

图书基本信息

书名：<<C++ Primer 中文版 (第5版)>>

13位ISBN编号：9787121155352

10位ISBN编号：7121155354

出版时间：2013-9

出版时间：电子工业出版社

作者：[美] Stanley B. Lippman,[美] Jos é e Lajoie,[美] Barbara E. Moo

译者：王刚,杨巨峰

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

内容概要

这本久负盛名的 C++ 经典教程，时隔八年之久，终迎来史无前例的重大升级。

除令全球无数程序员从中受益，甚至为之迷醉的——C++ 大师 Stanley B. Lippman 的丰富实践经验，C++ 标准委员会原负责人 Jos é e Lajoie 对 C++ 标准的深入理解，以及 C++ 先驱 Barbara E. Moo 在 C++ 教学方面的真知灼见外，更是基于全新的 C++11 标准进行了全面而彻底的内容更新。

非常难能可贵的是，《C++ Primer 中文版(第5版)》所有示例均全部采用 C++11 标准改写，这在经典升级版中极其罕见——充分体现了 C++ 语言的重大进展及其全面实践。

书中丰富的教学辅助内容、醒目的知识点提示，以及精心组织的编程示范，让这本书在 C++ 领域的权威地位更加不可动摇。

无论是初学者入门，或是中、高级程序员提升，本书均为不容置疑的首选。

作者简介

Stanley B. Lippman目前是微软公司 Visual C++ 团队的架构师。他从1984年开始在贝尔实验室与C++的设计者Bjarne Stroustrup一起从事C++的设计与开发。他在迪士尼和梦工厂从事动画制作，还担任过JPL的高级顾问。

Jos é e Lajoie曾经是IBM加拿大研究中心C/C++编译器开发团队的成员，在ISO C++标准委员会工作了7年，担任过ISO核心语言工作组的主席和C++ Report杂志的专栏作家。

Barbara E. Moo是拥有25年软件经验的独立咨询顾问。在AT&T，她与Stroustrup、Lippman一起管理过复杂的C++开发项目。

书籍目录

第1章 开始	1
1.1 编写一个简单的C++程序	2
1.1.1 编译、运行程序	3
1.2 初识输入输出	5
1.3 注释简介	8
1.4 控制流	10
1.4.1 while语句	10
1.4.2 for语句	11
1.4.3 读取数量不定的输入数据	13
1.4.4 if语句	15
1.5 类简介	17
1.5.1 Sales_item类	17
1.5.2 初识成员函数	20
1.6 书店程序	21
小结	23
术语表	23
第 部分 C++基础	27
第2章 变量和基本类型	29
2.1 基本内置类型	30
2.1.1 算术类型	30
2.1.2 类型转换	32
2.1.3 字面值常量	35
2.2 变量	38
2.2.1 变量定义	38
2.2.2 变量声明和定义的关系	41
2.2.3 标识符	42
2.2.4 名字的作用域	43
2.3 复合类型	45
2.3.1 引用	45
2.3.2 指针	47
2.3.3 理解复合类型的声明	51
2.4 const限定符	53
2.4.1 const的引用	54
2.4.2 指针和const	56
2.4.3 顶层const	57
2.4.4 constexpr和常量表达式	58
2.5 处理类型	60
2.5.1 类型别名	60
2.5.2 auto类型说明符	61
2.5.3 decltype类型指示符	62
2.6 自定义数据结构	64
2.6.1 定义Sales_data类型	64
2.6.2 使用Sales_data类	66
2.6.3 编写自己的头文件	67
小结	69

