

THOMSON



C++

游戏编程入门教程

BEGINNING

C++ GAME PROGRAMMING

〔美〕Michael Dawson 著
徐刚 薛雷 于健 译



人民邮电出版社
POSTS & TELECOM PRESS

C++游戏编程入门教程

[美] Michael Dawson 著

徐刚 薛雷 于健 译

人民邮电出版社

图书在版编目 (CIP) 数据

C++游戏编程入门教程 / (美) 道森 (Dawson, M.) 著; 徐刚, 薛雷, 于健译.

—北京: 人民邮电出版社, 2006.1

ISBN 7-115-13731-5

I. C... II. ①道...②徐...③薛...④于... III. C 语言—程序设计—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2005) 第 131647 号

版 权 声 明

Michael Dawson

Beginning C++ Game Programming

ISBN: 1-59200-205-6

Copyright © 2004 by Thomson Course Technology PTR.

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。

PTPress 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.

本书中文简体字翻译版由汤姆森学习出版集团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。

981-254-664-2

Thomson Learning (A division of Thomson Asia Pte Ltd)

5 Shenton Way #01-01 UIC Building Singapore 068808

C++游戏编程入门教程

-
- ◆ 著 [美] Michael Dawson
 - 译 徐刚 薛雷 于健
 - 责任编辑 李岚

 - ◆ 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>
 - 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: 20
 - 字数: 433 千字 2006 年 1 月第 1 版
 - 印数: 1—4 000 册 2006 年 1 月北京第 1 次印刷

著作权合同登记号 图字: 01-2005-1959 号

ISBN 7-115-13731-5/TP · 4851

定价: 39.00 元 (附光盘)

读者服务热线: (010) 67132705 印装质量热线: (010) 67129223

本书的结构

本书从 C++ 语言和游戏编程最基础的部分开始，假设读者不具有这两方面的任何经验。随着各章的进展，将在之前内容的基础上介绍更高级的主题。

在各章中，将介绍一个或几个相关的主题。本书循序渐进地介绍各个概念，编写非常小的与游戏有关的程序来说明概念。在每一章的最后，有一个单独的游戏，结合了本章最重要的概念。本书的最后一章以最庞大的项目结束，这个项目应用了在全书中介绍的所有主要概念。

除了学习 C++ 和游戏编程之外，读者还将学习如何组织自己的工作、将问题划分为易于管理的块以及改进代码等。读者随时会面临挑战，但是决不会过于困难。最重要的是，读者将在学习的过程中感受到乐趣。在此过程中，读者将创建一些非常好的计算机游戏并深入了解游戏编程的技术。

第 1 章 类型、变量和标准 I/O: Lost Fortune。本章将向读者介绍游戏业的标准语言 C++ 的基础知识。读者将学习如何在控制台窗口中显示输出、执行算术运算、使用变量以及从键盘获得玩家的输入。

第 2 章 事实、分支和游戏循环: Guess My Number。读者将编写根据某种条件执行、略过或者重复代码块的程序，从而创建更有趣的游戏。读者将学习如何生成随机数，以便向游戏添加一些不可预测性。读者还将学习游戏循环——一种组织游戏以使操作得以继续的基本方式。

第 3 章 for 循环、字符串和数组: Word Jumble。读者将学习序列和使用字符串——非常适合文字游戏的字符序列。读者还将学习软件对象——可以用来在游戏中表示物体（例如外星飞船、治疗药水或者甚至玩家自己）的实体。

第 4 章 标准模板库: Hangman。本章将向读者介绍一个功能强大的库——这是一个工具箱，游戏程序员（以及非游戏程序员）用它来管理对象（例如玩家装备库中的物品）的集合。读者还将学习一些有助于读者设计大型游戏程序的技术。

