

SAMS
PUBLISHING

汤姆·斯旺C++编程秘诀

Tom Swan'Code Secrets

【美】TOM SWAN 著

宋建云 王理 陈晓明 陈亚琼 郑莉萍 译



最新
畅销书！

電子工業出版社

77312
927

汤姆·斯旺 C++ 编程秘诀

[美] Tom Swan 著

宋建云 王 理 陈晓明 译
陈亚琼 郑莉萍

电子工业出版社

(京)新登字 055 号

JS168/27
内 容 提 要

汤姆·斯旺C++编程秘诀是为培养最优秀的C++程序员而编写的,所讲的编程技巧是作者丰富的经验和体会的结晶。主要内容有I/O流和可流类的奥秘;函数模板和类模板的探讨;对象构造和对象析构方面的技术;读写多态的面向对象数据库文件中的持久对象的方法等。本书还提供了较完善的基本类模板、抽象类模板、容器类模板和迭代器类模板的全特性类库。附带的磁盘即有完整的源程序代码清单,也有C++类和成员函数库、屏幕设计和编辑库及串行通信库。

本书英文版由美国Sams Publishing出版,中文版于1994年8月由Macmillan Publishing授权电子工业出版社在中国独家出版。未经出版者书面允许,不得以任何方式或手段复制或抄袭本书内容。

Copyright © 1994 by Sams Publishing.

Tom Swan's Code Secrets

Tom Swan

* * *

汤姆·斯旺C++编程秘诀

[美]汤姆·斯旺 著

宋建云 王 理 陈晓明 译

陈亚琼 郑莉萍

特约编辑 张成全

电子工业出版社出版

电子工业出版社发行 各地新华书店经销

北京顺义天竺颖华印刷厂印刷

* * *

开本:787×1092毫米 1/16 印张:28.25 字数:560千字

1994年10月第一版 1994年10月北京第一次印刷

印数:0,001- 6000册 定价:52.00元

ISBN 7-5053-2793-3/TP·898

前 言

假设有两名学计算机科学的学生,他们年龄相仿,能力相当。现在教他们学习 C++ 语言,给他们同样的教科书并配备完全一样的当今最先进的计算机。结果其中一名学生能写出富有创新的软件应用程序,并由此赢得声誉和财富,而另一名学生却连单个的工作程序都完成不了。

为什么? 奥秘究竟在哪里? 为什么一些程序员能更轻松自如地编写计算机程序而另外一些程序员却不能呢? 难道他们更有天赋,更聪明或者只不过完全靠运气吗? 当前面所述的第一位学生用版税钱购买了新宅并在其游泳池边休生养息的时候,那第二位学生或许还在迷惑不解:“我到底什么地方做得不好? 怎么样才能获知一些奥秘来帮助我编写出成功的代码呢?”

假设我现在是在杂耍表演中兜售蛇油,或许我会两眼盯住你并向你建议:去阅读《编程秘诀》一书(译者注:即指本书),由此你也将会在计算机软件事务中获得巨大的报偿。不过你是不会上我的当的,事实是:我不知道什么是成功的奥秘,也并不存在这样的奥秘。

然而,我从事了 15 年程序设计及撰写有关程序设计方面书籍的工作,我注意到了成功的程序员的共同特征,尽管他们具备不同方面的能力,但是毫无疑问这些最优秀的程序员都具备这样一种性格:即对教科书及课堂外的技术有着全身心的永不满足的求知欲望。

我试图在这本书中精确地注入这样一类信息——即大多数程序员成年累月才能获得的(有关程序设计方面的)告诫和技术。然而,这本书不是百科全书——只有兜售蛇油的庸医骗子才许诺包治百病。但是一旦我完成了这本书的工作,那么其中的高级 C++ 技术将会激发你针对今后的任务去寻找你自己的解决问题的答案,如果真有一个秘诀能使你在软件开发中获得成功,那么只能是你的主观努力。

Tom Swan (汤姆·斯旺)
Compu Serve ID73627,3241

出版者的话

为了促进我国计算机技术的发展,我社与美国的MACMILLAN出版公司达成协议,近期即将翻译出版一批美国新出版的计算机软件图书。其中在1994年年底及1995年年初即将出版发行的有:

