

CENGAGE
Learning™

• 游戏开发经典丛书 •

(美) Michael Dawson 著
万 千 译

通过游戏编程实战 教新手学C++编程

Beginning C++ Through Game Programming, Third Edition

Beginning C++
Through Game Programming,
Third Edition

CENGAGE
Learning™
圣智学习

清华大学出版社

• 游戏开发经典丛书 •

通过游戏编程实战 教新手学 C++ 编程

(美) Michael Dawson 著

万 千 译

清华大学出版社

北 京

北京市版权局著作权合同登记号 图字：01-2011-2581

Beginning C++ Through Game Programming, Third Edition

Michael Dawson 著，万千 译

Copyright © 2011 by 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。

Tsinghua University 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)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。

ISBN 978-7-302-27195-6

Cengage Learning Asia Pte. Ltd.

5 Shenton Way, # 01-01 UIC Building, Singapore 068808

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

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

图书在版编目(CIP)数据

通过游戏编程实战教新手学 C++编程/(美)道森(Dawson, M.)著；万千译。

—北京：清华大学出版社，2011.11

(游戏开发经典丛书)

书名原文：Beginning C++ Through Game Programming, Third Edition

ISBN 978-7-302-27195-6

I. 通… II. ①道… ②万… III. ①C 语言—程序设计 ②游戏—应用程序—程序设计 IV. ①TP312 ②G899

中国版本图书馆 CIP 数据核字(2011)第 221472 号

责任编辑：王 军 赵利通

装帧设计：牛艳敏

责任校对：邱晓玉

责任印制：李红英

出版发行：清华大学出版社

地 址：北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编：100084

社 总 机：010-62770175

邮 购：010-62786544

投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者：北京密云胶印厂

装 订 者：三河市新茂装订有限公司

经 销：全国新华书店

开 本：185×260 印 张：19.75 字 数：481 千字

版 次：2011 年 11 月第 1 版 印 次：2011 年 11 月第 1 次印刷

印 数：1~4000

定 价：39.00 元

产品编号：041891-01

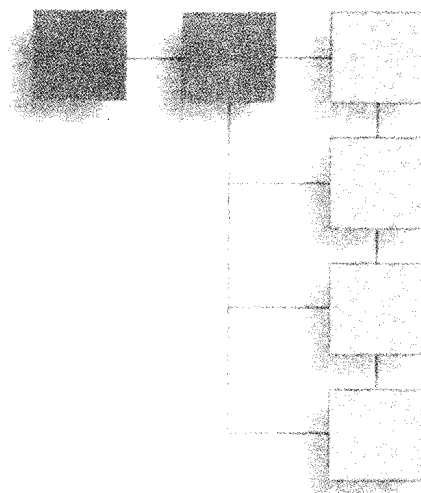
作者简介

Michael Dawson 是一位游戏编程图书的作者。他为 UCLA Extension、The Digital Media Academy 和 The Los Angeles Film School 开设并执教游戏编程课程，指导学生游戏编程的艺术与科学。另外，他的多部著作已成为国内多所大学的必读书目。

Mike 以游戏制作人和设计师的身份进入游戏行业，同时还在一个玩家冒险游戏中担任主角——玩家控制的角色名字就叫 Mike Dawson。在游戏中，Dawson 的大脑中被植入了外星人的胚胎，玩家要指挥数字化形象的 Dawson 在胚胎诞生之前阻止外星人的入侵。

Mike 的著作包括《通过游戏编程实战教新手学 C++编程》、*Python Programming for the Absolute Beginner*、*C++ Projects: Programming with Text-Based Games* 以及 *Guide to Programming with Python*。他在南加州大学获得计算机科学学士学位。关于 Mike 的更多信息，以及要获得他的著作的支持，请访问他的网站 www.programgames.com。

致 谢



您所读过的每一本书都在延续一个谎言。在此，我要指出出版界的一个小秘密：每本书都不是由一个人来完成的。书的封面上确实只有一个名字(本书也不例外)，但实际上每本书的背后都有一个乐于奉献的团队。仅凭作者一个人无法完成图书出版，我也不例外。因此，我要感谢所有帮助本书出版的人们。

