SAMS Teach Yourself C++ in 21 Days

经典C++教程
全美销量超过25万册

21天学通C++

（第五版）

〔美〕Jesse Liberty Bradley Jones 著
李佩乾 杨小珂 译

人民邮电出版社
POSTS & TELECOM PRESS

21天学通C++（第五版）

[美] Jesse Liberty　Bradley Jones　著

李佩乾　杨小珂　译

人民邮电出版社

图书在版编目（CIP）数据

21 天学通 C++：第五版 /（美）利伯帝（Liberty, J.），（美）琼斯（Jones, B.）著；李佩乾，杨小珂译. —北京：人民邮电出版社，2005.9

ISBN 7-115-13692-0

Ⅰ. 2… Ⅱ. ①利…②琼…③李…④杨… Ⅲ. C 语言－程序设计 Ⅳ. TP312

中国版本图书馆 CIP 数据核字（2005）第 097601 号

版 权 声 明

21 天学通 C++（第五版）

◆ 著 [美] Jesse Liberty Bradley Jones

译 李佩乾 杨小珂

责任编辑 李 际

◆ 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：787×1092 1/16

印张：40.5

字数：1 317 千字 2005 年 9 月第 1 版

印数：1－5 000 册 2005 年 9 月北京第 1 次印刷

著作权合同登记号 图字：01-2004-5456 号

ISBN 7-115-13692-0/TP・4812

定价：59.00 元

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

作者简介

Jesse Liberty 编著了大量有关软件开发的图书，其中包括 C++ 和 .NET 方面的畅销书。他是 Liberty Associates 公司（http://www.LibertyAssociates.com）的总裁，该公司致力于为客户提供编程、咨询和培训方面的服务。

Bradley Jones 是 Microsoft Visual C++ MVP，他身兼网站管理员、经理、编码大师、执行编辑等职，其主要精力放在众多软件开发网站和频道上，其中包括 Developer.com、CodeGuru.com、DevX、VBForums、Gamelan 以及 Jupitermedia 的其他网站。这些影响力在不断扩大的网站每月为 250 万开发人员提供信息。

他最擅长的是大 C 语言领域：C、C++和 C#，但在 PowerBuilder、VB、Java、ASP 和 COBOL I/II 开发以及各种现在没有人提起的古老技术方面也拥有丰富的经验。他还从事过咨询人员、分析员、项目负责人、联合发行人和作者等工作，他最近参与编著的图书包括《21 天学通 C#》和《21 天学通 C 语言（第六版）》等。他还是 Indianapolis .NET Developers Association（成员超过 700 的 INETA 集团的一家连锁公司）的联合创始人兼总裁。你经常可以在 CodeGuru.com 和 VBForums.com 论坛上看到他发表的贴子，他还每周在 CodeGuru 上发表一篇通信稿，其读者为数以万计的开发人员。

前　言

本书旨在帮助读者学习如何使用 C++进行编程。没有人仅在三个星期内就能学好一种严谨的编程语言，但本书每章的内容都可以在几小时内阅读完毕。

只需 21 天，读者就能学习诸如控制输入/输出、循环和数组、面向对象编程、模板和创建 C++应用程序等基本知识，所有这些内容都被组织成结构合理、易于理解的章节。每章都提供范例程序清单，并辅以范例输出和代码分析以演示该章介绍的主题。

为加深读者对所学内容的理解，每章最后都提供了常见问题及其答案以及测验和练习。读者可对照附录 D 提供的测验和练习答案，了解自己对所学内容的掌握程度。

本书针对的读者

通过阅读本书来学习 C++时，读者不需要有任何编程经验。本书从入门开始，既介绍 C++语言，又讨论使用 C++进行编程涉及的概念。本书提供了大量语法实例和详细的代码分析，它们是引导读者完成 C++编程之旅的优秀向导。无论读者是刚开始学习编程还是已经有一些编程经验，书中精心安排的内容都将让你的 C++学习变得既快速又轻松。

