SAMS

畅销全球的
经典C++教程

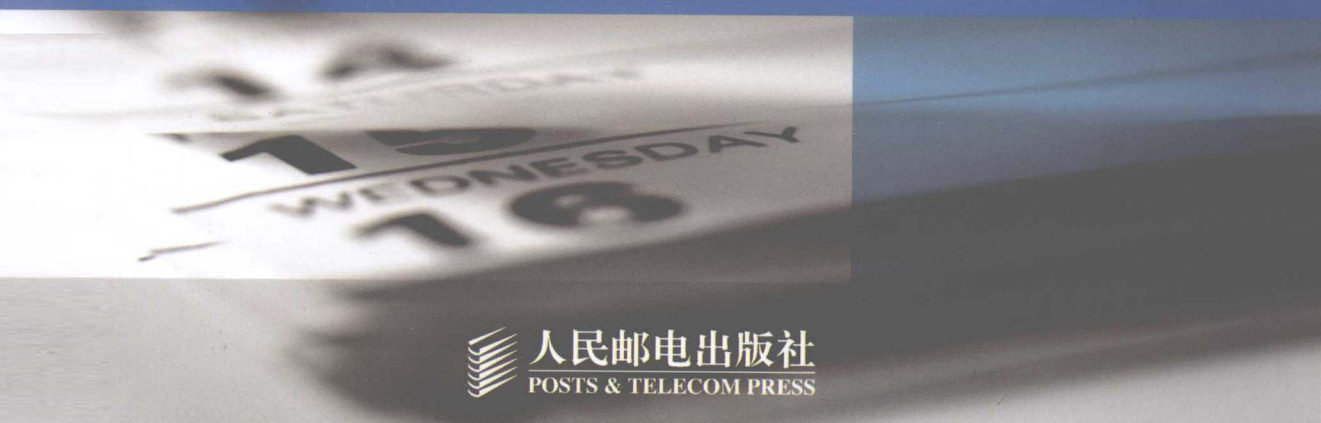
Sams Teach Yourself C++
in One Hour a Day
Seventh Edition

