

SAMS
Teach
Yourself

- 全球销量逾百万册的系列图书
- 连续十余年打造的经典品牌
- 直观、循序渐进的学习教程
- 掌握关键知识的最佳起点
- 秉承Read Less, Do More (精读多练)的教学理念
- 以示例引导读者完成最常见的任务

每章内容针对初学者精心设计, **1** 小时轻松阅读学习,
24 小时彻底掌握关键知识

每章 **案例与练习题** 助你轻松完成常见任务,
通过 **实践** 提高应用技能, 巩固所学知识

C++

入门经典 (第5版·修订版)

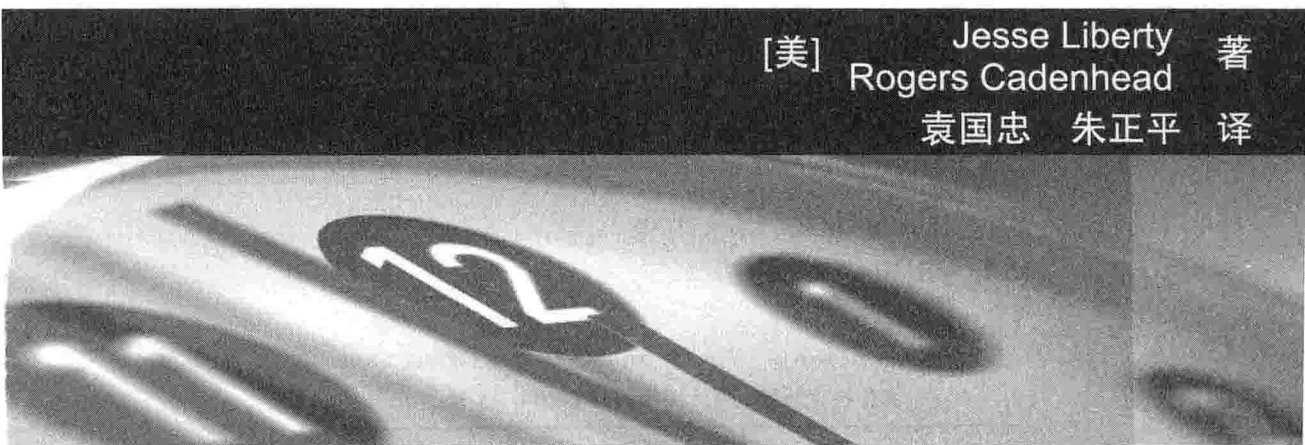
[美] Jesse Liberty 著
Rogers Cadenhead
袁国忠 朱正平 译

 人民邮电出版社
POSTS & TELECOM PRESS

C++

入门经典 (第5版·修订版)

[美] Jesse Liberty 著
Rogers Cadenhead
袁国忠 朱正平 译



人民邮电出版社

图书在版编目 (C I P) 数据

C++入门经典：第5版修订版 / (美) 丽波缇 (Liberty, J.) , (美) 卡登海德 (Cadenhead, R.) 著 ; 袁国忠, 朱正平译. — 2版. — 北京 : 人民邮电出版社, 2015. 2

ISBN 978-7-115-38202-3

I. ①C… II. ①丽… ②卡… ③袁… ④朱… III. ①C语言—程序设计 IV. ①TP312

中国版本图书馆CIP数据核字(2015)第001849号

版 权 声 明

Jesse Liberty and Rogers Cadenhead: Sams Teach Yourself C++ in 24 Hours

ISBN: 0672333317

Copyright © 2010 by Sams Publishing.

Authorized translation from the English languages 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。

-
- ◆ 著 [美] Jesse Liberty Rogers Cadenhead
 - 译 袁国忠 朱正平
 - 责任编辑 傅道坤
 - 责任印制 张佳莹 焦志炜
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
 - 邮编 100164 电子邮件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 大厂聚鑫印刷有限责任公司印刷
 - ◆ 开本: 787×1092 1/16
 - 印张: 20.5
 - 字数: 510 千字 2015 年 2 月第 2 版
 - 印数: 6 001 - 9 000 册 2015 年 2 月河北第 1 次印刷
 - 著作权合同登记号 图字: 01-2010-6748 号
-

定价: 39.00 元

读者服务热线: (010)81055410 印装质量热线: (010)81055316

反盗版热线: (010)81055315

内容提要

本书通过大量短小精悍的程序详细阐述了基本的 C++ 编程概念，包括 C++ 程序的组成部分、变量、常量、表达式、语句、运算符、函数、循环、数组、指针、引用、函数和运算符重载、继承、C++0x 新增功能、面向对象分析和设计、模板、错误和异常处理等主题。每章都提供了示例程序清单，并辅以示例输出和代码分析，以阐述该章介绍的主题。为加深读者对所学内容的理解，每章末尾都提供了常见问题及其答案以及练习和测验。