感谢身兼项目编辑与文字编辑两职的Jenny Davidson，她让本书的写作能按时完成，并确保了文字的流畅。

感谢技术审阅者 Maneesh Sethi，他帮助确保本书的程序能正确运行。

感谢校对 Michael Beady，他确保了本书文字的正确性。

我还要感谢高级策划编辑 Emi Smith 给予我的鼓励。

最后，我还要感谢那些编写出在我成长过程中玩过的所有游戏的游戏程序员们。他们激发了我进入这一行业以及编写游戏的兴趣。我希望本书也能对您起到同样的激励作用。

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。如果您去您最喜爱的商场的 PC 游戏区逛逛，并随手拿起一款游戏，它主要或完全用 C++ 编写的可能性非常大。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。

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，您将创建一些极妙的计算机游戏，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，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程序中的基本逻辑单元：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，以及如何通过面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

本书采用了如下约定：

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.courseptr.com/或 <http://www.tupwk.com.cn/downpage> 上下载到。您可以通过检索本书的 ISBN 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 C++ 2010 Express Edition，因为它包含了一个现代编译器，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 C++ 2010 Express Edition 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。

目 录

第 1 章 类型、变量与标准

I/O: Lost Fortune	1
1.1 C++简介	1
1.1.1 使用 C++编写游戏	1
1.1.2 生成可执行文件	2
1.1.3 错误处理	3
1.1.4 理解 ISO 标准	3
1.2 编写第一个 C++程序	4
1.2.1 Game Over 程序简介	4
1.2.2 注释	5
1.2.3 使用空白字符	5
1.2.4 包含其他文件	5
1.2.5 定义 main()函数	6
1.2.6 通过标准输出显示文本	6
1.2.7 语句的终止	7
1.2.8 从 main()函数返回值	7
1.3 使用 std 名称空间	8
1.3.1 Game Over 2.0 程序简介	8
1.3.2 使用 using 指令	8
1.3.3 Game Over 3.0 程序简介	8
1.3.4 使用 using 声明	9
1.3.5 使用 using 的时机	9
1.4 使用算术运算符	9
1.4.1 Expensive Calculator 程序简介	10
1.4.2 加法、减法与乘法	11
1.4.3 理解整型与浮点型除法	11
1.4.4 使用取模运算符	11
1.4.5 运算符的优先级	11
1.5 声明和初始化变量	12
1.5.1 Game Stats 程序简介	12
1.5.2 基本类型	13
1.5.3 类型修饰符	13
1.5.4 变量声明	14
1.5.5 变量命名	15
1.5.6 变量的赋值	15
1.5.7 变量初始化	16
1.5.8 显示变量值	16
1.5.9 获取用户输入	17
1.5.10 为类型定义新名称	17
1.5.11 类型的选择	17
1.6 使用变量进行算术运算	18
1.6.1 Game Stats 2.0 程序简介	18
1.6.2 修改变量值	19
1.6.3 使用组合赋值运算符	19
1.6.4 递增运算符与递减 运算符	20
1.6.5 整数的溢出处理	21
1.7 使用常量	21
1.7.1 Game Stats 3.0 程序简介	21

1.7.2 使用常量	22	2.7.2 使用 do 循环	43
1.7.3 使用枚举类型	23	2.8 使用 break 和 continue	
1.8 Lost Fortune 简介	23	语句	44
1.8.1 创建程序	24	2.8.1 Finicky Counter 程序	
1.8.2 从玩家获取信息	24	简介	44
1.8.3 讲故事	25	2.8.2 创建 while(true) 循环	45
1.9 本章小结	26	2.8.3 使用 break 语句退出	
1.10 问与答	26	循环	45
1.11 问题讨论	27	2.8.4 使用 continue 语句	
1.12 习题	27	跳转到循环开始	45
第 2 章 真值、分支与游戏循环:		2.8.5 使用 break 和 continue	
Guess My Number	29	的时机	46
2.1 理解真值	29	2.9 使用逻辑运算符	46
2.2 使用 if 语句	30	2.9.1 Designers Network	
2.2.1 Score Rater 程序简介	30	程序简介	46
2.2.2 验证真与假	32	2.9.2 使用逻辑与运算符	48
2.2.3 值的真与假	33	2.9.3 使用逻辑或运算符	49
2.2.4 使用关系运算符	33	2.9.4 使用逻辑非运算符	49
2.2.5 if 语句的嵌套	34	2.9.5 运算符的优先级	50
2.3 使用 else 子句	34	2.10 随机数的生成	50
2.3.1 Score Rater 2.0 程序简介	34	2.10.1 Die Roller 程序简介	51
2.3.2 创建分支	36	2.10.2 调用 rand() 函数	51
2.4 使用带 else 子句的 if 语句		2.10.3 为随机数生成器	
序列	36	确定种子	52
2.4.1 Score Rater 3.0 程序简介	36	2.10.4 在一定范围内计算	52
2.4.2 创建带 else 子句的 if		2.11 理解游戏主循环	53
语句序列	38	2.12 Guess My Number	
2.5 使用 switch 语句	38	游戏简介	54
2.5.1 Menu Chooser 程序简介	39	2.12.1 采用游戏主循环	54
2.5.2 创建多路分支	40	2.12.2 初始化游戏	55
2.6 使用 while 循环	40	2.12.3 创建游戏主循环	56
2.6.1 Play Again 游戏简介	41	2.12.4 游戏结束	56
2.6.2 使用 while 循环	42	2.13 本章小结	56
2.7 使用 do 循环	42	2.14 问与答	57
2.7.1 Play Again 2.0 程序简介	42	2.15 问题讨论	58
		2.16 习题	59