第 5 章 函数: Mad Lib。读者将学习将游戏程序划分为更小、更易于管理的代码块。读者将通过学习函数（游戏程序中的基本逻辑单位）来实现这一点。

第 6 章 引用: Tic-Tac-Toe。读者将学习如何有效而清晰地与程序的不同部分共享信息。读者还将看到 AI（Artificial Intelligence，人工智能）的一个简单例子，并学习如何使计算机对手具有点个性。

第 7 章 指针: Tic-Tac-Toe 2.0。读者将开始了解 C++ 的一些最低级的强大特性，例如如何直接寻址以及管理计算机的内存。

第 8 章 类: Critter Caretaker。读者将学习如何通过面向对象编程创建自己的对象类型并定义它们彼此交互的方式。在此过程中，读者将创建自己的需要照顾的宠物。

第 9 章 高级类和动态内存：Game Lobby。读者将扩展自己与计算机的直接联系，学习获得和释放游戏程序需要的内存。读者还将看到使用这种“动态”内存的危险并了解如何避免这些危险。

第 10 章 继承和多态：Blackjack。读者将学习如何在其他对象的基础上定义对象。然后，读者将已经学习过的所有技术结合到一个最终的大型游戏中。通过创建经典的卡西诺游戏 Blackjack（没有包括俗气的绿毡）的一个多玩家版本，读者将看到如何设计和实现一个相当大的项目。

本书使用的约定

在全书中使用了许多特殊的元素，它们的作用如下。

提示：它们是有助于使读者成为更好的游戏程序员的好想法。

陷阱：它们指出了很容易犯错误的地方。

技巧：它们指出了一些技巧和快捷方法，可以使游戏程序员的工作变得更轻松。

实际情况：它们是有关现实游戏编程的事实。

配套光盘的内容

本书附带的光盘包括了以下内容：

- 本书中介绍的程序和游戏的所有源代码；
- Bloodshed Dev-C++ IDE，它使用 GCC 的 MinGW 版本作为编译器；
- 初步介绍如何使用 Dev-C++ IDE 来创建第一个程序；
- SGI 的标准模板库编程指南；
- 有关 C++、游戏编程以及行业新闻和信息的有用链接。

关于编译器

在这里谈论编译器可能有一点超前了，但是这个问题非常重要，因为“编译器”将你编写的源代码转换为计算机能够运行的程序。我强烈推荐使用本书配套光盘上的 Dev-C++ IDE。

它包括了一个现代的 C++编译器，易于使用，最好的一点在于是免费的。我在光盘上提供了一个使用它的初步介绍（附录 A “创建第一个 C++程序”），解释了如何使用 Dev-C++编译第一个 C++程序。你还等什么呢？赶紧拿起光盘，安装 Dev-C++并看看这个初步介绍吧！

陷阱：我讨厌在任何特定的编译器或者软件公司上挑三拣四，但必须指出，我不推荐 Microsoft Visual C++ 6.0。它的编译器没能正确实现 C++的一些重要方面。本书中的一些程序无法在 Visual C++ 6.0 下编译。当本书中的程序存在这些问题时，我会尽力指出来。如果读者想要使用 Microsoft 的产品，那么我推荐他们最新的 Visual Studio.NET 产品，它相当忠实地实现了 C++标准。

目 录

第 1 章 类型、变量和标准 I/O: Lost Fortune	1
1.1 C++简介	1
1.1.1 将 C++用于游戏	1
1.1.2 创建可执行文件	2
1.1.3 处理错误	3
1.1.4 了解 ISO 标准	4
1.2 编写第一个 C++程序	4
1.2.1 Game Over 程序简介	4
1.2.2 为代码添加注释	5
1.2.3 使用空白	5
1.2.4 包括其他文件	6
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 结束语句	7
1.2.8 从 main()返回一个值	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	9
1.3.4 使用 using 声明	9
1.3.5 何时使用 using	9
1.4 使用算术操作符	10