术语表	69
第3章 字符串、向量和数组	73
3.1 命名空间的using声明	74
3.2 标准库类型string	75
3.2.1 定义和初始化string对象	76
3.2.2 string对象上的操作	77
3.2.3 处理string对象中的字符	81
3.3 标准库类型vector	86
3.3.1 定义和初始化vector对象	87
3.3.2 向vector对象中添加元素	90
3.3.3 其他vector操作	91
3.4 迭代器介绍	95
3.4.1 使用迭代器	95
3.4.2 迭代器运算	99
3.5 数组	101
3.5.1 定义和初始化内置数组	101
3.5.2 访问数组元素	103
3.5.3 指针和数组	105
3.5.4 C风格字符串	109
3.5.5 与旧代码的接口	111
3.6 多维数组	112
小结	117
术语表	117
第4章 表达式	119
4.1 基础	120
4.1.1 基本概念	120
4.1.2 优先级与结合律	121
4.1.3 求值顺序	123
4.2 算术运算符	124
4.3 逻辑和关系运算符	126
4.4 赋值运算符	129
4.5 递增和递减运算符	131
4.6 成员访问运算符	133
4.7 条件运算符	134
4.8 位运算符	135
4.9 sizeof运算符	139
4.10 逗号运算符	140
4.11 类型转换	141
4.11.1 算术转换	142
4.11.2 其他隐式类型转换	143
4.11.3 显式转换	144
4.12 运算符优先级表	147
小结	149
术语表	149
第5章 语句	153
5.1 简单语句	154
5.2 语句作用域	155

5.3 条件语句	156
5.3.1 if语句	156
5.3.2 switch语句	159
5.4 迭代语句	165
5.4.1 while语句	165
5.4.2 传统的for语句	166
5.4.3 范围for语句	168
5.4.4 do while语句	169
5.5 跳转语句	170
5.5.1 break语句	170
5.5.2 continue语句	171
5.5.3 goto语句	172
5.6 TRY语句块和异常处理	172
5.6.1 throw表达式	173
5.6.2 try语句块	174
5.6.3 标准异常	176
小结	178
术语表	178
第6章 函数	181
6.1 函数基础	182
6.1.1 局部对象	184
6.1.2 函数声明	186
6.1.3 分离式编译	186
6.2 参数传递	187
6.2.1 传值参数	187
6.2.2 传引用参数	188
6.2.3 const形参和实参	190
6.2.4 数组形参	193
6.2.5 main：处理命令行选项	196
6.2.6 含有可变形参的函数	197
6.3 返回类型和return语句	199
6.3.1 无返回值函数	200
6.3.2 有返回值函数	200
6.3.3 返回数组指针	205
6.4 函数重载	206
6.4.1 重载与作用域	210
6.5 特殊用途语言特性	211
6.5.1 默认实参	211
6.5.2 内联函数和constexpr函数	213
6.5.3 调试帮助	215
6.6 函数匹配	217
6.6.1 实参类型转换	219
6.7 函数指针	221
小结	225
术语表	225
第7章 类	227
7.1 定义抽象数据类型	228

- 7.1.1 设计Sales_data类 228
- 7.1.2 定义改进的Sales_data类 230
- 7.1.3 定义类相关的非成员函数 234
- 7.1.4 构造函数 235
- 7.1.5 拷贝、赋值和析构 239
- 7.2 访问控制与封装 240
 - 7.2.1 友元 241
- 7.3 类的其他特性 243
 - 7.3.1 类成员再探 243
 - 7.3.2 返回*this的成员函数 246
 - 7.3.3 类类型 249
 - 7.3.4 友元再探 250
- 7.4 类的作用域 253
 - 7.4.1 名字查找与类的作用域 254
- 7.5 构造函数再探 257
 - 7.5.1 构造函数初始值列表 258
 - 7.5.2 委托构造函数 261
 - 7.5.3 默认构造函数的作用 262
 - 7.5.4 隐式的类类型转换 263
 - 7.5.5 聚合类 266
 - 7.5.6 字面值常量类 267
- 7.6 类的静态成员 268
- 小结 273
- 术语表 273
- 第 部 C++标准库 275
- 第8章 IO库 277
 - 8.1 IO类 278
 - 8.1.1 IO对象无拷贝或赋值 279
 - 8.1.2 条件状态 279
 - 8.1.3 管理输出缓冲 281
 - 8.2 文件输入输出 283
 - 8.2.1 使用文件流对象 284
 - 8.2.2 文件模式 286
 - 8.3 string流 287
 - 8.3.1 使用istringstream 287
 - 8.3.2 使用ostringstream 289
- 小结 290
- 术语表 290
- 第9章 顺序容器 291
 - 9.1 顺序容器概述 292
 - 9.2 容器库概览 294
 - 9.2.1 迭代器 296
 - 9.2.2 容器类型成员 297
 - 9.2.3 begin和end成员 298
 - 9.2.4 容器定义和初始化 299
 - 9.2.5 赋值和swap 302
 - 9.2.6 容器大小操作 304

