

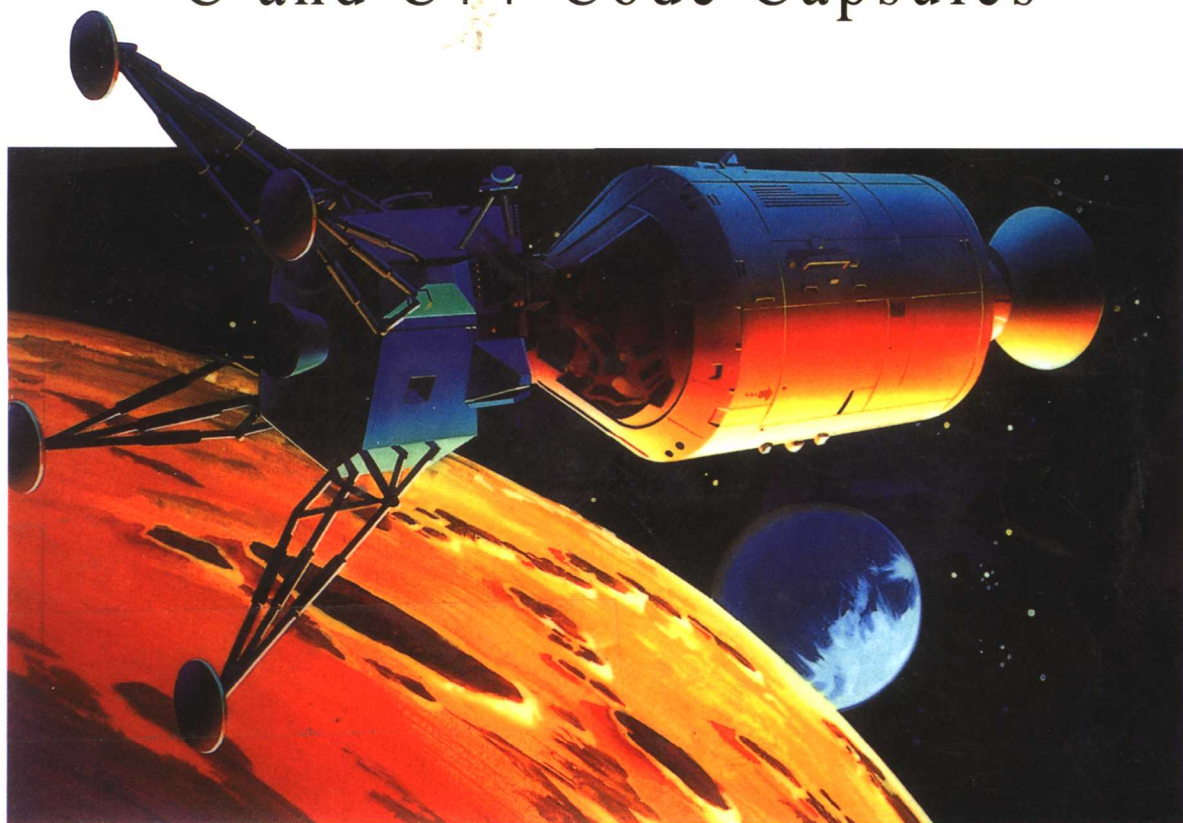


C 和 C++ 实务精选

PTR  
PH

# C 和 C++ 代码精粹

C and C++ Code Capsules



Chuck Allison 著  
董慧颖 译

人民邮电出版社  
POSTS & TELECOMMUNICATIONS PRESS

TP312C  
2A124

C和C++实务精选

# C和C++代码精粹

---

Chuck Allison 著

董慧颖 译

人民邮电出版社

## 图书在版编目 (CIP) 数据

C 和 C++代码精粹 / (美) 阿林森 (Allison, C.) 著; 董慧颖译. —北京: 人民邮电出版社, 2003.4  
ISBN 7-115-10625-8

I. C... II. ①阿... ②董... III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2002) 第 089462 号

## 版 权 声 明

Chuck Allison: C & C++ Code Capsules

Authorized translation from the English language edition published by Prentice Hall PTR.

