



C++游戏

编程入门 (第4版)

BEGINNING C++ THROUGH GAME
PROGRAMMING, 4TH EDITION

[美] Michael Dawson 著 李军 译



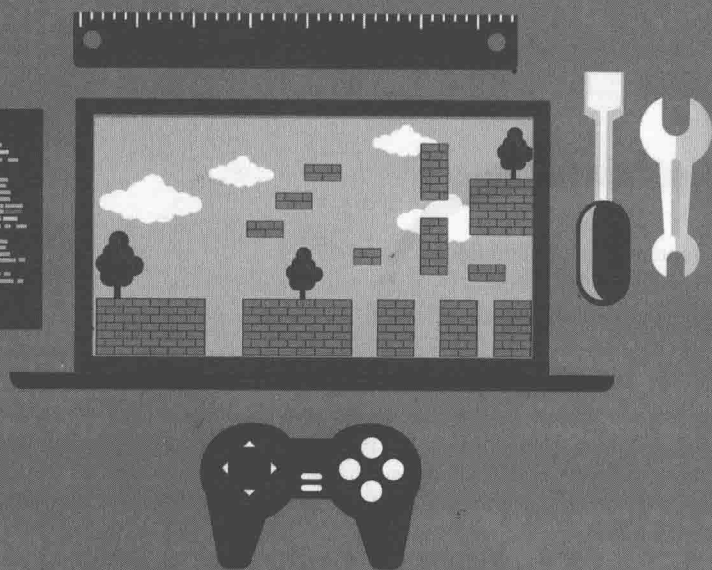
中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



游戏设计与开发



C++ 游戏

编程入门 (第4版)

[美] Michael Dawson 著 李军 译

人民邮电出版社
北京

图书在版编目 (C I P) 数据

C++游戏编程入门 : 第4版 / (美) 道森
(Dawson, M.) 著 ; 李军译. — 北京 : 人民邮电出版社,
2015.8

ISBN 978-7-115-39639-6

I. ①C… II. ①道… ②李… III. ①游戏程序—C语言—程序设计 IV. ①TP311.5

中国版本图书馆CIP数据核字(2015)第146379号

版权声明

Beginning C++ Through Game Programming, Fourth Edition

Michael Dawson

Copyright © 2015 Course Technology, a part of Cengage Learning.

Original edition published by Cengage Learning. All Rights reserved.

本书原版由圣智学习出版公司出版。版权所有, 盗印必究。

Posts & Telecom Press is authorized by Cengage 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.

本书中文简体字翻译版由圣智学习出版公司授权人民邮电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)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。

978-7-115-39639-6

Cengage Learning Asia Pte. Ltd.

151 Lorong Chuan, #02-08 New Tech Park, Singapore 556741

本书封面贴有 Cengage Learning 防伪标签, 无标签者不得销售。

-
- ◆ 著 [美] Michael Dawson
 - 译 李 军
 - 责任编辑 陈冀康
 - 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>
 - 北京鑫正大印刷有限公司印刷
 - ◆ 开本: 800×1000 1/16
 - 印张: 22
 - 字数: 412 千字 2015 年 8 月第 1 版
 - 印数: 1—3 000 册 2015 年 8 月北京第 1 次印刷
 - 著作权合同登记号 图字: 01-2015-2392 号
-

定价: 59.00 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

本书从游戏编程的角度介绍 C++ 语言，既独具匠心又妙趣横生。

全书共 10 章，每章介绍 C++ 语言的一个或数个重要的知识领域，同时通过一个游戏示例项目的开发进行实践和讲解。每章的结尾，会在一个游戏项目中将一些最重要的概念组合起来。最后一章的游戏将综合运用本书介绍的概念与技巧，创建一个相对复杂的游戏，涵盖了本书介绍过的所有主要概念。随着学习的深入，读者将学会如何组织编程项目，如何将问题分解为可管理的子问题块，以及如何精炼代码。

本书适合任何想编写游戏的读者，主要针对初学者，并假设读者之前没有任何编程经验。通过阅读本书，并在实验中实践，读者将为掌握 C++ 这门语言并为游戏编程打下坚实的基础。