1. Tom Swan's Code Secret
2. Programmer's Problem Solver (Second Edition)
3. C++ By Example
4. Visual Basic for Applications By Example
5. The Data Recovery Bible
6. Tricks of The Windows 3.1 Masters
7. Teach Yourself visual C++ 1.5 in 21 Days (Revised Editions)
8. DOS Secrets Unleashed
9. Tricks of the Graphic Gurus
10. Visual C++ Object Oriented Programming
11. Using Novell DOS 7
12. Visual C++ Developer's Guide

作为领导世界计算机图书潮流的MACMILLAN公司所出版的以上图书内容新颖、通俗易懂、图文并茂、讲解生动。这些书基本上都是美国一些计算机编程大师们的经验结晶。我社翻译出版这些图书,希望对我国从事计算机应用、研究、开发的有关人员业务提高有所帮助。热忱欢迎广大读者朋友踊跃选购。

我们还应提出的是:美国CAPAO (Chinese American Publishing Company) 的Michael Howard先生为了开展中美文化交流,在电子工业出版社与Macmillan公司签订出书合同方面起到了牵线搭桥作用,在此深表感谢。

引 言

这里用短短的一章向您介绍《编程秘诀》一书,它将概述本书的各个部分,列出有关要求并提出一些充分利用各章节内容的建议。你没有必要逐字逐句地阅读本引言,但你应该浏览一下以便熟悉本书的框架。

0.1 快! ——

我已经急不可耐了!

转到“安装软盘”一节,然后直接跳到第一章“转向流机制”,快去!

0.2 干吗着急?

我想要一个完整的描述!

《编程秘诀》涉及种种编程方面的问题,这些都会在介绍高级 C++ 技术的上下文中出现。阅读下面内容以了解使用该书的要求及建议。

0.2.1 要求

你需要有一套符合 ANSI C++ 标准草案的编译器以及一台能运行该编译器的计算机。如果能满足这两个要求,那么你将可以使用任意一台计算机或操作系统来运行本书中的范例程序。

《编程秘诀》不打算从零开始教授 C++ 程序设计语言。为能理解本书各章节的内容,你应该已经完成了 C++ 程序设计的初级课程。或许你通过学习某一本 C++ 的教程,如我所编写的《学习 C++》,《掌握 Borland C++》或《Tom Swan's C++ 入门》等,你已经具备所要求的 C++ 基础知识。你应该熟悉面向对象的程序设计的有关概念,但并不要求你已经是一个面向对象的程序设计方面的行家里手。



注:如果这些术语如类、封装、继承和虚成员函数听起来还非常陌生的话,那么你应该先专心地学习 C++ 的基础知识,否则本书对你来说没有任何意义。

所有的程序清单和代码片段都符合目前可用的 ANSI C++ 标准草案。我已用 Borland C++ (3.1 版)和 Turbo C++ (3.0 版)测试了该书中的所有程序,当然你应该能在更高版本或最新版本的 C++ 编译器中使用这些程序。如果你在对程序进行编译的过程中遇到麻烦,可参考附录 A 的内容。

附在封底的软盘已做过格式化,可用于 100% 兼容的 IBM PC 机,软盘上的程序清单以 8 位 ASCII 文本方式存放,每一行都以回车换行结束,除

这些控制码外,这些文件不包含任何格式化指令或嵌入的制表符。

0.3 安装软盘



注:软盘上的文件以压缩形式存放。若要使用这些文件,则必须用所提供的实用程序 LHA.EXE 对文件进行解压缩处理。该盘已做过格式化,可用于 IBM PC 机或与之 100% 兼容的机器。

- (1) 在硬盘上创建一个目录以存放非压缩文件(在此称这个目录为 SECRETS,也可以取其它名字)。为把这些文件装到硬盘 C 上,可键入下列命令:

C:

md \secrets

- (2) 把软盘上所有的文件拷到这个新创建的目录下,软盘插在驱动器 A 中(或其它的软盘驱动器中),然后输入下列命令:

copy a:*.* c:\secrets

- (3) 拿掉软盘。你已经不再需要它了,不过它充当了这样一个相当昂贵的使者,以致你可能想把它安全地保存起来。转到硬盘这个新建目录下并运行所提供的 UNPACK.BAT 程序来完成安装。输入命令:

C:

cd \secrets

unpack

- (4) 看到显示的信息“Done(完成)”后,便可删除以.LZH 为后缀的压缩文件和 UNPACK.BAT 批处理文件。输入命令:

del *.lzh

del unpack.bat

- (5) 若要知道其它的注意事项及最新消息,可阅读 README.TXT 文件(可把它装入到任何 ASCII 文本编辑器中)



注:软盘上所提供的所有程序都以文本的形式存放,这种文本形式即本书中印刷程序清单的形式。因此,在运行这些程序之前,必须先编译这些程序。

如果你急于翻过本引言,则可以跳到第一章“转向流机制”否则,继续往下阅读关于本书内容的描述。

0.4 怎样使用这本书

本书不是描写犯罪的小说——既没有凶手也没有受害人(译者注:原文

butler 疑为 butcher 之误),因此无论你怎么杂乱无序地阅读各章节也不会影响你的理解。如果你很好地掌握了 C++, 则可以根据兴趣挑选要阅读章节的内容;如果你刚开始学习 C++, 那么你最好从头至尾地阅读,就象你学习 C++ 教程一样。

为了能使你从本书中获得最大的收益,这里再提出一些其它的建议:

- 手头保留一本 C++ 参考手册以查阅所不熟悉的术语和结构。我假设你已经具备了 C++ 的基础知识,因此为节约时间和空间,我不再定义有关术语及概念,一本好的 C++ 教程对这些术语及概念应该是作了解释的。
- 运行每章中用于说明概念的范例语句。如同不湿身就学不会游泳一样,不在计算机上运行程序就不可能学会 C++ 程序设计。
- 如果你不明白某条语句的用途,可以先删掉它,然后对修改后的代码进行编译和运行,并观察其结果。从程序中移去某一段代码的做法比捧着书本冥思苦想往往能更多更好地揭示这些语句的意义。
- 提出问题,并通过修改范例程序来回答这些问题,可以利用编译器来做一些“如果——会怎样”的游戏,例如,你可以提出这样一个问题:“如果在这个位置使用指针那么会怎么样?会发生什么现象?”你不要去“想”这个问题,而应该去修改程序代码从运行结果中找到答案。熟练地解决你自己提出的难题是扩展你的程序设计知识的一个关键秘诀。
- 在一些有重要意义的地方插入输出语句。一个插得恰到好处的输出语句(输出如“ Surprise(令人吃惊)!”之类的字串)也许比我在书本中所做的任何解释能更好地说明程序的组织结构。
- 在调试程序中运行已编译的程序。利用调试程序的命令检查变量、单步执行程序语句及在断点挂起程序等。一个调试程序可以充当一个非常好的老师——它的作用不仅仅在于捕捉程序隐错。如果你对系统的调试程序还不熟悉,则应该留出一个下午的时间来专门学习该调试程序的命令,将来你会为今天所花的这么一点时间而深感庆幸。

0.5 关于本书的章节

下面是对本书各大篇及章节的简短描述。阅读本节内容有助于在本书中找到特定的专题。如果你急于寻觅到一份提示或论据,则也可以查阅目录、索引及封面里的“Classified Secrets”(“秘诀分类”)。

0.5.1 篇

《编程秘诀》一书分为三大篇,每一篇都致力于一个有关 C++ 编程的高级技术专题,我选择这些专题的原因是因为它们具有实用性而且被绝大多数 C++ 初级教程所忽视。这几大篇分别说明如下:

- 第一篇 C++ 流的奥秘。I/O 流所具备的真正的能力已经在软件开发中被人们所认识。本篇中的各章将介绍 C++ 流及关于流、重载运算符、可流类、操纵符的及文件处理的高级使用法。

- 第二篇 C++数据结构的奥秘。有效地利用C++类的关键是把它们理解为新的数据类型。本篇中的各章将通过作为类来实现的种种数据类型来说明这个概念。其中的范例程序还广泛使用了C++模板——这是ANSI C++标准草案所采纳的最新特性之一。在这一篇中,你还可以完全依据模板上开发出一种复杂的串类以及全特性的容器类库。
- 第三篇 C++内存和数据管理的奥秘。C++给程序员提供控制内存分配的选项。本篇的各章将说明如何重载 new 和 delete 运算符来利用这种能力。在本篇中,你还可以学会如何构造定制堆管理器,如何处理存储错误、如何使用稀疏和三角矩阵以及如何在面向对象的数据库文件中存储“持久”对象。