Copyright © 1998 Prentice Hall PTR

ISBN: 0135917859

All rights reserved. No part of the book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the publisher.

Chinese Simplified language edition published by Posts & Telecommunications Press.

本书英文版由 Prentice Hall PTR 出版。人民邮电出版社取得授权翻译出版中文简体版。

未经出版者许可, 对本书任何部分不得以任何方式或任何手段复制和传播。

版权所有, 侵权必究。

C 和 C++实务精选

## C 和 C++ 代码精粹

- 
- ◆ 著           Chuck Allison  
   译           董慧颖  
   责任编辑   陈冀康
  - ◆ 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 010-67132705  
   北京汉魂图文设计有限公司制作  
   北京顺义振华印刷厂印刷  
   新华书店总店北京发行所经销
  - ◆ 开本: 800×1000 1/16  
   印张: 34.75  
   字数: 778 千字                               2003 年 4 月第 1 版  
   印数: 1-5 000 册                           2003 年 4 月北京第 1 次印刷

著作权合同登记 图字: 01-2002-2445 号

ISBN 7-115-10625-8/TP · 3082

定价: 59.00 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

---

本书基于作者备受好评的 *C/C++ User Journal* 杂志上的每月专栏，通过大量完全符合 ISO 标准 C++ 的程序集合，说明了 C++ 真正强大的威力，是 C 和 C++ 职业程序员的实践指南。

全书分为 3 篇共 20 章，分别从指针、预处理器、C 标准库、抽象、模板、异常、算法、容器、文件处理、动态内存管理等不同层次的话题展开讨论。书中的精粹代码，对于 C 和 C++ 程序员具有很好的使用价值和启发意义。

本书可以帮助有一定经验的 C 和 C++ 程序员深入学习这两种密切相关的语言，对书中代码的参悟和应用，可以帮助他们从根本上提高使用程序的效率。

# 序

我第一次听说 Chuck 是在 1993 年 3 月,当时他的位处理类、bitset 和 bitstring 被标准 C++ 库所采纳 (bitstring 是在后来加入 STL 时被吸收的)。虽然我本人间或为 C++ 标准加进两三个极小的位,但是他那种通过 ANSI/ISO C++ 委员会非常严格的考验成功地运行一个完整的类的想法给我留下了深刻的印象 (我们都是委员会成员,均悉内情)。

尽管计算机业界言过其实的事情不乏其例,以致于我必须尴尬地依靠我的直觉而不是智慧来辨别真相。因此,在这个领域中,给我留下较深刻印象的是那些能用简明扼要和不夸张的方式来阐明问题的人,更确切地说,这样的人就是一位了不起的导师。Chuck 就是这样的导师。你可以通过他全力所从事的一系列活动,如写作、教学、编辑、答疑中看出这一点。当我发现这类人——当我见到他们对听众演讲时,我便能对他们深信不疑——因此,我说服他们在软件开发 (Software Development) 会议上讲演 (我在该会议上担任 C++ 和 Java 的专题主席)。Chuck 已经成为会议上一位定期演讲的固定人物,他令双方的听众都满意。

在最后一次 SD 会议上 (1997 年秋,华盛顿 DC),当我们得知当天是 Chuck 的生日后,我们一伙人便邀他出去共进晚餐。全体就座后,我环顾四周,才意识到我们都是技术图书的作者: Bjarne Stroustrup (C++ 的创始人和 *The C++ Programming Language* 一书的作者)、Dan Saks (C++ 专栏作家、演说家、顾问和 ANSI/ISO C++ 委员会长期秘书)、Bobby Schmidt (*CUJ* 专栏作家、演讲家)、Marco Cantu (*Mastering Delphi* 一书和 C++ 丛书的作者)、Tim Gooch (Cobb Group 出版社 C++ 的编辑,现任 Java 的编辑) 以及我本人,这些人都非常敬重 Chuck,所以才请他吃晚餐。