- 9.2.7 关系运算符 304
- 9.3 顺序容器操作 305
 - 9.3.1 向顺序容器添加元素 305
 - 9.3.2 访问元素 309
 - 9.3.3 删除元素 311
 - 9.3.4 特殊的forward_list操作 312
 - 9.3.5 改变容器大小 314
 - 9.3.6 容器操作可能使迭代器失效 315
- 9.4 vector对象是如何增长的 317
- 9.5 额外的string操作 320
 - 9.5.1 构造string的其他方法 321
 - 9.5.2 改变string的其他方法 322
 - 9.5.3 string搜索操作 325
 - 9.5.4 compare函数 327
 - 9.5.5 数值转换 327
- 9.6 容器适配器 329
- 小结 332
- 术语表 332
- 第10章 泛型算法 335
 - 10.1 概述 336
 - 10.2 初识泛型算法 338
 - 10.2.1 只读算法 338
 - 10.2.2 写容器元素的算法 339
 - 10.2.3 重排容器元素的算法 342
 - 10.3 定制操作 344
 - 10.3.1 向算法传递函数 344
 - 10.3.2 lambda表达式 345
 - 10.3.3 lambda捕获和返回 349
 - 10.3.4 参数绑定 354
 - 10.4 再探迭代器 357
 - 10.4.1 插入迭代器 358
 - 10.4.2 istream迭代器 359
 - 10.4.3 反向迭代器 363
 - 10.5 泛型算法结构 365
 - 10.5.1 5类迭代器 365
 - 10.5.2 算法形参模式 367
 - 10.5.3 算法命名规范 368
 - 10.6 特定容器算法 369
 - 小结 371
 - 术语表 371
- 第11章 关联容器 373
 - 11.1 使用关联容器 374
 - 11.2 关联容器概述 376
 - 11.2.1 定义关联容器 376
 - 11.2.2 关键字类型的要求 378
 - 11.2.3 pair类型 379
 - 11.3 关联容器操作 381

11.3.1 关联容器迭代器	382
11.3.2 添加元素	383
11.3.3 删除元素	386
11.3.4 map的下标操作	387
11.3.5 访问元素	388
11.3.6 一个单词转换的map	391
11.4 无序容器	394
小结	397
术语表	397
第12章 动态内存	399
12.1 动态内存与智能指针	400
12.1.1 shared_ptr类	400
12.1.2 直接管理内存	407
12.1.3 shared_ptr和new结合使用	412
12.1.4 智能指针和异常	415
12.1.5 unique_ptr	417
12.1.6 weak_ptr	420
12.2 动态数组	423
12.2.1 new和数组	423
12.2.2 allocator类	427
12.3 使用标准库：文本查询程序	430
12.3.1 文本查询程序设计	430
12.3.2 文本查询程序类的定义	432
小结	436
术语表	436
第 部分 类设计者的工具	437
第13章 拷贝控制	439
13.1 拷贝、赋值与销毁	440
13.1.1 拷贝构造函数	440
13.1.2 拷贝赋值运算符	443
13.1.3 析构函数	444
13.1.4 三/五法则	447
13.1.5 使用=default	449
13.1.6 阻止拷贝	449
13.2 拷贝控制和资源管理	452
13.2.1 行为像值的类	453
13.2.2 定义行为像指针的类	455
13.3 交换操作	457
13.4 拷贝控制示例	460
13.5 动态内存管理类	464
13.6 对象移动	470
13.6.1 右值引用	471
13.6.2 移动构造函数和移动赋值运算符	473
13.6.3 右值引用和成员函数	481
小结	486
术语表	486
第14章 操作重载与类型转换	489

