

国外计算机科学经典教材

THOMSON

C++ Programming

From Problem Analysis to Program Design

Second Edition

C++ 基础教程

——从问题分析到程序设计

(第2版)

(美) D. S. Malik 著
曹蓉蓉 宋红 译
郑岩 宋美娜 审校



清华大学出版社

国外计算机科学经典教材

C++基础教程——从问题分析到程序设计

(第 2 版)

(美) D. S. Malik 著
曹蓉蓉 宋红 译
郑岩 宋美娜 审校

清华大学出版社

北 京

D. S. Malik

C++ Programming: From Problem Analysis to Program Design, Second Edition

EISBN: 0-619-16042-x

Copyright © 2004 by Course Technology, a division of Thomson Learning.

Original language published by Thomson Learning (a division of Thomson Learning Asia Pte Ltd). All rights reserved.

本书原版由汤姆森学习出版集团出版。版权所有，盗印必究。

Tsinghua University Press is authorized by Thomson Learning to publish and distribute exclusively this Simplified Chinese edition. This edition is authorized for sale in the People's Republic of China only (excluding Hong Kong, Macao SAR and Taiwan). Unauthorized export of this edition is a violation of the Copyright Act. No part of this publication may be reproduced or distributed by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

本中文简体字翻译版由汤姆森学习出版集团授权清华大学出版社独家出版发行。此版本仅限在中华人民共和国境内(不包括中国香港、澳门特别行政区及中国台湾地区)销售。未经授权的本书出口将被视为违反版权法的行为。未经出版者预先书面许可，不得以任何方式复制或发行本书的任何部分。

981-265-890-4

北京市版权局著作权合同登记号 图字: 01-2004-4751

版权所有，翻印必究。举报电话: 010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

本书防伪标签采用特殊防伪技术，用户可通过在图案表面涂抹清水，图案消失，水干后图案复现；或将表面膜揭下，放在白纸上用彩笔涂抹，图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

C++基础教程——从问题分析到程序设计(第2版)/(美) 马里克(Malik,D.S.) 著; 曹蓉蓉, 宋红 译; 郑岩, 宋美娜 审校. —北京: 清华大学出版社, 2006.4

书名原文: C++ Programming: From Problem Analysis to Program Design, Second Edition

(国外计算机科学经典教材)

ISBN 7-302-12168-0

I. C… II. ①马…②曹…③宋…④郑…⑤宋… III. C 语言—程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(2005)第 139697 号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦
http://www.tup.com.cn 邮 编: 100084
社 总 机: 010-62770175 客户服务: 010-62776969

组稿编辑: 曹 康

文稿编辑: 李 阳

封面设计: 康 博

版式设计: 康 博

印 刷 者: 清华大学印刷厂

装 订 者: 三河市李旗庄少明装订厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印张: 63.5 字数: 1626 千字

版 次: 2006 年 4 月第 1 版 2006 年 4 月第 1 次印刷

书 号: ISBN 7-302-12168-0/TP·7846

印 数: 1~3000

定 价: 118.00 元

出版说明

近年来,我国高等学校的计算机学科教育进行了较大的改革,急需一批门类齐全、具有国际水平的计算机经典教材,以适应当前的教学需要。引进国外经典教材,可以了解并吸收国际先进的教学思想和教学方法,使我国的计算机学科教育能够与国际接轨,从而培育更多具有国际水准的计算机专业人才,增强我国信息产业的核心竞争力。Pearson、Thomson、McGraw-Hill、Springer、John Wiley 等出版集团都是全球最有影响的图书出版机构,它们在高等教育领域也都有着不凡的表现,为全世界的高等学校计算机教学提供了大量的优秀教材。为了满足我国高等学校计算机学科的教学需要,我社计划从这些知名的国外出版集团引进计算机学科经典教材。

为了保证引进版教材的质量,我们在全国范围内组织并成立了“清华大学计算机外版教材编审委员会”(以下简称“编委会”),旨在对引进教材进行审定、对教材翻译质量进行评审。

“编委会”成员皆为全国各类重点院校教学与科研第一线的知名教授,其中许多教授为各校相关院、系的院长或系主任。“编委会”一致认为,引进版教材要能够满足国内各高校计算机教学与国际接轨的需要,要有特色风格,有创新性、先进性、示范性和一定的前瞻性,是真正的经典教材。为了保证外版教材的翻译质量,我们聘请了高校计算机相关专业教学与科研第一线的教师及相关领域的专家担纲译者,其中许多译者为海外留学回国人员。为了尽可能地保留与发扬教材原著的精华,在经过翻译和编辑加工之后,由“编委会”成员对文稿进行审定,以最大程度地弥补和修正在前面一系列加工过程中对教材造成的误差和瑕疵。

由于时间紧迫和能力所限,本套外版教材在出版过程中还可能存在一些不足和遗憾,欢迎广大师生批评指正。同时,也欢迎读者朋友积极向我们推荐各类优秀的国外计算机教材,共同为我国高等学校的计算机教育事业贡献力量。

清华大学出版社

国外计算机科学经典教材

编审委员会

主任委员:

孙家广 清华大学教授

副主任委员:

周立柱 清华大学教授

委员(按姓氏笔画排序):