当然, C++ 教程式的书不在少数。我有时感到自己正坚持不懈地在写这样的一本书 (我的最终成果是 *Thinking in C++*)。然而,当你已经理解了基础知识并想更加深入时会如何呢? 虽然书是有了,但是它们常常是以专家的口吻写的 (一种令人瞠目结舌的语言) 要不然就是它们所覆盖的主题太深奥莫测。而这本书架起了通往高深主题的桥梁,它给了你所需要的而不会在前进的过程中难倒你。

Chuck 的书写得既明了又准确,准确是我非常喜欢的一件事情。一本书的瑕疵太多会令我厌倦 (过去我们不得不容忍这样的书,但现在精心地编写的 C++ 书籍足够多了,因此没有理由再浪费时间)。我非常喜爱这本书的另一个原因是它的每一章都很简练,每章都集中讲一个主题,你可以很快地领会它并且得到完整的概念 (我注意力集中的时间可不很长)。这是一本经得起时间考验的好书,它会带给你一个又一个顿悟。

Bruce Eckel

<http://www.EckelObjects.com>

1997 年 10 月

# 前言

本书适合于那些 C 和 C++ 的职业程序员。假如你已熟悉这两种语言的语法和基本结构, 这本书能够为你创建有效的、实用的程序提供实践性的指导。每一个代码范例或程序范例均标明行之有效的用法和技术, 这些用法和技术对 C/C++ 这两种重要编程语言的性能发挥起着重要的作用。

对于那些希望在工作中加强自身技术和提高效率的人来说, 本书可以算是一本经验之谈。尽管目前人们面对面向对象模式的推崇到了白热状态 (本书也包括这方面的丰富内容), 可是我没有理由不对 C++ 的基础——C 表示尊崇。我发现太多的程序开发者由于培训不当而不能掌握 C++, 因为他们缺乏对一些基本概念如指针、作用域、连接和静态类型检查的全面理解, 也许所有这些之中最大的缺陷就是缺乏对标准 C 库的熟悉。开发者浪费了大量时间去编写库中已提供的函数确实令人感到悲哀。那些 C++ 的新手因为热衷 C++ 的那些“令人兴奋”的特征, 如继承、异常或重载运算符 `new`, 甚至当一些特征还没有得到验证的时候就迫不及待地放弃了简单的 C 语言。我深信每个人都可以从本书中学到一些东西。严格地说, 第 1 章和第 13~16 章讲的都是 C++, 第 4~6 章讲的仅仅是 C 语言, 而所有其他章节则包含了与 C 和 C++ 各自主题都相关的内容。

可以说, 这主要是一本关于 C++ 的书。当本书 (指出版于 1998 年的英文原书) 要印刷的时候, C++ 标准化的努力已进入最后冲刺中, 第二届公共委员会草案 (CD2) 的整套方案已经完成, 仅剩下次要的编辑工作。我自 1991 年初作为该委员会的成员以来, 目睹了标准化的文档从 200 页增加到 750 多页。我们已在语言中加入了异常、模板、名字空间、运行期类型识别 (RTTI) 以及其他一些特征, 而在库 (通常称作标准模板库或 STL) 中加入了复杂的、相互关联算法的模板化系统、容器以及迭代结构。与其他标准化的努力方向不同, 委员会对新发明和标准化现存的实践同等重视。C++ 的过于复杂引起了一个网上冲浪者发表了下面的观点: “如果 C 语言给了你足够长的绳子让你上吊, 那么 C++ 则给了你更长的绳子, 足够让

你周围的每一个人上吊，还够你升起一艘小帆船的帆，剩余的还够让你上吊”。我不辞辛苦地用这种方法说明并解释标准 C++和它的库，就是希望你可以更明智地运用你的绳子。

“C++标准的制定”是我对 Bjarne Stroustrup 采访的摘录，记录了他对 C++成为标准的感受。本书的其他部分则分成了 3 个部分。

## 第一部分：基础篇

在对 C++作了简短的介绍之后，这些章节弥补 C 程序员在准备使用 C++之前可能存在的差距。第 2 章“指针”基于我在 1993 年的 *C Users Journal* 上发表后反响不错的三个系列，第 4 章到第 6 章包含了每个职业程序员都应该知道的标准 C 库，这是标准 C++极其重要的一部分。