本书是针对 C++ 初学者编写的，不要求读者有 C 语言方面的背景知识，既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。

作者简介

 Jesse Liberty 编写了大量有关软件开发的图书，包括 C++ 和 .NET 方面的畅销书。他是 Liberty Associates 公司 (<http://www.libertyassociates.com>) 的总裁，负责编程、咨询和培训工作。

 Rogers Cadenhead 是作者、计算机程序员和 Web 开发人员，编著过 23 本图书，包括《Sams Teach Yourself Java in 21 Days》和《Sams Teach Yourself Java in 24 Hours》。他运营的 Drudge Retort 等网站每年的访问量超过 2200 万人次。

献 辞

谨以此书献给 Edythe、Stacey 以及 Robin 和 Rachel，他们分别给予我生命、与我分享生活以及赋予我生活的意义。

——Jesse Liberty

谨以此书献给我的父亲，他当前正教我比编程难得多的东西：在脊椎手术后重新行走。在持续数月的术后恢复期内，是他一直在不断给我以鼓舞，没有人像他那样对我的康复充满信心。

——Rogers Cadenhead

致 谢

每本书付梓时，都有机会感谢那些提供了支持和帮助的人，没有他们本书就不可能完成。在这些人中，最重要的是 Stacey、Robin 和 Rachel Liberty。

——Jesse Liberty

本书的完成需要付出艰辛的劳动，也与很多人的帮助分不开，其中大部分是 Sams 出版公司的员工，这里对他们深表感谢，尤其是 Keith Cline、Mandie Frank、Songlin Qiu、Mark Taber 和 Jon Upchurch。最重要的是，要感谢我的妻子 Mary 以及儿子 Max、Eli 和 Sam。

——Rogers Cadenhead

前 言

祝贺您！当您阅读到这里时，离学习最重要的编程语言之一——C++又近了 20 秒。

如果您再花 23 小时 59 分 40 秒，就将掌握 C++编程语言的基本知识。只需 24 个课程（每个课程不超过 1 小时），就将学会 C++基本知识，如管理 I/O、创建循环和数组、使用模板进行面向对象编程以及创建 C++程序。

我们将所有这些主题组织成了结构完美、易于理解的课程。在每章中，都将通过项目、输出和代码分析，演示相关的主题。另外，还清楚地标出了语法示例，以方便参考。

每章末尾还列出了常见问题及其答案，帮助您更深入地掌握学到的知识。

本书针对的读者

通过阅读本书来学习 C++时，读者不需要有任何编程经验。

本书从基本知识开始，既介绍 C++语言，又讨论使用 C++进行编程涉及的概念。无论读者是刚开始学习编程还是已经有一些编程经验，本书都将让您能够快速而轻松地学习 C++语言。

是否要先学 C 语言

不需要先学 C 语言。Bjarne Stroustrup 创立的 C++语言是 C 语言的继任者，功能更强大，更多才多艺。先学 C 语言会让您养成一些编程习惯，在您使用 C++编程时，这些习惯容易导致错误。本书不要求读者熟悉 C 语言。

为何要学习 C++

您可学习众多其他的语言，但 C++语言最值得学习，因为它经受了时间的考验，当前仍是一种深受欢迎的编程语言。

虽然面世于 1979 年，但鉴于 C++语言的强大功能和灵活性，当前仍被用于开发专业软

件。新版本即将面世，其工作名为 C++0x，这将让 C++ 语言更有用。

诸如 Java 等其他语言的灵感来自 C++ 语言，学习 C++ 语言也将让您对这些语言有深入认识。掌握 C++ 语言后，便可将这些技能用于当今的任何平台，从个人计算机到 Linux 和 UNIX 服务器，再到大型机和移动设备。

本书使用的约定

本书包含如下特殊元素。

**By the
Way**

注意：提供与读者阅读的内容相关的额外信息。

**Watch
Out!**

警告：提醒读者注意在特定情况下可能发生的问题或副作用。

**Did you
Know?**

提示：突出那些可以使 C++ 编程更有效的信息。

目 录

第一部分 C++入门

第 1 章 编写第一个程序..... 2

- 1.1 使用 C++.....2
- 1.2 寻找编译器.....3
- 1.3 编译和链接源代码.....5
- 1.4 创建您的第一个程序.....5
- 1.5 总结.....6
- 1.6 问与答.....7
- 1.7 作业.....7
 - 1.7.1 测验.....7
 - 1.7.2 答案.....8
 - 1.7.3 练习.....8