1.4.1	Expensive Calculator 程序简介	10
1.4.2	加法、减法和乘法	11
1.4.3	整数除法和浮点数除法	11
1.4.4	使用模操作符	12
1.4.5	操作的顺序	12
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	14
1.5.3	类型修饰符	14
1.5.4	声明变量	15
1.5.5	命名变量	15
1.5.6	向变量赋值	16
1.5.7	初始化变量	17
1.5.8	显示变量值	17
1.5.9	获得用户输入	18
1.5.10	为类型定义新名称	18
1.5.11	使用哪一种类型	18
1.6	使用变量执行算术运算	19
1.6.1	Game Stats 2.0 程序简介	19
1.6.2	改变变量的值	20
1.6.3	使用组合赋值操作符	20
1.6.4	使用加 1 和减 1 操作符	21
1.6.5	处理整型数环绕	22
1.7	使用常量	22
1.7.1	Game Stats 3.0 程序简介	23
1.7.2	使用常量	24
1.7.3	使用枚举	24
1.8	Lost Fortune 简介	25
1.8.1	准备程序	26
1.8.2	从玩家那里获取信息	26
1.8.3	讲故事	27
1.9	小结	27
1.10	问题与答案	28
1.11	讨论题	29
1.12	练习题	30

第 2 章 事实、分支和游戏循环: Guess My Number	31
2.1 理解事实	31
2.2 使用 if 语句	32
2.2.1 Score Rater 程序简介	32
2.2.2 测试 true 和 false	34
2.2.3 将值解释为 true 或 false	34
2.2.4 使用关系操作符	35
2.2.5 嵌套 if 语句	35
2.3 使用 else 子句	36
2.3.1 Score Rater 2.0 程序简介	36
2.3.2 创建两个分支路径	37
2.3.3 使 else 子句与正确的 if 语句配对	38
2.4 使用 switch 语句	39
2.4.1 Menu Chooser 程序简介	39
2.4.2 创建多个分支路径	41
2.5 使用 while 循环	41
2.5.1 Play Again 程序简介	41
2.5.2 使用 while 循环实现循环	42
2.6 使用 do 循环	42
2.6.1 Play Again 2.0 程序简介	43
2.6.2 使用 do 循环实现循环	44
2.7 使用 break 和 continue 语句	44
2.7.1 Finicky Counter 程序简介	44
2.7.2 创建 while (true) 循环	45
2.7.3 使用 break 语句退出循环	46
2.7.4 使用 continue 语句跳回循环顶部	46
2.7.5 何时使用 break 和 continue	46
2.8 使用逻辑操作符	46
2.8.1 Designers Network 程序简介	47
2.8.2 使用逻辑与操作符	49
2.8.3 使用逻辑或操作符	50
2.8.4 使用逻辑非操作符	50
2.8.5 操作符的顺序	51
2.9 生成随机数	51
2.9.1 Die Roller 程序简介	52

2.9.2 调用 rand()函数	52
2.9.3 生成随机数生成器种子	53
2.9.4 计算某个范围内的数字	54
2.10 游戏循环	54
2.11 Guess My Number 简介	55
2.11.1 应用游戏循环	56
2.11.2 开始游戏	57
2.11.3 创建猜测循环	57
2.11.4 结束游戏	58
2.12 小结	58
2.13 问题与答案	59
2.14 讨论题	60
2.15 练习题	60
第3章 for 循环、字符串和数组: 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 嵌套 for 循环	64
3.2 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 串联 string 对象	68
3.3.4 使用 size()成员函数	69
3.3.5 索引 string 对象	69
3.3.6 遍历 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 索引数组	74
3.4.4 访问数组元素的成员函数	75