王成山	天津大学教授
王 珊	中国人民大学教授
冯少荣	厦门大学教授
冯全源	西南交通大学教授
刘乐善	华中科技大学教授
刘腾红	中南财经政法大学教授
吉根林	南京师范大学教授
孙吉贵	吉林大学教授
阮秋琦	北京交通大学教授
何 晨	上海交通大学教授
吴百锋	复旦大学教授
李 彤	云南大学教授
沈钧毅	西安交通大学教授
邵志清	华东理工大学教授
陈 纯	浙江大学教授
陈 钟	北京大学教授
陈道蓄	南京大学教授
周伯生	北京航空航天大学教授
孟祥旭	山东大学教授
姚淑珍	北京航空航天大学教授
徐佩霞	中国科学技术大学教授
徐晓飞	哈尔滨工业大学教授
秦小麟	南京航空航天大学教授
钱培德	苏州大学教授
曹元大	北京理工大学教授
龚声蓉	苏州大学教授
谢希仁	中国人民解放军理工大学教授

前言

本书专门为计算机科学(CS1)C++课程而编写,为教师和学生学生提供了大量新内容。我们的主要目标是启发并激励所有学生学习编程的积极性。学习动机激发学习的兴趣,学习动机和兴趣是学生获得成功的关键因素。本书是 50 多个学期的成功编程教学课堂记录的综合和结晶。

本书首先介绍了一些简短示例、练习题和较长的编程示例,然后列举了一些大型编程示例,这些示例构成了本书的主要内容。实际上,本书根据学生的要求提供素材,简洁明了且具有较强的可读性。书中的大多数示例都来自于课堂上学生的交互练习。

不管从事任何职业,都必须进行不断的实践。学习烹饪的学员需要实践他们的食谱。刚开始学习小提琴的学生需要练习如何拉准音调。刚开始学习编程的学员必须实践解决问题和编写代码。本书并不是一本 C++ 示例书。书中没有简单地列出 C++ 语法,举出示例;而是仔细分析了所有概念背后的“原因”。对于首次介绍的每个主题,都会回答“原因”中的关键问题。这种技巧提供了学习 C++ 的桥梁。学生只有理解“原因”后才能获得学习的兴趣。

过去,学习 C++ 的初级编程人员需要了解另一种编程语言的工作原理。本书假设读者先前不具备任何编程经验。不过,学习本书,读者最好还是需要掌握一些适当的数学背景,例如高等数学的知识。

0.1 第 2 版中的更新

相对于第 1 版,本书有如下改进:

- 第 2 章中包括了创建 C++ 程序的新章节。在学习了 C++ 的基本元素后,读者可以快速学会如何编写 C++ 程序。
- 第 5 章中添加了各种示例,用于说明如何使用嵌套循环来获得所需的结果。
- 在第 7 章中,为了便于阅读,添加了关于值参数和引用参数的章节。
- 在第 1 版中,第 10 章中介绍了二维数组。在本书中,第 9 章中介绍了二维数组。因此,第 9 章现在包括一维数组和多维数组以及字符串。在第 10 章中,添加了排序算法、冒泡排序以及 vector 类的介绍。
- 第 1 版中在第 11 章介绍了递归,本书则将其放在第 17 章介绍且独立成章,该章可以放在第 10 章后面的任何位置。
- 在第 12 章中,介绍了类和数据抽象、UML 图和各种面向对象的设计(OOD)的术语,例如存取器(accessor)函数和增变(mutator)函数。添加了关于类中静态成员的章节。
- 第 16 章“异常处理”是新增的章节。这一章详细介绍了异常处理。
- 在附录 E 中,扩展了文件 I/O 的介绍。在第 3 章中介绍过文件 I/O 后,该附录介绍了二进制和随机访问文件。
- 新增的附录 H 介绍了标准模板库(STL)。

0.2 本书内容

编程语言 C++ 从 C 语言发展而来, 它不再是只针对行业内的语言。许多大学和学院使用 C++ 作为首要的编程语言课程。C++ 结合了结构化编程和面向对象编程, 本书将介绍这两种编程方式。

本书可以简单地划分为两个部分: 结构化编程和面向对象编程。前 11 章组成了结构化编程部分; 第 12~19 章组成面向对象编程部分。然而, 前 11 章是后面学习面向对象编程部分的基础。

在 1998 年 7 月, ANSI/ISO 标准 C++ 被正式批准。本书主要关注 ANSI/ISO 标准 C++。即使标准 C++ 的语法和 ANSI/ISO 标准 C++ 的语法非常类似, 但第 8 章中讨论了一些 ANSI/ISO 标准 C++ 的特性, 这些特性不可用于标准 C++ 中。

第 1 章简要回顾了计算机和编程语言的历史。读者可以快速浏览本章, 并且熟悉计算机的一些硬件组成部分和软件组成部分。本章也描述了结构化编程和面向对象编程。

完成第 2 章的学习后, 学生应该熟悉 C++ 的一些基本原理, 并且准备编写可以执行一些计算的复杂程序。输入/输出是任何编程语言的基础。第 3 章会详细介绍输入/输出。