第 2 章 程序的组成部分..... 9

- 2.1 使用 C++的原因.....9
 - 2.1.1 编程风格.....10
 - 2.1.2 C++和面向对象编程.....11
- 2.2 程序的组成部分.....12
 - 2.2.1 预处理器编译指令.....12
 - 2.2.2 源代码行.....13
- 2.3 注释.....13
- 2.4 函数.....14
- 2.5 总结.....16
- 2.6 问与答.....16

- 2.7 作业.....17
 - 2.7.1 测验.....17
 - 2.7.2 答案.....18
 - 2.7.3 练习.....18

第 3 章 创建变量和常量..... 19

- 3.1 变量是什么.....19
 - 3.1.1 在内存中存储变量.....20
 - 3.1.2 无符号变量和带符号变量.....21
 - 3.1.3 变量类型.....21
- 3.2 定义变量.....22
- 3.3 给变量赋值.....23
- 3.4 使用类型定义.....24
- 3.5 常量.....25
 - 3.5.1 定义常量.....26
 - 3.5.2 枚举常量.....26
- 3.6 总结.....26
- 3.7 问与答.....27
- 3.8 作业.....27
 - 3.8.1 测验.....27
 - 3.8.2 答案.....28
 - 3.8.3 练习.....28

第 4 章 使用表达式、语句和运算符..... 29

- 4.1 语句.....29
 - 4.1.1 空白.....29
 - 4.1.2 复合语句.....30

4.2 表达式	30	第 6 章 控制程序流程	55
4.3 运算符	31	6.1 循环	55
4.3.1 赋值运算符	31	6.2 while 循环	55
4.3.2 数学运算符	31	6.2.1 退出循环	56
4.3.3 组合运算符	32	6.2.2 跳到下一次循环	57
4.3.4 递增和递减运算符	32	6.3 do-while 循环	58
4.3.5 前缀运算符和后缀运算符	32	6.4 for 循环	59
4.3.6 运算符优先级	34	6.4.1 高级 for 循环	61
4.3.7 关系运算符	36	6.4.2 嵌套循环	61
4.4 if-else 条件语句	36	6.5 switch 语句	62
4.4.1 else 子句	37	6.6 总结	64
4.4.2 复合 if 语句	37	6.7 问与答	64
4.5 逻辑运算符	39	6.8 作业	65
4.5.1 与运算符	39	6.8.1 测验	65
4.5.2 或运算符	39	6.8.2 答案	65
4.5.3 非运算符	39	6.8.3 练习	66
4.5.4 关系运算符和 逻辑运算符的优先级	40	第 7 章 使用数组和字符串存储信息	67
4.6 棘手的表达式值	40	7.1 数组是什么	67
4.7 总结	40	7.2 写入时超过数组末尾	69
4.8 问与答	41	7.3 初始化数组	69
4.9 作业	41	7.4 多维数组	70
4.9.1 测验	41	7.4.1 初始化多维数组	71
4.9.2 答案	42	7.4.2 内存简介	72
4.9.3 练习	42	7.5 字符数组	72
第 5 章 调用函数	43	7.6 复制字符串	74
5.1 函数是什么	43	7.7 总结	75
5.2 声明和定义函数	43	7.8 问与答	75
5.3 在函数中使用变量	45	7.9 作业	75
5.3.1 局部变量	45	7.9.1 测验	75
5.3.2 全局变量	46	7.9.2 答案	76
5.4 函数参数	47	7.9.3 练习	76
5.5 从函数返回值	48		
5.6 默认函数参数	50		
5.7 函数重载	51		
5.8 总结	52		
5.9 问与答	53		
5.10 作业	53		
5.10.1 测验	53		
5.10.2 答案	54		
5.10.3 练习	54		
		第二部分 类	
		第 8 章 创建基本类	78
		8.1 类型是什么	78
		8.2 创建新类型	78
		8.3 类和成员	79
		8.3.1 声明类	79
		8.3.2 创建对象	80

8.4	访问类成员	80
8.5	实现成员函数	81
8.6	创建和删除对象	83
8.6.1	默认构造函数	84
8.6.2	编译器提供的构造函数	84
8.7	总结	86
8.8	问与答	87
8.9	作业	87
8.9.1	测验	87
8.9.2	答案	87
8.9.3	练习	88
第9章	高级类	89
9.1	const 成员函数	89
9.2	接口和实现	90
9.3	组织类声明和函数定义	90
9.4	内联实现	90
9.5	将其他类用作成员数据的类	92
9.6	总结	96
9.7	问与答	96
9.8	作业	96
9.8.1	测验	96
9.8.2	答案	97
9.8.3	练习	97