3.4.5 注意数组边界	75
3.5 C 风格的字符串	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 索引多维数组	79
3.7 Word Jumble 简介	79
3.7.1 开始程序	80
3.7.2 挑选要打乱的单词	80
3.7.3 打乱单词	81
3.7.4 欢迎玩家	81
3.7.5 进入游戏循环	82
3.7.6 说再见	82
3.8 小结	82
3.9 问题与答案	83
3.10 讨论题	84
3.11 练习题	85
第 4 章 标准模板库: Hangman	86
4.1 标准模板库简介	86
4.2 使用矢量	87
4.2.1 Hero's Inventory 2.0 程序简介	87
4.2.2 准备使用矢量	89
4.2.3 声明矢量	89
4.2.4 使用 push_back()成员函数	89
4.2.5 使用 size()成员函数	90
4.2.6 索引矢量	90
4.2.7 调用元素的成员函数	91
4.2.8 使用 pop_back()成员函数	91
4.2.9 使用 clear()成员函数	91
4.2.10 使用 empty()成员函数	91
4.3 使用迭代器	92
4.3.1 Hero's Inventory 3.0 程序简介	92
4.3.2 声明迭代器	93
4.3.3 遍历矢量	94
4.3.4 更改矢量元素的值	96

4.3.5	访问矢量元素的成员函数	96
4.3.6	使用 insert() 矢量成员函数	97
4.3.7	使用 erase() 矢量成员函数	97
4.4	使用算法	98
4.4.1	High Scores 程序简介	98
4.4.2	准备使用算法	100
4.4.3	使用 random_shuffle() 算法	100
4.4.4	使用 sort() 算法	100
4.4.5	使用 merge() 算法	101
4.5	了解矢量性能	102
4.5.1	了解矢量的扩大	102
4.5.2	了解元素插入和删除	103
4.6	了解其他 STL 容器	104
4.7	规划程序	104
4.7.1	使用伪代码	105
4.7.2	使用逐步求精方法	105
4.8	Hangman 简介	106
4.8.1	设计游戏	106
4.8.2	开始程序	107
4.8.3	初始化变量和常量	107
4.8.4	进入主循环	108
4.8.5	获得玩家的猜测	108
4.8.6	结束游戏	109
4.9	小结	109
4.10	问题与答案	110
4.11	讨论题	111
4.12	练习题	111
第 5 章	函数: Mad Lib	112
5.1	创建函数	112
5.1.1	Instructions 程序简介	112
5.1.2	声明函数	113
5.1.3	定义函数	114
5.1.4	调用函数	114
5.1.5	抽象	115
5.2	使用参数和返回值	115

5.2.1	Yes or No 程序简介	115
5.2.2	返回一个值	117
5.2.3	使用参数接受值	117
5.2.4	封装	119
5.3	软件重用	119
5.4	了解范围	120
5.4.1	Scoping 程序简介	120
5.4.2	处理独立的范围	121
5.4.3	使用嵌套的范围	122
5.5	使用全局变量	123
5.5.1	Global Reach 程序简介	123
5.5.2	声明全局变量	125
5.5.3	访问全局变量	125
5.5.4	隐藏全局变量	125
5.5.5	改变全局变量	126
5.5.6	尽可能少使用全局变量	126
5.6	使用全局常量	126
5.7	使用默认参数	127
5.7.1	Give Me a Number 程序简介	127
5.7.2	指定默认参数	128
5.7.3	将默认参数赋给参数	129
5.7.4	覆盖默认参数	129
5.8	重载函数	130
5.8.1	Triple 程序简介	130
5.8.2	创建重载的函数	131
5.8.3	调用重载的函数	132
5.9	内联函数	132
5.9.1	Taking Damage 程序简介	132
5.9.2	指定函数将要内联	133
5.9.3	调用内联的函数	134
5.10	Mad Lib 游戏简介	134
5.10.1	开始程序	134
5.10.2	main()函数	135
5.10.3	askText()函数	136
5.10.4	askNumber()函数	136
5.10.5	tellStory()函数	136