第 4 章和第 5 章介绍了用于改变执行顺序流程的控制结构。第 6 章和第 7 章研究了用户定义的函数。对于不具备编程经验的读者, 推荐在第 6 章和第 7 章上多花费点时间学习。提供了一些示例来帮助读者理解参数传递和标识符作用域的概念。

第 8 章讨论了用户定义的简单数据类型(枚举类型)、ANSI/ISO 标准 C++ 的命名空间机制以及字符串类型。C 的早先版本没有包括枚举类型。枚举类型的使用范围非常有限; 它们的主要目标是提高程序的可读性。本书按照如下方式编排: 在第一次阅读时, 读者可以跳过关于枚举类型的部分, 而不会影响到连续性, 后面如有需要可以回头浏览这一部分。

第 9 章详细讨论了数组。第 10 章描述了各种查找和排序算法, 并且介绍了 `vector` 类。第 11 章介绍了记录(结构体)。本书中关于结构体的介绍类似于 C 中的结构体。这一章是可选章节; 它并不是学习后面章节的先决条件。

第 12 章开始研究面向对象的编程(OOP), 并且介绍了类。本章的前半部分显示了如何在程序中定义和使用类。本章的第二部分介绍了抽象数据类型(ADT)。本章显示了 C++ 中的类如何作为实现 ADT 的自然方式。第 13 章继续讨论面向对象设计(OOD)和 OOP 的基本原理, 并且讨论继承和组合。本章介绍了 C++ 中的类如何提供用于 OOD 的自然机制, 以及 C++ 如何支持 OOP。第 13 章也讨论了如何找出给定问题中的对象。

第 14 章详细研究了指针。介绍过指针以及如何在程序中使用它们后, 本章重点介绍了具有指针数据成员的类的特性, 以及如何避免它们。第 14 章也讨论了通过虚函数实现的一种多态性类型。

第 15 章继续研究 OOD 和 OOP。本章还特别研究 C++ 中的多态性, 专门讨论了两种类型的多态性: 重载和模板。

第 16 章是第 2 版的新增章节。本章详细讨论了异常处理。

第 17 章介绍并讨论了递归。第 17 章是第 1 版中的第 11 章。此外, 这是一个单独的章节, 因此可以在学习第 10 章后的任何时间研究这一章。

第 18 章和第 19 章专门研究了数据结构。第 18 章详细讨论了链表。第 19 章中则详细讨论栈和队列。这些章节中开发的编程代码具有一般性。这些章节有效使用了 OOD 的基本原理。

附录 A 列出了 C++ 中的关键字。附录 B 显示了 C++ 运算符的优先级和关联性。附录 C 列出了 ASCII(美国信息交换标准码)和 EBCDIC(扩充的二进制编码的十进制交换码)的字符集。附录 D 列出了可以重载的 C++ 运算符。

附录 E 有两个目标。首先,详细讨论了二进制文件。然后描述了 ANSI/ISO 标准 C++ 和标准 C++ 中头文件的命名约定。附录 F 讨论了一些最广泛使用的库例程,并且包括了标准 C++ 的头文件名。附录 G 中的程序输出系统上内置数据类型占用的内存大小。新增的附录 H 介绍了标准模板库,附录 I 则提供了书中部分练习题的答案。