- 14.1 基本概念 490
- 14.2 输入和输出运算符 494
 - 14.2.1 重载输出运算符> 495
- 14.3 算术和关系运算符 497
 - 14.3.1 相等运算符 497
 - 14.3.2 关系运算符 498
- 14.4 赋值运算符 499
- 14.5 下标运算符 501
- 14.6 递增和递减运算符 502
- 14.7 成员访问运算符 504
- 14.8 函数调用运算符 506
 - 14.8.1 lambda是函数对象 507
 - 14.8.2 标准库定义的函数对象 509
 - 14.8.3 可调用对象与function 511
- 14.9 重载、类型转换与运算符 514
 - 14.9.1 类型转换运算符 514
 - 14.9.2 避免有二义性的类型转换 517
 - 14.9.3 函数匹配与重载运算符 521
- 小结 523
- 术语表 523
- 第15章 面向对象程序设计 525
 - 15.1 OOP : 概述 526
 - 15.2 定义基类和派生类 527
 - 15.2.1 定义基类 528
 - 15.2.2 定义派生类 529
 - 15.2.3 类型转换与继承 534
 - 15.3 虚函数 536
 - 15.4 抽象基类 540
 - 15.5 访问控制与继承 542
 - 15.6 继承中的类作用域 547
 - 15.7 构造函数与拷贝控制 551
 - 15.7.1 虚析构函数 552
 - 15.7.2 合成拷贝控制与继承 552
 - 15.7.3 派生类的拷贝控制成员 554
 - 15.7.4 继承的构造函数 557
 - 15.8 容器与继承 558
 - 15.8.1 编写Basket类 559
 - 15.9 文本查询程序再探 562
 - 15.9.1 面向对象的解决方案 563
 - 15.9.2 Query_base类和Query类 567
 - 15.9.3 派生类 568
 - 15.9.4 eval函数 571
 - 小结 575
 - 术语表 575
- 第16章 模板与泛型编程 577
 - 16.1 定义模板 578
 - 16.1.1 函数模板 578

- 16.1.2 类模板 583
- 16.1.3 模板参数 592
- 16.1.4 成员模板 595
- 16.1.5 控制实例化 597
- 16.1.6 效率与灵活性 599
- 16.2 模板实参推断 600
 - 16.2.1 类型转换与模板类型参数 601
 - 16.2.2 函数模板显式实参 603
 - 16.2.3 尾置返回类型与类型转换 604
 - 16.2.4 函数指针和实参推断 607
 - 16.2.5 模板实参推断和引用 608
 - 16.2.6 理解std::move 610
 - 16.2.7 转发 612
- 16.3 重载与模板 614
- 16.4 可变参数模板 618
 - 16.4.1 编写可变参数函数模板 620
 - 16.4.2 包扩展 621
 - 16.4.3 转发参数包 622
- 16.5 模板特例化 624
- 小结 630
- 术语表 630
- 第 部分 高级主题 633
- 第17章 标准库特殊设施 635
 - 17.1 tuple类型 636
 - 17.1.1 定义和初始化tuple 637
 - 17.1.2 使用tuple返回多个值 638
 - 17.2 BITSET类型 640
 - 17.2.1 定义和初始化bitset 641
 - 17.2.2 bitset操作 643
 - 17.3 正则表达式 645
 - 17.3.1 使用正则表达式库 646
 - 17.3.2 匹配与Regex迭代器类型 650
 - 17.3.3 使用子表达式 653
 - 17.3.4 使用regex_replace 657
 - 17.4 随机数 659
 - 17.4.2 其他随机数分布 663
 - bernoulli_distribution类 665
 - 17.5 IO库再探 666
 - 17.5.1 格式化输入与输出 666
 - 17.5.2 未格式化的输入/输出操作 673
 - 17.5.3 流随机访问 676
- 小结 680
- 术语表 680
- 第18章 用于大型程序的工具 683
 - 18.1 异常处理 684
 - 18.1.1 抛出异常 684
 - 18.1.2 捕获异常 687