前言

好莱坞能带来最好视觉效果、声乐效果以及纯粹的兴奋，顶级的计算机游戏也完全可以与其媲美。但是游戏这种娱乐方式与其他方式不同，它们能让玩家一连好几个小时守在屏幕跟前。能让游戏如此与众不同并且引人入胜的原因在于交互性。在计算机游戏中，我们不是坐下来观看主人公如何与怪物搏斗，而是自己担任主角。实现这种交互性的关键在于编程。编程让外星生物、外星人进攻中队或整支敌人军队在不同情况下对玩家做出不同的反应。编程让游戏的故事情节能够以新的方式展开。实际上，作为编程的结果，游戏可能以编写者想象不到的方式与玩家进行交互。

尽管有数以千计的计算机编程语言，但是 C++ 是游戏行业的标准语言。如果您去您最爱的商场的 PC 游戏区逛逛，并随手拿起一款游戏，它很可能主要或完全是用 C++ 编写的。如果想要更加专业地编写计算机游戏，就必须了解 C++。

本书的目的在于从游戏编程的角度介绍 C++ 语言。尽管没有哪本书可以让人同时掌握 C++ 与游戏编程这两门高深的主题，但是本书可以作为这两个主题的入门读物。

本书读者对象

本书适合任何想编写游戏的读者，主要针对初学者，并假设读者之前没有任何编程经验。如果您可以熟练使用计算机，则马上可以开始您的游戏编程之旅。虽然本书是写给初学者的，但并不代表学习 C++ 与游戏编程很简单。您必须阅读本书，并在实验中实践。读完本书，您将为掌握 C++ 这门游戏编程语言打下坚实的基础。

本书的组织方式

本书从 C++ 与游戏编程的基础开始，假设读者对两者都没有经验。随着章节的推进，本书将在已学内容的基础上介绍更高级的内容。

本书的每一章介绍一个或几个相关主题。在介绍概念的同时会给出长度较短并且与游戏相关的程序来进行演示。每章的结尾会在一个游戏项目中将一些最重要的概念组合起来。本书最后一章以一个最具雄心的项目来结束，它涵盖了本书介绍过的所有主要概念。

除了介绍 C++ 与游戏编程之外，本书还介绍如何组织编程工作，如何将问题分解为可管理的子问题块，以及如何精炼代码。您有时会遇到一些困难，但总是可以克服。总体而

言,学习过程是充满乐趣的。在学习过程中,您将创建一些很酷的计算机游戏,并理解一些游戏编程的技巧。

第1章 类型、变量与标准 I/O: Lost Fortune

本章将介绍游戏行业标准语言 C++ 的基础知识,以及如何在控制台窗口显示输出、执行算术计算、使用变量以及从键盘获取用户输入。

第2章 真值、分支与游戏循环: Guess My Number

通过编程让程序根据某种条件执行、跳转以及重复执行代码块,您将创建一些更加有趣的的游戏。本章将介绍如何生成随机数来增加游戏的不可预测性,还将介绍游戏循环——组织游戏不断运行的基本方式。