0.3 如何阅读本书

可以通过各种方法使用本书。图 1 显示了各章节之间的相关性。

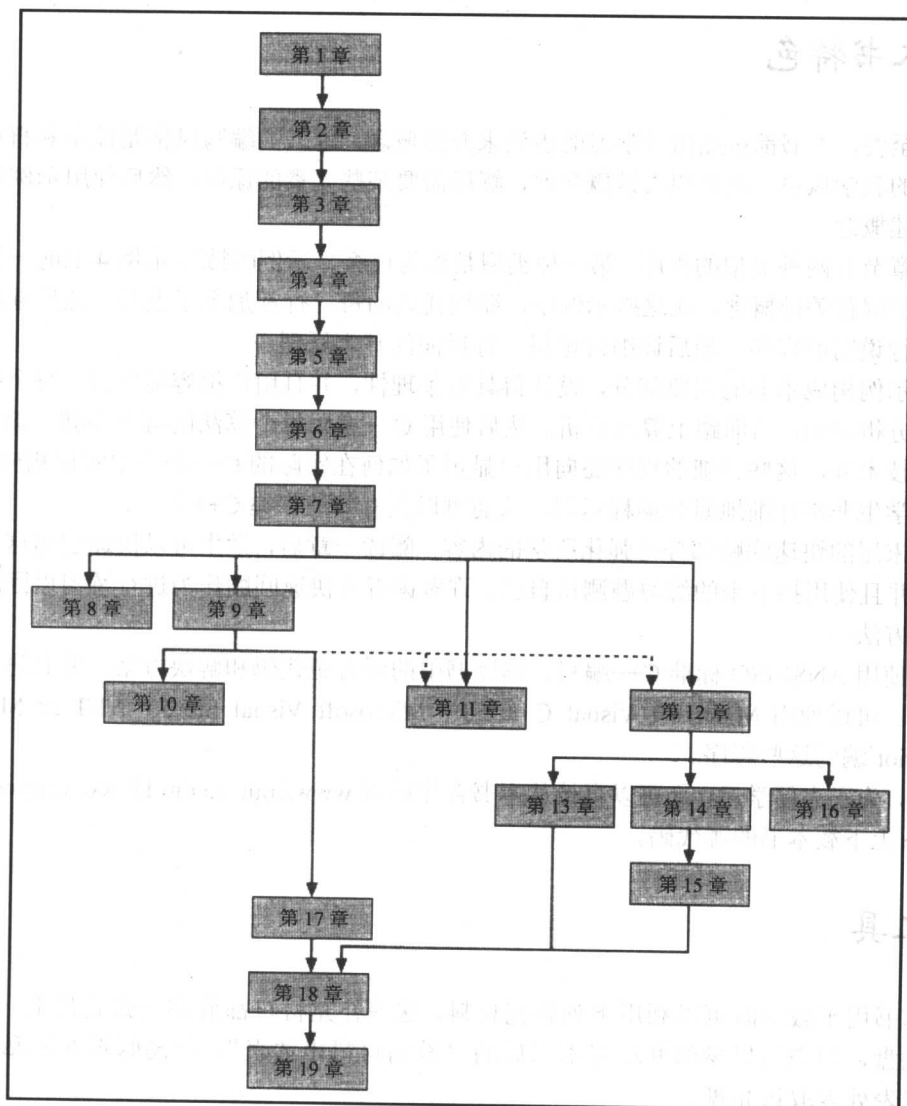


图 1 各章节之间的相关性图表

在图 1 中, 虚线表示前面的章节将在当前章节的一个部分中用到, 并且不是学习下一个章节的先决条件。例如, 第 9 章详细讨论了数组。在第 11 章和第 12 章中, 分别显示了数组和结构体以及数组和类之间的关系。然而, 如果在学习第 9 章之前就先学习第 12 章, 则可以跳过第 12 章中处理数组的部分, 而不会影响连续性。可以在研究第 9 章后再来学习这个特定的部分。

推荐读者按顺序阅读前 7 章。学习前 7 章后, 如果读者希望尽早学习 OOD 和 OOP, 则可以在第 7 章后直接学习第 12 章。可以在学习第 7 章后的任何时间学习第 8 章。按顺序学习前 7 章后, 一些学习方法如下:

- (1) 按如下顺序学习章节: 9、10、11、12、13、14、15、16、17、18、19。
- (2) 按如下顺序学习章节: 9、12、14、15、13、17、18、19、10、16。
- (3) 按如下顺序学习章节: 12、9、10、14、15、13、17、18、19、16。
- (4) 按如下顺序学习章节: 12、9、14、15、13、17、18、19、10、16。

0.4 本书特色

自始至终, 本书都按照指导学习的步调来介绍概念。本书的编写风格是简单和直观, 类似于课堂上的教学风格。在介绍关键概念前, 解释需要某些元素的原因。然后使用示例和较小的程序来描述概念。

每个章节有两种类型的程序。第一种类型是作为已编号示例(例如, 示例 4-1)的一部分的小程序, 用于解释关键概念。在这些示例中, 编程代码的每一行都加上了编号。然后逐行解释通过示例运行说明的程序。最后详细讨论每一行后面的基本原理。

编程示例构成本书的主要部分, 设计得具有条理性, 并且用户很容易掌握。每个编程示例都从问题分析开始, 后面跟上算法分析。然后使用 C++ 编码每个算法的每个步骤。除了教导解决问题的技术外, 这些详细的程序还向用户显示了如何在实际的 C++ 程序中实现相应的概念。强烈推荐学生非常仔细地研究编程示例, 从而可以更有效地掌握 C++。

每章末尾的快速回顾部分可强化所学的内容。阅读一章后, 学生可以快速遍历这一章的重点部分, 并且使用接下来的练习题测试自己。许多读者将快速回顾作为进行练习以回顾本章内容的简单方法。

本书使用 ANSI/ISO 标准 C++ 编写、编译书中的所有源代码和解决方案, 并且进行了质量保证测试。可以使用 Microsoft Visual C++ 6.0、Microsoft Visual Studio .NET 或 Metrowerks CodeWarrior 编写这些程序。

另外, 为了方便学习, 您可以直接从本书合作站点 www.course.com 或 www.tupwk.com.cn/downloadpage 上下载本书的源代码。

教学工具

当本书用于教学时可以利用下列补充材料, 这些补充材料都放在一张光盘上。本书并未附带该光盘, 但您可以参阅并填写本书后的“教辅材料申请表”, 向汤姆森亚洲私人有限公司北京代表处索取该光盘。

电子教学手册。本书附带的教师手册包含以下内容:

- 备课的辅助教学材料，包括对授课主题的建议。
- 所有各章末尾习题的答案，包括“编程练习”。

ExamView。本书附带的 ExamView 是一种强大的测试软件包，使用它，教师可以创建并管理笔试、机试(基于 LAN)和 Internet 考试。ExamView 包含与本书主题相配套的数百道试题，使学生可以定制一些包含了页面索引在内的详细的学习指南，以供进一步复习使用。这些基于计算机和 Internet 的考试组件使学生可以通过他们的计算机参加考试，由于每次考试后都能自动进行评分，因此节省了教师的时间。