第3章 for 循环、字符串与数组:

Word Jumble	61
3.1 使用 for 循环	61
3.1.1 Counter 程序简介	62
3.1.2 使用 for 循环计数	63
3.1.3 在 for 循环中使用空 语句	64
3.1.4 for 循环的嵌套	64
3.2 理解对象	65
3.3 使用 string 对象	66
3.3.1 String Tester 程序简介	67
3.3.2 创建 string 对象	68
3.3.3 string 对象的连接	68
3.3.4 使用 size()成员函数	69
3.3.5 索引 string 对象	69
3.3.6 循环访问 string 对象	70
3.3.7 使用 find()成员函数	70
3.3.8 使用 erase()成员函数	71
3.3.9 使用 empty()成员函数	71
3.4 使用数组	72
3.4.1 Hero's Inventory 程序 简介	72
3.4.2 创建数组	73
3.4.3 数组的索引	74
3.4.4 使用数组元素的成员 函数	75
3.4.5 数组边界	75
3.5 理解 C 风格字符串	76
3.6 使用多维数组	77
3.6.1 Tic-Tac-Toe Board 程序简介	77
3.6.2 创建多维数组	78
3.6.3 多维数组的索引	79
3.7 Word Jumble 程序简介	79
3.7.1 创建程序	80
3.7.2 选择单词	80
3.7.3 单词乱序	81

3.7.4 欢迎界面	81
3.7.5 进入游戏主循环	82
3.7.6 游戏结束	82
3.8 本章小结	82
3.9 问与答	83
3.10 问题讨论	84
3.11 习题	85
第4章 标准模板库: Hangman	87
4.1 标准模板库简介	87
4.2 使用 vector	88
4.2.1 Hero's Inventory 2.0 程序简介	88
4.2.2 使用向量的准备工作	90
4.2.3 向量的声明	90
4.2.4 使用 push_back()成员 函数	91
4.2.5 使用 size()成员函数	91
4.2.6 向量的索引	91
4.2.7 调用元素的成员函数	92
4.2.8 使用 pop_back()成员 函数	92
4.2.9 使用 clear()成员函数	92
4.2.10 使用 empty()成员函数	92
4.3 使用迭代器	93
4.3.1 Hero's Inventory 3.0 程序简介	93
4.3.2 迭代器的声明	94
4.3.3 循环访问向量	95
4.3.4 修改向量元素的值	97
4.3.5 访问向量元素的成员 函数	97
4.3.6 使用向量的成员函数 insert()	98
4.3.7 使用向量的成员函数 erase()	98
4.4 使用算法	99
4.4.1 High Scores 程序简介	99