0.5.2 章

每一篇中均有若干章覆盖相应专题的内容范围,下面是对每章内容的简短介绍。

0.5.2.1 第一篇 C++流的奥秘

- 第一章 转向流机制。这一章介绍C++流,列出其优缺点并讨论C++流与标准的I/O函数之间的差异。
- 第二章 理解汇、源和 iostream 类。这一章勾画了一个概括性的 iostream 类层次结构图,并说明如何充分利用库的汇和源对象(如 cout 和 cin)。
- 第三章 调用流成员函数。这一章更深层次地研究C++流,探讨可用于构造输入输出流的 iostream 类成员函数。
- 第四章 塑造操纵符。这一章说明如何使用操纵符及如何为操纵符编制程序以直接在流语句中形成输入和输出。
- 第五章 创建可流类。这一章说明如何把可流性添加到C++类中,实际上是对输入输出流重新编写程序以识别你自己定义的数据类型。
- 第六章 读写文件流。这一章详细描述了使用文件流来读写存储在文件中的数据,包括扩展标准的纯文本流来读写文件中二进制对象的指导。

0.5.2.2 第二篇 C++数据结构的奥秘

- 第七章 精制函数模板。这一章说明如何利用模板来创建编译时能自动形成接纳不同数据类型对象的函数。
- 第八章 定型类模板。这一章继续探讨模板,详细描述创建和使用编译器可重塑的类模板,这些类模板可用于实际中任何类型的对象和数值。
- 第九章 向量化数组。这一章说明如何利用类模板创建能容纳任一类型的数值和类对象的数组或向量。
- 第十章 构造串类。这一章指导如何逐步地开发复杂串类——这也是如何形成今后 ANSI C++正式标准的有争议的问题之一。
- 第十一章 实现基本数据结构。这一章提出了几种使用模板和类来开发面向对象形式的公共数据结构(如表、树)的方法。
- 第十二章 开发抽象容器类。这一章说明如何使用基本数据结构来实现

高级抽象容器结构,如数组、栈、队列、双队列和集合等。

0.5.2.3 第三篇 C++内存和数据管理的奥秘

- 第十三章 管理动态内存。这一章回顾了堆上存储 C++ 类对象及其它数据的内存管理技术,并针对如何充分利用 C++ 堆运算符即 new 和 delete 给出了几点提示。
- 第十四章 重载 new 和 delete。这一章介绍了通过重载 new 和 delete 来控制对象一内存分配的技术,并说明如何构造定制堆内存管理器。
- 第十五章 有效地存储对象。这一章介绍如何改进对象存储效率,包括指针、繁衍、位置语法、可变大小对象及复制对象引用等五种技术。
- 第十六章 利用稀疏矩阵赢得空间。这一章着重考虑使用稀疏数组、三角稀疏矩阵及矩形稀疏矩阵来节省内存资源的几种方法。范例程序说明如何利用这些“洞穴产生”技术在相对小的空间创建巨大稀疏结构。
- 第十七章 保存持久对象。这一章讨论了在多态数据库文件中程序运行生命期之外的存留的或持久的对象,并说明如何把不同种类的对象存储到某个文件,及如何把这些对象安全地读回到内存中。

0.6 结束语

《编程秘诀》在介绍指导高级的 C++ 技术的上下文中涉及了种种程序设计的问题,附带的软盘提供了本书中的所有程序清单,而且这些程序均可以用符合 ANSI C++ 标准草案的编译器进行编译,我已经用 Bortand C++ 和 Turbo C++ 测试了所有程序,你当然也可以用其它的符合标准的编译器来编译这些程序。