第3章 for 循环、字符串与数组: Word Jumble

本章将介绍序列与字符串(非常适合单词游戏的字符序列),以及软件对象(用于表示游戏中对象的实体,譬如外星飞行器、生命药水甚至玩家自身)。

第4章 标准模板库: Hangman

本章将介绍一个功能强大的库(游戏程序员甚至非游戏程序员依赖它来存储对象的集合,譬如玩家物品栏中的物品),还有协助程序员规划更大型游戏程序的技术。

第5章 函数: Mad Lib

本章将介绍如何将游戏程序分解为更小的、可管理的代码块。分解方法是使用游戏中的基本逻辑单元——函数。

第6章 引用: Tic-Tac-Toe

本章将介绍如何在程序的不同部分使用高效与清晰的方式共享信息,一个简单的 AI(人工智能)例子,以及为计算机对手赋予一些人格的方法。

第7章 指针: Tic-Tac-Toe 2.0

本章将介绍 C++ 的一些最底层与最强大的特征,如直接对计算机内存进行寻址与操作的方法。

第8章 类: Critter Caretaker

本章将介绍如何创建自定义类型的对象,以及如何通过向对象编程来定义对象之间的交互方式。在学习过程中,您将创建要照顾的自己的动物。

第9章 高级类与动态内存: Game Lobby

本章将介绍如何扩展与计算机之间直接的连接,以及按照游戏程序需求获取与释放内存的方法,还有使用“动态”内存的隐患与避免方法。

第10章 继承与多态: Blackjack

本章将介绍根据其他对象来定义新对象的方法,随后将所有学过的知识融汇到一个大

型游戏中。通过创建经典的赌博游戏 Blackjack，我们将介绍大型项目的设计与实现方法。

本书约定

在本书中，我使用了一些特殊约定，如下所示。

提示

这部分内容给出一些好的建议，帮助您成为更好的游戏程序员。

陷阱

这部分内容指出容易犯错的地方。

技巧

这部分内容给出一些技巧，可以让游戏编程更简单。

现实世界

这部分内容是一些关于游戏编程的真实情况。

本书程序的源代码

本书的所有源代码都可以从 www.cengageptr.com/downloads 下载到。您可以通过检索本书的 ISBN（978-1-305-10991-9）来获得相关源代码。

关于编译器

现在讨论编译器或许为时过早，但编译器很重要，因为是它们将编写的源代码翻译成计算机可以运行的程序。如果使用的是 Windows 操作系统，建议使用 Microsoft 的 Visual Studio Express 2013 for Windows Desktop，因为它包含了一个现代 C++ 编译器，并且是免费的。软件安装好之后，请查看本书的附录 A，其中介绍了使用 Visual Studio Express 2013 for Windows Desktop 编译 C++ 程序的方法。如果您使用其他的编译器或 IDE，请查阅其文档。

致谢

你曾阅读的每本书都延续一个弥天大谎。我要透露出出版行业的一个小秘密——一个人是

“完成”不了一本书的。没错，许多书（包括本书）的封面上只有一个名字，但是最终完成著作却要靠一个敬业的团队。其他作者不能独立完成著作，我也同样无法做到。所以，我想感谢所有对本书新版本提供帮助的人。

感谢 Dan Foster，他身兼两职，既是项目编辑，也是文字编辑。Dan 看过本书的多个版本，并且让本书得到了进一步的提升。

感谢技术评审 Joshua Smith，他确保程序能够正确运行。

感谢校对 Kelly Talbot，她让本书从字面上看上去很棒。

还要感谢资深策划编辑 Emi Smith 对我的鼓励。

最后，我要感谢那些创建了伴随我成长的游戏的所有游戏程序员。他们启发我做自己的小游戏，并最终在游戏行业工作。希望我能激励一些读者做同样的事。

作者简介

Michael Dawson 是一名游戏程序作家，也是一名教授学生编写自己游戏的艺术和科学老师。Mike 曾经在 UCLA Extension、Digital Media Academy 和 Los Angeles Film School 开发和教授游戏编程课程。此外，在全球的许多高校中，他的著作都是必读的。

Mike 在游戏行业曾是一名制作人和设计师，也在冒险游戏中充当玩家控制的主要角色 Mike Dawson。在游戏中，玩家指导 Dawson 的数字影像，他必须在外星人胚胎植入自己的大脑之前阻止外星人入侵。

现实生活中，Mike 是 *Beginning C++ Through Game Programming*、*Python Programming for the Absolute Beginner*、*C++ Projects: Programming with Text-Based Games* 和 *Guide to Programming with Python* 的作者。他在南加利福尼亚大学获得计算机科学的学士学位。可以访问他的个人网页 www.programgames.com，来获得关于他的著作的更多支持。

目录

第 1 章 类型、变量与标准 I/O:

Lost Fortune	1	1.4.3 理解整型与浮点型 除法	13
1.1 C++简介	1	1.4.4 使用模除运算符	14
1.1.1 使用 C++编写游戏	1	1.4.5 运算符的优先级	14
1.1.2 生成可执行文件	2	1.5 声明和初始化变量	14
1.1.3 错误处理	3	1.5.1 Game Stats 程序简介	14
1.1.4 理解 ISO 标准	4	1.5.2 基本类型	16
1.2 编写第一个 C++程序	4	1.5.3 类型修饰符	16
1.2.1 Game Over 程序简介	4	1.5.4 变量声明	17
1.2.2 注释	5	1.5.5 变量命名	18
1.2.3 使用空白字符	6	1.5.6 变量的赋值	19
1.2.4 包含其他文件	6	1.5.7 变量初始化	20
1.2.5 定义 main() 函数	7	1.5.8 显示变量值	20
1.2.6 通过标准输出显示 文本	7	1.5.9 获取用户输入	20
1.2.7 语句的终止	8	1.5.10 为类型定义新名称	21
1.2.8 从 main() 函数返回值	8	1.5.11 类型的选择	21
1.3 使用 std 名称空间	9	1.6 使用变量进行算术运算	21
1.3.1 Game Over 2.0 程序 简介	9	1.6.1 Game Stats 2.0 程序简介	22
1.3.2 使用 using 指令	10	1.6.2 修改变量值	23
1.3.3 Game Over 3.0 程序简介	10	1.6.3 使用组合赋值运算符	23
1.3.4 使用 using 声明	11	1.6.4 递增运算符与递减 运算符	24
1.3.5 使用 using 的时机	11	1.6.5 整数的溢出处理	25
1.4 使用算术运算符	12	1.7 使用常量	26
1.4.1 Expensive Calculator 程序简介	12	1.7.1 Game Stats 3.0 程序简介	26
1.4.2 加法、减法与乘法	13	1.7.2 使用常量	27
		1.7.3 使用枚举类型	27

1.8	Lost Fortune 简介	28	2.6.1	Play Again 游戏简介	48
1.8.1	创建程序	29	2.6.2	使用 while 循环	49
1.8.2	从玩家获取信息	30	2.7	使用 do 循环	50
1.8.3	讲故事	30	2.7.1	Play Again 2.0 程序简介	50
1.9	本章小结	31	2.7.2	使用 do 循环	51
1.10	问与答	32	2.8	使用 break 和 continue 语句	52
1.11	问题讨论	33	2.8.1	Finicky Counter 程序简介	52
1.12	习题	34	2.8.2	创建 while(true)循环	53
第 2 章 真值、分支与游戏循环:			2.8.3	使用 break 语句退出循环	53
	Guess My Number	35	2.8.4	使用 continue 语句跳转到循环开始	54
2.1	理解真值	35	2.8.5	使用 break 和 continue 的时机	54
2.2	使用 if 语句	36	2.9	使用逻辑运算符	54
2.2.1	Score Rater 程序简介	36	2.9.1	Designers Network 程序简介	55
2.2.2	验证真与假	38	2.9.2	使用逻辑与运算符	57
2.2.3	值的真与假	39	2.9.3	使用逻辑或运算符	58
2.2.4	使用关系运算符	39	2.9.4	使用逻辑非运算符	58
2.2.5	if 语句的嵌套	40	2.9.5	运算符的优先级	59
2.3	使用 else 子句	40	2.10	随机数的生成	60
2.3.1	Score Rater 2.0 程序简介	41	2.10.1	Die Roller 程序简介	60
2.3.2	两种创建分支的方法	42	2.10.2	调用 rand()函数	61
2.4	使用带 else 子句的 if 语句序列	43	2.10.3	为随机数生成器确定种子	61
2.4.1	Score Rater 3.0 程序简介	43	2.10.4	在一定范围内计算	62
2.4.2	创建带 else 子句的 if 语句序列	44	2.11	理解游戏主循环	63
2.5	使用 switch 语句	45	2.12	Guess My Number 游戏简介	64
2.5.1	Menu Chooser 程序简介	46			
2.5.2	创建多路分支	48			
2.6	使用 while 循环	48			