本书约定

提示：提供使读者进行 C++编程时更高效、更有效的信息。

注意：提供与读者阅读的内容相关的信息。

FAQ：对 C++语言的用法进行了深入剖析，澄清一些容易混淆的问题。

警告：提醒读者注意在特定情况下可能出现的问题或副作用。

应该：提供当前章介绍的基本原理的摘要。

不应该：提供一些有用的信息。

在程序清单中，在每行代码中都加上了行号；没有行号的代码行是前一行的续行（有些代码行太长，无法在一行中列出）。这种情况下，应将两行作为一行输入，不能将它们分开。

本书的范例代码

本书正文及附录 D 中的范例代码可从 Sams 网站下载，其网址为 http://www.samspublishing.com。在文本框“Search”中输入本书英文版的 ISBN（0672327112），单击 Search 按钮，然后单击原版书名（Sams Teach Yourself C++ in 21 Days, 5th Edition）便可链接到可下载范例代码的页面，点击 Downloads 即可下载。

目　录

第1周课程简介

第 1 周复习

第 2 周课程简介

第 3 周复习

第 1 周课程简介

开始学习如何使用 C++进行编程之前，读者需要有几样东西：编译器、编辑器以及本书。即使没有 C++编译器和编辑器，读者仍可通过本书学习 C++，但如果能做些练习，将学到更多的知识。

学习编程的最好方法就是编写程序！在每章最后都有作业，其中包括测验和一些练习。请务必花时间回答所有的问题，并尽可能客观地为自己打分。后面的章节都是以前面的内容为基础的，因此继续阅读后面的内容之前，一定要完全理解当前的知识。

C 程序员的注意事项

C 程序员一定很熟悉本书前 5 章的内容，然而，如果要遵循 C++标准，将存在一些细微的差别。请务必浏览这些内容并完成练习，以确保能够快速地进入第 6 章的学习。

本周内容简介

本周简要地介绍编程（尤其是 C++编程）所需的一些知识。第 1 章和第 2 章介绍有关编程和程序流程的基本概念；第 3 章介绍有关变量和常量以及如何在程序中使用数据的知识；第 4 章讨论程序如何根据运行时提供的数据和遇到的条件执行不同的分支；第 5 章介绍函数是什么以及如何使用它们；第 6 章介绍类和对象；第 7 章介绍更深入地探讨有关程序流程的知识。经过本周的学习后，读者将能够编写真正的面向对象的程序。

第 1 章　绪　　论

欢迎使用本书！今天你将迈出成为高级 C++程序员的第一步。

本章将学习：

- 为何 C++是软件开发的标准？
- 开发 C++程序的步骤。
- 如何输入、编译和链接你的第一个 C++程序。

1.1　C++简史

第一代电子计算机诞生于二战期间，用来帮助计算武器弹道数据。从此以后，计算机语言经历了翻天覆地的变化。起初，程序员们使用最原始的计算机指令即机器语言来编程。这些指令是由 0 和 1 组成的很长的字符串。很快，人们就发明了汇编语言，将机器指令映射为人们可以阅读和易于处理的助记符，如 ADD 和 MOV。

随后出现了高级语言，如 BASIC 和 COBOL。这些语言让人们能够用一些类似于单词或句子的源代码来编程，如 Let I=100。这些指令再通过解释器和编译器转换为机器语言。

解释器翻译并执行程序，它读取程序指令（源代码），直接将其转换为操作。

编译器先将源代码转换为中间格式；这通常被称为编译，它生成目标文件。然后，编译器调用链接器，将目标文件组合成为可执行程序。

由于解释器按原样读取代码并当场执行代码，因此程序员使用起来比较方便。当前，大部分解释型程序被称为脚本，解释器被称为脚本引擎。

有些语言（如 Visual Basic 6）将解释器称为运行库；其他语言（如 Visual Basic .NET 和 Java）有另一个组件——虚拟机（Virtual Machine，VM）或运行库。它也是解释器，然而它不是源代码解释器——将人类可读的语言转换为依赖于计算机的机器码；而是对一种编译后的独立于计算机的语言（中间语言）进行解释和执行。

编译器增加了一个步骤：将人类能够理解的源代码编译成机器能够理解的目标代码。这个步骤看似不太方便，但由于将源代码转换为机器语言这样一个耗时的任务已经在编译阶段完成，因此编译型程序的运行速度非常快。由于转换工作已经完成，因此执行程序时无需做这样的工作。