中文版
累计销量
超50000册

全面涵盖C++11新标准
帮助读者编写高效的C++应用程序

# 21天学通C++（第7版）

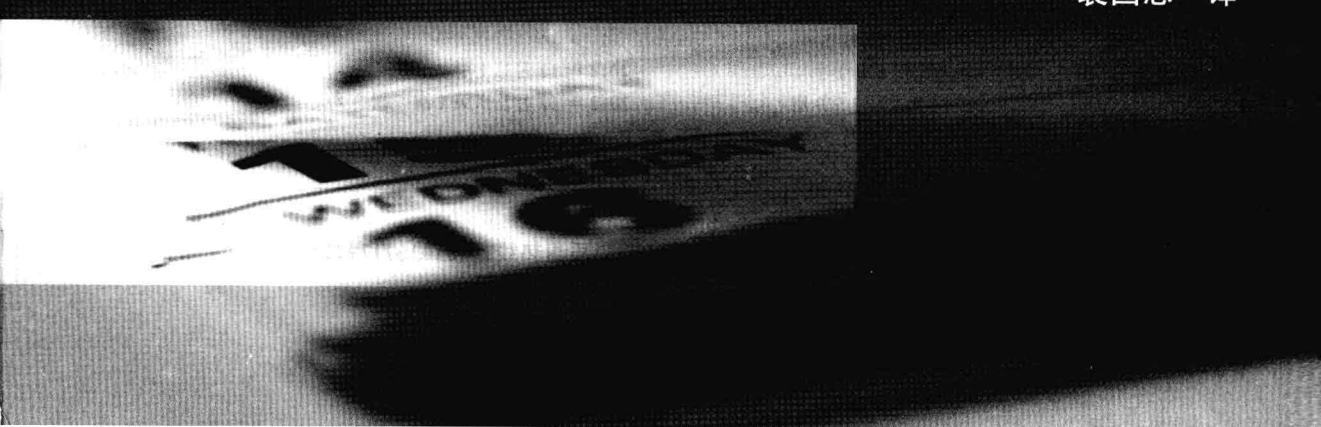
[美] Siddhartha Rao 著
袁国忠 译




人民邮电出版社
POSTS & TELECOM PRESS

# 21天学通C++（第7版）

[美] Siddhartha Rao 著
袁国忠 译



**图书在版编目（CIP）数据**

21天学通C++ : 第7版 / (美) 罗奥 (Rao, S.) 著 ; 袁国忠译. -- 北京 : 人民邮电出版社, 2012.12
ISBN 978-7-115-29624-5

Ⅰ. ①2… Ⅱ. ①罗… ②袁… Ⅲ. ①C语言－程序设计 Ⅳ. ①TP312

中国版本图书馆CIP数据核字(2012)第235664号



**21 天学通 C++（第 7 版）**

◆ 著 [美] Siddhartha Rao
译 袁国忠
责任编辑 傅道坤

◆ 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 电子邮件 315@ptpress.com.cn
网址 http://www.ptpress.com.cn
北京昌平百善印刷厂印刷

◆ 开本：787×1092 1/16
印张：29.5
字数：870 千字 2012 年 12 月第 1 版
印数：1–3 500 册 2012 年 12 月北京第 1 次印刷

著作权合同登记号 图字：01-2012-5025 号
ISBN 978-7-115-29624-5

定价：59.00 元

读者服务热线：(010)67132692 印装质量热线：(010)67129223
反盗版热线：(010)67171154
广告经营许可证：京崇工商广字第 0021 号

# 内容提要

本书通过大量短小精悍的程序，详细而全面地阐述了C++基本概念和技术以及C++11新增的功能，包括管理输入/输出、循环和数组、面向对象编程、模板、使用标准模板库以及lambda表达式等。这些内容被组织成结构合理、联系紧密的章节，每章都可在1小时内阅读完毕；每章都提供了示例程序清单，并辅以示例输出和代码分析，以阐述该章介绍的主题。为加深读者对所学内容的理解，每章末尾都提供了常见问题及其答案以及练习和测验。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。

本书是针对C++初学者编写的，不要求读者有C语言方面的背景知识，可作为高等院校教授C++课程的教材，也可供初学者自学C++时使用。

# 作者简介

Siddhartha Rao 是全球领先的企业软件提供商 SAP AG 的技术专家。作为 SAP Product Security India 的负责人，其主要职责包括招募产品安全领域的专家以及制定软件开发最佳实践，以保持 SAP 软件的全球竞争力。作为一位 Microsoft Visual C++ MVP，他深信 C++11 有助于编写速度更快、更简洁、更高效的 C++应用程序。

Siddhartha 酷爱旅游，不放过任何一次探索新文化的机会。例如，本书就是在 4 个不同的国度创作而成的，其中包括法国布列塔尼一个面朝大西洋的奇特村庄。他期待着您对这部全球之作提出宝贵的建议。

# 献词

谨将本书献给我的父母和我的妹妹，他们是我坚强的后盾。

# 致谢

当我为了编写本书而通宵达旦地忙碌时，是我的朋友承担起了我的后勤工作，对此深表谢意。谢谢本书的所有编辑人员，正是他们的勤劳付出，才让本书出现在各位读者的书架上。

# 前　言

对C++来说，2011是个很特别的年份。在这一年，C++11终于获批成为新标准，它新增了一些可提高编程效率的关键字和结构，让您能够编写更优质的代码。本书旨在帮助您循序渐进地学习C++11，其中的章节经过仔细编排，从实用的角度介绍这种面向对象的编程语言的基本知识。读者只需每天花1小时，在学完本书后，就能掌握C++11。