第三部分 内存管理

第10章	创建指针	100
10.1	理解指针及其用途	100
10.1.1	在指针中存储地址	102
10.1.2	间接运算符（间接地使用指针）	103
10.1.3	指针、地址和变量	104
10.1.4	使用指针操作数据	105
10.1.5	查看存储在指针中的地址	105
10.1.6	为何使用指针	107
10.2	栈和堆	107
10.2.1	使用关键字 new	108
10.2.2	使用关键字 delete	109
10.2.3	避免内存泄露	110
10.3	总结	111

10.4	问与答	111
10.5	作业	111
10.5.1	测验	111
10.5.2	答案	112
10.5.3	练习	112
第11章	开发高级指针	113
11.1	在堆中创建对象	113
11.2	删除对象	113
11.3	使用指针访问数据成员	114
11.4	堆中的数据成员	115
11.5	this 指针	117
11.6	悬摆指针	118
11.7	const 指针	118
11.8	const 指针和 const 成员函数	119
11.9	总结	120
11.10	问与答	121
11.11	作业	121
11.11.1	测验	121
11.11.2	答案	121
11.11.3	练习	122
第12章	创建引用	123
12.1	什么是引用	123
12.2	创建引用	123
12.3	将地址运算符用于引用	124
12.4	可引用的目标	126
12.5	空指针和空引用	127
12.6	按引用传递函数参数	127
12.6.1	使用指针实现 swap()	128
12.6.2	使用引用实现 swap()	129
12.7	理解函数头和原型	130
12.8	返回多个值	130
12.9	总结	133
12.10	问与答	133
12.11	作业	133
12.11.1	测验	133
12.11.2	答案	134
12.11.3	练习	134
第13章	高级引用和指针	135
13.1	按引用传递以提高效率	135

13.2	传递 const 指针	137
13.3	作为指针替代品的引用	139
13.4	什么情况下使用引用以及 什么情况下使用指针	141
13.5	不要返回不在作用域内的 引用	141
13.6	返回指向堆中对象的引用	142
13.7	谁拥有指针	143
13.8	总结	144
13.9	问与答	144
13.10	作业	144
13.10.1	测验	144
13.10.2	答案	145
13.10.3	练习	145

第四部分 高级 C++

第 14 章	高级函数	148
14.1	重载成员函数	148
14.2	使用默认值	150
14.3	初始化对象	151
14.4	复制构造函数	152
14.5	总结	155
14.6	问与答	155
14.7	作业	155
14.7.1	测验	156
14.7.2	答案	156
14.7.3	练习	156
第 15 章	运算符重载	157
15.1	重载运算符	157
15.1.1	编写递增方法	158
15.1.2	重载后缀运算符	160
15.1.3	重载加法运算符	161
15.1.4	对运算符重载的限制	162
15.1.5	赋值运算符	163
15.2	转换运算符	165
15.3	总结	167
15.4	问与答	168
15.5	作业	168
15.5.1	测验	168

15.5.2	答案	168
15.5.3	练习	169

第五部分 继承和多态

第 16 章	使用继承扩展类	172
16.1	什么是继承	172
16.1.1	继承和派生	173
16.1.2	动物和继承	173
16.1.3	派生语法	173
16.2	私有和保护	175
16.3	构造函数和析构函数	176
16.4	将参数传递给基类构造函数	178
16.5	重写函数	181
16.5.1	重载和重写	183
16.5.2	隐藏基类方法	183
16.5.3	调用基类方法	184
16.6	总结	185
16.7	问与答	185
16.8	作业	185
16.8.1	测验	186
16.8.2	答案	186
16.8.3	练习	186
第 17 章	使用多态和派生类	187
17.1	使用虚函数实现多态	187
17.2	虚成员函数的工作原理	190
17.2.1	不能通过基类指针访问 派生类特有的方法	191
17.2.2	切除	191
17.2.3	虚析构函数	193
17.2.4	虚复制构造函数	193
17.2.5	使用虚成员函数的代价	196
17.3	总结	196
17.4	问与答	196
17.5	作业	196
17.5.1	测验	196
17.5.2	答案	197
17.5.3	练习	197
第 18 章	使用高级多态	198
18.1	单继承存在的问题	198