PowerPoint 演示文稿。本书为每章都提供了 Microsoft PowerPoint 幻灯片。这些幻灯片可作为教学辅助工具供课堂演示使用，使学生可以通过网络查看章节内容，或打印出来在课堂上分发。教师可以制作授课所需的其他主题的幻灯片。

远程教学。Course Technology 的得意之作就是在 WebCT 和 Blackboard 及它自己的课程增强工具 MyCourse.com 中提供了一些联机课程。当您将联机内容添加到课程中时，将为您添加下列多种功能：自我测试、链接、词汇表，最重要的是它提供了通向 21 世纪最重要信息资源的途径。无论其在联机还是脱机状态下，我们都希望学生能够充分利用这些课程。如果您想知道有关如何将远程教学引入课程的详细信息，请与当地的 Course Technology 销售代表联系。

源代码。可从站点 www.course.com 上可以找到源代码。运行某些程序所需的输入文件也包含在这些源代码中。

习题答案文件。所有编程练习的习题答案文件可在 www.course.com 中找到。运行某些程序所需的输入文件也包含在习题答案文件中。

目 录

第 1 章 计算机和编程语言概述	1	2.8 预处理指令	48
1.1 简介	1	2.9 创建 C++ 程序	50
1.2 概述计算机的发展史	2	2.10 编程风格和方式	53
1.3 计算机系统的描述	2	2.10.1 语法	53
1.3.1 硬件	2	2.10.2 文档	55
1.3.2 软件	4	2.11 更多的赋值语句	56
1.4 计算机语言	4	2.12 编程示例：换算长度	57
1.5 编程语言的发展史	5	2.12.1 问题分析和算法设计	57
1.6 高级语言程序的执行	6	2.12.2 变量	58
1.7 问题分析-编码-执行周期的编程	8	2.12.3 指定常量	58
1.8 编程方法	12	2.12.4 主算法	58
1.8.1 结构化编程	12	2.12.5 综合	58
1.8.2 面向对象编程	12	2.13 编程示例：美分找零	60
1.9 ANSI/ISO 标准 C++	13	2.13.1 问题分析和算法设计	60
1.10 快速回顾	14	2.13.2 变量	61
1.11 练习题	15	2.13.3 指定常量	61
第 2 章 C++ 的基本元素	17	2.13.4 主算法	62
2.1 C++ 基础知识	18	2.14 快速回顾	63
2.1.1 特殊字符	19	2.15 练习题	65
2.1.2 关键字	19	2.16 编程练习	71
2.1.3 标识符	19	第 3 章 输入/输出	74
2.2 数据类型	20	3.1 I/O 流和标准 I/O 设备	74
2.2.1 基本数据类型	21	3.2 在程序中使用预定义函数	79
2.2.2 浮点数据类型	23	3.2.1 cin 和 ignore 函数	81
2.2.3 string 数据类型	24	3.2.2 putback 和 peek 函数	82
2.3 算术运算符和运算符的优先级	25	3.2.3 注意 I/Ostream 变量和 I/O 函数 之间的点号	84
2.4 表达式	28	3.3 输入错误	84
2.4.1 混合表达式	28	3.4 输出和格式化输出	87
2.4.2 类型转换(强制转换)	30	3.4.1 操作符 setprecision	88
2.5 输入	31	3.4.2 操作符 fixed	88
2.5.1 为常量和变量分配内存	31	3.4.3 操作符 showpoint	88
2.5.2 对变量赋值	33	3.4.4 操作符 setw	90
2.6 递增运算符和递减运算符	40	3.4.5 操作符 flush	92
2.7 输出	42	3.5 附加的输出格式化工具	93

3.5.1	操作符 setfill	93
3.5.2	操作符 left 和 right	95
3.6	输入/输出和 string 类型	96
3.7	文件输入/输出	97
3.8	编程示例: 售电影票和向慈善机构捐赠	100
3.8.1	问题分析和算法设计	101
3.8.2	变量	101
3.8.3	格式化输出	102
3.8.4	主算法	102
3.8.5	完整的程序清单	103
3.9	编程示例: 学生分数	105
3.9.1	问题分析和算法设计	105
3.9.2	变量	105
3.9.3	主算法	106
3.9.4	完整的程序清单	106
3.10	快速回顾	107
3.11	练习题	109
3.12	编程练习	111
第 4 章	控制结构 I	114
4.1	控制结构	114
4.2	关系运算符	115
4.3	关系运算符和基本数据类型	116
4.4	关系运算符和 string 类型	118
4.5	逻辑(布尔)运算符和逻辑表达式	119
4.6	优先顺序	120
4.6.1	快速求值法	124
4.6.2	int 数据类型和逻辑(布尔)表达式	124
4.6.3	bool 数据类型和逻辑(布尔)表达式	125
4.7	选择结构: if 和 if...else 语句	126
4.7.1	单选	126
4.7.2	双选	128
4.7.3	复合语句(块)	131
4.7.4	多选: 嵌套 if 语句	131
4.7.5	将 if...else 语句与 if 语句进行比较	134