4.4.2	使用算法的准备工作	101	5.4	使用作用域	120
4.4.3	使用 find() 算法	101	5.4.1	Scoping 程序简介	120
4.4.4	使用 random_shuffle() 算法	101	5.4.2	使用独立的作用域	122
4.4.5	使用 sort() 算法	102	5.4.3	使用嵌套作用域	123
4.5	理解向量的性能	102	5.5	使用全局变量	124
4.5.1	向量的增长	103	5.5.1	Global Reach 程序简介	124
4.5.2	元素的插入与删除	104	5.5.2	声明全局变量	125
4.6	其他 STL 容器	104	5.5.3	访问全局变量	125
4.7	对程序进行规划	105	5.5.4	隐藏全局变量	125
4.7.1	使用伪代码	105	5.5.5	修改全局变量	126
4.7.2	逐步细化	106	5.5.6	尽量少使用全局变量	126
4.8	Hangman 简介	106	5.6	使用全局常量	126
4.8.1	游戏规划	107	5.7	使用默认参数	127
4.8.2	创建程序	107	5.7.1	Give Me a Number 程序简介	127
4.8.3	变量与常量的初始化	108	5.7.2	指定默认参数	128
4.8.4	进入游戏主循环	108	5.7.3	为形参设置默认参数	129
4.8.5	获取玩家的猜测	108	5.7.4	重写默认参数	129
4.8.6	游戏结束	109	5.8	函数重载	130
4.9	本章小结	110	5.8.1	Triple 程序简介	130
4.10	问与答	110	5.8.2	创建重载函数	131
4.11	问题讨论	111	5.8.3	调用重载函数	131
4.12	习题	111	5.9	内联函数	132
第 5 章	函数: Mad Lib	113	5.9.1	Taking Damage 程序简介	132
5.1	创建函数	113	5.9.2	函数内联的指定	133
5.1.1	Instructions 程序简介	113	5.9.3	调用内联函数	133
5.1.2	函数声明	114	5.10	Mad Lib 游戏简介	134
5.1.3	函数定义	115	5.10.1	创建程序	134
5.1.4	函数调用	115	5.10.2	main() 函数	135
5.1.5	理解抽象	115	5.10.3	askText() 函数	135
5.2	使用形参和返回值	116	5.10.4	askNumber() 函数	135
5.2.1	Yes or No 程序简介	116	5.10.5	tellStory() 函数	136
5.2.2	返回值	117	5.11	本章小结	136
5.2.3	传递参数值	118	5.12	问与答	137
5.2.4	理解封装	119	5.13	问题讨论	138
5.3	理解软件重用	120	5.14	习题	138

第 6 章 引用: Tic-Tac-Toe139**6.1 使用引用 139**

6.1.1 Referencing 程序简介.....139

6.1.2 创建引用140

6.1.3 访问被引用的值141

6.1.4 修改被引用的值141

6.2 通过传递引用改变实参 142

6.2.1 Swap 程序简介142

6.2.2 按值传递参数144

6.2.3 按引用传递参数144

6.3 传递引用以提高效率 1456.3.1 Inventory Displayer
程序简介145

6.3.2 引用传递的陷阱146

6.3.3 以常量引用声明参数146

6.3.4 传递常量引用147

6.4 如何传递实参 147**6.5 返回引用 148**6.5.1 Inventory Referencer
程序简介148

6.5.2 返回一个引用149

6.5.3 显示返回的引用的值150

6.5.4 将返回的引用赋值给
引用1506.5.5 将返回的引用赋值给
变量1506.5.6 通过返回的引用修改
对象150**6.6 Tic-Tac-Toe 游戏简介 151**

6.6.1 游戏规划151

6.6.2 创建程序153

6.6.3 main()函数154

6.6.4 instructions()函数155

6.6.5 askYesNo()函数155

6.6.6 askNumber()函数156

6.6.7 humanPiece()函数156

6.6.8 opponent()函数156

6.6.9 displayBoard()函数 157

6.6.10 winner()函数 157

6.6.11 isLegal()函数 158

6.6.12 humanMove()函数 159

6.6.13 computerMove()函数 159

6.6.14 announceWinner()函数 161

6.7 本章小结 162**6.8 问与答 162****6.9 问题讨论 164****6.10 习题 164****第 7 章 指针: Tic-Tac-Toe 2.0 165****7.1 指针基础 165**