学习C++的最佳方式是动手实践。本书包含丰富的代码示例，有助于读者提高编程技能，请务必亲自动手尝试这些代码。这些代码片段都使用了（在本书编写时）最新版本的编译器进行了测试，具体地说是Microsoft Visual C++ 2010和GNU C++编译器4.6版，它们都支持大量的C++11功能。

## 针对的读者

本书从最基本的C++知识开始介绍，读者只需具备学习C++的愿望及了解工作原理的好奇心即可；虽然具备一些C++知识会有所帮助，但这并非必需的。本书也可供熟悉C++但想了解C++11新增功能的读者参考；如果你是专业程序员，第3部分“学习标准模板库”可以帮助你创建更优质、更实用的C++应用程序。

## 本书内容

读者可根据自己对C++的熟练程度，阅读感兴趣的部分。本书包含5部分。

- 第1部分“基本知识”，引导读者编写一些简单的C++应用程序，并介绍一些在C++的未妥协类型安全变量的代码中最常见的关键字。
- 第2部分“C++面向对象编程基础”，介绍类的概念，您将学习C++如何支持封装、抽象、继承和多态等重要的面向对象编程原则。第9章将介绍C++11新增的移动构造函数，而第12章将介绍移动赋值运算符。这些功能有助于避免不必要的复制步骤，从而提升应用程序的性能。第14章是一个跳板，助您编写功能强大的C++通用代码。
- 第3部分“学习标准模板库”，将帮助您使用STL string类和容器编写高效而实用的C++代码。您将了解到，使用std::string可安全而轻松地拼接字符串，您不再需要使用C风格字符串（char*）。您可使用STL动态数组和链表，而无需自己编写这样的类。
- 第4部分“再谈STL”，专注于算法，您将学习如何通过迭代器对vector等容器进行排序。在这部分，您将发现，通过使用C++11新增的关键字auto，可极大地简化冗长的迭代器声明。第22章将介绍C++11新增的lambda表达式，这可极大地简化使用STL算法的代码。
- 第5部分“高级C++概念”，阐述智能指针和异常处理等C++功能。对C++应用程序来说，这些功能并非必需的，但可极大地提高应用程序的稳定性和品质。在这部分的最后，简要地介绍了有助于编写杰出C++应用程序的最佳实践。

## 本书体例

本书使用了下述提供更多信息的元素：

**注意** 提供与读者阅读的内容相关的信息。

**C++11**

突出 C++11 新增的功能。要使用这些功能，可能需要使用较新的编译器版本。

**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。

**提示** 提供 C++编程最佳实践。

| 应该 | 不应该 |
| --- | --- |
| 提供当前章介绍的基本原理的摘要。 | 提供一些有用的信息。 |

本书使用不同的字体来区分代码和正文，全书都用特殊字体呈现代码、命令以及与编程相关的术语。

# 目　录

# 第1章 绪 论

欢迎使用本书！通过阅读本章，您将迈出成为高级C++程序员的第一步。

在本章中，您将学习：

- 为何C++是软件开发的标准；
- 输入、编译和链接第一个C++程序；
- C++11新增的功能。

## 1.1 C++简史

编程语言旨在让人更容易使用计算资源。C++并非一种新语言，但仍被广泛采用，并在不断改进。2011年，最新的C++标准获得了ISO标准委员会的批准，名为C++11。

### 1.1.1 与C语言的关系

C++最初由Bjarne Stroustroup于1979年在贝尔实验室开发，被设计为C语言的继任者。C语言是一种过程型语言，程序员使用它定义执行特定操作的函数，而C++是一种面向对象的语言，实现了继承、抽象、多态和封装等概念。C++支持类，而类包含成员数据以及操作数据的成员方法（方法类似于C语言中的函数）。其结果是，程序员需要考虑数据以及要用它们来做什么。一直以来，C++编译器都支持C语言，这具有向后与既有代码兼容的优势，但也存在缺点，那就是编译器非常复杂，因为随着C++的发展，编译器既要实现所有的新功能，又要向程序员提供这种向后兼容的功能。