4.7.6	使用伪代码开发、测验和调试程序	135
4.7.7	输入失败和 if 语句	137
4.7.8	等于运算符(==)和赋值运算符(=)之间的混淆	140
4.7.9	条件运算符(?:)	141
4.8	switch 语句	141
4.9	利用 assert 函数终止程序	147
4.10	编程示例: 有线电视公司的收费单	148
4.10.1	问题分析和算法设计	149
4.10.2	变量	149
4.10.3	指定的常量	150
4.10.4	公式	150
4.10.5	主算法	151
4.11	快速回顾	153
4.12	练习题	154
4.13	编程练习	159
第 5 章	控制结构 II(循环)	161
5.1	为什么需要循环语句	161
5.2	while 循环结构	162
5.2.1	案例 1: 使用控制计数器的 while 循环	165
5.2.2	案例 2: 使用标记控制的 while 循环	167
5.2.3	案例 3: 标志控制的 while 循环	171
5.2.4	案例 4: EOF 控制的 while 循环	171
5.3	编程示例: 检查支票账户余额	173
5.3.1	问题分析和算法设计	173
5.3.2	主算法	176
5.3.3	完整的程序清单	177
5.4	编程示例: 斐波纳契数列	180
5.4.1	问题分析和算法设计	181
5.4.2	主算法	181
5.4.3	完整的程序清单	182
5.5	for 循环结构	184
5.6	编程示例: 分类数字	189

5.6.1 问题分析和算法设计	189	7.10 带默认参数的函数	271
5.6.2 主算法	190	7.11 编程示例: 分类数字	273
5.6.3 完整的程序清单	190	7.11.1 主算法	275
5.7 do...while 循环结构	192	7.11.2 完整的程序清单	275
5.8 break 语句和 continue 语句	195	7.12 编程示例: 数据比较	277
5.9 嵌套控制结构	196	7.12.1 问题分析和算法设计	278
5.10 快速回顾	203	7.12.2 主算法: 函数 main	281
5.11 练习题	204	7.12.3 完整的程序清单	282
5.12 编程练习	214	7.12.4 条形图	285
第 6 章 用户定义的函数 I	216	7.12.5 函数 printBar	286
6.1 预定义函数	216	7.13 快速回顾	287
6.2 用户定义的函数	218	7.14 练习题	288
6.3 返回值函数	219	7.15 编程练习	294
6.3.1 return 语句	221	第 8 章 用户定义的简单数据类型、	
6.3.2 函数原型	224	命名空间及字符串类型	297
6.3.3 程序的执行流程	229	8.1 枚举类型	297
6.4 编程示例: 最大数	229	8.1.1 声明变量	299
6.4.1 问题分析和算法设计	230	8.1.2 赋值	299
6.4.2 完整的程序清单	230	8.1.3 枚举类型的操作	299
6.5 编程示例: 有线电视公司	231	8.1.4 关系运算符	300
6.5.1 问题分析和算法设计	231	8.1.5 枚举类型及循环	300
6.5.2 主算法(main 函数):	233	8.1.6 枚举类型的输入和输出	300
6.5.3 完整的程序清单	233	8.1.7 函数和枚举类型	302
6.6 快速回顾	236	8.1.8 在定义枚举类型时声明变量	303
6.7 练习题	237	8.1.9 匿名数据类型	303
6.8 编程练习	241	8.1.10 typedef 语句	304
第 7 章 用户定义的函数 II	243	8.2 编程示例: 石头、纸和剪刀的	
7.1 void 函数	243	游戏	305
7.1.1 不带参数的 void 函数	243	8.2.1 问题分析和算法设计	305
7.1.2 带参数的 void 函数	246	8.2.2 主算法	310
7.2 值参数	249	8.2.3 完整的程序清单	310
7.3 将引用变量作为参数	251	8.3 命名空间	314
7.4 值参数和引用参数以及内存		8.4 字符串类型	319
分配	254	8.5 编程示例: 大拉丁字符串	328
7.5 引用参数和返回值函数	265	8.5.1 问题分析和算法设计	329
7.6 标识符的作用域	265	8.5.2 主算法	331
7.7 全局变量的副作用	268	8.5.3 完整的程序清单	331
7.8 静态变量和自动变量	269	8.6 快速回顾	333
7.9 函数重载概述	270	8.7 练习题	335