7.1.1 Pointing 程序简介 166

7.1.2 指针的声明 167

7.1.3 指针的初始化 168

7.1.4 将地址赋值给指针 168

7.1.5 指针的解引用 169

7.1.6 指针的重新赋值 169

7.1.7 使用对象的指针 170

7.2 指针和常量 171

7.2.1 使用常量指针 171

7.2.2 使用指向常量的指针 172

7.2.3 使用指向常量的常量
指针 172

7.2.4 常量与指针小结 173

7.3 传递指针 1737.3.1 Swap Pointer Version
程序简介 173

7.3.2 值传递 175

7.3.3 传递常量指针 175

7.4 返回指针 1767.4.1 Inventory Pointer
程序简介 176

7.4.2 返回指针 178

7.4.3 使用返回的指针显示值 179

7.4.4 将返回的指针赋值给
指针 179

7.4.5	将返回的指针指向的值 赋值给变量.....	179	8.4.3	访问静态数据成员	199
7.4.6	通过返回的指针修改 对象.....	180	8.4.4	声明与定义静态成员 函数.....	200
7.5	理解指针与数组的关系	181	8.4.5	调用静态成员函数	200
7.5.1	Array Passer 程序简介.....	181	8.5	Critter Caretaker 游戏简介	201
7.5.2	将数组名用作常量指针	182	8.5.1	游戏规划	201
7.5.3	数组的传递与返回	182	8.5.2	规划伪代码	202
7.6	Tic-Tac-Toe 2.0 程序简介	183	8.5.3	Critter 类.....	202
7.7	本章小结	184	8.5.4	main()函数.....	205
7.8	问与答	185	8.6	本章小结	206
7.9	问题讨论	186	8.7	问与答	207
7.10	习题	186	8.8	问题讨论	208
			8.9	习题	208
第 8 章	类: Critter Caretaker.....	187	第 9 章	高级类与动态内存:	
8.1	定义新类型	187		Game Lobby.....	211
8.1.1	Simple Critter 程序简介	187	9.1	使用聚合体	211
8.1.2	类的定义	189	9.1.1	Critter Farm 程序简介.....	212
8.1.3	成员函数的定义	189	9.1.2	使用对象数据成员	213
8.1.4	对象的实例化	190	9.1.3	使用容器数据成员	214
8.1.5	数据成员的访问	190	9.2	使用友元函数与运算符 重载	215
8.1.6	成员函数的调用	190	9.2.1	Friend Critter 程序简介.....	215
8.2	使用构造函数	191	9.2.2	创建友元函数	217
8.2.1	Constructor Critter 程序简介	191	9.2.3	运算符重载	217
8.2.2	构造函数的声明与定义	192	9.3	动态分配内存	218
8.2.3	构造函数的自动调用	193	9.3.1	Heap 程序简介.....	218
8.3	设置成员访问级别	193	9.3.2	使用 new 运算符.....	220
8.3.1	Private Critter 程序简介	193	9.3.3	使用 delete 运算符.....	221
8.3.2	指定公有与私有访问 级别	195	9.3.4	避免内存泄漏	221
8.3.3	定义访问器成员函数	196	9.4	使用数据成员与堆	223
8.3.4	定义常量成员函数	197	9.4.1	Heap Data Member 程序简介	223
8.4	使用静态数据成员与 静态成员函数.....	197	9.4.2	声明指向堆中值的 指针数据成员	225
8.4.1	Static Critter 程序简介.....	197	9.4.3	声明与定义析构函数	226
8.4.2	声明与初始化静态 数据成员	199	9.4.4	声明与定义拷贝 构造函数	227