在安装完软盘及浏览了一遍关于本书概貌的介绍以后,你便可以转到本书第一篇阅读介绍输入输出流及说明它们与标准 I/O 函数之间的差异的有关内容。

目 录

前 言

引 言

第一篇 C++流的奥秘

第一章 转向流机制	(3)
1.1 垂死的标准	(3)
1.2 新生的流	(4)
1.2.1 标准 I/O	(4)
1.2.2 流 I/O	(5)
1.2.3 流的优点	(7)
1.2.4 流的缺点	(9)
1.3 iostream 库	(10)
1.4 结束语	(11)
第二章 理解汇、源和 iostream 类	(12)
2.1 汇和源	(12)
2.1.1 cin	(12)
2.1.2 cout	(13)
2.1.3 cerr	(14)
2.1.4 clog	(15)
2.2 流打结	(15)
2.3 类	(15)
2.4 在内存格式化	(17)
2.5 结束语	(20)
第三章 调用流成员函数	(21)
3.1 在 iostream 类的内部	(21)
3.2 ios 成员函数	(22)
3.3 ostream 成员函数	(26)
3.4 istream 成员函数	(27)
3.5 流状态	(33)
3.6 结束语	(36)
第四章 塑造操纵符	(37)
4.1 内部操纵符	(37)
4.1.1 作为过滤器的操纵符	(38)
4.1.2 标准操纵符	(38)
4.1.3 使用操纵符	(39)

4.1.4	将输入解析为字	(40)
4.1.5	带参数的操纵符	(41)
4.2	定制操纵符	(43)
4.2.1	作为函数的操纵符	(43)
4.2.2	输入和输出操纵符	(45)
4.2.3	作为对象的操纵符	(45)
4.3	结束语	(47)
第 5 章	建立可流(Streamable)类	(48)
5.1	重载流运算符	(48)
5.1.1	输出重载	(48)
5.1.2	输入重载	(51)
5.2	可流类	(52)
5.2.1	重载流运算符	(52)
5.2.2	友元与成员	(54)
5.2.3	一个可流类的范例	(56)
5.2.4	重载输出(<<)	(57)
5.2.5	重载输入(>>)	(60)
5.3	多态流	(64)
5.4	结束语	(67)
第 6 章	读写文件流	(68)
6.1	文件流类	(68)
6.2	文本文件流	(69)
6.2.1	建立和写文本文件	(69)
6.2.2	一次一个字符地读文本文件	(72)
6.2.3	一次一个字符地写文本文件	(73)
6.2.4	一次一行地读文本文件	(75)
6.2.5	一次一行地写文本文件	(77)
6.3	二进制文件流	(78)
6.3.1	使用二进制文件流	(78)
6.3.2	错误的解决办法	(79)
6.3.3	正确的解决办法	(80)
6.3.4	用于二进制文件 I/O 的类	(84)
6.3.5	写二进制文件	(88)
6.3.6	在二进制文件中存入其它对象	(89)
6.3.7	在流中搜寻	(92)
6.4	其它的文件流技术	(95)
6.4.1	打开和关闭文件对象	(95)
6.4.2	获取文件描述符	(96)
6.4.3	从同一个文件中读写	(97)
6.5	结束语	(99)

第 二 篇 C++ 数据结构的奥秘

第 七 章	精制函数模板	(103)
--------------	---------------	-------

7.1 什么是模板?	(103)
7.2 函数模板	(104)
7.2.1 函数模板语法	(104)
7.2.2 min 和 max 函数模板	(105)
7.2.3 前向函数模板	(108)
7.2.4 显示函数-模板原型	(109)
7.2.5 特定模板函数	(111)
7.2.6 排序函数模板	(113)
7.2.7 在何处存储函数模板	(117)
7.3 结束语	(117)
第八章 定型类模板	(118)
8.1 回顾模板	(118)
8.2 类模板	(118)
8.2.1 类模板语法	(119)
8.2.2 类模板的对象	(120)
8.2.3 多重模板函数	(122)
8.2.4 复杂模板	(124)
8.3 实用类模板	(125)
8.3.1 类模板中的成员函数	(125)
8.3.2 派生类和类模板	(127)
8.4 结束语	(130)
第九章 向量化数组	(131)
9.1 单下标向量	(131)
9.1.1 快速却低劣的向量类	(132)
9.1.2 改进的向量类	(134)
9.1.3 错误处理	(136)
9.2 成品向量类模板	(137)
9.2.1 对模板的类	(137)
9.2.2 vector.h 头文件	(139)
9.2.3 向量受保护数据成员	(142)
9.2.4 向量构造函数及析构函数	(143)
9.2.5 向量初始化	(146)
9.2.6 向量内联成员函数	(146)
9.2.7 向量重载运算符	(147)
9.3 向量存储要求	(148)
9.3.1 类对象的向量	(148)
9.3.2 间接存储	(150)
9.4 排序向量模板	(151)
9.5 结束语	(155)
第十章 构造字符串类	(156)
10.1 串作为向量	(156)
10.2 字符串包装类	(158)