2.12.1 采用游戏主循环	64	3.4.4 使用数组元素的成员 函数	87
2.12.2 初始化游戏	65	3.4.5 数组边界	87
2.12.3 创建游戏主循环	66	3.5 理解 C 风格字符串	88
2.12.4 游戏结束	67	3.6 使用多维数组	89
2.13 本章小结	67	3.6.1 Tic-Tac-Toe Board 程序 简介	90
2.14 问与答	68	3.6.2 创建多维数组	91
2.15 问题讨论	69	3.6.3 多维数组的索引	92
2.16 习题	69	3.7 Word Jumble 程序简介	92
第 3 章 for 循环、字符串与数组:		3.7.1 创建程序	93
Word Jumble	71	3.7.2 选择单词	93
3.1 使用 for 循环	71	3.7.3 单词乱序	94
3.1.1 Counter 程序简介	72	3.7.4 欢迎界面	95
3.1.2 使用 for 循环计数	73	3.7.5 进入游戏主循环	95
3.1.3 在 for 循环中使用空 语句	74	3.7.6 游戏结束	96
3.1.4 for 循环的嵌套	74	3.8 本章小结	96
3.2 了解对象	75	3.9 问与答	97
3.3 使用 string 对象	77	3.10 问题讨论	99
3.3.1 String Tester 程序简介	77	3.11 习题	99
3.3.2 创建 string 对象	79	第 4 章 标准模板库: Hangman	100
3.3.3 string 对象的连接	79	4.1 标准模板库简介	100
3.3.4 使用 size() 成员函数	79	4.2 使用 vector	101
3.3.5 索引 string 对象	80	4.2.1 Hero's Inventory 2.0 程序 简介	101
3.3.6 循环访问 string 对象	81	4.2.2 使用向量的准备工作	103
3.3.7 使用 find() 成员函数	81	4.2.3 向量的声明	103
3.3.8 使用 erase() 成员函数	82	4.2.4 使用 push_back() 成员 函数	104
3.3.9 使用 empty() 成员函数	83	4.2.5 使用 size() 成员函数	104
3.4 使用数组	83	4.2.6 向量的索引	105
3.4.1 Hero's Inventory 程序 简介	83	4.2.7 调用元素的成员函数	105
3.4.2 创建数组	85		
3.4.3 数组的索引	86		

4.2.8 使用 pop_back()成员函数	106	4.8.1 游戏规划	123
4.2.9 使用 clear()成员函数	106	4.8.2 创建程序	124
4.2.10 使用 empty()成员函数	106	4.8.3 变量与常量的初始	124
4.3 使用迭代器	107	4.8.4 进入游戏主循环	125
4.3.1 Hero's Inventory 3.0 程序简介	107	4.8.5 获取玩家的猜测	125
4.3.2 迭代器的声明	109	4.8.6 游戏结束	126
4.3.3 循环访问向量	110	4.9 本章小结	127
4.3.4 修改向量元素的值	111	4.10 问与答	128
4.3.5 访问向量元素的成员函数	112	4.11 问题讨论	129
4.3.6 使用向量的成员函数 insert()	113	4.12 习题	129
4.3.7 使用向量的成员函数 erase()	113	第 5 章 函数: Mad Lib	131
4.4 使用算法	114	5.1 创建函数	131
4.4.1 High Scores 程序简介	114	5.1.1 Instructions 程序简介	131
4.4.2 使用算法的准备工作	116	5.1.2 函数声明	132
4.4.3 使用 find()算法	116	5.1.3 函数定义	133
4.4.4 使用 random_shuffle() 算法	117	5.1.4 函数调用	134
4.4.5 使用 sort()算法	117	5.1.5 理解抽象	134
4.5 理解向量的性能	118	5.2 使用形参和返回值	134
4.5.1 向量的增长	118	5.2.1 Yes or No 程序简介	135
4.5.2 元素的插入与删除	120	5.2.2 返回值	136
4.6 其他 STL 容器	120	5.2.3 传递参数值	137
4.7 对程序进行规划	121	5.2.4 理解封装	138
4.7.1 使用伪代码	121	5.3 理解软件重用	139
4.7.2 逐步细化	122	5.4 使用作用域	140
4.8 Hangman 简介	123	5.4.1 Scoping 程序简介	140
		5.4.2 使用独立的作用域	141
		5.4.3 使用嵌套作用域	142
		5.5 使用全局变量	144
		5.5.1 Global Reach 程序简介	144
		5.5.2 声明全局变量	145
		5.5.3 访问全局变量	145