9.4.5 赋值运算符的重载	229	10.3.5 调用基类成员函数	251
9.5 Game Lobby 程序简介	231	10.4 在派生类中使用重载赋值 运算符与拷贝构造函数	252
9.5.1 Player 类	232	10.5 多态简介	253
9.5.2 Lobby 类	233	10.5.1 Polymorphic Bad Guy 程序简介	253
9.5.3 Lobby::AddPlayer() 成员函数	234	10.5.2 使用基类指针指向 派生类对象	255
9.5.4 Lobby::RemovePlayer() 成员函数	236	10.5.3 定义虚析构函数	256
9.5.5 Lobby::Clear()成员函数	237	10.6 使用抽象类	257
9.5.6 operator<<()成员函数	237	10.6.1 Abstract Creature 程序简介	257
9.5.7 main()函数	237	10.6.2 声明纯虚函数	258
9.6 本章小结	238	10.6.3 从抽象类派生类	259
9.7 问与答	239	10.7 Blackjack 游戏简介	259
9.8 问题讨论	240	10.7.1 类的设计	260
9.9 习题	240	10.7.2 规划游戏的逻辑	263
第 10 章 继承与多态: Blackjack	241	10.7.3 Card 类	264
10.1 继承简介	241	10.7.4 Hand 类	265
10.1.1 Simple Boss 程序 简介	242	10.7.5 GenericPlayer 类	267
10.1.2 从基类派生	244	10.7.6 Player 类	268
10.1.3 从派生类实例化 对象	245	10.7.7 House 类	269
10.1.4 使用继承成员	245	10.7.8 Deck 类	270
10.2 继承访问权的控制	246	10.7.9 Game 类	272
10.2.1 Simple Boss 2.0 程序简介	246	10.7.10 main()函数	275
10.2.2 对类成员使用访问 修饰符	247	10.8 本章小结	277
10.2.3 使用访问修饰符 修饰派生类	247	10.9 问与答	277
10.3 调用与重写基类成员 函数	248	10.10 问题讨论	278
10.3.1 Overriding Boss 程序简介	248	10.11 习题	279
10.3.2 调用基类构造函数	250	附录 A 创建第一个 C++ 程序	281
10.3.3 声明虚基类成员函数	250	附录 B 运算符优先级	285
10.3.4 重写虚基类成员函数	251	附录 C 关键字	287
		附录 D ASCII 字符表	289
		附录 E 转义序列	295

第 1 章



类型、变量与标准 I/O: Lost Fortune

游戏编程的要求很高。它要求程序员和硬件将其能力都发挥到极致。但即便是没有做到极致，游戏也能让玩家非常满意。本章将介绍编写一流游戏的标准语言——C++的基础知识。具体而言，本章内容如下：

- 在控制台窗口中显示输出
- 执行算术运算
- 使用变量对数据进行存储、操作和检索
- 获取用户输入
- 使用常量与枚举类型
- 使用字符串

1.1 C++简介

全世界数以百万计的程序员都在使用 C++。它是编写计算机应用程序的最流行的语言之一，而且是编写大预算计算机游戏的最流行的语言。

Bjarne Stroustrup 发明的 C++是 C 语言的直系后代。实际上，C++作为 C 语言的超集，几乎包含它的所有内容。不仅如此，C++还提供了更好的问题解决方式和一些全新的功能。

1.1.1 使用 C++编写游戏

游戏程序员选择 C++的原因各种各样，下面列出其中一些：

- **高速。**经过精良编写的 C++程序速度明显要快。C++的设计目标之一就是实现高性能。如果您想从程序中获取更多的性能提升，可以在 C++中使用汇编语言(一种最底层的、人类可读的编程语言)来与计算机硬件直接通信。
- **灵活。**C++是一种支持包括面向对象编程在内的不同编程方式的多范型语言。与其他一些现代语言不同，C++并不会强制程序员使用某一特定编程方式。
- **良好的支持。**源于C++在游戏行业的悠久历史，现在有大量资源库可供 C++游戏程序员使用。这些资源包括图形 API、2D、3D、物理以及声音引擎。为了尽可能加快游戏开发的进度，C++程序员可以使用所有这些已有的代码。

1.1.2 生成可执行文件

不管某个程序是游戏还是商业应用程序，启动该程序所运行的文件即为可执行文件。从 C++源代码(C++语言指令的集合)生成可执行文件包含以下几个步骤。生成过程如图 1-1 所示。

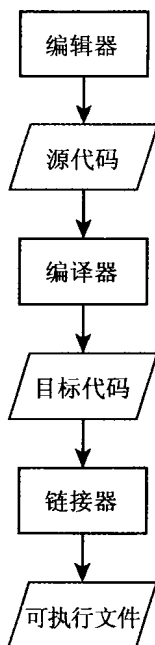


图 1-1 从 C++源代码生成可执行文件

(1) 首先，程序员使用编辑器编写 C++源代码，通常为以.cpp 为扩展名的文件。编辑器就好比程序员的字处理器，它能帮助程序员创建、编辑以及保存源代码。

(2) 程序员保存好源文件后，调用 C++编译器——一种读取源代码并将其翻译成目标文件的应用程序。目标文件的扩展名通常为.obj。

(3) 接下来，链接器将目标文件链接到任何必要的外部文件，然后生成可执行文件。其扩展名通常为.exe。至此，用户(或玩家)就可以通过启动可执行文件来运行该程序了。