析构函数	284
10.3.1	声明和定义构造函数	284
10.3.2	使用构造函数	285
10.3.3	默认构造函数	285
10.3.4	析构函数	286
10.3.5	改进 Stock 类	287
10.3.6	构造函数和析构函数小结	292
10.4	this 指针	292
10.5	对象数组	297
10.6	类作用域	299
10.6.1	作用域为整个类的常量	300
10.7	抽象数据类型	300
10.8	总结	304
10.9	复习题	305
10.10	编程练习	305
第 11 章	使用类	308
11.1	操作符重载	309
11.2	计算时间	310
11.2.1	添加加法操作符	312
11.2.2	重载限制	314
11.2.3	其他重载操作符	315
11.3	友元	318
11.3.1	创建友元	319
11.3.2	常用的友元: 重载 << 操作符	319
11.4	重载操作符: 成员函数和非成员函数之比较	324
11.5	再谈重载: 矢量类	324
11.5.1	使用状态成员	330
11.5.2	重载其他的操作符	331
11.5.3	对实现的说明	333
11.5.4	使用 Vector 类来模拟随机行走	333
11.6	类的自动转换和强制类型转换	335
11.6.1	转换函数	339
11.6.2	转换和友元	342
11.7	总结	345
11.8	复习题	346
11.9	编程练习	346
第 12 章	类和动态内存分配	348
12.1	动态内存和类	349
12.1.1	复习范例和静态类成员	349
12.1.2	隐式成员函数	355

12.1.3	改进后的新字符串类	360
12.1.4	在构造函数中使用 new 时应注意的事项	368
12.1.5	使用指向对象的指针	370
12.1.6	复习各种技术	374
12.2	队列模拟	374
12.2.1	队列类	375
12.2.2	客户类	382
12.2.3	模拟	385
12.3	总结	389
12.4	复习题	390
12.5	编程练习	391
第 13 章	类继承	394
13.1	一个简单的基类	395
13.1.1	派生一个类	396
13.1.2	构造函数：访问权限的考虑	398
13.1.3	使用派生类	400
13.2	特殊关系	402
13.3	继承——is-a 关系	403
13.4	多态公有继承	404
13.4.1	开发两个类	405
13.4.2	静态联编和动态联编	414
13.4.3	指针和引用类型兼容性	414
13.4.4	虚拟成员函数和动态联编	414
13.4.5	为什么有两种类型的联编	415
13.4.6	虚拟函数的工作原理	416
13.4.7	虚拟函数注意事项	417
13.5	访问控制——protected	418
13.6	抽象基类	419
13.6.1	应用 ABC 概念	421
13.7	继承和动态内存分配	424
13.7.1	第一种情况：派生类不使用 new	425
13.7.2	第二种情况：派生类使用 new	425
13.7.3	使用动态内存分配和友元的继承范例	427
13.8	类设计回顾	431
13.8.1	编译器生成的成员函数	431
13.8.2	其他的类方法	432
13.8.3	公有继承的考虑因素	434
13.8.4	类函数小结	436
13.9	总结	436
13.10	复习题	437
13.11	编程练习	438
第 14 章	C++ 中的代码重用	440
14.1	包含对象成员类	441

14.1.1	ArrayDb 类	442
14.1.2	Student 类范例	446
14.2	私有继承	450
14.2.1	Student 类范例 (新版本)	450
14.3	多重继承	455
14.3.1	有多少 Worker	459
14.3.2	哪个方法	461
14.3.3	多重继承小结	469
14.4	类模板	470
14.4.1	定义类模板	470
14.4.2	使用模板类	472
14.4.3	深入探讨模板类	474
14.4.4	数组模板范例和非类型参数	478
14.4.5	模板多功能性	480
14.4.6	模板的具体化	483
14.4.7	成员模板	485
14.4.8	将模板用作参数	487
14.4.9	模板类和友元	488
14.5	总结	493
14.6	复习题	494
14.7	编程练习	495
第 15 章	友元、异常和其他	498
15.1	友元	499
15.1.1	友元类	499
15.1.2	友元成员函数	503
15.1.3	其他友好关系	505
15.2	嵌套类	506
15.2.1	嵌套类和访问权限	508
15.2.2	模板中的嵌套	509
15.3	异常	511
15.3.1	异常机制	514
15.3.2	异常的多功能性	516
15.3.3	异常和类	521
15.3.4	异常和继承	526
15.3.5	exception 类	529
15.3.6	bad_alloc 异常和 new	530
15.3.7	异常何时会迷失方向	531
15.3.8	有关异常的注意事项	533
15.4	RTTI	534
15.4.1	RTTI 的用途	534
15.4.2	RTTI 的工作原理	534
15.5	类型转换操作符	541
15.6	总结	543
15.7	复习题	543