5.5.4	隐藏全局变量	146	6.1.1	Referencing 程序简介	162
5.5.5	修改全局变量	146	6.1.2	创建引用	164
5.5.6	尽量少使用全局 变量	147	6.1.3	访问被引用的值	165
5.6	使用全局常量	147	6.1.4	修改被引用的值	165
5.7	使用默认参数	148	6.2	通过传递引用改变实参	166
5.7.1	Give Me a Number 程序 简介	148	6.2.1	Swap 程序简介	166
5.7.2	指定默认参数	149	6.2.2	按值传递参数	167
5.7.3	为形参设置默认参数	150	6.2.3	按引用传递参数	168
5.7.4	重写默认参数	150	6.3	传递引用以提高效率	169
5.8	函数重载	151	6.3.1	Inventory Displayer 程序 简介	169
5.8.1	Triple 程序简介	151	6.3.2	引用传递的陷阱	170
5.8.2	创建重载函数	152	6.3.3	以常量引用声明参数	171
5.8.3	调用重载函数	153	6.3.4	传递常量引用	171
5.9	内联函数	153	6.4	如何传递实参	172
5.9.1	Taking Damage 程序 简介	153	6.5	返回引用	172
5.9.2	函数内联的指定	155	6.5.1	Inventory Referencer 程序 简介	172
5.9.3	调用内联函数	155	6.5.2	返回一个引用	174
5.10	Mad Lib 游戏简介	156	6.5.3	显示返回的引用 的值	175
5.10.1	创建程序	156	6.5.4	将返回的引用赋值给 引用	175
5.10.2	main()函数	157	6.5.5	将返回的引用赋值给 变量	175
5.10.3	askText()函数	157	6.5.6	通过返回的引用修改 对象	176
5.10.4	askNumber()函数	158	6.6	Tic-Tac-Toe 游戏简介	176
5.10.5	tellStory()函数	158	6.6.1	游戏规划	177
5.11	本章小结	159	6.6.2	创建程序	179
5.12	问与答	160	6.6.3	main()函数	179
5.13	问题讨论	161	6.6.4	instructions()函数	180
5.14	习题	161			
第 6 章	引用: Tic-Tac-Toe	162			
6.1	使用引用	162			

6.6.5	askYesNo()函数	181	7.3.1	Swap Pointer Version 程序简介	202
6.6.6	askNumber()函数	181	7.3.2	值传递	204
6.6.7	humanPiece()函数	182	7.3.3	传递常量指针	204
6.6.8	opponent()函数	182	7.4	返回指针	205
6.6.9	displayBoard()函数	182	7.4.1	Inventory Pointer 程序简介	205
6.6.10	winner()函数	183	7.4.2	返回指针	207
6.6.11	isLegal()函数	184	7.4.3	使用返回的指针显示值	208
6.6.12	humanMove()函数	185	7.4.4	将返回的指针赋值给指针	208
6.6.13	computerMove()函数	185	7.4.5	将返回的指针指向的值赋值给变量	209
6.6.14	announceWinner()函数	188	7.4.6	通过返回的指针修改对象	210
6.7	本章小结	188	7.5	理解指针与数组的关系	210
6.8	问与答	189	7.5.1	Array Passer 程序简介	211
6.9	问题讨论	191	7.5.2	将数组名用作常量指针	212
6.10	习题	191	7.5.3	数组的传递与返回	213
第 7 章	指针: Tic-Tac-Toe 2.0	192	7.6	Tic-Tac-Toe 2.0 程序简介	214
7.1	指针基础	192	7.7	本章小结	214
7.1.1	Pointing 程序简介	193	7.8	问与答	215
7.1.2	指针的声明	194	7.9	问题讨论	217
7.1.3	指针的初始化	195	7.10	习题	217
7.1.4	将地址赋值给指针	195	第 8 章	类: Critter Caretaker	218
7.1.5	指针的解引用	196	8.1	定义新类型	218
7.1.6	指针的重新赋值	197	8.1.1	Simple Critter 程序简介	218
7.1.7	使用对象的指针	198	8.1.2	定义一个类	220
7.2	指针和常量	199			
7.2.1	使用常量指针	199			
7.2.2	使用指向常量的指针	200			
7.2.3	使用指向常量的常量指针	200			
7.2.4	常量与指针小结	201			
7.3	传递指针	202			