诸如 C++等编译型语言的另一个优点是，可以向没有编译器的用户提供可执行程序。使用解释型语言时，必须有解释器才能运行程序。

C++通常是一种编译型语言，虽然存在一些 C++解释器。和众多其他编译型语言一样，C++以能够生成快速而功能强大的程序著称。

事实上，很多年来，计算机程序员的一个主要目标是编写能够快速执行的简短代码。程序必须短小，因为内存昂贵；同时必须快速运行，因为处理能力也很昂贵。随着计算机的体积越来越小、价格越来越便宜、速度越来越快，内存的价格不断下降，这些优先考虑的因素发生了变化。当前，程序员的时间费用已远远超过了大多数商用计算机的费用。精心编写和易于维护的代码成了编程的首要目标。易于维护是指当程序需求

发生变化时，无需花太大的代价就可以对程序进行扩展和改进。

注意：程序一词通常有两种含义：一种是指程序员编写的指令（源代码），另一种是指可执行软件。这种区分可能会引起巨大的混乱，因此本书尽可能将源代码和可执行文件区分开来。

1.1.1 解决问题

当前，程序员们面临的问题与20年前完全不同。在20世纪80年代，人们编写程序是为了管理和处理大量的原始数据。那时，编写代码的人和使用程序的人都是计算机专业人员。现在，越来越多的人开始使用计算机，其中大部分人对计算机和程序的工作原理知之甚少。人们通常只愿意将计算机作为一种工具来解决商务问题，而无心去钻研它。

具有讽刺意味的是，为了满足这些新用户易于使用的要求，程序变得越来越复杂。那种需要输入神秘的命令，结果却只看到一大串枯燥的数据的时代已一去不复返。当前的程序大都采用“用户友好界面”，其中包括多个窗口、菜单、对话框以及非常形象的标识，对此我们已经非常熟悉了。

随着万维网的发展，计算机跨入了一个新的市场渗透时代；使用计算机的人比任何时候都多，他们对计算机的期望也非常高。万维网的易用性也提高了人们的期望，他们期望程序能够充分利用万维网提供的信息。

在过去的几年中，应用程序的运行平台已扩展到其他设备，不再限于桌面PC，而被用于手机、个人数字助理（PDA）、平板PC和其他设备。

在本书第一版面世后的几年中，为满足用户的需求，程序员编写的程序变得越来越庞大，越来越复杂。因此，对有助于应对这种复杂性的编程技术的需求显得越来越迫切。

随着编程要求的改变，用于编写程序的语言和技术都得到了发展，以帮助程序员应对这种复杂性。虽然完整的发展历史令人神往，但本书只介绍其中的重要组成部分：从过程化编程到面向对象编程的转变。

1.1.2 过程化编程、结构化编程和面向对象编程

直到不久前，计算机程序还被看作是一系列处理数据的过程。过程（函数或方法）是指一组依次执行的指令。数据与过程是分离的，编程技巧是跟踪哪些函数调用了其他函数以及哪些数据被修改了。为避免发生，结构化编程应运而生。

结构化编程的主要思想是分而治之。可将计算机程序看作由一系列任务组成。任何过于复杂、无法简单描述的任务都将分解为一系列较小的子任务，直至每个任务都很小，很容易理解。

例如，计算公司职员的平均工资是一项较复杂的任务。然而，可将其划分成下面几个子任务：

（1）确定公司员工数。

（2）确定每个员工的工资。

（3）计算工资总额。

（4）将工资总额除以员工数。

计算工资总额还可分成以下几步：

（1）读取每个职员的记录。

（2）读取工资额。

（3）将工资额添加到工资总额中。

（4）读取下一个职员的记录。

读取每个职员的记录又可分为以下几步：

（1）打开职员文件。

（2）找到正确的记录。

（3）读取数据。

结构化编程仍是一种非常成功的、处理复杂问题的方法。然而，到20世界80年代后期，它的许多不足逐渐暴露出来了。

首先，一种自然而然的愿望是，将数据（如职员记录）及其操作（排序、编辑等）看作一个整体。不幸