## 第二部分：关键概念

这一部分详尽地揭示和阐述了 C++语言的概念和特征。第 7 章通过类介绍数据抽象；第 8 章覆盖了由 C++模板机制实现的类型抽象，模板对有效地使用 C++来说和对象一样重要，甚至可能更重要；第 14 章不仅论述了继承和多态，也说明了面向对象的设计和重用，如使用当今关系数据库管理系统的用于对象持久性框架的描述。中间的章节深入介绍了许多开发者都容易忽略的重要基本概念。

## 第三部分：标准库的强大作用

第 15 章到第 20 章介绍了如何使用和享受标准 C++库卓越的组件，同时还阐明了在第 4、5、6 章范围之外的标准 C 库的一些更为复杂的特征；第 15 章和第 16 章解释了作为 C++库子集的 STL 库为什么会是这样，以及如何有效地使用它；第 19 章包含了一个有用的甚至能够处理部分日期的日期组件，可以满足普通的商业数据处理的需要。

总之，这是一本关于如何编程的书，我已尝试通过深度和广度的合理平衡来说明“最好的实践”，引导读者放弃“已经掌握了 C++”的想法。这本书在标准委员会通过 ISO C++的最终草案仅一周之后就付诸印刷，而我已经谨慎地避免了继续存在的任何死角（所有的语言和环境都有）。我自信本书中的所有材料在未来几年中都是适用的。

## 致谢

此书的实际写作时间比我的计划多花了很多年。写作始于 1984 年，当时我正担任亚利桑那州图森市皮玛社区学院计算机科学系主任，我的同事 Claire Hamlet 劝我在那儿开设用 C 语

言编程的第一门课程。从此我开始收集 C 程序的例子，并与在 Hughes Aircraft 公司和在盐湖城 Church of Jesus Christ of Latter-day Saints 的世界总部的一些雇员们共同分享。“代码精粹” (code capsules)，即关于特殊主题带有例子的短小程序，源自于我努力使 COBOL “受难者”在学习 C 时感到有趣并且不枯燥。我足够幸运的是在担任 C 语言支持委员会主席的时候，曾一度得到了委员会管理上的支持，该委员会有一支经验丰富的一流的程序员队伍 (David Coombs、John Pearson、Lorin Lords、Kent Olsen、Bill Owens、Drew Terry 以及 Mike Terrazas)，是他们开发了这样顶尖的课程并有效地培训了 100 多名 Church 雇员。

“代码精粹”这个名字是在我跟 Lee Copeland 在 Church 食堂吃早餐时想到的 (他使我有准备行动的路线)。Mike Terrazas 审查这些短小程序的早期版本，并建议我把它们发表在 *C Users Journal* 上，这些都是 1992 年 3 月在伦敦举行的 C++ 标准化委员会会议上，我将这些短小程序展示给 P.J. Bill 是不经意的，作为一位资深的编辑，他提议我成为这一期刊的专栏作者。“代码精粹”这一专栏从 1992 年 10 月开办直至 1995 年 5 月，后来我由于时间有限而不得不放弃。令人钦佩的良师益友 Bill 在鼓励我将这些代码精粹整理成书方面起了很大作用。Bruce Eckel 欣然阅读了本书的部分内容，Pete Becker 通览了全部的手稿，去掉了一些错误和不一致的内容。然而，当提及那种所需的无形的精神动力时，我必须像几乎每一位作者那样对我的家人表示感谢，只有 Sandy、James 和 Kim 才能体会到为了将本书出版大家所付出的巨大努力。21 岁的 James 最近从他呆了两年的英格兰来信说：“终于，爸爸完成了他的著作！他从我记事起就一直做这项工作。”