8.1.3	成员函数的定义	221	8.6	本章小结	240
8.1.4	对象的实例化	221	8.7	问与答	241
8.1.5	数据成员的访问	221	8.8	问题讨论	243
8.1.6	成员函数的调用	222	8.9	习题	243
8.2	使用构造函数	222	第 9 章	高级类与动态内存:	
8.2.1	Constructor Critter 程序 简介	223		Game Lobby	244
8.2.2	构造函数的声明与 定义	224	9.1	使用聚合体	244
8.2.3	构造函数的自动调用	225	9.1.1	Critter Farm 程序 简介	244
8.3	设置成员访问级别	225	9.1.2	使用对象数据成员	247
8.3.1	Private Critter 程序 简介	226	9.1.3	使用容器数据成员	247
8.3.2	指定公有与私有访问 级别	227	9.2	使用友元函数与运算符 重载	248
8.3.3	定义访问器成员函数	228	9.2.1	Friend Critter 程序 简介	249
8.3.4	定义常量成员函数	229	9.2.2	创建友元函数	250
8.4	使用静态数据成员与静态成员 函数	230	9.2.3	运算符重载	251
8.4.1	Static Critter 程序 简介	230	9.3	动态分配内存	252
8.4.2	声明与初始化静态数据 成员	232	9.3.1	Heap 程序简介	252
8.4.3	访问静态数据成员	232	9.3.2	使用 new 运算符	254
8.4.4	声明与定义静态成员 函数	233	9.3.3	避免内存泄漏	256
8.4.5	调用静态成员函数	233	9.4	使用数据成员与堆	257
8.5	Critter Caretaker 游戏简介	234	9.4.1	Heap Data Member 程序 简介	258
8.5.1	游戏规划	235	9.4.2	声明指向堆中值的指针 数据成员	260
8.5.2	规划伪代码	236	9.4.3	声明与定义析构 函数	261
8.5.3	Critter 类	236	9.4.4	声明与定义拷贝构造 函数	262
8.5.4	main() 函数	239	9.4.5	赋值运算符的重载	265
			9.5	Game Lobby 程序简介	267

9.5.1	Player 类	268	10.3.2	调用基类构造函数	289
9.5.2	Lobby 类	269	10.3.3	声明虚基类成员 函数	290
9.5.3	Lobby::AddPlayer()成员 函数	271	10.3.4	重写虚基类成员 函数	290
9.5.4	Lobby::RemovePlayer() 成员函数	272	10.3.5	调用基类成员函数	291
9.5.5	Lobby::Clear()成员 函数	274	10.4	在派生类中使用重载赋值运算 符与拷贝构造函数	292
9.5.6	operator<<()成员 函数	274	10.5	多态简介	292
9.5.7	main()函数	275	10.5.1	Polymorphic Bad Guy 程序简介	293
9.6	本章小结	275	10.5.2	使用基类指针指向派 生类对象	295
9.7	问与答	276	10.5.3	定义虚析构函数	296
9.8	问题讨论	278	10.6	使用抽象类	297
9.9	习题	278	10.6.1	Abstract Creature 程序 简介	297
第 10 章	继承与多态: Blackjack	279	10.6.2	声明纯虚函数	299
10.1	继承简介	279	10.6.3	从抽象类派生类	299
10.1.1	Simple Boss 程序 简介	280	10.7	Blackjack 游戏简介	300
10.1.2	从基类派生	282	10.7.1	类的设计	301
10.1.3	从派生类实例化对象	283	10.7.2	规划游戏的逻辑	304
10.1.4	使用继承成员	283	10.7.3	Card 类	304
10.2	继承访问权的控制	284	10.7.4	Hand 类	306
10.2.1	Simple Boss 2.0 程序 简介	284	10.7.5	GenericPlayer 类	308
10.2.2	对类成员使用访问修 饰符	286	10.7.6	Player 类	309
10.2.3	使用访问修饰符修饰 派生类	286	10.7.7	House 类	311
10.3	调用与重写基类成员函数	287	10.7.8	Deck 类	312
10.3.1	Overriding Boss 程序 简介	287	10.7.9	Game 类	314
			10.7.10	main()函数	317
			10.8	本章小结	319
			10.9	问与答	320

10.10 问题讨论.....	321	附录 C 关键字.....	331
10.11 习题.....	321	附录 D ASCII 字符表.....	333
附录 A 创建第一个 C++程序.....	323	附录 E 转义序列.....	335
附录 B 运算符优先级.....	329		