18.2	抽象数据类型	201
18.2.1	纯虚函数	203
18.2.2	实现纯虚函数	204
18.2.3	复杂的抽象层次结构	206
18.2.4	哪些类是抽象的	209
18.3	总结	210
18.4	问与答	210
18.5	作业	210
18.5.1	测验	210
18.5.2	答案	211
18.5.3	练习	211
第 19 章	使用链表存储信息	212
19.1	链表和其他结构	212
19.2	链表案例研究	213
19.2.1	委托	213
19.2.2	链表的组成部分	213
19.3	作为对象的链表	219
19.4	总结	220
19.5	问与答	220
19.6	作业	220
19.6.1	测验	221
19.6.2	答案	221
19.6.3	练习	221

第六部分 特殊主题

第 20 章	使用特殊的类、函数和指针	224
20.1	静态成员数据	224
20.2	静态成员函数	226
20.3	将其他类对象作为成员	227
20.3.1	访问被包含类的成员	231
20.3.2	禁止访问包含类的成员	232
20.3.3	按引用还是按值复制	232
20.4	友元类和友元函数	232
20.5	函数指针	233
20.5.1	函数指针数组	235
20.5.2	将函数指针传递给 其他函数	237
20.5.3	将 typedef 用于函数 指针	239
20.5.4	成员函数指针	239

20.5.5	成员函数指针数组	241
20.6	总结	242
20.7	问与答	243
20.8	作业	243
20.8.1	问与答	243
20.8.2	答案	244
20.8.3	练习	244

第 21 章 使用 C++0x 新增的功能

21.1	下一版 C++	245
21.2	空指针常量	245
21.3	编译阶段常量表达式	247
21.4	自动确定类型的变量	248
21.5	新的 for 循环	250
21.6	总结	250
21.7	问与答	251
21.8	作业	251
21.8.1	测验	251
21.8.2	答案	252
21.8.3	练习	252

第 22 章 面向对象分析和设计

22.1	开发周期	253
22.2	模拟报警系统	253
22.2.1	概念化	254
22.2.2	分析和需求	254
22.2.3	高级设计和详细设计	254
22.2.4	其他对象	255
22.2.5	设计类	255
22.2.6	添加其他类	256
22.2.7	事件循环	256
22.3	案例研究	258
22.3.1	三思而后行	259
22.3.2	分而治之	259
22.3.3	邮件格式	259
22.3.4	初步类设计	260
22.3.5	同根和不同根	261
22.3.6	设计接口	262
22.3.7	建立原型	263
22.3.8	80/80 规则	263
22.3.9	设计 PostMasterMessage 类	264
22.3.10	应用程序编程接口	264

第一部分

C++入门

第 1 章 编写第一个程序

第 2 章 程序的组成部分

第 3 章 创建变量和常量

第 4 章 使用表达式、语句和运算符

第 5 章 调用函数

第 6 章 控制程序流程

第 7 章 使用数组和字符串存储信息

第 1 章

编写第一个程序

本章介绍如下内容：

- C++是如何发明的以及发明它的原因。
- 如何寻找 C++编译器？
- 如何创建并编译第一个程序？
- 如何链接并运行程序？

1.1 使用 C++

1979 年，美国贝尔实验室的一位丹麦计算机科学家开始着手改进 C 编程语言，Bjarne Stroustrup 在其个人网站上解释说，他希望有一种可以高效而优雅地编写程序的语言。

这也是很多人的愿望。

Stroustrup 将其作品取名为 C++，几十年来，它一直占据着世界顶级编程语言的宝座。多年来，很多编程语言如流星般闪过，但对台式机、智能手机和 MP3 播放器等嵌入式设备以及众多其他计算环境的软件开发而言，C++始终是不错的选择。

C++是一种可移植的语言，适用于 Microsoft Windows、Apple Mac OS、Linux 和 UNIX 系统。要学习这种语言，最佳的方式是编写程序，而不考虑运行程序的操作系统。

本书从实用的角度介绍 C++，没有对使用的操作系统做任何假设。之所以能够做到这一点，是因为本书介绍的是标准 C++（也被称为 ANSI/ISO C++），这是国际上达成一致的版本，可移植到任何平台和开发环境。

本书的所有代码都是标准 ANSI/ISO C++，可在任何遵循最新 C++标准的开发环境中运行。

另外，也介绍了下一个版本（C++0x）的新功能。该版本将与 2012 年初发行，但有些流行的开发环境已经支持其中一些最有用的功能。

C++程序是使用一组系统工作的工具开发的，这些工具称为编译器和链接器。