Chuck Allison

<http://www.freshsources.com>

1997 年 11 月

## 第一部分 预备知识

|                         |    |
|-------------------------|----|
| 第1章 更好的 C .....         | 3  |
| 1.1 两种语言简述 .....        | 3  |
| 1.2 循序渐进 .....          | 4  |
| 1.3 类型系统 .....          | 4  |
| 1.4 函数原型 .....          | 5  |
| 1.5 类型安全连接 .....        | 9  |
| 1.6 引用 .....            | 10 |
| 1.7 类型安全 I/O .....      | 11 |
| 1.8 标准流 .....           | 12 |
| 1.9 格式化 .....           | 14 |
| 1.10 操纵器 .....          | 18 |
| 1.11 函数重载和函数模板 .....    | 22 |
| 1.12 运算符重载 .....        | 22 |
| 1.13 内联函数 .....         | 24 |
| 1.14 缺省参数 .....         | 25 |
| 1.15 new 和 delete ..... | 26 |
| 1.16 语句声明 .....         | 26 |
| 1.17 标准库特征 .....        | 27 |

|                                   |           |
|-----------------------------------|-----------|
| 1.18 C 的兼容性.....                  | 27        |
| 1.19 小结.....                      | 29        |
| <b>第2章 指针</b> .....               | <b>31</b> |
| 2.1 容易出错的编程.....                  | 31        |
| 2.2 基础.....                       | 31        |
| 2.3 指针运算.....                     | 35        |
| 2.4 传引用语义.....                    | 38        |
| 2.5 普通指针.....                     | 39        |
| 2.6 const 指针.....                 | 40        |
| 2.7 指针和一维数组.....                  | 42        |
| 2.8 数组作为参数.....                   | 47        |
| 2.9 字符串数组.....                    | 49        |
| 2.10 指针和多维数组.....                 | 50        |
| 2.11 更高深的内容.....                  | 53        |
| 2.12 指向函数的指针.....                 | 56        |
| 2.13 指向成员函数的指针.....               | 59        |
| 2.14 封装和不完全类型.....                | 61        |
| 2.15 小结.....                      | 65        |
| <b>第3章 预处理器</b> .....             | <b>67</b> |
| 3.1 #include 指令.....              | 67        |
| 3.2 其他的预处理指令.....                 | 68        |
| 3.3 预定义宏.....                     | 70        |
| 3.4 条件编译.....                     | 72        |
| 3.5 预处理运算符.....                   | 73        |
| 3.6 实现 assert.....                | 75        |
| 3.7 宏的魅力.....                     | 76        |
| 3.8 字符集、三字符运算符和双字符运算符.....        | 79        |
| 3.9 翻译阶段.....                     | 82        |
| 3.10 小结.....                      | 82        |
| <b>第4章 C 标准库之一：面向合格的程序员</b> ..... | <b>83</b> |
| 4.1 <ctype.h>.....                | 84        |

|              |                               |            |
|--------------|-------------------------------|------------|
| 4.2          | <stdio.h> .....               | 87         |
| 4.3          | <stdlib.h> .....              | 91         |
| 4.4          | <string.h> .....              | 100        |
| <b>第 5 章</b> | <b>C 标准库之二：面向熟练的程序员 .....</b> | <b>103</b> |
| 5.1          | <assert.h> .....              | 103        |
| 5.2          | <limits.h> .....              | 104        |
| 5.3          | <stddef.h> .....              | 107        |
| 5.4          | <time.h> .....                | 110        |
| 5.5          | 字符集 .....                     | 112        |
| 5.6          | 代码页 .....                     | 114        |
| 5.7          | 字符集标准 .....                   | 115        |
| 5.8          | ISO 10646 .....               | 115        |
| 5.9          | 统一字符编码 .....                  | 116        |
| <b>第 6 章</b> | <b>C 标准库之三：面向优秀的程序员 .....</b> | <b>117</b> |
| 6.1          | <float.h> .....               | 117        |
| 6.2          | <math.h> .....                | 119        |
| 6.3          | <error.h> .....               | 123        |
| 6.4          | <locale.h> .....              | 124        |
| 6.5          | <setjmp.h> .....              | 126        |
| 6.6          | <signal.h> .....              | 127        |
| 6.7          | <stdarg.h> .....              | 128        |
| 6.8          | va_list 作为参数 .....            | 130        |
| 6.9          | 应用 .....                      | 131        |
| 6.10         | 结论 .....                      | 135        |
| 6.11         | 浮点数系统 .....                   | 135        |