10.2.1 数据成员	(158)
10.2.2 错误控制	(159)
10.2.3 构造函数与析构函数	(161)
10.2.4 类的调试	(171)
10.2.5 运算符	(173)
10.2.6 成员函数	(182)
10.3 最终的 String 类	(184)
10.4 字串和模板	(187)
10.5 结束语	(188)
第十一章 实现基本数据结构	(189)
11.1 基本数据结构	(189)
11.1.1 重温向量	(190)
11.1.2 用继承解决问题	(191)
11.2 链表类	(193)
11.2.1 直接链表	(194)
11.2.2 间接链表	(194)
11.2.3 链表类声明	(195)
11.2.4 Carrier<T> 实现	(197)
11.2.5 链表占据权(ownership)	(198)
11.2.6 List<T> 实现	(199)
11.3 迭代器(iterator)类	(205)
11.3.1 链表迭代器	(206)
11.3.2 向量迭代器	(215)
11.4 树类	(219)
11.5 结束语	(231)
第十二章 开发抽象容器类	(232)
12.1 介绍容器类	(232)
12.2 表容器和向量容器	(233)
12.2.1 表容器和向量容器	(234)
12.2.2 错误处理	(243)
12.3 介绍抽象数据类型	(243)
12.4 数组类型	(243)
12.4.1 实用的 ArrayAsVector<T>	(245)
12.4.2 排序数组	(247)
12.5 栈类型	(250)
12.6 队列类型	(258)
12.7 双队列类型	(262)
12.8 集合类型	(266)
12.9 结束语	(273)

第三篇 C++内存及数据管理的奥秘

第十三章 管理动态内存	(277)
--------------------------	-------

13.1	动态对象	(277)
13.1.1	运算符 new 和 delete	(278)
13.1.2	可靠的对象构造	(280)
13.1.3	坏的引用	(284)
13.1.4	好的引用	(285)
13.2	安全的对象构造	(286)
13.3	结束语	(288)
第十四章	重载的 new 和 delete	(289)
14.1	进入重载	(289)
14.1.1	全局的 new 和 delete 运算符	(289)
14.1.2	类的 new 和 delete 运算符	(291)
14.1.3	继承	(294)
14.1.4	虚析构造函数	(295)
14.1.5	公有还是私有	(296)
14.1.6	数组与重载的 new 和 delete	(299)
14.2	利用 new 和 delete 进行调试	(299)
14.3	标记-释放内存管理	(301)
14.4	设置 new 处理器	(306)
14.4.1	内存溢出	(306)
14.4.2	浅内存池	(306)
14.5	结束语	(310)
第十五章	有效地存储对象	(312)
15.1	对象和指针	(312)
15.2	对象繁衍	(317)
15.2.1	错误的繁衍方式	(317)
15.2.2	正确的繁衍方式	(319)
15.3	对象和位置语法	(325)
15.3.1	把标志传递给 new	(326)
15.3.2	把对象放到特定位置	(327)
15.4	可变大小对象	(331)
15.5	重复对象引用	(336)
15.6	结束语	(346)
第十六章	利用稀疏矩阵赢得空间	(347)
16.1	稀疏数组	(347)
16.2	稀疏三角矩阵	(350)
16.3	稀疏直方矩阵	(357)
16.4	结束语	(368)
第十七章	保存持久的对象	(369)
17.1	文件中的对象	(369)
17.1.1	难题	(370)
17.1.2	目标	(371)
17.1.3	解答	(374)

17.2	持久性的彻底解决.....	(379)
17.3	结束语.....	(399)
附录 A	程序清单编译	(400)
A.1	普遍错误	(400)
A.2	使用自动的 MAKE 文件	(402)
A.3	建立函数库	(407)
A.4	Borland C++ IDE	(410)
A.5	Turbo C++ IDE	(413)
A.6	MAKEFILE 文件	(413)
附录 B	磁盘目录	(427)
B.1	目录和文件	(427)
附录 C	磁盘上的附赠程序	(432)
C.1	ObjectEase	(432)
C.2	Turbo Screen Edit	(433)
C.3	ProModem	(434)
C.4	LHA	(434)

第一篇 C++流的奥秘