5.11 小结	137
5.12 问题与答案	138
5.13 讨论题	139
5.14 练习题	139
第6章 引用: Tic-Tac-Toe	140
6.1 使用引用	140
6.1.1 Referencing 程序简介	140
6.1.2 创建引用	141
6.1.3 访问引用的值	142
6.1.4 更改引用的值	142
6.2 传递引用以更改参数	143
6.2.1 Swap 程序简介	143
6.2.2 按值传递	145
6.2.3 按引用传递	145
6.3 传递引用以提高效率	146
6.3.1 Inventory Displayer 程序简介	146
6.3.2 了解引用传递的缺点	147
6.3.3 将参数声明为常量引用	147
6.3.4 传递常量引用	148
6.4 决定如何传递参数	148
6.5 返回引用	149
6.5.1 Inventory Reference 程序简介	149
6.5.2 返回一个引用	150
6.5.3 显示返回的引用的值	151
6.5.4 将返回的引用赋给另一个引用	151
6.5.5 将返回的引用赋给变量	151
6.5.6 通过返回的引用更改对象	152
6.6 Tic-Tac-Toe 游戏简介	152
6.6.1 规划游戏	153
6.6.2 开始程序	155
6.6.3 main()函数	156
6.6.4 Instruction()函数	156
6.6.5 askYesNo()函数	157
6.6.6 askNumber()函数	157
6.6.7 humanPiece()函数	158

6.6.8	opponent()函数	158
6.6.9	displayBoard()函数	158
6.6.10	winner()函数	159
6.6.11	isLegal()函数	160
6.6.12	humanMove()函数	160
6.6.13	computerMove()函数	161
6.6.14	announceWinner()函数	163
6.7	小结	163
6.8	问题与答案	164
6.9	讨论题	165
6.10	练习题	166
第7章	指针: Tic-Tac-Toe 2.0	167
7.1	指针基础知识	167
7.1.1	Pointing 程序简介	168
7.1.2	声明指针	169
7.1.3	初始化指针	170
7.1.4	将地址赋给指针	170
7.1.5	解除指针的引用	171
7.1.6	重新为指针赋值	172
7.1.7	使用对象指针	172
7.2	指针和常量	173
7.2.1	使用常量指针	173
7.2.2	使用指向常量的指针	174
7.2.3	使用指向常量的常量指针	175
7.2.4	总结常量和指针	175
7.3	传递指针	176
7.3.1	Swap Pointer Version 程序简介	176
7.3.2	按值传递	177
7.3.3	传递常量指针	178
7.4	返回指针	179
7.4.1	Inventory Pointer 程序简介	179
7.4.2	返回指针	180
7.4.3	使用返回的指针显示值	181
7.4.4	将返回的指针赋给另一个指针	181
7.4.5	将返回的指针指向的值赋给变量	182

7.4.6 通过返回的指针改变对象	183
7.5 理解指针与数组之间的关系	183
7.5.1 Array Passer 程序简介	183
7.5.2 使用数组名称作为常量指针	185
7.5.3 传递和返回数组	185
7.6 Tic-Tac-Toe 2.0 游戏简介	186
7.7 小结	187
7.8 问题与答案	188
7.9 讨论题	189
7.10 练习题	189
第 8 章 类: Critter Caretaker	190
8.1 定义新类型	190
8.1.1 Simple Critter 程序简介	190
8.1.2 定义类	192
8.1.3 定义成员函数	193
8.1.4 实例化对象	193
8.1.5 访问数据成员	193
8.1.6 调用成员函数	194
8.2 使用构造函数	194
8.2.1 Constructor Critter 程序简介	194
8.2.2 声明和定义构造函数	196
8.2.3 自动调用构造函数	197
8.3 设置成员访问级别	197
8.3.1 Private Critter 程序简介	197
8.3.2 指定公共和私有访问级别	199
8.3.3 定义访问器成员函数	199
8.3.4 定义常量成员函数	200
8.4 使用静态数据成员和成员函数	201
8.4.1 Static Critter 程序简介	201
8.4.2 声明和初始化静态数据成员	202
8.4.3 访问静态数据成员	203
8.4.4 定义静态成员函数	203
8.4.5 调用静态成员函数	204
8.5 Critter Caretaker 游戏简介	204
8.5.1 设计游戏	204