8.8	编程练习	338	10.1.1	查找	400
第 9 章	数组以及字符串	340	10.1.2	排序列表: 冒泡法排序	404
9.1	数组	341	10.1.3	排序列表: 选择排序	407
9.1.1	访问数组元素	342	10.1.4	对有序列表进行顺序查找	413
9.1.2	一维数组的处理	344	10.1.5	二分查找法	415
9.1.3	数组索引越界	348	10.2	vector 类型(类)	419
9.1.4	声明时初始化数组	348	10.3	编程示例: 选举结果	423
9.1.5	处理数组时的一些限制	349	10.3.1	问题分析和算法设计	424
9.1.6	将数组作为函数的参数	350	10.3.2	完整的程序清单	432
9.1.7	整型数据类型以及数组索引	355	10.4	快速回顾	437
9.2	C-string(字符数组)	356	10.5	练习题	438
9.2.1	字符串比较	358	10.6	编程练习	440
9.2.2	字符串的读写	359	第 11 章	记录(结构体)	442
9.2.3	字符串输入	359	11.1	记录(结构体)	442
9.2.4	字符串输出	360	11.1.1	访问结构体成员	444
9.2.5	在执行期间指定输入/输出 文件	360	11.1.2	赋值	445
9.4	平行数组	361	11.1.3	比较(关系运算符)	446
9.3	二维数组和多维数组	362	11.1.4	输入/输出	447
9.4.1	访问数组元素	364	11.1.5	结构体变量和函数	447
9.4.2	声明时初始化二维数组	365	11.1.6	数组与结构体的比较	448
9.4.3	二维数组和枚举类型	365	11.1.7	结构体中的数组	448
9.4.4	二维数组的处理	367	11.1.8	数组中的结构体	450
9.4.5	将二维数组作为参数传递给 函数	372	11.1.9	结构体中的结构体	452
9.4.6	字符串数组	374	11.2	编程示例: 销售数据分析	455
9.4.7	声明二维数组的其他方法	376	11.2.1	问题分析和算法设计	456
9.5	多维数组	377	11.2.2	主算法	463
9.6	编程示例: 代码检测	377	11.2.3	完整的程序清单	464
9.6.1	问题分析和算法设计	378	11.3	快速回顾	469
9.6.2	完整的程序清单	381	11.4	练习题	470
9.7	编程示例: 文本处理	383	11.5	编程练习	472
9.7.1	问题分析和算法设计	384	第 12 章	类和数据抽象	475
9.7.2	完整的程序清单	386	12.1	类	475
9.8	快速回顾	389	12.1.1	统一建模语言图	478
9.9	练习题	390	12.1.2	变量(对象)声明	478
9.10	编程练习	395	12.1.3	访问类成员	479
第 10 章	数组和 vector 类型的应用	400	12.1.4	对类执行的内置操作	480
10.1	列表处理	400	12.1.5	赋值运算符和类	480
			12.1.6	类的作用域	481
			12.1.7	函数和类	481

12.1.8	成员函数的实现	482	13.4.2	主程序	573
12.1.9	访问函数和赋值函数	485	13.4.3	程序清单	575
12.1.10	类公有成员和私有成员的 次序	491	13.5	快速回顾	581
12.1.11	构造函数	493	13.6	练习题	581
12.1.12	调用构造函数	494	13.7	编程练习	587
12.1.13	类和构造函数	499	第 14 章	指针、类和虚函数	590
12.1.14	类对象(变量)数组和构造 函数	500	14.1	指针数据类型和指针变量	590
12.1.15	析构函数	501	14.2	取地址运算符(&)	591
12.2	数据抽象化、类和抽象数据 类型	502	14.3	复引用运算符(*)	592
12.3	结构体和类的比较	504	14.4	类、结构体和指针变量	597
12.4	信息隐藏	504	14.5	初始化指针变量	600
12.5	可执行代码	507	14.6	动态变量	600
12.6	类的静态成员	510	14.6.1	运算符 new	600
12.7	编程示例:糖果机	515	14.6.2	运算符 delete	601
12.7.1	问题分析和算法设计	515	14.7	指针变量操作	603
12.7.2	主程序	519	14.8	动态数组	605
12.7.3	完整的程序清单	522	14.8.1	函数和指针	607
12.8	快速回顾	527	14.8.2	指针和函数返回值	607
12.9	练习题	529	14.9	表层与深层复制和指针	608
12.10	编程练习	533	14.10	类和指针的一些特性	610
第 13 章	继承和组合	536	14.10.1	析构函数	610
13.1	继承	536	14.10.2	赋值运算符	611
13.1.1	重定义(重写)基类的成员 函数	538	14.10.3	复制构造函数	612
13.1.2	派生类和基类的构造函数	544	14.11	继承、指针和虚函数	619
13.1.3	派生类的头文件	550	14.12	取地址运算符和类	625
13.1.4	头文件的多次包含	551	14.13	快速回顾	627
13.1.5	C++流类	552	14.14	练习题	629
13.1.6	类的保护成员	553	14.15	编程练习	635
13.1.7	public、protected 或 private 继承	553	第 15 章	重载和模板	636
13.2	组合	556	15.1	为什么需要运算符重载	636
13.3	面向对象设计(OOD)和面向 对象的程序设计(OOP)	560	15.2	运算符重载	637
13.4	编程示例:成绩单	562	15.2.1	运算符函数的语法	637
13.4.1	问题分析和算法设计	564	15.2.2	重载运算符的一些限制	638
			15.2.3	指针 this	638
			15.2.4	类的友元函数	643
			15.2.5	作为成员函数和非成员函 数的运算符函数	645
			15.2.6	重载二元运算符	648