- 18.1.3 函数try语句块与构造函数 689
- 18.1.4 noexcept异常说明 690
- 18.1.5 异常类层次 693
- 18.2 命名空间 695
 - 18.2.1 命名空间定义 695
 - 18.2.2 使用命名空间成员 701
 - 18.2.3 类、命名空间与作用域 705
 - 18.2.4 重载与命名空间 708
- 18.3 多重继承与虚继承 710
 - 18.3.1 多重继承 711
 - 18.3.2 类型转换与多个基类 713
 - 18.3.3 多重继承下的类作用域 715
 - 18.3.4 虚继承 717
 - 18.3.5 构造函数与虚继承 720
- 小结 722
- 术语表 722
- 第19章 特殊工具与技术 725
 - 19.1 控制内存分配 726
 - 19.1.1 重载new和delete 726
 - 19.1.2 定位new表达式 729
 - 19.2 运行时类型识别 730
 - 19.2.1 dynamic_cast运算符 730
 - 19.2.2 typeid运算符 732
 - 19.2.3 使用RTTI 733
 - 19.2.4 type_info类 735
 - 19.3 枚举类型 736
 - 19.4 类成员指针 739
 - 19.4.1 数据成员指针 740
 - 19.4.2 成员函数指针 741
 - 19.4.3 将成员函数用作可调用对象 744
 - 19.5 嵌套类 746
 - 19.6 union：一种节省空间的类 749
 - 19.7 局部类 754
 - 19.8 固有的不可移植的特性 755
 - 19.8.1 位域 756
 - 19.8.2 volatile限定符 757
 - 19.8.3 链接指示：extern "C" 758
 - 小结 762
 - 术语表 762
- 附录A 标准库 765
 - A.1 标准库名字和头文件 766
 - A.2 算法概览 770
 - A.2.1 查找对象的算法 771
 - A.2.2 其他只读算法 772
 - A.2.3 二分搜索算法 772
 - A.2.4 写容器元素的算法 773
 - A.2.5 划分与排序算法 775

- A.2.6 通用重排操作 776
- A.2.7 排列算法 778
- A.2.8 有序序列的集合算法 778
- A.2.9 最小值和最大值 779
- A.2.10 数值算法 780
- A.3 随机数 781
 - A.3.1 随机数分布 781
 - A.3.2 随机数引擎 783
- C++11的新特性
 - 2.1.1 long long类型 31
 - 2.2.1 列表初始化 39
 - 2.3.2 nullptr常量 48
 - 2.4.4 constexpr变量 59
 - 2.5.1 类型别名声明 60
 - 2.5.2 auto类型指示符 61
 - 2.5.3 decltype类型指示符 62
 - 2.6.1 类内初始化 65
 - 3.2.2 使用auto或decltype缩写类型 79
 - 3.2.3 范围for语句 82
 - 3.3 定义vector对象的vector (向量的向量) 87
 - 3.3.1 vector对象的列表初始化 88
 - 3.4.1 容器的cbegin和cend函数 98
 - 3.5.3 标准库begin和end函数 106
 - 3.6 使用auto和decltype简化声明 115
- 4.2 除法的舍入规则 125
- 4.4 用大括号包围的值列表赋值 129
- 4.9 将sizeof用于类成员 139
 - 5.4.3 范围for语句 168
- 6.2.6 标准库initializer_list类 197
- 6.3.2 列表初始化返回值 203
- 6.3.3 定义尾置返回类型 206
- 6.3.3 使用decltype简化返回类型定义
- 6.5.2 constexpr函数 214
- 7.1.4 使用=default生成默认构造函数 237
- 7.3.1 类对象成员的类内初始化 246
- 7.5.2 委托构造函数 261
- 7.5.6 constexpr构造函数 268
- 8.2.1 用string对象处理文件名 284
- 9.1 array和forward_list容器 293
 - 9.2.3 容器的cbegin和cend函数 298
 - 9.2.4 容器的列表初始化 300
 - 9.2.5 容器的非成员函数swap 303
 - 9.3.1 容器insert成员的返回类型 308
 - 9.3.1 容器的emplace成员的返回类型 308
- 9.4 shrink_to_fit 318
- 9.5.5 string的数值转换函数 327
- 10.3.2 Lambda表达式 346