### 1.1.2 C++的优点

C++是一种中级编程语言，这意味着使用它既可以高级编程方式编写应用程序，又可以低级编程方式编写与硬件紧密协作的库。在很多程序员看来，C++既是一种高级语言，让他们能够开发复杂的应用程序，又提供了极大的灵活性，让开发人员能够控制资源的使用和可用性，从而最大限度地提高性能。

虽然有更新的编程语言面世，如Java以及其他基于.NET的语言，但C++始终深受欢迎并在不断发展。较新的语言因提供了某些功能（如通过垃圾收集管理内存）让一些程序员钟爱有加，但在需要精确控制应用程序的性能时，他们还是会选择C++。当前，常常使用C++编写Web服务器，并使用HTML、Java或.NET编写前端应用程序。

### 1.1.3 C++标准的发展历程

经过多年的发展，C++得到了广泛接受和采纳，但存在多种不同版本，因为有很多不同的编译

器，它们风格各异。鉴于C++广受欢迎，且不同的版本之间存在差异，这导致了众多互操作性和移植方面的问题，需要对其进行标准化。

1998年，第一个C++标准获得了ISO标准委员会的批准，这就是ISO/IEC 14882:1998。2003年进行了修订，即为ISO/IEC 14882:2003。最新的C++标准于2011年8月获批，其官方名称为C++11（ISO/IEC 14882:2011），它包含一些雄心勃勃的改进。

**注意**

网上的很多文档仍称这个C++标准为C++0x。人们最初预期新标准将于2008或2009年获批，因此用x表示获批的年份。但提议的新标准到2011年才获批，因此将其称为C++11。换句话说，C++11是新的C++0x。

### 1.1.4 哪些人使用C++程序

无论你是什么人，做什么工作（无论是经验丰富的程序员，还是将计算机用于特定目的的人），都可能经常会用到C++应用程序和库。C++常用于开发操作系统、设备驱动程序、办公软件、Web服务器、基于云的应用程序和搜索引擎，甚至用于编写新编程语言编译器。

## 1.2 编写C++应用程序

当您在计算机上启动Notepad或VI时，实际上是命令处理器运行该程序的可执行文件。可执行文件是可运行的成品，应按程序员期望的那样做。

### 1.2.1 生成可执行文件的步骤

要创建可在操作系统中运行的可执行文件，第一步是编写一个C++程序。创建C++应用程序的基本步骤如下。

1. 使用文本编辑器编写C++代码。
2. 使用C++编译器对代码进行编译，将代码转换为包含在目标文件中的机器语言版本。
3. 使用链接程序链接编译器的输出，生成一个可执行文件（如Windows中的.exe文件）。

您在编程时创建的是文本文件，但微处理器无法处理这样的文件。在编译过程中，C++代码（通常包含在.CPP文本文件中）被转换为处理器能够处理的字节码。编译器每次转换一个代码文件，生成一个扩展名为.o或.obj的目标文件，并忽略这个CPP文件可能对其他文件中代码的依赖。解析这些依存关系的工作由链接程序负责。除将各种目标文件组合起来外，链接程序还建立依存关系，如果链接成功，则创建一个可执行文件，供程序员执行和分发。

### 1.2.2 分析并修复错误

大多数复杂应用程序很少能够一次通过编译并完美地运行，由众多程序员合作开发的应用程序尤其如此。无论使用什么语言（包括C++）编写，庞大或复杂的应用程序都需要运行很多次，以分析问题和发现Bug。修复Bug后，重新生成程序，再重复上述过程。因此，除编写、编译和链接等三个步骤外，开发过程通常还包括调试步骤。在这个步骤中，程序员使用工具（如监视点）和调试功能（如逐行执行应用程序）对应用程序中的异常和错误进行分析。

### 1.2.3 集成开发环境

很多程序员都喜欢使用集成开发环境（Integrated Development Environment，IDE）。集成开