## 第二部分 主要概念

|              |                 |            |
|--------------|-----------------|------------|
| <b>第 7 章</b> | <b>抽象 .....</b> | <b>145</b> |
| 7.1          | 数据抽象 .....      | 145        |
| 7.2          | 运算符重载 .....     | 154        |
| 7.3          | 具体的数据类型 .....   | 158        |

|                                 |            |
|---------------------------------|------------|
| 7.4 类型抽象 .....                  | 164        |
| 7.5 函数抽象 .....                  | 166        |
| 7.6 小结 .....                    | 167        |
| <b>第 8 章 模板 .....</b>           | <b>169</b> |
| 8.1 泛型编程 .....                  | 171        |
| 8.2 函数模板 .....                  | 171        |
| 8.3 类模板 .....                   | 173        |
| 8.4 模板参数 .....                  | 177        |
| 8.5 模板特化 .....                  | 179        |
| 8.6 小结 .....                    | 183        |
| <b>第 9 章 位操作 .....</b>          | <b>185</b> |
| 9.1 按位运算符 .....                 | 185        |
| 9.2 访问单独的位 .....                | 187        |
| 9.3 大型置位 .....                  | 193        |
| 9.4 位字符串 .....                  | 205        |
| 9.5 Wish List .....             | 205        |
| 9.6 bitset 模板 .....             | 206        |
| 9.7 vector<bool>模板特化 .....      | 208        |
| 9.8 小结 .....                    | 209        |
| <b>第 10 章 类型转换和强制类型转换 .....</b> | <b>211</b> |
| 10.1 整数的升级 .....                | 211        |
| 10.2 降级 .....                   | 215        |
| 10.3 算术类型转换 .....               | 216        |
| 10.4 函数原型 .....                 | 218        |
| 10.5 显式类型转换 .....               | 220        |
| 10.6 函数风格强制类型转换 .....           | 221        |
| 10.7 const 的正确性 .....           | 221        |
| 10.8 用户定义的类型转换 .....            | 223        |
| 10.9 加强运算符[] .....              | 229        |
| 10.10 新风格强制类型转换 .....           | 231        |
| 10.11 小结 .....                  | 233        |

|                    |     |
|--------------------|-----|
| <b>第 11 章 可见性</b>  | 235 |
| 11.1 名字中包含什么       | 235 |
| 11.2 作用域           | 235 |
| 11.3 最小的作用域        | 238 |
| 11.4 类的作用域         | 239 |
| 11.5 嵌套类           | 244 |
| 11.6 局部类           | 246 |
| 11.7 典型的名字空间       | 247 |
| 11.8 名字空间的作用域      | 249 |
| 11.9 生存期           | 250 |
| 11.10 临时对象的生存期     | 253 |
| 11.11 连接           | 254 |
| 11.12 类型安全连接       | 258 |
| 11.13 “语言”连接       | 258 |
| 11.14 小结           | 259 |
| <b>第 12 章 控制结构</b> | 261 |
| 12.1 结构化编程         | 261 |
| 12.2 分支            | 270 |
| 12.3 非局部分支         | 273 |
| 12.4 信号            | 277 |
| 12.5 小结            | 282 |
| 12.6 参考文献          | 282 |
| <b>第 13 章 异常</b>   | 283 |
| 13.1 可选择的错误处理方法    | 283 |
| 13.2 堆栈展开          | 288 |
| 13.3 异常捕捉          | 290 |
| 13.4 标准异常          | 293 |
| 13.5 资源管理          | 293 |
| 13.6 构造函数和异常       | 298 |
| 13.7 内存管理          | 303 |
| 13.8 异常规范          | 304 |

|                            |            |
|----------------------------|------------|
| 13.9 错误处理策略 .....          | 306        |
| 13.10 小结 .....             | 309        |
| <b>第 14 章 面向对象编程</b> ..... | <b>311</b> |
| 14.1 继承 .....              | 314        |
| 14.2 不同种类的集合 .....         | 316        |
| 14.3 虚函数和多态 .....          | 316        |
| 14.4 抽象基类 .....            | 319        |
| 14.5 实例研究：一个对象持续的框架 .....  | 322        |
| 14.6 数据库访问 .....           | 324        |
| 14.7 映射对象到相关模式 .....       | 326        |
| 14.8 PFX 的结构 .....         | 327        |
| 14.9 一个代码的预排 .....         | 330        |
| 14.10 小结 .....             | 349        |