15.2.7 重载流插入运算符(<<)和流 提取运算符(>>).....	653	17.1.1 直接和间接递归.....	743
15.2.8 重载赋值运算符(=).....	657	17.1.2 无限递归.....	743
15.2.9 重载一元运算符.....	664	17.2 使用递归解决问题.....	743
15.2.10 重载自增(++)和自减(--) 运算符.....	664	17.3 递归或迭代各自的用途.....	752
15.2.11 运算符重载: 成员和 非成员.....	670	17.4 编程示例: 将一个数从二进制 转化为十进制.....	753
15.2.12 类和指针数据成员(回顾).....	670	17.5 编程示例: 将一个数从十进制 转化为二进制.....	756
15.2.13 运算符重载: 小结.....	671	17.6 快速回顾.....	759
15.3 编程示例 clockType.....	671	17.7 练习题.....	759
15.4 编程示例: 复数.....	677	17.8 编程练习.....	762
15.5 重载数组索引(下标) 运算符([]).....	682	第 18 章 链表.....	765
15.6 编程示例: newString.....	683	18.1 链表.....	765
15.7 函数重载.....	689	18.1.1 链表的属性.....	766
15.8 模板.....	689	18.1.2 遍历链表.....	768
15.8.1 函数模板.....	689	18.1.3 项插入和删除.....	769
15.8.2 类模板.....	691	18.1.4 构建链表.....	772
15.9 快速回顾.....	698	18.2 作为 ADT 的链表.....	776
15.10 练习题.....	700	18.2.1 列表的长度.....	781
15.11 编程练习.....	704	18.2.2 检索第一个节点的数据.....	781
第 16 章 异常处理.....	710	18.2.3 检索最后一个节点的数据.....	781
16.1 处理程序中的异常.....	710	18.2.4 查找列表.....	781
16.1.1 C++ 的异常处理机制.....	714	18.2.5 插入第一个节点.....	782
16.1.2 try/catch 块.....	714	18.2.6 插入最后一个节点.....	783
16.1.3 在程序中使用 try/catch 块.....	716	18.2.7 复制列表.....	787
16.1.4 使用 C++ 异常类.....	720	18.2.8 析构函数.....	789
16.2 创建自己的异常类.....	723	18.2.9 复制构造函数.....	789
16.3 异常处理技巧.....	732	18.2.10 重载赋值运算符.....	789
16.3.1 终止程序.....	732	18.3 有序链表.....	790
16.3.2 修复错误并继续执行.....	732	18.4 双向链表.....	801
16.3.3 记录错误并继续执行.....	734	18.4.1 默认的构造函数.....	803
16.4 栈展开.....	734	18.4.2 isEmptyList.....	804
16.5 快速回顾.....	737	18.4.3 销毁列表.....	804
16.6 练习题.....	738	18.4.4 初始化列表.....	804
16.7 编程练习.....	740	18.4.5 列表的长度.....	804
第 17 章 递归.....	741	18.4.6 输出列表.....	805
17.1 递归定义.....	741	18.4.7 反向输出列表.....	805
		18.4.8 查找列表.....	805
		18.4.9 第一个和最后一个元素.....	806

18.5	编程示例：录像带出租店	810	19.8.3	服务器	903
18.5.1	录像带成员	811	19.8.4	服务器列表	906
18.5.2	顾客成员	821	19.8.5	等待顾客队列	909
18.5.3	主程序	823	19.8.6	主程序	911
18.5.4	主程序清单	824	19.9	快速回顾	917
18.6	快速回顾	828	19.10	练习题	918
18.7	练习题	828	19.11	编程练习	921
18.8	编程练习	832	附录 A	保留字	924
第 19 章	栈和队列	835	附录 B	运算符优先级	925
19.1	栈	835	附录 C	字符集	926
19.2	将栈实现为数组	839	C.1	ASCII(美国标准信息交换码)	926
19.2.1	复制栈	845	C.2	EBCDIC(扩展二进制编码 十进制互换代码)	927
19.2.2	构造函数和析构函数	846	附录 D	运算符重载	928
19.2.3	复制构造函数	846	附录 E	其他 C++ 主题	929
19.2.4	重载赋值运算符(=)	847	E.1	关于文件输入/输出的更多 内容	929
19.2.5	栈的头文件	847	E.1.1	二进制文件	929
19.3	编程示例：最高 GPA	852	E.1.2	随机文件访问	934
19.3.1	程序分析和算法设计	852	E.2	ANSI/ISO 标准 C++ 和标准 C++ 中头	941
19.3.2	完整的程序清单	853	附录 F	头文件	943
19.4	栈的链实现	855	F.1	头文件 cassert(assert.h)	943
19.4.1	返回栈顶元素	861	F.2	头文件 ctype(ctype.h)	943
19.4.2	复制栈	863	F.3	头文件 cfloat(float.h)	944
19.4.3	从类 linkedListType 派生 的栈	867	F.4	头文件 climits(limits.h)	946
19.5	栈的应用：后缀表达式 计算器	868	F.5	头文件 cmath(math.h)	947
19.5.1	主算法	871	F.6	头文件 cstdint(stddef.h)	948
19.5.2	完整的程序清单	873	F.7	头文件 cstring(string.h)	948
19.6	删除递归：使用非递归算法 反向输出链表	877	F.8	头文件 string	948
19.7	队列	882	附录 G	系统上的内存大小和随机数 生成器	951
19.7.1	队列操作	882	附录 H	标准模板库(STL)	953
19.7.2	将队列实现为数组	883	H.1	STL 的组成部分	953
19.7.3	队列的链实现	892	H.2	容器类型	953
19.7.4	从类 linkedListType 派生的 队列	896	H.2.1	顺序容器	953
19.8	队列应用：模拟	899			
19.8.1	设计排队系统	900			
19.8.2	顾客	901			