- 10.3.3 Lambda表达式中的尾置返回类型 353
- 10.3.4 标准库bind函数 354
- 11.2.1 关联容器的列表初始化 377
- 11.2.3 列表初始化pair的返回类型 380
- 11.3.2 pair的列表初始化 384
- 11.4 无序容器 394
- 12.1 智能指针 400
 - 12.1.1 shared_ptr类
 - 12.1.2 动态分配对象的列表初始化 407
 - 12.1.2 auto和动态分配 408
 - 12.1.5 unique_ptr类 417
 - 12.1.6 weak_ptr类 420
 - 12.2.1 范围for语句不能应用于动态分配数组 424
 - 12.2.1 动态分配数组的列表初始化 424
 - 12.2.1 auto不能用于分配数组 424
 - 12.2.2 allocator::construct可使用任意构造函数 428
- 13.1.5 将=default用于拷贝控制成员 449
- 13.1.6 使用=default阻止拷贝类对象 449
- 13.5 用移动类对象代替拷贝类对象 469
 - 13.6.1 右值引用 471
 - 13.6.1 标准库move函数 472
 - 13.6.2 移动构造函数和移动赋值 473
 - 13.6.2 移动构造函数通常应该是noexcept 473
 - 13.6.2 移动迭代器 480
 - 13.6.3 引用限定成员函数 483
- 14.8.3 function类模板 512
- 14.9.1 explicit类型转换运算符 516
- 15.2.2 虚函数的override指示符 530
- 15.2.2 通过定义类为final来阻止继承 533
- 15.3 虚函数的override和final指示符 538
- 15.7.2 删除的拷贝控制和继承 553
- 15.7.4 继承的构造函数 557
- 16.1.2 声明模板类型形参为友元 590
- 16.1.2 模板类型别名 590
- 16.1.3 模板函数的默认模板参数 594
- 16.1.5 实例化的显式控制 597
- 16.2.3 模板函数与尾置返回类型 605
- 16.2.5 引用折叠规则 609
- 16.2.6 用static_cast将左值转换为右值 612
- 16.2.7 标准库forward函数 614
- 16.4 可变参数模板 618
 - 16.4 sizeof...运算符 619
 - 16.4.3 可变参数模板与转发 622
- 17.1 标准库Tuple类模板 636
- 17.2.2 新的bitset运算 643
- 17.3 正则表达式库 645
- 17.4 随机数库 659

17.5.1	浮点数格式控制	670
18.1.4	noexcept异常指示符	690
18.1.4	noexcept运算符	691
18.2.1	内联名字空间	699
18.3.1	继承的构造函数和多重继承	712
19.3	有作用域的enum	736
19.3	说明类型用于保存enum对象	738
19.3	enum的提前声明	738
19.4.3	标准库mem_fn类模板	746
19.6	类类型的联合成员	751

版权说明

本站所提供下载的PDF图书仅提供预览和简介, 请支持正版图书。

更多资源请访问:<http://www.tushu007.com>