### 第三部分 使用标准库

|                            |            |
|----------------------------|------------|
| <b>第 15 章 算法</b> .....     | <b>353</b> |
| 15.1 复杂度 .....             | 354        |
| 15.2 通用算法 .....            | 356        |
| 15.3 函数对象 .....            | 360        |
| 15.4 函数种类 .....            | 361        |
| 15.5 函数对象适配器 .....         | 362        |
| 15.6 算法种类 .....            | 363        |
| 15.7 小结 .....              | 366        |
| 15.8 参考文献 .....            | 366        |
| <b>第 16 章 容器和迭代器</b> ..... | <b>367</b> |
| 16.1 标准容器 .....            | 370        |
| 16.2 迭代器 .....             | 371        |
| 16.3 迭代器种类 .....           | 372        |
| 16.4 特殊用途的迭代器 .....        | 373        |
| 16.5 容器适配器 .....           | 376        |
| 16.6 关联容器 .....            | 377        |

|                             |            |
|-----------------------------|------------|
| 16.7 应用.....                | 377        |
| 16.8 非标准模板库容器.....          | 385        |
| 16.9 小结.....                | 387        |
| <b>第 17 章 文本处理</b> .....    | <b>389</b> |
| 17.1 scanf.....             | 389        |
| 17.2 printf.....            | 394        |
| 17.3 子字符串.....              | 397        |
| 17.4 标准 C++ 字符串类.....       | 403        |
| 17.5 字符串流.....              | 406        |
| 17.6 宽字符串.....              | 408        |
| 17.7 小结.....                | 408        |
| <b>第 18 章 文件处理</b> .....    | <b>409</b> |
| 18.1 过滤器.....               | 409        |
| 18.2 二进制文件.....             | 412        |
| 18.3 记录处理.....              | 414        |
| 18.4 临时文件.....              | 419        |
| 18.5 可移植性.....              | 420        |
| 18.6 POSIX.....             | 420        |
| 18.7 文件描述符.....             | 421        |
| 18.8 通过描述符来拷贝文件.....        | 421        |
| 18.9 读目录条目.....             | 423        |
| 18.10 重定向标准错误.....          | 427        |
| 18.11 封装重定向操作.....          | 431        |
| 18.12 小结.....               | 436        |
| <b>第 19 章 时间和日期处理</b> ..... | <b>437</b> |
| 19.1 Julian 日期编码.....       | 442        |
| 19.2 用于实际工作的日期类.....        | 460        |
| 19.3 计算年的星期数.....           | 486        |
| 19.4 小结.....                | 487        |
| 19.5 参考文献.....              | 487        |

|                            |     |
|----------------------------|-----|
| <b>第 20 章 动态内存管理</b> ..... | 489 |
| 20.1 参差数组 .....            | 489 |
| 20.2 在标准 C 中使用堆 .....      | 491 |
| 20.3 C++ 的自由存储 .....       | 495 |
| 20.4 浅拷贝与深拷贝 .....         | 496 |
| 20.5 处理内存分配失败 .....        | 499 |
| 20.6 重载 new 和 delete ..... | 499 |
| 20.7 配置 new .....          | 501 |
| 20.8 堆的管理 .....            | 503 |
| 20.9 避免内存管理 .....          | 504 |
| 20.10 小结 .....             | 510 |

## 附 录

|                              |     |
|------------------------------|-----|
| <b>附录 A C/C++ 的兼容性</b> ..... | 513 |
| <b>附录 B 标准 C++ 算法</b> .....  | 515 |
| <b>附录 C 函数对象和适配器</b> .....   | 525 |
| <b>附录 D 有注解的参考书目</b> .....   | 529 |
| <b>附录 E C++ 标准的制定</b> .....  | 531 |

# 第一部分

## 预备知识